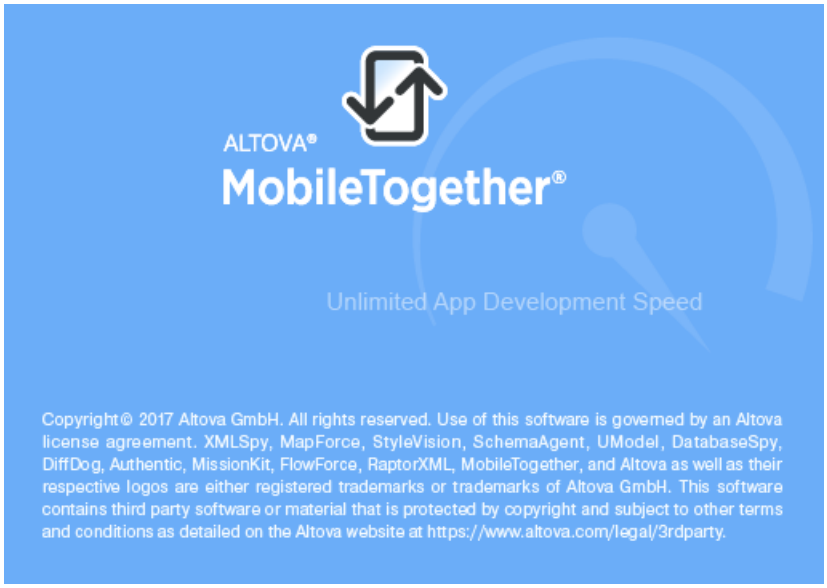
ユーザーマニュアル


ALTOVA®
MobileTogether®
Unlimited App Development Speed
Copyright© 2017 Altova GmbH. All rights reserved. Use of this software is governed by an Altova license agreement. XMLSpy, MapForce, StyleVision, SchemaAgent, UModel, DatabaseSpy, DiffDog, Authentic, MissionKit, FlowForce, RaptorXML, MobileTogether, and Altova as well as their respective logos are either registered trademarks or trademarks of Altova GmbH. This software contains third party software or material that is protected by copyright and subject to other terms and conditions as detailed on the Altova website at https://www.altova.com/legal/3rdparty.

Altova MobileTogether Designer
ユーザー マニュアル

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/ or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

発行日: 2018

(C) 2018 Altova GmbH

目次

1 Altova MobileTogether Designer 3

2 新機能 6

2.1 バージョン 3 ... 10
2.2 バージョン 2 ... 13
2.3 バージョン 1 ... 16

3 はじめに 20

3.1 MobileTogether 概要 ... 21
3.2 用語 Q&A ... 23
3.3 デザインのステップ ... 25
3.4 MobileTogether 内の XPath ... 27

4 チュートリアル 32

4.1 クイックスタート(パート1) ... 34
4.1.1 新しいデザインの作成 ... 35
4.1.2 ページのセットアップ ... 36
4.1.3 ページ データソースの追加 ... 37
4.1.4 デザインのフォーマット ... 41
4.1.5 コントロールの追加: コンボ ボックス ... 42
4.1.6 コントロールの追加: イメージ ... 45
4.1.7 コントロール アクションの定義 ... 47
4.1.8 プロジェクトの検証 ... 49
4.1.9 シミュレーションの実行 ... 50
4.1.10 サーバーにデプロイ ... 53
4.2 クイックスタート(パート2) ... 55
4.2.1 ファイルからデータをロード ... 56
4.2.2 ソース ノードの変更 ... 59
4.2.3 シミュレーションの実行 ... 61
4.2.4 コンボ ボックス エントリにファイル データを使用 ... 63
4.2.5 データファイルをデフォルトファイルとして設定 ... 65
4.2.6 Web ページへの動的リンクの作成 ... 67

4.2.7 データをファイルに戻して保存 ... 69
4.3 シンプルなデータベース ... 70
4.3.1 DB データソース ... 71
4.3.2 ユーザーの入力を保管するための永続的なツリー ... 73
4.3.3 ユーザーの選択に基づき DB データをロードする ... 75
4.4 詳細データベース ... 80
4.4.1 ページソース: DB と Persistent ... 82
4.4.2 可視性の管理 ... 86
4.4.3 DB 内のイメージ ... 87
4.4.4 レコードの編集 ... 89
4.4.5 レコードの追加 ... 91
4.5 データベースとチャート ... 93
4.5.1 プロジェクトの構造 ... 94
4.5.2 メインページ ... 95
4.5.3 メインページのデータソース ... 97
4.5.4 コンボ ボックス ... 101
4.5.5 表形式レポート ... 104
4.5.6 チャート ... 105
4.5.7 オフィス テーブルの編集 ... 109
4.5.8 セールステーブルの編集 ... 113
4.6 サブページと可視性 ... 118
4.6.1 デザイン構成 ... 120
4.6.2 データソース リスト ... 122
4.6.3 トップページ: データソース ... 124
4.6.4 トップページ: カスタマーテーブル ... 126
4.6.5 トップページ: アクショングループ、サブページへ移動 ... 127
4.6.6 トップページ: すべてのオーダー アクションを表示する ... 130
4.6.7 サブページ: データソース ... 131
4.6.8 サブページ: オーダー テーブル ... 133
4.6.9 サブページ: 可視性プロパティ ... 135
4.6.10 サブページ:XPath 内の総額 ... 136
4.6.11 シミュレーションとテスト ... 137
4.7 レコードの追加と編集 ... 138
4.7.1 デザインページ ... 139
4.7.2 データソース ... 140
4.7.3 新規レコードを追加する ... 142
4.7.4 新規レコードデータを入力する ... 144
4.7.5 全てのレコードを表示する ... 145
4.7.6 既存のレコードを編集する ... 147
4.8 SOAP リクエスト ... 148

4.8.1 XML ページソース ... 150
4.8.2 デザインコンポーネント ... 154
4.8.3 ページの更新 ... 159
4.9 位置情報の共有 ... 162
4.9.1 位置情報の読み込みと共有 ... 164
4.9.2 Try/ Catch/ Throw 例外の使用 ... 166
4.10 スクロールすることのできるテーブル ... 168
4.10.1 スクリーン全体の高さで表示されるテーブル ... 170
4.10.2 特定の高さを持つテーブル ... 172

5 ユーザー インターフェイス 176

5.1 メイン ウィンドウ ... 178
5.1.1 ページ デザイン ... 179
5.1.2 DB クエリ ... 181
5.2 ページ ペイン ... 182
5.3 ファイル ペイン ... 184
5.4 コントロール ペイン ... 186
5.5 ページソース ペイン ... 187
5.6 概要ペイン ... 189
5.7 スタイル ＆プロパティペイン ... 190
5.8 メッセージ ペイン ... 192

6 プロジェクト 196

6.1 クライアントとサーバー間のインタラクション ... 197
6.2 プロジェクトファイルの場所 ... 198
6.3 プロジェクトのデプロイ ... 200
6.4 プロジェクトプロパティ ... 204
6.5 ローカライズ ... 211
6.6 名前空間 ... 212
6.7 グローバル リソース ... 213
6.8 パフォーマンス ... 214
6.8.1 デザイン ファイルに XML を埋め込む ... 215
6.8.2 XQuery 3.1 を使用したデータ クエリ ... 216
6.8.3 サーバー上のデータ ストレージ ... 217
6.8.4 クライアント側の永続データ ストレージ ... 219

7 データソース 222

7.1 ページデータソースの追加 ... 224
7.2 データソースの種類 ... 225
7.3 ページソースオプション ... 231
7.4 HTTP/ FTP、REST および SOAP リクエスト ... 233
7.4.1 HTTP/ FTP リクエスト設定 ... 234
REST リクエスト設定 ... 236
7.4.2 SOAP リクエスト設定 ... 243
7.5 ルートノード ... 245
7.6 ページソース ツリー ... 248
7.6.1 ツリー構造 ... 251
7.6.2 ツリーデータ ... 253
7.7 プロジェクト内の名前空間 ... 258
7.8 キャッシュ ... 259
7.9 コンテキストメニュー ... 262

8 ページ 278

8.1 プロジェクトのページ ... 279
8.2 ページのデータソース ... 281
8.3 ページ プロパティ ... 283
8.4 ページイベント ... 286
8.4.1 ページのロード時 ... 288
8.4.2 ページの更新時 ... 289
8.4.3 戻るボタンのクリック時 ... 291
8.4.4 送信ボタンのクリック時 ... 292
8.4.5 サーバーへの接続エラー時 ... 293
8.4.6 埋め込まれたメッセージ ... 296

9 コントロールとコントロールイベント 300

9.1 コントロール ... 301
9.1.1 アサーション メッセージ ... 306
9.1.2 ボタン ... 312
9.1.3 チャート ... 322
9.1.4 チェック ボックス ... 328
9.1.5 コンボ ボックス ... 337
9.1.6 日付 ... 347

9.1.7 日時 (iOS) ... 356
9.1.8 編集フィールド ... 363
9.1.9 水平線 ... 372
9.1.10 水平線スライダー ... 375
9.1.11 イメージ ... 378
9.1.12 ラベル ... 389
9.1.13 ラジオボタン ... 398
9.1.14 リッチテキスト ... 408
9.1.15 署名フィールド ... 413
9.1.16 スペース ... 421
9.1.17 切り替え ... 423
9.1.18 テーブル ... 431
9.1.19 時刻 ... 443
9.1.20 垂直線 ... 451
9.1.21 ビデオ ... 454
9.2 コントロール イベント ... 462

10 アクション 466

10.1 ユーザー インタラクション ... 469
10.1.1 メッセージ ボックス ... 470
10.1.2 電子メールの送信 ... 472
10.1.3 共有 ... 477
10.1.4 SMS を送信する ... 480
10.1.5 通話を発信する ... 481
10.1.6 URL/ ファイル を開く ... 482
10.1.7 印刷する ... 485
10.1.8 MapForce 転送 ... 490
10.1.9 待機カーソル ... 493
10.1.10 連絡先の読み取り ... 494
10.1.11 カレンダーへのアクセス ... 495
10.1.12 ユーザーによる日付の選択 ... 499
10.1.13 ユーザーによる時刻の選択 ... 500
10.2 イメージ、オーディオ、ビデオ ... 501
10.2.1 ユーザーのイメージ選択 ... 502
10.2.2 イメージのロード/保存 ... 503
10.2.3 イメージの表示 ... 508
10.2.4 ユーザーによるバーコードのスキャン ... 509
10.2.5 オーディオ ... 511
10.2.6 オーディオの録音 ... 515
10.2.7 読み上げ ... 518

10.2.8 ビデオ ... 519
10.3 位置情報サービス ... 520
10.3.1 位置情報トラッキング開始 /停止 ... 521
10.3.2 位置情報データの読み込み ... 522
10.3.3 位置情報の表示 ... 526
10.4 NFC ... 529
10.4.1 NFC の開始 /停止 ... 530
10.4.2 NFC プッシュ ... 532
10.5 プッシュ通知 ... 536
10.5.1 プッシュ通知の送信 ... 537
10.5.2 外部プッシュ通知キーの登録（登録の解除） ... 541
10.5.3 外部プッシュ通知トピックの登録（登録の解除） ... 542
10.6 ページ ... 543
10.6.1 ページへ移動 ... 544
10.6.2 サブページへ移動 ... 545
10.6.3 サブページを閉じる ... 549
10.6.4 スクロール ... 550
10.6.5 キーボードを隠す ... 551
10.6.6 表示の更新 ... 552
10.6.7 ページタイマーの再開 /停止 ... 553
10.7 ページ ソース ... 554
10.7.1 再ロード ... 556
10.7.2 ファイルをロード /保存 ... 557
10.7.3 バイナリファイルのロード /保存 ... 563
10.7.4 HTTP/ FTP をロード /保存 ... 569
10.7.5 文字列をロード /保存 ... 571
10.7.6 SOAP からロード ... 573
10.7.7 保存 ... 575
10.7.8 ファイル /フォルダーの削除 ... 577
10.7.9 リセット ... 578
10.7.10 SOAP リクエストの実行 ... 579
10.7.11 REST リクエストの実行 ... 581
10.7.12 ファイル情報の取得 ... 582
10.7.13 フォルダーの読み取り ... 584
10.7.14 ページソースの保存 /復元 ... 586
10.8 データベース ... 587
10.8.1 DB トランザクションの開始 ... 588
10.8.2 DB 実行 ... 590
10.8.3 DB バルク挿入 ... 594
10.8.4 DB コミット トランザクション ... 596

10.8.5 DB ロールバック トランザクション ... 597
10.9 その他 ... 598
10.9.1 コメント ... 599
10.9.2 実行 ... 600
10.9.3 アクションの実行のキャンセル ... 601
10.9.4 ユーザーキャンセルの振る舞い ... 602
10.9.5 ソリューションの実行 ... 604
10.9.6 言語の設定 ... 605
10.9.7 埋め込まれたメッセージをバックする ... 606
10.10 データの更新 ... 607
10.10.1 ノードの更新 ... 608
10.10.2 ノードの挿入 ... 612
10.10.3 ノードの追加 ... 615
10.10.4 ノードの削除 ... 619
10.10.5 ノードの置換 ... 620
10.11 If、Loop、Let、トライ /キャッチ、スロー ... 622
10.11.1 If- Then ... 623
10.11.2 If- Then- Else ... 624
10.11.3 Loop ... 625
10.11.4 Let ... 627
10.11.5 スロー ... 629
10.11.6 例外の トライ /キャッチ ... 630
10.11.7 サーバー接続エラーの トライ /キャッチ ... 632
10.11.8 リターン ... 633
10.12 アクショングループ ... 634
10.12.1 アクショングループの管理 ... 635
10.12.2 アクションを再使用するためのアクショングループ ... 637
10.12.3 パラメーターを持つアクショングループ ... 639
10.12.4 アクショングループパラメーターを持つアクショングループ ... 641
10.12.5 変数とアクショングループの結果 ... 644

11 デザインオブジェクト/機能 648

11.1 テーブル ... 649
11.1.1 静的なテーブル ... 651
11.1.2 繰り返しテーブル ... 653
11.1.3 動的な行 ... 657
11.1.4 動的な列 ... 661
11.1.5 テーブルプロパティ ... 665
11.1.6 テーブルのコンテキストメニュー ... 671
11.2 イメージ ... 674

11.2.1 イメージ ソース ... 675
11.2.2 Base64 エンコードイメージ ... 677
11.2.3 エクスチェンジャブル イメージ ファイル フォーマット(Exif) ... 679
11.2.4 エンドユーザーにより選択されたイメージ ... 685
11.2.5 イメージの変換 ... 690
11.2.6 データベースのイメージ ... 691
11.3 オーディオ、ビデオ ... 692
11.3.1 オーディオの再生 ... 693
11.3.2 オーディオの録音 ... 696
11.3.3 読み上げ ... 698
11.3.4 ビデオの再生 ... 699
11.3.5 オーディオ /ビデオフォーマット ... 701
11.4 NFC ... 703
11.4.1 NFC タグの検出と読み取り ... 705
11.4.2 他のデバイスにデータをプッシュする ... 706
11.4.3 NFC に関連したイベント ... 707
11.4.4 NFC のためのデザインコンポーネント ... 709
11.5 プッシュ通知 ... 711
11.5.1 ソリューションの送信 ... 712
11.5.2 ソリューションの受信 ... 714
11.5.3 AppStore App 内のプッシュ通知 ... 716
11.5.4 プッシュ通知のシミュレーション ... 719
11.6 チャート ... 721
11.6.1 チャートの作成と構成 ... 722
11.6.2 チャートデータの選択 ... 724
チャートデータの選択: シンプル ... 727
チャートデータの選択: フレキシブル ... 731
オーバーレイチャート ... 738
11.6.3 チャートの設定と外観 ... 741
チャートの基本設定 ... 742
チャートの詳細設定 ... 748
動的な XPath 設定 ... 765
11.7 スタイルシート ... 767
11.7.1 スタイルシートの型とスコープ ... 769
11.7.2 スタイルシート内の優先順位 ... 770
11.7.3 スタイルシート全体の優先順位 ... 774
11.7.4 ユーザーにより作成されたスタイルシートの適用 ... 776
11.7.5 スタイルシートプロパティ ... 777
11.8 リッチテキスト ... 778
11.8.1 リッチテキストコントロール ... 779

11.8.2 リッチテキストスタイルシート: セットアップ ... 780
11.8.3 リッチテキストスタイルシート: スタイル ... 782
11.8.4 リッチテキストコンテンツの編集 ... 786
11.9 ソリューションへのハイパーリンク ... 789

12 XPath/ XQuery: 式、関数、変数 794

12.1 XPath/ XQuery 式と関数 ... 795
12.1.1 XPath/ XQuery 条件式の編集ダイアログ ... 796
12.1.2 MobileTogether 拡張関数 ... 798
12.1.3 ユーザー定義 XPath/ XQuery 関数 ... 814
12.1.4 XPath/ XQuery についてよくある質問 ... 816
12.1.5 グローバル変数 ... 817
静的なグローバル変数 ... 819
動的なローカル変数 ... 822
ユーザー変数 ... 825

13 データベース 828

13.1 データソースとしての DB ... 830
13.2 データベースへの接続 ... 834
13.2.1 データベース接続ウィザードの開始 ... 836
13.2.2 データベースドライバー概要 ... 837
13.2.3 ADO 接続の設定 ... 840
既存の Microsoft Access データベースへの接続 ... 842
新規の Microsoft Access データベースの作成 ... 843
SQL Server データリンクプロパティの設定 ... 844
Microsoft Access データリンクプロパティの設定 ... 845
13.2.4 ADO.NET 接続のセットアップ ... 847
Visual Studio 内で接続文字列を作成する ... 849
サンプル ADO.NET 接続文字列 ... 851
ADO.NET サポートに関するメモ ... 853
13.2.5 ODBC 接続の設定 ... 854
13.2.6 JDBC 接続の設定 ... 856
CLASSPATH の構成 ... 858
13.2.7 使用可能な ODBC ドライバーの表示 ... 860
13.2.8 PostgreSQL 接続のセットアップ ... 861
13.2.9 SQLite 接続の設定 ... 863
新規の SQLite データベースの作成 ... 864
外部キーの制約 ... 865
13.2.10 グローバルリソースからの接続の使用 ... 866

13.2.11 データベース接続のサンプル ... 867
Firebird (ODBC) への接続 ... *868*
Firebird (JDBC) への接続 ... *871*
IBM DB2 (ODBC) への接続 ... *873*
IBM DB2 for i (ODBC) への接続 ... *878*
IBM Informix (JDBC) への接続 ... *881*
MariaDB (ODBC) への接続 ... *883*
Microsoft Access (ADO) への接続 ... *885*
Microsoft SQL Server (ADO) への接続 ... *887*
Microsoft SQL Server (ODBC) への接続 ... *890*
MySQL (ODBC) への接続 ... *892*
Oracle (ODBC) への接続 ... *895*
Oracle (JDBC) への接続 ... *900*
PostgreSQL (ODBC) への接続 ... *902*
Progress OpenEdge (ODBC) への接続 ... *904*
Progress OpenEdge (JDBC) への接続 ... *907*
Sybase (JDBC) への接続 ... *909*
Teradata (ODBC) への接続 ... *910*
Teradata (JDBC) への接続 ... *914*
13.3 Linux とmacOS でのデータベース接続 ... 916
13.3.1 Linux とmacOS 上での SQLite 接続 ... 917
13.3.2 Linux とmacOS 上での JDBC 接続 ... 918
13.3.3 OS X Yosemite 上での Oracle 接続 ... 919
13.4 データソースとして DB オブジェクトを選択する ... 920
13.5 DB データの編集 ... 922
13.6 DB にデータを保存する ... 926
13.7 DB 実行アクション ... 930
13.8 DB データの表示 ... 934
13.9 データベース クエリ ... 936
13.9.1 GUI 概要とツールバー ... 938
13.9.2 データソースへの接続 ... 940
13.9.3 ブラウザー ペイン ... 942
13.9.4 クエリペイン: 説明 ... 945
13.9.5 クエリペイン: 作業方法 ... 949
13.9.6 結果とメッセージ ペイン ... 950

14 Altova グローバルリソース 954

14.1 グローバル リソースの定義 ... 955
14.1.1 ファイル ... 957

14.1.2 フォルダー ... 961
14.1.3 データベース ... 963
14.2 グローバルリソースを使用する ... 966
14.2.1 ファイルとフォルダーの割り当て ... 967
14.2.2 データベースの割り当て ... 968
14.2.3 構成の切り替え ... 969

15 シミュレーション 972

15.1 MobileTogether Designer でのシミュレーション ... 973
15.2 サーバーでのシミュレーション ... 978
15.3 クライアントでのシミュレーション ... 984
15.4 位置情報の設定 ... 986
15.5 NFC サンプルファイル ... 990
15.6 プッシュ通知のシミュレーション ... 992
15.7 連絡先のサンプルファイル ... 994
15.8 カレンダーのサンプルファイル ... 995
15.9 サービストリガーのシミュレーション ... 996
15.10 メッセージ ペイン ... 998

16 埋め込まれた Web ページソリューション 1000

16.1 Web ページにソリューションを埋め込む ... 1003
16.2 Web ページとサーバー間の通信 ... 1006
16.2.1 ポスティング: Web ページからサーバーへ ... 1007
16.2.2 リッスン: ソリューションから Web ページへ ... 1008
16.3 認証 ... 1010
16.3.1 匿名ログイン ... 1011
16.3.2 ユーザーログイン ... 1012
16.3.3 JWT 認証 ... 1013
対称キー: 共有された秘密 ... 1015
非対称キー: 公開キー ... 1017
16.4 サンプル ... 1018
16.4.1 ソリューションの埋め込み ... 1019
16.4.2 JSON データの送信 /受信 ... 1020
16.4.3 XML データの送信 /受信 ... 1027
16.4.4 JSON ページソースの事前設定 ... 1034
16.4.5 JWT 認証 ... 1039

17 自動化されたテスト 1044

17.1 テストケースの記録 ... 1046
17.2 テストケースの再生 ... 1047
17.3 テストケースとテストランの管理 ... 1050
17.4 サーバーへテストケースをデプロイする ... 1054
17.5 テストの実行の比較 ... 1058

18 AppStore App 1062

18.1 プロジェクトからプログラムコードを生成する ... 1063
18.2 サーバーにワークフローをデプロイする ... 1075
18.3 プログラムコードのコンパイル ... 1077
18.3.1 Android ... 1078
18.3.2 iOS ... 1080
18.3.3 Windows アプリ ... 1082
18.4 SPL テンプレート ... 1084
18.4.1 SPL 構文 ... 1086
18.4.2 文字列のメカニズム ... 1088
18.4.3 $Options のプロパティ ... 1090
18.4.4 $Application のプロパティ ... 1092
18.4.5 その他のオブジェクト ... 1093

19 サーバーサービス 1096

19.1 サービスの作成 ... 1097
19.2 サービスのデプロイ ... 1099
19.3 サービスの実行 ... 1100

20 メニュー コマンド 1102

20.1 ファイル ... 1103
20.1.1 新規作成 ... 1104
20.1.2 新規のサービス ... 1105
20.1.3 開く ... 1106
20.1.4 再ロード ... 1110
20.1.5 閉じる、すべて閉じる、非アクティブをすべて閉じる ... 1111
20.1.6 保存、名前を付けて保存、コピーに名前を付けて保存、すべて保存 ... 1112
20.1.7 MobileTogether Server へデプロイ ... 1117

20.1.8 MobileTogether Server から開く ... 1119
20.1.9 MobileTogether Server から削除する ... 1120
20.1.10 モバイルアプリのためのコード生成 ... 1122
20.1.11 メールで送信 ... 1123
20.1.12 印刷 ... 1124
20.1.13 印刷プレビュー、印刷設定 ... 1125
20.1.14 最近使用したファイル、閉じる ... 1127
20.2 編集 ... 1128
20.2.1 元に戻す、やり直し ... 1129
20.2.2 切り取り、コピー、貼り付け、削除 ... 1130
20.2.3 すべて選択 ... 1132
20.3 プロジェクト ... 1133
20.3.1 検証 ... 1134
20.3.2 ページソース構造の再ロード ... 1135
20.3.3 ワークフローのシミュレーション ... 1136
20.3.4 クライアントでの試行 ... 1137
20.3.5 ワークフローのシミュレーションにサーバーを使用 ... 1138
20.3.6 新しいテストケースの記録 ... 1139
20.3.7 テストケースの再生 ... 1140
20.3.8 クライアント上でのテストケースのトライアルラン ... 1141
20.3.9 テストケースとテストランの管理 ... 1142
20.3.10 グローバル変数 ... 1143
20.3.11 すべてのグローバル変数の使用方法のリスト ... 1145
20.3.12 すべてのページソース変数の使用方法のリスト ... 1146
20.3.13 XPath/ XQuery 関数 ... 1147
20.3.14 すべてのユーザー定義 XPath/ XQuery 関数のリスト ... 1149
20.3.15 アクション グループ ... 1150
20.3.16 すべてのアクション グループの使用法のリスト ... 1152
20.3.17 スタイルシート ... 1153
20.3.18 スタイルシートのすべての使用方法をリストする ... 1155
20.3.19 リッチテキストスタイルシート ... 1156
20.3.20 キャッシュの概要 ... 1157
20.3.21 ローカライズ ... 1158
20.3.22 シミュレーションする言語 ... 1164
20.3.23 すべての外部データ参照のリスト ... 1165
20.3.24 使用されていない関数、ユーザー変数およびアクション グループのリスト ... 1166
20.3.25 OAuth 設定の維持 ... 1167
20.3.26 OAuth 設定のインポート ... 1169
20.3.27 iOS プッシュ通知ボタンセット ... 1170
20.4 ページ ... 1171
20.4.1 ページ アクション ... 1172

20.4.2 アクションの概要 ... 1173
20.4.3 コントロールへジャンプ ... 1174
20.4.4 タブの順序の表示 /定義 ... 1175
20.4.5 テキストサイズの自動調整グループのリスト ... 1177
20.5 テーブル ... 1178
20.5.1 テーブルの挿入 /削除 ... 1179
20.5.2 行 /列の挿入 /追加 /削除 ... 1180
20.5.3 セルのジョイン /分割 ... 1181
20.5.4 追加 /削除ボタンの表示 ... 1182
20.5.5 ヘッダー /フッター、先行 /末尾列の追加 ... 1183
20.5.6 ヘッダー /フッター、先行 /末尾列を削除する ... 1185
20.5.7 行を繰り返し /静的な行に変換する ... 1187
20.5.8 繰り返し /非繰り返しテーブルへの変換 ... 1188
20.5.9 列を繰り返し /非繰り返し列へ変換する ... 1189
20.6 表示 ... 1190
20.6.1 ステータス バー とペイン ... 1191
20.6.2 ズームのレベル ... 1192
20.7 ツール ... 1193
20.7.1 グローバル リソース ... 1194
20.7.2 アクティブな構成 ... 1195
20.7.3 ユーザー定義ツール ... 1196
20.7.4 カスタマイズ ... 1197
コマンド ... 1198
ツールバー ... 1200
ツール ... 1202
キーボード ... 1204
メニュー ... 1206
オプション ... 1208
20.7.5 ツールバーとウィンドウの復元 ... 1209
20.7.6 オプション ... 1210
20.8 ウィンドウ ... 1218
20.8.1 重ねて表示と並べて表示 ... 1219
20.8.2 閉じる、全て閉じる、非アクティブを全て閉じる ... 1220
20.8.3 現在開かれているウィンドウのリスト ... 1221
20.9 ヘルプ ... 1222
20.9.1 目次、インデックス、検索 ... 1223
20.9.2 アクティベーション、注文フォーム、登録、最新情報のチェック ... 1224
20.9.3 他のコマンド ... 1227

21 よくある質問 1230

22 参照 1232

22.1 XSLT と XPath/ XQuery 関数 1233
22.1.1 Altova 拡張関数 1235
XPath/ XQuery 関数: 日付と時刻 1236
XPath/ XQuery 関数: 位置情報 1249
XPath/ XQuery 関数: イメージに関連 1257
XPath/ XQuery 関数: 数値 1266
XPath/ XQuery 関数: シーケンス 1269
XPath/ XQuery 関数: 文字列 1275
XPath/ XQuery 関数: その他 1281
22.2 ライセンス情報 1282
22.2.1 電子的なソフトウェアの配布 1283
22.2.2 ソフトウェアのアクティベーションとライセンスの計測 1284
22.2.3 知的所有権 1285
22.2.4 Altova MobileTogether Designer エンドユーザー使用許諾契約書 1286

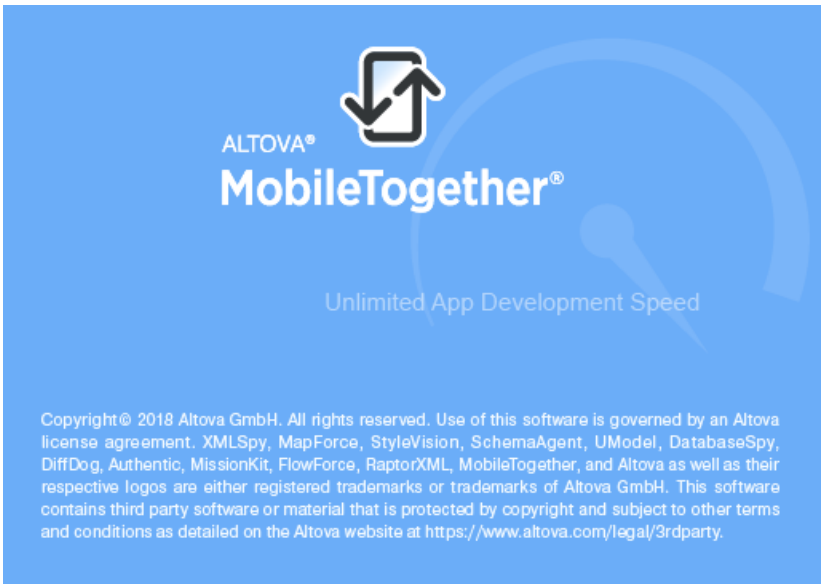
インデックス 1293

チャプター 1

Altova MobileTogether Designer

1 Altova MobileTogether Designer

MobileTogether Designer は希望通りモバイル ソリューションを作成する、Windows マシンのための無料製品です。理解しやすいアプローチにより、ドラッグ アンド ドロップ機能を使用してエレガントなモバイル ソリューションを作成することができます。MobileTogether Designer には、完全なモバイル シミュレーターが搭載されており、モバイル ソリューションをデザイナーで即時実行することができます。また、モバイル ソリューションを直接モバイルデバイスで実行しプロジェクトをリアルタイムで見ることができます。



MobileTogether Designer ビデオデモ

Altova Web サイトでMobileTogether Designer を使用して多種のMobileTogether ソリューションの作成方法の**ビデオデモ**を観ることができます。これらのビデオはMobileTogether Designer のパワフルな機能性への簡単手短な紹介です。

このドキュメントについて

このドキュメントはMobileTogether Designer のユーザーマニュアルです。このドキュメントは以下のセクションにより整理されています:

- はじめに
- チュートリアル
- ユーザー インターフェイス
- プロジェクト
- ページ
- データソース
- コントロールとコントロールイベント
- アクション
- デザインオブジェクトと機能

- XPath/XQuery: 式、関数、変数
- データベース
- Altova グローバルリソース
- シミュレーション
- 埋め込まれたWeb ページソリューション
- 自動化されたテスト
- AppStore App
- サーバー サービス
- メニューコマンド
- よくある質問
- 付録

現在のバージョン: 4.1

最終更新日: 2018年 02月 26日

(C) 2018 Altova GmbH

チャプター 2

新機能

2 新機能

MobileTogether Designer **バージョン4.1** の新規機能は以下にリストされるとおりです。バージョン4 の前のリリースの新規の機能を含むリストが後にリストされています。

バージョン 4.1

MobileTogether Designer **バージョン4.1** の新機能とアップデートは以下の通りです。

- サーバーサービスはソリューション(`.mtd` ファイル) として **MobileTogether Server Advanced Edition** にデプロイされるMobileTogether Designer アクションのセットです。MobileTogether Server 条件のセットが満たされるとサービスはサーバー上で実行されます (これらのサーバー条件はMobileTogether Server Advanced Edition の管理者インターフェイス内で定義されます) 。
- MobileTogether Designer の **「ファイル」 「新規のサービス」** を使用して開くことのできるサーバーサービス デザイン内でサーバーサービスは定義されます。
- MobileTogether Designer 内でサーバーサービスを作成する方法は、サーバーサービスのセクションで説明されています。
- **`$MT_SERVICE`** ページソースは、サービスデザインが作成されると自動的に作成されます。これはサービス トリガーに関するランタイムのデータを保持しています。
- **`$MT_SERVICE`** ページソースはサービス トリガーに関するランタイムデータをシミュレートするために手動でファイルすることができます。

リッチテキスト

- 新規の リッチテキスト コントロール により、ページソースからのテキストを(すべてのクライアント上での)書式設定と(Windows とWeb クライアント上での)編集と共に表示することができます。書式設定はXML ページソース内でのスタイルマークアップを基に、または、追加することができます。どちらの場合でも、リッチテキスト スタイルシート内でルールは指定されています。
- 各プロジェクト(デザイン) のために、リッチテキスト スタイルシート ダイアログを使用して複数のリッチテキスト スタイルシートを定義することができます。リッチテキスト コントロールにこれらのスタイルシートを割り当てることができ、コントロール内に表示されるテキストは、選択されたスタイルシートのルールに従い書式設定することができます。
- この機能に関する詳細に関しては、リッチテキストのセクションを参照してください。

アクション

- サブページへ移動 アクションは、サーブページをモーダルダイアログとして開くオプションを持つように拡張されました(具体的には、現在のページの上に個別のウィンドウを表示します)。これは、サブページをもつ現在のページを置き換える代替の表示です。
- ページソース保存 復元 アクションにより、仮にページソースを保存することができ、条件が満たされるか否かにより更に変更を受け入れるか破棄するかを決定することができます。
- カレンダーへのアクセス アクションにより、デバイスのカレンダーに関する情報および **`$MT_CALENDAR`** ページソースへのカレンダーイベントに関する情報を保存することができます。デバイス上のカレンダーにイベントを書き込むことができます。シミュレーションに関しては、Microsoft Outlook のカレンダー またはXML ファイルを使用することができます。
- ノードの置換 アクションは、ページソースのノードからノードを削除するし新規のノードを追加するメカニズムを提供します。

コントロール

- (コントロール **複数の選択** プロパティを使用して) コンボボックス コントロールは ユーザーが複数のオプションを選択できるように拡張されました。

- コントロールには、コントロールの幅にテキストサイズが調整される **テキストサイズ** プロパティと **テキストサイズの自動調整** プロパティが追加されました。コントロールは、グループに割り当てられ、すべてのコントロールが自動的に統一された、適切なサイズに調整されます。自動調整プロパティが設定されているページのすべてのコントロールは、メニューコマンド **テキストサイズの自動調整グループ** を使用してメッセージペイン内にリストすることができます。
- (Web とWindows クライアント上で) デザイン内でコントロールの一部は、エンドユーザーが **Tab** キーを繰り返しクリックすると、タブの順序のシーケンスに割り当てられます。ソリューションのフォーカスは指定されたタブ順序内のコントロールに移動されます。タブ順序の全体は、**「ページ | タブの順序の表示 /定義」** メニューコマンドにより設定されます。個別のコントロールのシーケンス内の位置は、コントロールの **タブの順序** プロパティ内で設定することができます。タブの順序シーケンス内で位置を割り当てることのできるコントロールは以下のとおりです: ボタン、チェックボックス、コンボボックス、日付、編集フィールド、ラジオボタン、切り替えコントロール、時刻コントロール。
- (Web およびWindows クライアント上では) **クリック時** イベントを持つコントロール (ボタン、チャート、イメージ、および ラベル) はクライアントの **入力** または **エスケープ** キーを使用してクリックイベントをトリガーすることができます。設定は、コントロールの 入力 /エスケープ プロパティを使用して、または、コントロールの **クリック時** イベント アクションを定義するダイアログ内でおこなうことができます。対応するコントロールの各セクションを参照してください。

XPath 拡張関数

- 2つの新規のMobileTogether XPath 拡張関数: (i) **`mt-client-ip-address`** (デバイスのIP アドレスを取得するため); (ii) **`mt-image-width-and-height`** (与えられたBase64-インコードイメージのディメンションを取得するため)。
- 新規のAltova XPath 拡張子 **`generate-guid`** は、ID として使用することのできる一意のGUID 文字列を生成します。

その他

- ライトのテーマの強制: プロジェクトプロパティペイン内で、プロジェクトのページがライトの背景(薄い背景に濃いテキスト)を持つかを指定することができます。デフォルトの値の`false` は、クライアント固有のテーマを指定します。
- 連絡先マネジャーとMicrosoft Outlook のカレンダーは、連絡先の読み取りとカレンダーへのアクセス アクションのシミュレーションに使用することができます。オプションダイアログ内で対応するアイテムを選択してこれをおこなうことができます。

バージョン 4.0

MobileTogether Designer **バージョン4.0** の新機能とアップデートは以下の通りです。

プッシュ通知

- プッシュ通知 (PN) はMobileTogether ソリューションがインストールされているモバイルデバイスから送信されるテキストメッセージです。プッシュ通知が受信されると、受け取るソリューション内でこのアクションのセットがトリガーされます。プッシュ通知 機能の概要に関しては、次を参照してください: プッシュ通知.
- プッシュ通知の送信 アクションはソリューションの送信時に指定されています。送信されるプッシュ通知の多種のパラメーターを定義します。
- ソリューションを受け取る場合、**プッシュ通知の受信時** イベントのためのこのアクションはプッシュ通知が受信されると実行されます。
- テキストメッセージ以外にも、プッシュ通知はペイロードを実行することができます。ペイロードは自動的に受け取るソリューション **`$MT_PUSHNOTIFICATION`** ページソースに転送されます。
- プッシュ通知にはボタンが含まれています。プッシュ通知ボタンは送信するソリューションのプッシュ通知の送信 アクション内で指定されています。プッシュ通知の送信 アクション内で直接非 iOS デバイスのためのボタンを定義することができますが、iOS デバイスでは、ボタンは **「プロジェクト | iOS プッシュ通知 ボタンセット** コマンドによりソリューション内で定義することができます。
- 外部 プッシュ通知キー はモバイルデバイスを識別するために使用されるテキスト文字列です。外部プッシュ通知キーの登録 アクションは、モバイルデバイスを指定する文字列と関連付けます。外部 プッシュ通知キーはプッシュ通知を受信するモバイルデバイスのセットを識別するために使用されます。逆のアクション外部プッシュ通知キーの登録の

解除 も使用することができます。

- プッシュ通知 トピックはトピックの名前を与えるテキスト文字列です。プッシュ通知 トピックの登録 アクションはモバイルデバイスを1つまたは複数のプッシュ通知に関連付けます。プッシュ通知がプッシュ通知 トピックに送信されると、トピックに関連する全てのデバイスはそのプッシュ通知を受信します。逆のアクション プッシュ通知 トピックの登録の解除 を使用することもできます。
- プッシュ通知が異なるソリューションの受け取り先に送信される場合、ソリューションの受け取りのシミュレーションを成功させるために、プッシュ通知をシミュレートする必要があります。シミュレーター内でプッシュ通知の受信をシミュレートすることができます。詳細は、セクション プッシュ通知のシミュレーション を参照してください。
- プッシュ通知を使用するMobileTogether ソリューションをAppStore App にコンパイルすることができます。AppStore Apps をコンパイルするには追加ステップが必要になります。AppStore Apps 内のプッシュ通知 内でこれらのステップは説明されています。

埋め込まれた Web ページ ソリューション

- 新規の 埋め込まれたWeb ページ ソリューション 機能はIFrames を使用してWeb ページ内にソリューションを埋め込むことができます。Web ページと埋め込まれたソリューション間でデータを交換することができ、ソリューション自身はMobileTogether Server とデータをやり取りし、データをWeb ページに送り返します。JSON Web Tokens (JWT) を使用した認証 により埋め込まれたWeb ページ ソリューションが既存のシステムに統合されます。
- **埋め込まれたメッセージ** イベントは、サーバー上のソリューションのワークフローが埋め込まれたソリューションからメッセージを受け取るとトリガーされます。
- **`$MT_EMBEDDEDMESSAGE`** JSON ページソース(構造とデータ) は **埋め込まれたメッセージ** イベントがトリガーされると作成されます。
- 文字列からコード アクションは文字列を解析し、(JSON/XML) ページソースを生成します。
- 文字列に保存 アクションは選択された(JSON/XML) ページソースをシリアル化し、シリアル化された文字列を指定された場所に保存します。
- 埋め込まれたメッセージのバック アクションは、シリアル化されたJSON 文字列を現在のソリューションをロードしたIFrame へのメッセージイベントとして送信します。

新規のアクション

- MapForce 転送 アクションはMapForce Server Execution ファイル(MFX ファイル) をMapForce Server に処理のために提供します。入力データ構造のセットは、このように、データ構造 (MapForce Server の出力)の新規のセットに転送されます。変更することのできないレガシーデータ構造？ または、他のデータ構造をMobileTogether デザイン内で使用することができるようになります。
- フォルダーの読み取り アクションは、指定されたフォルダーのコンテンツを読み取り、各フォルダーアイテムに関するメタデータを **`$MT_FILEINFO`** ページソースの個別のノードにパスします。
- 言語の設定 アクションによりユーザーはソリューションの言語を変更することができます。これにより、ソリューションは特定のイベントがトリガーされると、代替の言語で表示されます。
- 文字列からコード アクションは文字列を解析し、(JSON/XML) ページソースを生成します。
- 文字列に保存 アクションは、選択された(JSON/XML) ページソースをシリアル化します。シリアル化された文字列を指定された場所に保存します。
- 埋め込まれたメッセージをバック アクションはシリアル化されたJSON 文字列をメッセージイベントとして、現在のソリューションをロードしたメッセージイベントとしてIFrame に送信します。
- プッシュ通知の送信 アクションは、送信されるプッシュ通知の多種のパラメーターを定義します。
- 外部プッシュ通知キーの登録（登録の解除） アクションは、文字列をモバイルデバイス上のソリューションの外部プッシュ通知キーとしてテキスト文字列を登録します。 詳細に関しては、プッシュ通知を参照してください。
- 外部プッシュ通知キーの登録（登録の解除） アクションは、1つまたは複数のトピックのプッシュ通知を受け取るためにデバイスを登録します。詳細に関しては、プッシュ通知を参照してください。

その他

- MobileTogether Server インストールにはサーバー上の統計にアクセスし表示する既にデプロイ済みのパワフルな

ソリューションが含まれています。例えば、特定のソリューションにアクセスしているデバイスの数、とデバイスの種類などが表示されます。**Statistics** ソリューションに関する詳細は、MobileTogether Server ドキュメントを参照してください。

- カスタマイズダイアログのツールタブ内でユーザー定義ツールを作成することができます。**「ツール」 ユーザー定義ツール」** メニュー内のコマンドからアクセスすることができます。
- テーブルに関連したコマンドを提供する新規のテーブルメニューは素早くデザインし、テーブル構造を編集する手助けを行います。
- 新規コマンド **スタイルシートのすべての使用方法をリストする** は、プロジェクト内で定義されている全てのスタイルシートを表示し、これらの(使用されていないスタイルシートを含む)スタイルシートを使用するページ、テーブル、および、コントロールインスタンスを表示します。使用されていないスタイルシートのリストは **使用されていない関数、ユーザー関数、スタイルシートとアクショングループをリスト** コマンドがクリックされると表示されます。
- ビューポートよりも広範囲の水平スクロールテーブルを左右にスクロールして確認することができます。
- シミュレーターメニューは次のモバイルデバイス機能をシミュレートするオプションを提供します: (i) カメラアプリ、(ii) ギャラリー、(iii) マイクロフォン、(iv) NFC、(v) GPS ロケーション、(vi) 連絡先、(vii) 通話サービス、(viii) SMS サービス。これらのオプションにより、デバイス上で使用することのできるサービスを必要とするデザインシナリオをテストすることができます。
- 抑制、または、必要に応じて有効化が可能な特定のイベントの特定のアクションに関連する(メッセージペインに表示される)メッセージのログ。
- ページソースデータはソリューションがページを移動すると自動的にリセットされます。データソースのコンテキストメニュー内で使用することのできる **データのリセット** コマンドを使用して行います。
- (全ての型の) ファイルを保存する場合、任意で、デフォルトのファイル拡張子を指定することができます。ファイル名と共に拡張子が指定されていない場合、この拡張子が使用されます。
- シミュレーション中に、ソースノードのXPath ロケーター式をクリップボードにコピーすることができます。

2.1 バージョン 3

バージョン**3** リリースの新機能のリストは以下のとおりです。

バージョン 3.2

MobileTogether Designer **バージョン3.2** の新機能とアップデートは以下の通りです。

近距離無線通信 (NFC)

- 新規のNFC 機能 は、NFC を使用してメッセージを受信し送信することができます。また、Android デバイスでは、Android ビームを使用してファイルを送信することができます。この機能を実装したデザインコンポーネントの概要に関しては、NFC のためのデザインコンポーネントを参照してください。
- アクションをトリガーするためのNFC に関連したイベント: **メッセージのプッシュの完了時** と**NFC タグの検出時**。
- 新規のMobileTogether 拡張関数 は、NFC が開始されたかをチェックします: `mt-nfc-started`。更に、テキストとBase64 をhexBinary に変換する関数 (NFC メッセージペイロードはhexBinary でエンコードされているため)。
- NFC サンプルファイル はNFC タグの検出するシミュレーションを可能にします。

テキストの読み上げ

- テキストの読み上げ アクション をベースにした、新規の テキストの読み上げ機能 により、テキスト文字列がスピーチに変換され、再生されます。
- テキストの読み上げ機能 に関連した、情報を提供する新規のMobileTogether 拡張関数 : `mt-text-to-speech-is-language-available` と`mt-text-to-speech-is-speaking`。

その他の新規のアクション

- 新規の連絡先の読み取り アクションにより、データソースツリー内のデバイスのアドレス帳の連絡先を保管することができます。
- 新規の ファイル情報の取得アクション により指定されたファイルのファイル(サイズ、作成日時、などの)情報をデータソースツリー内に保管することができます。
- 新規の待機カーソル により、プラットフォーム特有の待機カーソルをアクションの実行中に表示することができます。これは、完了に時間のかかるアクションに対して使用すると役にたちます。
- 新規の ユーザーによる日付選択 と ユーザーによる時刻選択 アクションによりエンドユーザーが日付と時刻を選択することができ、それぞれ、ページソースノードに保存されます。
- イメージの表示 アクションによりクライアントデバイス、または、データソースノード、または、イメージ/チャート/署名 コントロールからイメージを使用することができます。
- サーバーの接続エラーのトライ/キャッシュ アクションは特定のサーバートランザクション上の例外に対して使用することができます。接続エラーが発生した際に取られる適切なアクションを定義することができます。

既存のアクションの機能の強化

- 新規の コマンドは、デザイン内ので使用されているすべての アクション または アクショングループの使用を表示 します。
- オーディオ アクション ではクライアントデバイスで使用することのできる定義済みのサウンドの再生ができるようになりました。現在、16種類の定義済みのサウンドから選択することができます。
- (サーバーを使用して) 電子メールを送信アクション に返信の設定が含まれるようになりました。これにより「架空の」および「実際の送信者のアドレス」を用いてMobileTogether Server を使用して電子メールを選択することができます。
- リセット アクション は、(クライアント上の永続データのための) `$PERSISTENT` ツリー を含む全てのデータソースのリセットを有効化します。
- 位置情報の表示 アクションは、クライアントデバイスのマップアプリ上の位置情報のための住所を受け入れる事ができます。以前の機能は、緯度と経度の座標のみを使用していました。
- 前のリリースの一番下にスクロールするアクションはスクロール アクションに置き換えられています。新しい機能は、特

定のコントロール、または、指定できることのできるテーブルの上、または、下にスクロールすることができます。デザイン内で古い下にスクロールするアクションを使用し、新しいMobileTogether Designer のバージョンで開く場合は、アクションは、自動的に新しいアクションに置き換えられます。

- 新規のMobileTogether 拡張関数 は、位置情報のトラッキングの開始をチェックします: `mt-geolocation-started`.
- ページへ移動 とサブページへ移動 のターゲットページアクションは、XPath を使用して設定することができます。
- カスタムボタンを含むことのできるメッセージボックス を使用して、デバイスの **戻る** ボタンがタップされると、実行されるアクションを指定することができます。
- 前のリリースのトライ/キャッチ 式は、例外のトライ/キャッチに名前が変更されました。

テーブルの新機能

- 動的な列 をテーブルに挿入することができます。これは、デザイン内の列フィールドに対するインスタンスの数に基づいてテーブルの右側に列を動的に追加することができることを意味します。
- 動的な、ローカル変数、`MT_TableColumnContext` が追加されました。テーブルの生成中現在の列のコンテキストノードを与えます。使用方法に関しては、次を参照してください: 動的な列 。
- スクロールすることのできるテーブル内にロードすることのできる行数
- を **行グループチャンクサイズ** テーブルプロパティを使用して設定することができます。

コントロールの機能強化

- 有効化/無効化することのできるコントロールのために2つの新規のコントロールプロパティを使用することができます: テキストの色 (無効化済み) と背景色 (無効化済み)。これらにより、(有効化済み、または、無効化済み)状態により、コントロールのために異なる色を使用することができます。
- 追加されたボタンの外観 を使用することができます: インポートエクスポート、カレンダー、および時刻
- Base64 データとしてデザインファイル内に埋め込まれたイメージ は、簡単に再度埋め込むことができます。これは、バイナリからBase64 に変換し、デザイン内に保管することができます。これはイメージコントロールのコンテキストメニューから行う事ができます。この機能は、イメージが変更されると埋め込まれたイメージファイルの更新を簡単に行うことができます。

その他

- ページペイン内では、コンテキストメニュー コマンド **アクションの使用のリスト** を選択して、(デザイン内の)ページへの参照をチェックすることができます。
- XPath 式の編集ダイアログ からコピーされるテキストは、XPath としてスタイル & プロパティペイン内に貼り付けられます。
- アイテムの表示を管理するための スタイルシートダイアログ 内の追加ツールバーアイコンは以下のとおりです: (i) 全てのアイテムの展開、(ii) 全てのアイテムの折りたたみ、(iii) 空ではないアイテムの表示。

バージョン 3.0

MobileTogether Designer **バージョン3.0** の新機能とアップデートは以下の通りです。

- スタイルシート機能 により、プロジェクト、ページ、テーブルおよびコントロールレベルで適用することのできるグローバルなスタイルを定義することができます。これにより、プロジェクトのためのカスケードスタイルのレポジトリのステップを作成することができます。
- 印刷 アクションでは、Altova StyleVision Server を使用して、PDF、Word、およびRTF をXML データから生成することができます。
- URL を開くアクションは、機能が強化され、URL/ ファイルを開くアクションと改められました。以前は、このアクションは、クライアントデバイスのブラウザーでWeb ページを開くために使用されていました。このアクションを使用して、クライアントデバイス上のデフォルトのアプリケーションからそのファイルの型のファイルを開くことができるようになりました。
- ユーザーによりバーコードのスキャン アクションを使用して、クライアントのカメラのアプリケーションを開き、ユーザーがバーコードをスキャンできるようになりました。バーコードのデータはXML データツリーに入力され、更に処理されます。
- 2つの新規のプロパティにより、コントロールのレイアウトがより簡単になりました (i)(スタイル& プロパティ入力ヘルパー内で設定することのできるプロジェクトプロパティ内のプロジェクト設定で使用することができる) プロジェクト プロパティ

トップレベルの余白 を使用して、ページのためのマージンを設定します。(ii) テーブル プロパティ**テーブルパディング** により、iOS デバイス上のパディングをオン、または、オフに切り替えることができます。

- 自動化されたテスト 機能により2 件のテストケースを比較し、デザイン、ページソースデータ、およびソリューションの環境内の差異を検出できるようになりました。

2.2 バージョン 2

バージョン**2** リリースの新機能のリストは以下のとおりです。

バージョン 2.2

MobileTogether Designer **バージョン2.2** の新機能とアップデートは以下の通りです。

- エンドユーザーが選択することのできる、クライアントデバイス上の、クライアントファイルのロードおよび保存が以下のアクションで使用することができます: ファイルのロード/保存、イメージのロード/保存 、および バイナリのロード/保存。
- ビデオコントロール によりページでビデオを再生することができます。コントロールのプロパティとビデオに関連した MobileTogether 拡張関数 により再生とコントロールをカスタム化することができます。ビデオ機能の概要に関しては、オーディオ、ビデオのセクションを参照してください。
- ビデオアクション によりビデオを開始、一時停止、再開、停止、および検索することができます。特定の時間を定義されたセグメントも定義することができます。ビデオ機能の概要に関しては、オーディオ、ビデオのセクションを参照してください。
- オーディオアクション によりオーディオを開始、一時停止、再開、停止、および検索を5つのチャンネルから実行することができます。特定の時間を定義されたセグメントも定義することができます。オーディオ機能の概要に関しては、オーディオ、ビデオのセクションを参照してください。
- **`$MT_AudioChannel`** グローバル変数 には多種のアクションをトリガーするオーディオチャンネルがあります。
- オーディオの録音アクション によりクライアントデバイス上のファイルにオーディオを録音することができます。オーディオ機能の概要に関しては、オーディオ、ビデオのセクションを参照してください。
- オーディオとビデオファイルについての情報、および、オーディオとビデオに関連するアクションのための新規 MobileTogether 拡張関数 。
- オーディオの録音アクション によりオーディオクライアントデバイスのファイルに録音することができます。オーディオの機能の概要に関しては、オーディオ、ビデオを参照してください。
- オーディオとビデオファイルに関する情報を提供する新規 MobileTogether 拡張関数 と、オーディオおよびビデオに関連したアクション: `mt-audio-get-current-position`, `mt-audio-get-duration`, `mt-audio-is-playing`, `mt-audio-is-recording`, `mt-video-get-current-position`, `mt-video-get-duration`, `mt-video-height`, `mt-audio-is-playing`, と`mt-video-width`.
- 最後に使用されたクライアントファイルの情報を提供する新規 MobileTogether 拡張関数 : `mt-last-file-path`、`mt-extract-file-extension` および`mt-extract-file-name`。
- バイナリのロード/保存アクション により以下が可能になります: (i) Base64 エンコードXML コンテンツとしてすべての種類のバイナリファイルをソリューションにロードすることができます、また(ii) Base64 エンコードXML コンテンツをバイナリファイルとして保存することができます。
- 電子メールの送信 アクションにより、XML およびバイナリファイルに加え、テキストの添付を送信することができます。
- シミュレーター は、LAN 接続の有無をシミュレートできるように設定できます。接続の型の数 に追加され、現在使用できる接続の型は以下の通りです: モバイルネットワーク WiFi、および LAN。関連する MobileTogether 拡張関数 も搭載されています: (i) `mt-connected-via-lan`.
- テーブルのコンテキストメニュー を使用してテーブルヘッダーとフッターを動的テーブルに追加することができます。

バージョン 2.1

MobileTogether Designer **バージョン2.1** の新機能とアップデートは以下の通りです。

- XML と イメージ ファイルなどのデータファイルがクライアントデバイスからロードされ、クライアントデバイスに保存されます。
- 2つの新規コントロールが追加されました 垂直線 と水平スライダー。
- 次のアクションが追加されました
  - 共有

  - アクションのキャンセルの実行
  - ユーザーキャンセルの振る舞い
  - タイマーの再開/停止
  - ファイル/フォルダーの削除
  - DB バルク挿入
  - Let
  - トライ/キャッチ
  - スロー
  - リターン
- アクショングループの結果: アクショングループ内の リターン アクション は、Let アクション内で定義される変数の値として使用することのできるアクショングループの結果を生成します。
- アクショングループはパラメーターを取ることができます。また、アクショングループ自身はパラメーターの値として設定されることができます。
- サブページを閉じるアクション は、Let アクション内で定義される変数の値と使用できる値を返すように拡張されました。
- (地図上で)位置情報を表示 アクションは、2 箇所の場所のルートを表示するように強化されました。
- クライアントからの電子メールの送信 はHTML フォーマットであることができます。
- タイマーの正確性 は、ミリ秒の正確性まで向上されたページの更新イベントのために使用する事ができます。
- 次のプロパティのXPath 定義: (フィールドの編集コントロールの) `Keyboard`、`Horizontal Alignment` 、および`Vertical Alignment`。
- コントロール とテーブル列 の幅はピクセルで設定する事ができます。
- テーブル: テーブル全体またはテーブルの一部をスクロール可能にデザインすることができます。スクロールすることのできるテーブル は、スクリーン全体に表示されるように指定することができます。
- テーブル: スパンされた列と行のための個別の可視性の設定。
- テーブル: (セルに追加して)個別の行と列に背景色を割り当てることができます。
- テーブル: ネストされたテーブルは、水平方向の配置と垂直方向の配置 プロパティの値に割り当てられることができます。
- フィールドの編集コントロール のキーボード プロパティは、表示されるパスワードの値と共に強化されました。この結果、エンドユーザーが編集フィールドにパスワードを入力する場合、パスワードの表示と非表示を定義する事ができます。
- (ボタンの外観 プロパティで指定することのできる) ボタンコントロール には、透明なボタンを含む定義済みの外観が追加されました。
- 新しいMobileTogether 拡張関数: (i) `mt-connected-via-wifi`、(ii) `mt-control-width`、(iii) `mt-font-height`。**mt-font-height** 関数を使用する、XPath 式によりフォントサイズとピクセルの値は生成されます。
- DB に保存する場合、データが更新または挿入されるかにより、列を個別にフィルターする事ができます。
- ページタイトルバーの表示 ページプロパティはページのタイトルバーを表示または非表示にする事ができます。
- ユーザーにより生成されたAppStore App: アプリのUI 言語 は、英語、ドイツ語、フランス語、スペイン語、日本語から選択する事ができます。
- カスタムローカライズされた文字列の複製

バージョン 2.0

MobileTogether Designer **バージョン2.0** の新機能とアップデートは以下の通りです。

- デザイナーは、エンドユーザーがモバイルデバイスにダウンロードすることのできるMobileTogether カスタムアプリを作成することができます。これらのアプリはAppStore App と呼ばれます。AppStore App のセクションは、MobileTogether Designer からそのようなアプリのためのプログラムコードの生成方法を説明します。コードは、Android、iOS、Windows (タッチスクリーンが有効化されたデバイスとPC)、Windows Phone のために生成することができます。コードが生成された後、対応するAppStore App にコンパイルされることができます。
- モバイルデバイス上のソリューションを中断 (一時停止して最小化) することができます。新しいプロジェクトプロパティ

*他のソリューションに切り替え時*は、エンドユーザーが他のソリューションに切り替えると継続的にソリューションを中断するように設定することができます。エンドユーザーはMobileTogether Client の実行タブ内のアイコンをクリックしてソリューションを切り替えることができます。ソリューションをキャンセルまたは中断するには、ソリューションの実行アクションを利用して指定します。

- 署名フィールド コントロールは、エンドユーザーの署名をイメージとしてデータソースノードに保管することを有効化します。
- サーバー接続エラーが発生すると取るアクションを定義してテストすることができます。
- デザイン内定義されたアクションを列挙するシミュレーション機能が向上されました。例えば、サーバーの接続エラーは サーバーアクセスを回避するオプションでシミュレートすることができます。
- JSON データソースはページソースとして使用できます。
- REST リクエスト を使用して、ページデータにアクセスまたはページデータを保存することができます。このようなデータはページソース内で使用することができ、ページソースアクションを使用してアクセスまたは保存することができます。
- REST リクエストはOAuth 認証 をサポートします。各デザインには、ドキュメント内のすべての箇所で使用することのできる設定のプールがあります。設定は[OAuth 設定の維持] ダイアログで管理することができます。更に、設定は、他の開かれているMobileTogether Designer ドキュメントからアクティブなドキュメントにインポートすることができます。
- SOAP リクエストを使用してページデータにアクセスすることができます。このようなデータは、ページソース および ページソースアクションで使用することができます。
- 新規アクション: SOAP リクエストの実行 、REST リクエストの実行
- ページイベント サーバーの接続エラー が追加されました。
- ページソースのためのデータの保持 を使用することによりデータをクライアントまたはサーバーに保持する柔軟性が追加されました。
- 次の2 つの動的、ローカル変数 が追加されました **`MT_HTTPExecute_Result` および `MT_ServerConnectionErrorLocation`**.
- すべてのファイル、ディレクトリ および外部データソースのリストの作成コマンド をプロジェクト内で使用することができます。
- 繰り返しテーブル と動的テーブル のセルはページソールノードとXPath 式を使用して、関連付けることができます。この結果以前読み取り専用であったセルコンテンツを編集することができます。

2.3 バージョン 1

バージョン**1** リリースの新機能のリストは以下のとおりです。

バージョン 1.5

MobileTogether Designer **バージョン1.5** の新機能とアップデートは以下の通りです。

- 電子メールの送信 アクションによりソリューションの実行中に電子メールの送信が有効化されます。
- MobileTogether 拡張関数 **mt-email-attachment** により、電子メールの送信 アクションにより送信される電子メールのテキストとイメージの添付ファイルが作成されます。
- HTML であるリンクを電子メールの本文へ挿入することができます。 これらの リンクは、インターネットのページまたは MobileTogether ソリューションをターゲットにすることができます。
- MobileTogether ソリューションに移動するリンクはコントロールイベントとページイベントによりトリガーされます。 MobileTogether ソリューションをポイントするURL は、表示される特定のページコンテンツを許可するURL クエリ文字列を含むことができます。ソリューションへのハイパーリンクを参照してください。
- ターゲットのソリューションにURL クエリパラメーターをパスするハイパーリンク。 これらのパラメーターは、参照することが可能な場所から**$MT_InputParameters** グローバル変数に保管されることができます。
- リンクに関連した3 つのMobileTogether 拡張関数が追加されました **mt-run-solution-url**、**mt-run-solution-url-parameter**、および **mt-html-anchor**。
- パワフルな Loop アクションにより、ノードのセットが反復され、この結果、デザインの可能性とソリューション機能を与えることができます。
- 2 つの他のアクションが追加されました キーボードの非表示 および 表示の更新。
- 新しい ラジオボタン コントロールが導入されました。
- ソリューションが特定の言語に ローカライズ されている場合、ソリューションの文字列はモバイルデバイスの言語に自動的に表示されます。このバージョンでは、デフォルトとローカライズされた文字列は各言語のためにプロジェクト間および個別のXML ファイル間でエクスポート/インポート されることができます。この機能により、各翻訳家は独自にデフォルトの言語の文字列をターゲットの異なる言語に翻訳することができます。翻訳された各ファイルは個別にプロジェクトにインポートし直すことができます。
- [XPath/XQuery 式の編集] ダイアログ 内のXPath 式に**mt-load-string** 関数を入力する時、プロジェクト内で定義されたすべてカスタム文字列のがポップアップで表示されます。現在 MobileTogether Designer で選択されている、シミュレーションする言語内の値も表示されます。
- 新しい関数 **mt-localized-string-name** は、提出された(ローカライズされた) 文字列のコントロール名または文字列名を返します。
- ボタン コントロールには、既に定義されたアイコンの選択肢からボタンにアイコンを追加することのできる、新しいボタンルック プロパティがあります。
- 水平線 コントロールには以下の新しいプロパティがあります: ボタンルックス 余白、上余白、下余白。
- 全ての幅のコントロール は、(コントロールの コントロールの幅 プロパティを介して) ページの幅 のパーセントを指定することができます。
- クリックに関するイベントをユーザーがコントロールをクリックする時間により区別されるようになりました。コントロールのタップは クリック時 (クリック) イベントで、長押しは 長いクリック時 (長いクリック) イベントです。 クリックに関するイベントは以下のコントロールで使用することができます: ボタン、チャート、イメージ、および ラベル。
- ノードの挿入 および ノードの追加 アクションには、挿入された/追加されたノードをプロジェクトデータソース内の元の場所から削除するオプションがあります。
- アクションの追加のためのキーボードショートカット はイベントの定義に与えることができます。
- デザイン内のコントロール には、ブラウザー CSS クラス プロパティにより割り当てられた1 つ以上のクラス名が存在する場合があります。外部 CSS ファイル内で、サーバーにデプロイする必要のあるクラスセレクターのルールを定義することができます。この外部 CSS ファイルへの参照はプロジェクトの内でブラウザーの設定 で定義されます。
- プロジェクトのブラウザーの設定 の新しいダイアログは、クライアントモバイルデバイス内でのブラウザーの振る舞いを定義する設定を収集します。
- カスタムフォント はデザイン内に埋め込まれることができます。

- [XPath/XQuery 式] ダイアログへの拡張機能は、ポップアップで表示される関数と演算子の情報、およびグローバル変数とカスタム文字列に関する情報です。
- ユーザー定義 XPath/XQuery 関数は関数名の昇順/降順/ダイアログ順に並び替えることができます。
- クライアントデバイス上でサーバー設定を更新する。

バージョン 1.4

MobileTogether Designer **バージョン1.4** の新機能とアップデートは以下の通りです。

- 交通ベースのモバイルソリューションには欠かせない機能である、位置情報の取得と処理へのサポート。位置情報データをトラック 読み込み、表示するアクションがイベントのために定義することができます。更に、位置情報を操作するためのAltova XPath 拡張関数がデザインのXPath 式に対して使用できます。位置情報はデザイナーとサーバーシミュレーションに設定することができ、位置情報の入力がシミュレーターでテストすることができます。
- JSON フォーマットのマップ、列挙、データなどを使用する新機能を提供するXQuery 3.1 をサポートします。[XPath/XQuery 式の編集] ダイアログを使用して式を作成および確認することができます。
- 文字列のローカライズ(追加言語への翻訳)によりソリューションの文字列の翻訳をプロジェクトに保管することができます。ソリューションが実行される言語は、モバイルデバイスで選択されている言語と同じになるように自動的に選択されます。特定の言語でソリューションを実行することでローカライズされたソリューションをテストすることができます。
- HTTP リクエストに特定のヘッダーを追加することができます。これは、リクエスト内で指定することのできるパラメータに追加しての機能です。
- ソリューションは順々に連鎖的に実行することができます。[次へ] ソリューションはソリューションのキャンセルアクションのオプションで指定されているソリューションを実行します。[ソリューションのキャンセルアクションは、v2.0 でソリューションの実行 アクションが搭載されているため、使用することができません。]
- iOS7/8 レンダリングおよびXML ツリー編集のためにソリューション機能が拡張されました。シミュレーター内でツリーを変更し、シミュレーションで即時、変更結果を確認することにより、テストスピードを向上することができます。
- プロジェクト メニュー以下を表示するコマンドを含みます: (i) 使用されるグローバルおよびページソース変数; (ii) 使用されるユーザー定義 XPath/XQuery 関数; (iii) 使用されるアクショングループ; (iv) および使用されていない変数、関数、アクショングループ。この機能により大規模かつ複雑なソリューションの保持および開発を向上することができます。

チャプター 3

はじめに

3 はじめに

このセクションはMobileTogether とMobileTogether Designer の概要を説明し、以下のセクションから構成されています:

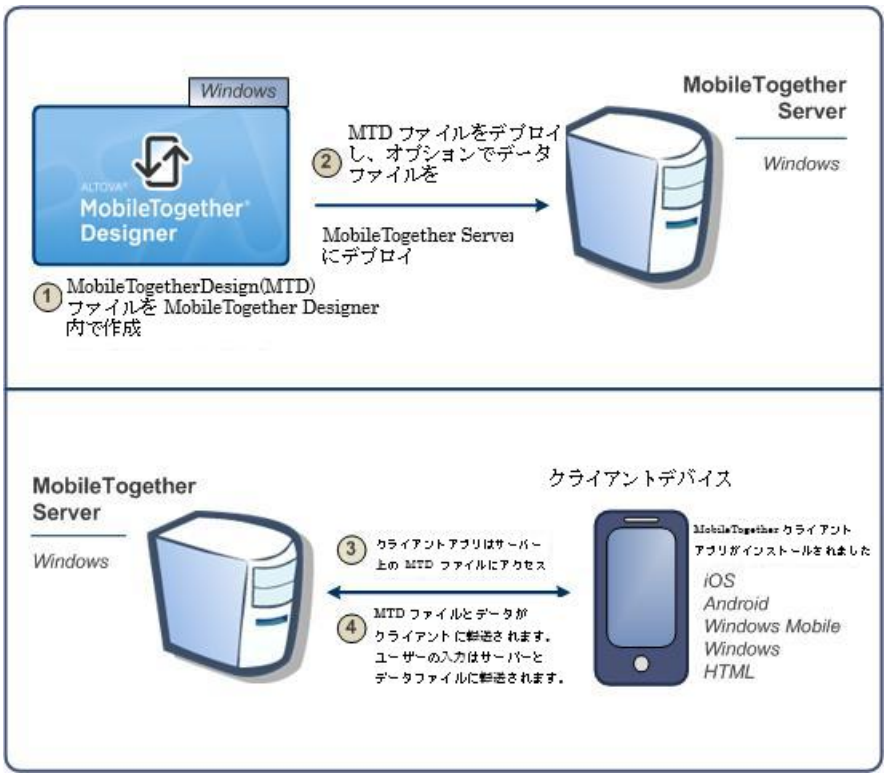
- MobileTogether 概要
- 用語 Q&A
- デザインのステップ
- MobileTogether 内のXPath

(C) 2018 Altova GmbH

3.1 MobileTogether 概要

MobileTogether は以下のモジュールにより構成されています:

- *MobileTogether Designer*は、モバイルクライアントのためのMobileTogether ソリューション(MTD ファイルは拡張子.mtd を持ちます) を作成します。これらのMobileTogether ソリューションは、MobileTogether Server にアップロードされます。
- *MobileTogether Server* はMobileTogether ソリューションをモバイル クライアントにサーブします。
- (iOS、Android、Windows Phone 8、Windows RT、Windows Metro、Web クライアント、Web ベースの スマートフォン/タブレットのための) *MobileTogether Client* アプリはエンドユーザーに受け取られ、MobileTogether ソリューション(**.mtd** ファイル) と相互作用し、MobileTogether Server により配信されます。



システム必要条件

▾ MobileTogether Designer

Windows	プラットフォーム更新済みのWindows 7 SP1、Windows 8、Windows 10
Windows Server	プラットフォーム更新済みのWindows Server 2008 R2 SP1 または以降

- MobileTogether Server

Windows	プラットフォーム更新済みのWindows 7 SP1、Windows 8、Windows 10
Windows Server	プラットフォーム更新済みのWindows Server 2008 R2 SP1 または以降
Linux	• CentOS 6 または以降 • RedHat 6 または以降 • Debian 7 または以降 • Ubuntu 12.04 または以降
(Mac) OS X, macOS	OS X10.10、10.11、macOS 10.12 または以降

- MobileTogether Client

iOS	Apple モバイルデバイスには9 以上
Android	Android モバイルデバイスには4.0 以上
Windows Mobile	Windows Phone にはWindows Phone 8.1
Windows RT、Metro	Windows 8.1、10; Windows RT タッチが有効化されているPC とタブレットコンピューター
HTML	その他すべてのモバイルデバイスにはHTML ブラウザー

3.2 用語 Q&A

MobileTogether システムのしくみ

- MobileTogether Designer で.mtd の拡張子を持つMobileTogether デザイン ファイル(MTD ファイル) を作成します。
- これらのファイルはMobileTogether Server にデプロイされ、モバイル クライアント デバイスにMobileTogether ソリューションとしてサーブされます。
- MTD ファイル内のデザインテンプレートを作成するために使用されるデータファイルは、元の場所に存在するか、MTD ファイルと共にMobileTogether Server にデプロイすることができます。
- モバイルクライアントでは、エンドユーザーはMTD ファイルに定義されたフォーマットでレポートを閲覧することができます。エンドユーザーは、(MobileTogether ソリューションを介して) データファイルをモバイル クライアント デバイスからアップデートすることもできます。

MobileTogether プロジェクトの MTD ファイルの内容

- MTD ファイルはネイティブのMobileTogether Designer ドキュメントです。
- 各 MTD ファイルはそれぞれ 1 つのMobileTogether プロジェクトを含んでいます。
- MobileTogether プロジェクトは 1 つ以上のページから構成されます。ページはエンドユーザーによりモバイル クライアント デバイスで閲覧されます。
- MTD ファイル内にページが 1 枚以上ある場合、ページは簡単なシーケンスで、最初のページから最後のページまで、つながっています。
- サブページは定義することができます。メイン ページの サブページへ移動 アクションによりサブページへアクセスすることができます。

ページの構成

- ページは、モバイルクライアントデバイスおよびユーザインタラクションのセットアップで閲覧できるようにフォーマットされている ページコントロール(略して[コントロール]) により構成されています。
- 各コントロールは異なるプロパティを持ちます。イベントのコントロール(略してコントロールイベント) がトリガーされている場合、これらのプロパティは関連したコンテンツ、フォーマット、およびアクションを定義します。
- 各ページは、ページの[ページ ソース] ペインでデータソースのセットが定義することができます。
- コントロールに関連付けられたコンテンツはこれらのデータソースから由来します。この様なデータはXPath または XQuery 言語を使用してアクセスすることができます。
- コントロールを介して、ページはデータをエンドユーザに提示され、データソースへの変更を受け入れることができます。

デザイン内の異なるイベントとアクション

- **コントロール イベント とアクション**: ページの各コントロールは、指定することのできるアクションをトリガーするイベントを持つことができます。例えば、コンボ ボックスのコントロールには、コンボ ボックスのドロップダウン リストからアイテムが選択された場合、発生する編集の終了時 イベントがあります。このイベントは、コンボ ボックスの選択の結果変更されるデータなど、希望するアクションをトリガーする場合、定義することができます。
- **ページ イベント とアクション**: (単一エントリとして) ページ自身は、アクションをトリガーするイベントに関連付けられ

ます。例えば、ページのロード時 はページイベントです。このイベントは、特定のデータファイルからページにデータをロードするなどの、希望するアクションをトリガーするように定義することができます。

3.3 デザインのステップ

以下にMobileTogether デザインファイル(MTD ファイル)の大まかな作成方法が順序を追って説明されています。

1. 新しい MTD ファイルの作成
   各 MTD ファイルは、簡単なシーケンスの1 枚以上のページで構成されたプロジェクトを提示しています。

   新しいMTD ファイルが作成されると、データソースの無いデフォルトのページが1 枚作成されます。デフォルトのページにデータソースを追加して、更にプロジェクトにページを追加することができます (*下記の点を参照してください*)。 [**ファイル**] [**新規作成**] コマンドを使用して新しいMTD ファイルを作成します。ファイルはメモリで作成されるので、[**ファイル**] [**保存**] コマンドを使用してディスクに保存されなければなりません。プロジェクト プロパティを定義し、他のプロジェクトに関連した設定を指定します。

2. ページ (ページソース) にデータソースを追加する
   各ページは、ページに表示されるデータの取得元であるデータソースに割り当てられています。ページのデータソースは[ページソース] ペインを介して追加され、ノードツリーとして表示されます。これらのノードからのデータは、表示または、通常 (チャートまたはイメージなどの) データのプレゼンテーションにつながる処理のためページデザインのコントロールで使用されます。データソースツリーのノードはXPath 式を用いてアドレス指定されます。希望する場合、クライアントデータ入力もデータソースに保存されることができます。詳細に関してはページソース のセクションを参照してください。

3. ページにコントロールを追加し、プロパティとイベントアクションを定義する
   ページコントロールは[コントロール] ペインからページに追加されます。各コントロールは([スタイル & プロパティ] ペインで定義された) プロパティのセットを持ち、関連した(データソースツリーからの) データを持ちます。コントロールには1 つ以上の定義済みのイベントが存在する場合もあります。イベントのコントロールがトリガーされた場合実行されるアクションを指定することができます。例えば、ボタンコントロールが ボタンのクリック時 というイベントを持つとします。そして、このイベントは関連した URL を開くというアクションを持つことができます。詳細に関しては、ページイベント とアクションを参照してください。更に、ページにはイベントがあり、ページイベントがトリガーされた場合実行されるアクションを指定することができます。例えば、ページが(ページイベントを) ロードして、指定された XML ファイルからデータを与えられたページデータソースにロードするアクションを指定することができます。

4. 必要な場合、プロジェクトに追加ページを追加してデザインする
   初期ページに追加ページを追加することができます。新しいページは[ページ] ペインのツールバーにある**[ページの追加]** アイコンをクリックして、トップページまたはサブページとして追加することができます。[ページ] ペイン内のトップページのシーケンスが、ワークフローのシーケンスを定義します。

5. トップページとサブページの間でのフローの作成
   サブページを使用して、ソリューションのワークフローを更に構成することができます。これらは、トップページ内から(コントロールまたはページイベントの) [サブページへ移動] アクション を使用してアクセスされます。他のページに関連したアクションにページ間の動きについて説明されています。

6. オプションとして、プロジェクトにユーザーに追加デザインと関連した機能性を追加する
   すべてのページが追加され、ワークフローの構成が完成すると、ページ デザインとワークフローを修正することができます。追加デザイン コンポーネントおよびアクションをプロジェクトに挿入することができます。

7. MobileTogether ソリューションのシミュレーションの実行

MobileTogether Designer 内で、ワークフローのシミュレーションを実行することによりデザインをテストすることができます。シミュレーションは（MobileTogether Designer 自身で）ワークフローがどのようにクライアントデバイスで実行されるか表示します。**[プロジェクト | ワークフローのシミュレーション]** を選択、または **[F5]** を押してシミュレーションを開始します。**[メッセージ]** ペインには、ワークフローアクティビティの詳細とステップバイステップのレポートが表示され、効果的かつ簡単なデバッグを行うことができます。

8. MTD ファイルをMobileTogether Server へデプロイする

   最終修正、ファイルの再テスト、および保存後、MobileTogether Server へファイルをデプロイします。MobileTogether ソリューションがモバイル クライアント デバイスからアクセスされる準備が整いました。

9. オプションでソリューションを AppStore App として作成する

   エンドユーザーがモバイルデバイスにダウンロードすることのできる MobileTogether カスタムアプリを作成することができます。これらのアプリは AppStore App と称されます。AppStore App のセクションで MobileTogether Designer プロジェクトからアプリのためのプログラムコード生成の方法が説明されています。コードは Android、iOS、Windows（タッチ有効化されたデバイスとPC）とWindows Phone のために生成されます。コードが生成された後、対応する AppStore App にコンパイルすることができます。

3.4 MobileTogether 内の XPath

XPath 言語はMobileTogether ソリューションのデザインの重要な役割を果たしています。XPath は、データを、ロケート、アクセス、操作、生成、および保存するために使用される多種のデータツリーで使用され、異なるデザインコンポーネントの機能を定義するために使用されます。MobileTogether デザイン内でXPath が使用される重要な使用方法は以下に説明されています。この概要はXPath の広義での柔軟性とMobileTogether デザイン内での使用方法を説明しています。

に関する詳細は、MobileTogether Designer でサポートされる言語の最新のバージョンである「W3C のXPath 3.1 に関する勧告」を参照してください。学習ベースのアプローチを使用するためには、以下を参考にすることができます:

- Altova のXPath の簡単な説明
- Altova のXPath 3.0 トレーニング (英語)
- W3C のXPath チュートリアル (英語)

ロケーター式

XPath 言語のロケーター式はツリー内のノードを検索するために使用されます。ロケーター式は通常、必要なノードを検索するパスにより構成されています。以下にいくつかのサンプルを紹介します:

- **`/Company/Office`**: は、トップレベルのドキュメントノードである`Company` 要素のすべての`Office` 子要素を検索します。 ルートノードの下で直接発生するため、最初のスラッシュで示される、`Company` 要素はトップレベルの要素です。
- **`/Company/Office[3]`**: は、`Company`要素の3 番目の`Office` 子要素を検索します。
- **`/Company/Office[3]/@location`**: は、`Company` 要素の3 番目の`Office` 子要素の`location` 属性を検索します。
- **`//Office[@location='US']`**: は、`US` の値を持つ`location` 属性を持つすべての`Office` 要素を検索します。

上のリストは基本的なロケーター式の例です。ロケーターを構築するには、多種の方法があります。

演算子

演算子により、フィルターの適用、条件の構築、選択とデータの操作を行うことができます。2 つの演算子の例を紹介します:

- **`if (Selection='US') then, //Office[@location='US'] else //Office[@location!='US']`**: この`if` 演算子は、`Selection` 子要素のコンテンツによりUS または非 US オフィスを選択します。
- **`for $i in //Office return $i[@location='US']`**: この`for` 演算子は、`US` の値を持つ`location` 属性を持つすべての`Office` 要素を返します。

XPath 関数

XPath 関数によりデータを操作、計算、生成することができます。例えば、関数は文字列を(関数の引数)入力とし、小文字に変換または文字列の一部を削除します。 MobileTogether デザイン内で使用することのできるXPath 関数には以下の種類があります:

- *内蔵された関数*

XPath 言語は、データおよびXML ツリーに関連したメタデータを抽出し、データを生成することのできる、大規模な内蔵された関数のライブラリを含んでいます。例えば

- **count(Office)**: は、Office 子要素の数を返します。
- **day-from-date("2015-04-26")**: は、関数の日付引数の一部である、数字 26 を返します。

内蔵された関数に関してのユーザーガイドと参照はインターネットで広く利用することができます。関数の完全なリストは「W3C のXPath 3.1 に関する勧告」で見つけることができます。

Altova 拡張関数

Altova が開発するXPath 拡張関数のセットは、開発者にXPath 内での機能性を提供します。現在、位置情報を提供する関数や整数を16進数に変換する関数、または16進数を整数に変換する関数など、およそ60 の拡張関数があります。例えば

- **format-geolocation(33.33, -22.22, 2)** は、xs:string "33.33N 22.22W" を返します。
- **hex-string-to-integer('1')** は、1 を返します。

Altova 拡張関数はすべてのMobileTogether デザインで使用することができます。使用方法の情報に関しては、MobileTogether Designer ユーザーマニュアルのAltova 拡張関数 セクションを参照してください。

MobileTogether 拡張関数

これらのXPath 拡張関数は、Altova により、MobileTogether デザイン内の具体的な使用のためにこのために開発されました。例えば

- **mt-has-server-access(10)** は、サーバーへのアクセスが関数の引数として指定されている秒内で可能な場合 true を返します。その他の場合はfalse を返します。
- **mt-load-string('MyCourier')** は、ソリューションの文字列プール内に保管されたローカライズされたMyCourier 文字列を返します。ローカライズの言語は、モバイルデバイスの言語に従い自動的に選択されます。

MobileTogether 拡張関数は、すべてのMobileTogether デザインで使用することができます。使用方法の情報に関しては、MobileTogether Designer ユーザーマニュアルのMobileTogether 拡張関数 セクションを参照してください。

ユーザー定義拡張関数

ユーザーは、上にリストされる関数ライブラリに適切な関数が見つからない場合、ユーザーは、XPath 拡張関数の一部を、具体的な目的のためにに定義することができます。例えば、セ氏と華氏の気温の違いを定義する関数が必要とします。ユーザー定義関数は、単一のMobileTogether プロジェクト内で定義され、その具体的なプロジェクト内で使用されます。この様なカスタム関数を定義する方法はMobileTogether Designer ユーザーマニュアルのユーザー定義 XPath/XQuery 関数 セクションで説明されています。

グローバル関数

グローバル関数はクライアントモバイルデバイスに関する情報を含んでいます。例えば、デバイスの種類を示す変数が1つあり、もう一つの変数はそのディメンションを示し、もう一つはデバイスの現在の向き(横長または縦長) を示します。これらの変数の値は、通常のモバイル通信の手順の一部としてクライアントデバイスから実行時に取得されます。その後、変数は

XPath/XQuery 式内で使用されることができます。結果、(サイズなど) デバイス固有の静的プロパティまたは(向きなどの) 変更可能な動的プロパティに対して条件的に、処理を指定することができます。

MobileTogether のグローバル関数は、グローバル関数 のセクションに定義済みで各変数の説明と使用可能な値と共にリストされています。下にリストされている`MT_iPad` グローバル変数 (使用可能な値: `true()`, `false()`) のサンプルはどの様にグローバル変数がXPath 式内でよびだされるかを示しています。 **$** シンボルは、XPath 内の変数を示す通常の方法である、グローバル変数の名前の後に何か続くかを示すために使用されます。

```
if ( $MT_iPad=true() ) then "Apple" else ""
```

チャプター 4

チュートリアル

4 チュートリアル

このセクションは、MobileTogether デザインを作成するために必要な基本ステップをガイドし、更に高度な MobileTogether の機能を理解するための手助けをするチュートリアルを含みます。

- クイックスタートチュートリアルは、基本から、MTD ファイルと関連ファイルを MobileTogether Server にデプロイするまでのステップをガイドします。
- シンプルなデータベースのチュートリアルでは、DB をベースにしたデザインの作成と、ユーザーの選択をベースにした DB 記録をロードする方法が説明されています。
- データベースとチャートのチュートリアルは既に完成したデザイン ファイルを使用して、データベースとの作業とチャートの作成を説明します。
- サブページと可視性 のチュートリアルは、トップページからサブページの開きかた、および `Visible` プロパティを使用してデータ構造の表示のフィルターの方法を説明します。
- レコードの追加と編集 のチュートリアルでは、MobileTogether Designer を使いこなすために一連の中間レベルの概念について説明されています。
- SOAP リクエスト チュートリアルは Web サービスからデータを取得するための SOAP の作成方法、またデザイン内で返されたデータを使用する方法を説明します。
- 位置情報の共有 チュートリアルは、位置情報機能の使用方法について説明しています。他の機能に関しては、共有 とエラーのキャッチで説明されています。
- スクロールすることのできるテーブル チュートリアルはスクロールすることのできるテーブルの主な機能について説明し、テーブルの作成方法を紹介します。

クイックスタート チュートリアルは、基本から、MTD ファイルと関連ファイルを MobileTogether Server にデプロイするまでのステップをガイドします。データベースとチャートのチュートリアルは既に完成したデザイン ファイルを使用して、データベースとの作業とチャートの作成を説明します。

チュートリアルで使用される既に完成したデザイン ファイル、およびデータソースとイメージ ファイルは、*(マイ ドキュメント フォルダー)* にあります: `Altova\MobileTogetherDesigner4\MobileTogetherDesignerExamples\Tutorials`。

`C:\MobileTogether` フォルダーを作成し、チュートリアルにあるフォルダーとファイルをコピーして **`C:\MobileTogether`** フォルダーに保存することが奨励されます。この手順の理由は、与えられたデザインファイル (**`.mtd`** ファイル) の一部は、**`C:\MobileTogether`** フォルダー内のターゲット ソースに絶対 URL を使用するからです。デザイン ファイルをサーバーにデプロイ中、(イメージファイルなどの) デプロイするリソースの一部を正確にロケートできない場合、(i) 各当するリソースファイルをデプロイするファイルのリストから削除し、(ii) 保存されているロケーションリソースを参照し、(iii) デプロイするファイルのリストにリソースファイルを(正確な場所と共に) 追加します 。

チュートリアルを完了した後、`MobileTogetherDesignerExamples` フォルダーから、更に高度なデザイン ファイル (**`.mtd`** ファイル) の例を開くことができます。これらの例は MobileTogether Designer の多種の機能の使用法を表示し、自身のデザインの基準点として使用することができます。

Windows 7、 Windows 8 および Windows 10 でのファイル パス

このドキュメントのファイル パスはすべてのオペレーティングシステムで同じではありません。以下の通信に注意してください:

- *(マイ ドキュメント フォルダー)*: デフォルトでは以下の場所に存在します。Example ファイルはこのフォルダのサブフォルダに存在します。

Windows 7/8/10	`C:\Users\<username>\Documents`

- *アプリケーション フォルダー*: アプリケーション フォルダーには Altova アプリケーションが存在します。デフォルトでの、

アプリケーションフォルダのパスは以下の通りです。

Windows 7/8/10	`C:\Program Files\Altova\`
64-bit OS 上での32 bit バージョン	`C:\Program Files (x86)\Altova\`

Example ファイル内のファイルURL
Example ファイル内のファイルURL は絶対 URL である可能性があります。その場合は、作業環境のディレクトリ構成に対応していない場合があります。与えられたExample ファイルを使用する場合、ファイル内のファイルURL ? とそのような URL を構築するXPath 式が? マシンまたはネットワークの正しいファイルの場所をポイントしていることを確認してください。

4.1 クイックスタート(パート1)

このクイックスタート チュートリアルは、MobileTogether デザインのMobileTogether ソリューション作成ための基本ステップを案内します。ソリューションは、エンドユーザーがコンボ ボックスから選択するAltova 製品のスプラッシュ スクリーンを表示しています。チュートリアルは、ページ、データソースのセットアップ、コントロールの追加、エンドユーザーの入力による条件付きスプラッシュ スクリーンの表示などを説明します。また、検証、MobileTogether Server へのデザインファイルのデプロイ方法、シミュレーションの実行方法についても説明します。このクイックスタート チュートリアルを終了すると、MobileTogether デザインの大まかな作動原則について十分理解できるはずです。この後、更に複雑なデザインに取り組むことができます。

このチュートリアルは、MobileTogether デザインの重要な側面を構成する以下のセクション別に整理されています。

- 新しいデザインの作成
- ページのセットアップ
- ページデータソースの追加
- デザインのフォーマット
- コントロールの追加: コンボ ボックス
- コントロールの追加: イメージ
- コントロール アクションの定義
- プロジェクトの検証
- シミュレーションの実行
- サーバーにデプロイ

チュートリアル ファイル

このチュートリアルのファイルは(マイ ドキュメント MobileTogether フォルダーにあります: **`MobileTogetherDesignerExamples\Tutorials\QuickStart`**。チュートリアルで使用されるスプラッシュスクリーンのイメージファイルは他の場所に保存して使用することができます。

- 使用するファイルは以下に類似しています: **`QuickStart01.mtd`**
- チュートリアルで使用されるイメージファイルは、フォルダーの **`*.bmp`** ファイルです。

新しいデザインの作成

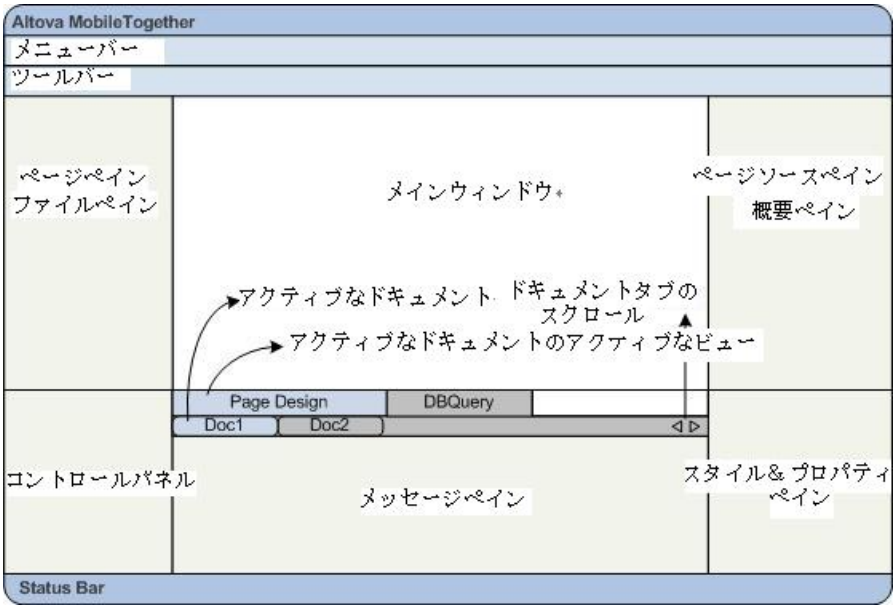
このパートでは以下を行います:

- 新しいMobileTogether デザインの作成
- デザインファイルの保存
- デザインのプレビュープロパティの設定: プレビュー デバイスと拡大率のレベル
- MobileTogether Designer のGUI に慣れ親しむ

MobileTogether Designer ネイティブの`.mtd` の拡張子を持つMobileTogether デザインはMTD (MobileTogether デザイン) ファイルで作成されます。

新しいMobileTogether デザインを作成するには、以下を行います:

1. **[ファイル| 新規作成]** をクリックします。これにより、新たしいデザインファイルがGUI (*以下の図*) のメインウィンドウに開かれます。このファイルはデフォルトで[新しいページ1] という名前を持ちます。



2. 新しいデザインは[新規デザイン1] というタブにあります。**[ファイル| 保存]** をクリックして、ファイルを希望する名前で希望する場所に保存します。デザインファイルが、スプラッシュスクリーンのイメージが保存されているフォルダーと同じフォルダーに保存されていることを確認してください。デザインが完成すると、Altova 製品のスプラッシュスクリーンを表示するので、例えば、`AltovaSplashScreens.mtd` と名付けることができます。`.mtd` の拡張子を持つ新しいファイル名は、ファイルタブの[新規デザイン1] を置き換えます。ファイル名はアプリケーションのタイトルバーにも表示されます。
3. MobileTogether Designer 主要なツールバーで**[プレビューデバイス]** を選択して、デザインの**[拡大率]** を設定できます。希望するオプションを選択します。モバイルデバイスは異なる背景の色とディメンションを持つので、適切なプレビューデバイスを選択することは、デザインをビジュアル化する際、役に立ちます。プレビューデバイスはデザインを他のデバイスでビジュアル化を希望すると随時変更することができます。これらの設定はシミュレーションにも適用されます。
4. メイン ウィンドウとタブおよびメイン ウィンドウの周りに存在する異なるペインに慣れ親しんでください。ユーザーインターフェイスに関しては、ユーザー インターフェイスのセクションで説明されています。
5. メイン ウィンドウのページ デザインとDB クエリタブを切り替えて、ペインの内容の変化を確認してください。

ページのセットアップ

MobileTogether デザインは1 枚以上のページから構成されています。ページとはエンドユーザーがモバイル クライアント デバイスで閲覧するものです。デザインに複数ページある場合、ページシーケンスは[ページ] ペインのシーケンスの順序により定義されます。ページ内で、(通常再利用可能なモジュールを含む) サブページへ移動するためのコントロールを設定することができます。チュートリアルのこのパートでは、手順を簡単にするために、1 ページから構成されるプロジェクトを作成します。ページに関する詳細は、[ページ] ペインに説明されています。

1 枚のページのデザインを作成します。このパートでは以下を行います:

- デフォルトのページに名前を与える
- ページのタイトルを表示するためにラベルのコントロールを追加する
- このラベルをフォーマットする

ページ名とラベルのコントロール

新しいデザインのデフォルトのページに以下の手順で名前を与えます:

1. [ページ] ペイン(*下のスクリーンショット*) で、新しいページ1 をダブルクリックして `SplashScreens` に名前を変更します。

2. [コントロール] ペイン から、[ラベル] コントロールをデザインにドラッグ アンド ドロップします。ページ上部にコントロールが置かれます。赤い感嘆符が表示されます。感嘆符をポイントすると、ラベルにコンテンツが無いことを知らせるメッセージが表示されます。
3. ラベルのコンテンツは静的または動的であることができます。動的な場合、コンテンツはXPath 式または、ページのデータソース1 つのノードから得ることができます。(`SplashScreens` ページのデータソースはまだ定義されていません。この手順は次のパートで行われます。) 静的コンテンツはラベルをダブルクリックしてテキスト、`Altova Splash Screens`、を入力し、**[Enter]** を押します。
4. テーブルツールバーで、**[中央揃え]** をクリックして、テキストを中央に揃えます。**[テキストの色]** および/または **[背景色]** をクリックして(またはテーブルツールバー内でも)、これらのプロパティを設定します。
5. 選択された[ラベル] コントロールにより、ラベルのプロパティは [スタイル & プロパティ] ペインに表示されます。希望する場合、このペインで、ラベルの下に余白を作成する下余白プロパティなどの利用可能なラベルのフォーマット プロパティを変更します。

ページ データソースの追加

ここまでの手順で、ページを作成し、タイトルを付けました。このパートでは以下を行います:

- このページに空白のXML データソースを追加する
- このデータソースには構成またはコンテンツがないため、直接 [ページソース] ペインに構成を作成する。
- デフォルトのXPath コンテキストノードを設定する。

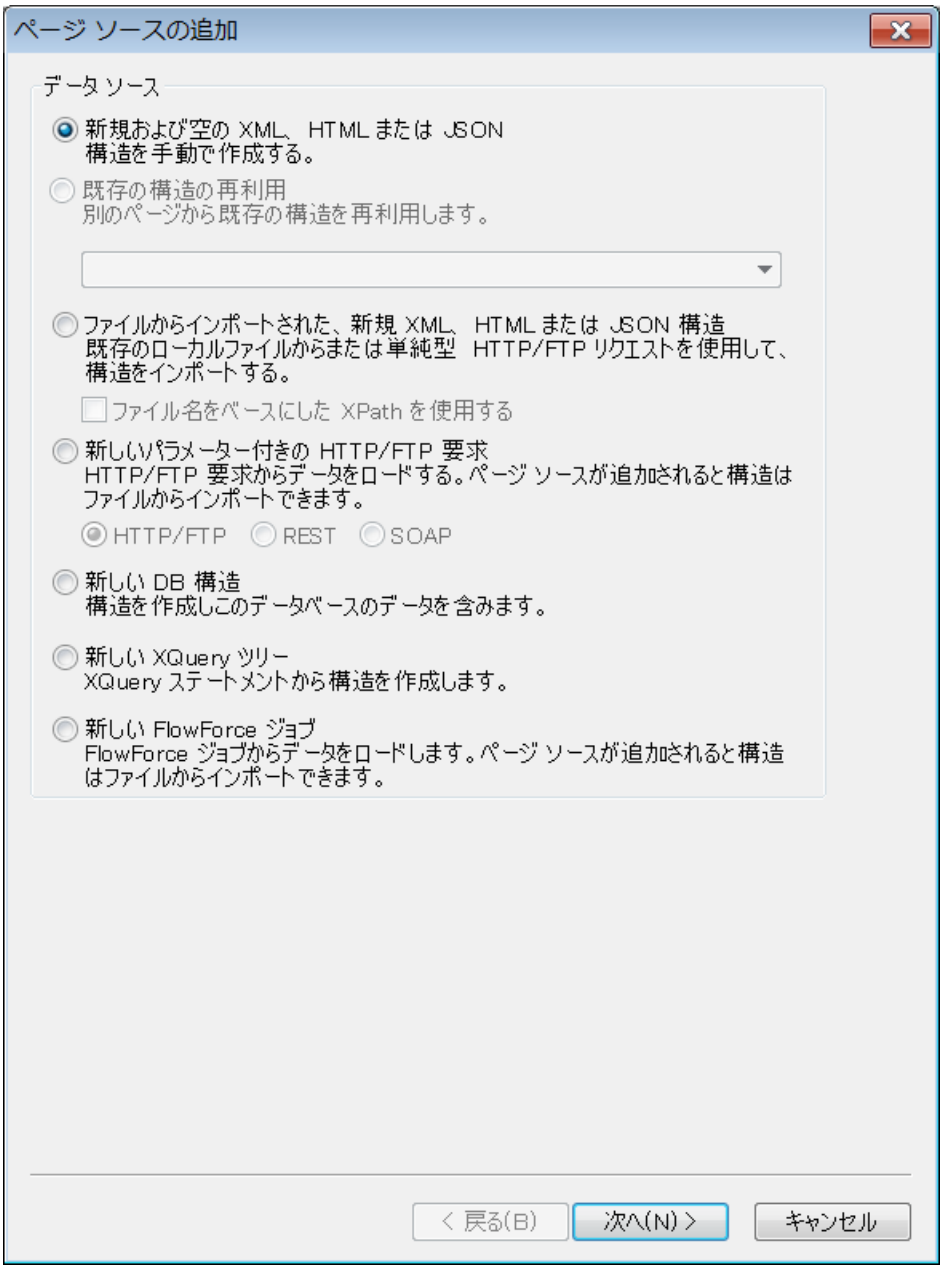
このセクションのボタン

ソースの追加

ページはデータの取得先であるページソースを1 つ以上持つことができます。データソースは、[ページソース] ペインで定義され、読み取り専用または編集可能を指定できます。データソースは外部ソース、または直接[ページソース] ペインで作成され静的データを含むことができます。使用できるデータソースの種類は、データソースのセクションで説明されています。このチュートリアルでは、編集可能な空白のXML ソースである1 種類のデータソースを追加し、直接 [ページソース] ペインで構成のコンテンツを作成します。

ページのデータソースを追加するには、以下を行います:

1. [ページソース] ペインの**[ページソースの追加]** ボタンをクリックして、ページソースの追加ダイアログを表示します (*下のスクリーンショット*)。



2. 新しい、空白の*XML* を選択して**[次へ]** をクリックします。
3. 次のスクリーンでデフォルトの設定を保持します(編集可能として、データソースが作成されると、含まれるデータを変更することができます)。**[OK]** をクリックします。`$XML1` と呼ばれるルートノードが[ページソース] ペインのページソースツリーに追加されます(*下のスクリーンショット*)。

4. ルートノード $XML1 のあるXML ツリーが空白なことに気が付きます。$XML1 を右クリックして、**[子の追加 | 要素]** を選択し、要素名 Products を入力します。
5. (右クリックして **[子の追加 | 要素]** を選択して) Products に子要素を追加し、子要素を Product と名付けます。要素の名前の変更を希望する場合は、要素の名前をダブルクリックして編集します。
6. 要素を右クリックして、**[ページのロード時に存在することを確認 (固定値)]** を選択します。Product 要素を右クリックして、**xmlspy** を入力します(*下のスクリーンショット参照*)。ページロード時の Product 要素のデフォルトのコンテンツとなります。

$XML1 に割り当てられたXML デフォルトのファイルは存在しません。デフォルトのファイルは、データソースツリーのノードに入るデータを提供します。デフォルトのファイルから提供されるデータは、手動でツリーノードに入力されるデータに上書きされます。このケースの場合、単一のデータノード **$XML1/Products/Product** があり、文字列 **xmlspy** をデフォルトコンテンツとして持ちます。XML データ構成は以下の通りです:

```
<Products>
  <Product>xmlspy</Product>
</Products>
```

ページ XPath コンテキスト ノードの設定

各ページは、そのページで定義された、相対 XPath 式 のコンテキストノードとして使用される、ページ XPath コンテキスト

ノードを持っています。ページXPath コンテキストノードは、ラベル *XPath コンテキスト* 付きの[ページソース] ペイン内で指定されています (*下のスクリーンショット参照*)。各ページは自身のページXPath コンテキストノードを持ちます。

ページXPath コンテキストノードを**Products** ノードに変更します。**Products** を右クリックして、**[XPath コンテキストとして設定]** を選択します。この要素 **Products** がページXPath コンテキスト ノードとして設定され、*XPath コンテキスト*(*下のスクリーンショット参照*) とラベルされて表示されます。

メモ 希望する場合、XPath 式内で、ページXPath コンテキストノードではない、データソースツリーのノードを参照し、そのノードの絶対パスを使用し、ツリーのルートノードから開始することができます。 例えば **$XML2/SomeRootElement/SomeOtherElement**。

デザインのフォーマット

デザインペインは(スプラッシュスクリーン) と呼ばれており、ディスプレイタイトル(Altova Splash Screens) とデータソース($XML1) を表示しています。このパートでは以下を行います:

- プレゼンテーションのためのテーブル コントロールの追加
- フォーマットの追加: 水平線コントロール、スペース、および色
- デザインの他の部分のコピー貼り付けコントロール

テーブルの追加

プレゼンテーションのためにテーブルを追加します。

1. [コントロール] ペインから、[テーブル] コントロールをデザインにドラッグして、ラベルの下にドロップします。
2. 表示された新しい[テーブルダイアログ] で、テーブルには2 つの静的な列と1 つの静的な行があるように指定し、**[OK]** をクリックします。作成されたテーブルには2 つの列と1 つの行があります。
3. [コントロール] ペインから、[ラベル] コントロールを左側のセルにドラッグ アンド ドロップし、静的テキストの値 `Altova Product` を与えます(*テーブルがハイライトされている下のスクリーンショット参照*)。

水平線、スペースおよび色の追加

タイトルとテーブルを分ける水平線の追加は以下の通り行います:

1. [コントロール] ペインから、水平線コントロールをタイトルとテーブルの間にドラッグ アンド ドロップします(*下のスクリーンショット参照。深い青色の水平線を、タイトルとテーブルの間にドラッグ アンド ドロップします*)。

2. デザインで選択するラインは、[スタイル & プロパティ] ペインで色と幅を設定することができます。
3. **[Ctrl+C]** を押して、ラインをコピーし、(**[Ctrl+V]** 使用して) テーブルの下に貼り付けます。ラインは定義したすべてのプロパティと共にコピーされます。
4. ラベル、テーブル、テーブルの個別のセルの背景色を変更することができます。カーソルをそれぞれのコンポーネントにあて、[スタイル & プロパティ] ペインで異なる背景色を選択します。
5. ラベルコントロールを選択して、[上余白] と[下余白] のプロパティを[スタイル & プロパティ] ペインで設定します。
6. デザインの色々なパートの異なるコンポーネントのコピーを試してください。

コントロールの追加: コンボ ボックス

前のセクションで作成されたテーブルの右側のセルにコンボ ボックスを追加します。その後、コンボ ボックスのプロパティを定義します。コンボボックスには2 つの重要なプロパティ定義が必要です:

- *コンボ ボックスの値*: これらの値はコンボ ボックスのドロップダウン リストを設定し、ユーザーが選択できるオプション (*可視エントリ*) を提供します。各可視エントリには、関連したXML ノードに入るために対応するXML 値があります。
- *関連したXML ノード*: このデータソースノードはコンボ ボックスのセクションと同期されています。コンボ ボックス セクションから値を受け取ります。初期の値は初期コンボ ボックスの選択を決定します。

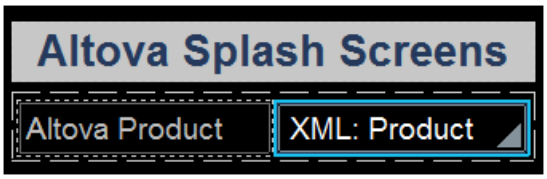
このセクションのボタン

... **追加ダイアログ**

コンボボックスの追加

コンボ ボックスを追加し、そのプロパティを定義するには以下を行います:

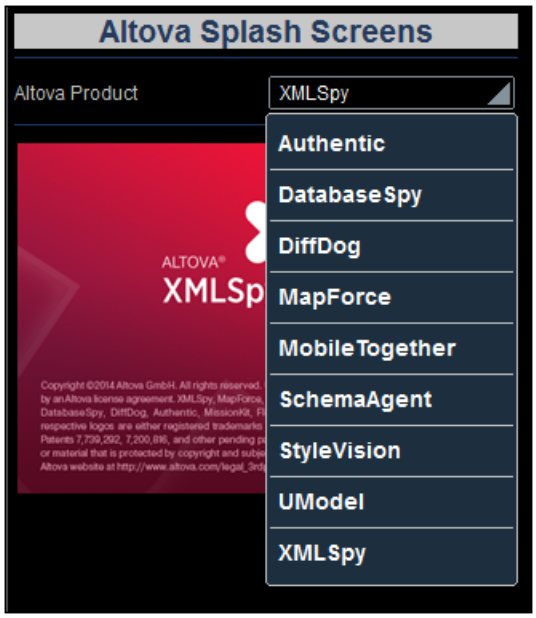
1. [コントロール] ペインからコンボ ボックス コントロールをテーブルの右側のセルに ドラッグ アンド ドロップします(*下のスクリーンショット参照*)。



2. [ページ ソース] ペインから`Product` ノードをドラッグしてコンボ ボックスにドロップします。`Product` ノードにコンボ ボックスを関連付けは、いわゆる**[ソースノード]** (またはページソースリンク)を作成します。エンドユーザーがコンボ ボックスからオプションを選択する際、選択されたオプションのXML 値が`Product` ノードにパスされ、`Product` ノードのコンテンツとなります。
3. コンボボックスのドロップダウン リストのアイテムを定義するには、コンボ ボックスを選択して、[スタイル & プロパティ] ペイン内の[コンボボックス入力値] プロパティの**[追加ダイアログ]** ボタンをクリックします。コンボ ボックスの編集ダイアログ (*下のスクリーンショット*) が表示されます。

4. [*値のリストを使用*] を選択して、[*異なるテキストとXML 値を使用*] を選択します。(コンボボックスのドロップダウンリストの)可視エントリの値と(`Product` ノードに書き込まれる)対応するXML 値を入力します。イメージコントロールの追加 の際、XML 値が小文字で記入されている理由を説明します。(ヒント: イメージ名は、例えば `xmlspy.bmp` の様に、小文字で記載されています。) 9 つあるAltova 製品を好きな順序で入力します: `XMLSpy`、`Authentic`、`StyleVision`、`MapForce`、`DiffDog`、`Databesespy`、`MobileTogether`、`SchemaAgent`、`UModel`。
5. ダイアログ下部の[値を並べ替え] のチェックボックスをチェックして、リストを並べ替え表示されるとを**[OK]** クリックして終了します。

完成したソリューションが実行され、または、完成したプロジェクトのシミュレーションが実行される際、ドロップダウンリスト(*下のスクリーンショット*) は、可視入力列の定義にリストされた値を表示します。(*上のスクリーンショット参照*)。



このプロジェクトの目的: エンドユーザーがドロップダウンリストからエントリを選択する際、(*上記コンボボックスの定義のスクリーンショット参照*) エントリに対応するXML 値がデータソースの`Product` ノードにパスされます。`Product` のデフォルトのコンテンツは(ノード作成時に定義された) `xmlspy` であることに注意してください。 ノード内の値を使用して、スプラッシュスクリーンのイメージを表示するURL を作成することができます。この点に関しては、チュートリアルの次のパートコントロールの追加: イメージで説明されます。

(C) 2018 Altova GmbH

コントロールの追加: イメージ

デザインにイメージを追加します。このイメージはエンドユーザーがコンボボックス(*前のセクション参照*) を選択するAltova製品のスプラッシュスクリーンとなります。最も重要なイメージのプロパティは イメージ URL プロパティと呼ばれ、表示されるイメージを選択します。作成されるURL は デザイン ファイルと同じフォルダー内のイメージ ファイルを探します。もちろん、絶対ファイルURL または インターネットのロケーションでのイメージを使用することもできます。

このパートでは以下を行います:

- イメージ コントロールの追加
- イメージ ソースプロパティを、コンボボックスの選択で動的に変更するURL を許可するXPath 式を使用して定義します。このチュートリアルで参照されるイメージは、(*マイ ドキュメントフォルダー*内のフォルダーの以下の場所にあります、`MobileTogetherDesignerExamples\Tutorials` 。希望する場合は、イメージを他のフォルダーに保存することができます。

このセクションのボタン

XPath

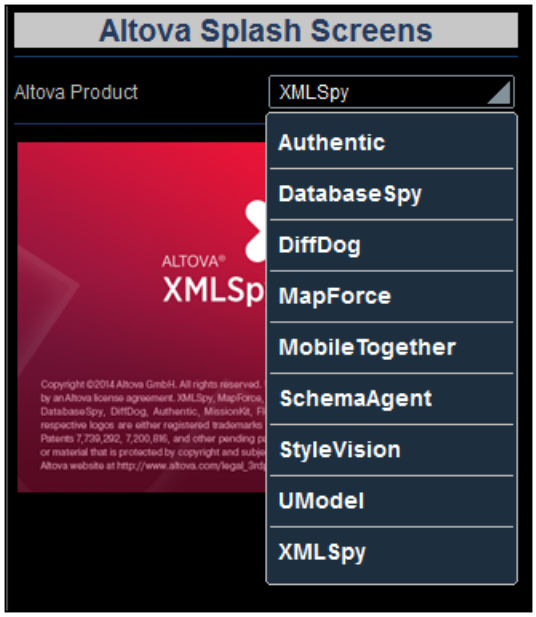
イメージの追加

イメージを追加、しそのプロパティを定義するには以下を行います:

1. [コントロール] ペインからイメージ コントロールをテーブルの下にドラッグ アンド ドロップします。
2. [スタイル & プロパティ] ペインでイメージの名前を、[名前] プロパティを[`Image: SplaschScreen`] と設定します。 .
3. **[イメージ ソース]** プロパティを選択して、[スタイル & プロパティ] ペインのツールバーの**[XPath]** ボタンをクリックすると、[XPath/XQuery 条件式の編集] ダイアログが表示されます。
4. XPath 式を入力します: concat(Product, 'bmp')、**[OK]** をクリックします。

このXPath 式は、デザインファイルと同じフォルダー内の **`.bmp`** ファイルを見つける**[相対 URL]** を作成します。イメージファイルが保存されている、正しい場所を指定する必要があります。 必要に応じて、絶対 URL (*下部の例を参照*) を使用することができます。ファイル名 (例えば、`xmlspy`) は、エンドユーザーのコンボボックスからの選択のコンテントを取得する `Product` ノードから取得することができます。のデフォルトの**`Product`** ノード値は**`xmlspy`** に設定されていますので、XPath 式とイメージURL の開始は以下の通りになります:

このXPath 式:	**`concat(Product, 'bmp')`**
はこの絶対 URL を作成します:	**`xmlspy`**`.bmp`



エンドユーザーがコンボボックス(*上のスクリーンショット参照*)のドロップダウンリストから製品を選択すると、例えば、**`MobileTogether`**、の場合、MobileTogether *のスプラッシュスクリーン*が表示されます(*下のスクリーンショット*)。これは、MobileTogether の選択が**`mobiletogether`** であることに対応したXML 値のためです。このXML 値は、**`Product`** ノードにパスされ、XPath 式が動的に対応イメージURL: **`mobiletogether.bmp`** を作成するために使用されます。

イメージが再ロードされたことを確認するために、コンボボックスの選択がなされると、アクションの再ロードがコンボボックスで設定されていなければなりません。手順は次のセクション[コントロール アクションの定義](#)で説明されています。

コントロール アクションの定義

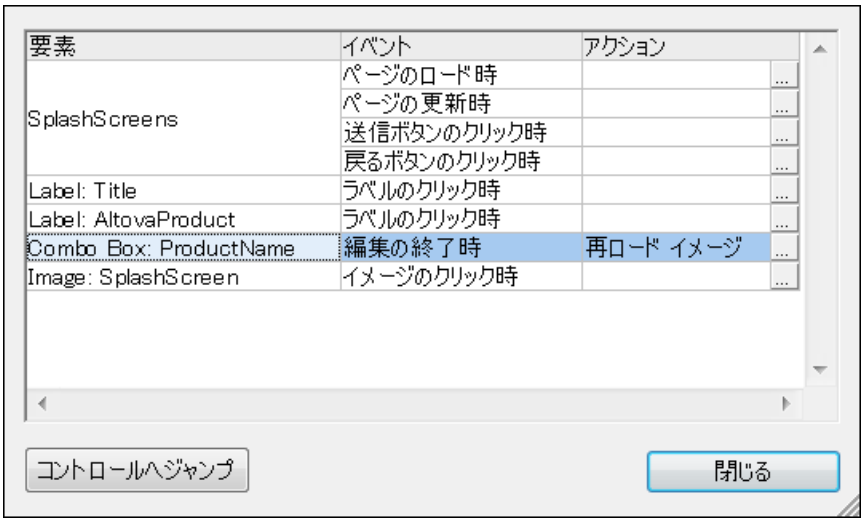
コントロール アクションは ボタンのクリック コンボボックスの選択などのイベントに対して、どのようなアクションをコントロールがとるか定義します。アクションは以下を含みます: データノードのアップデート、ページソースの再ロードまたは保存、データベースコマンドの実行。

このパートでは以下を行います:

- *ページのために定義されたページ アクションとコントロールを確認する*
- *コンボ ボックスが編集されるとイメージをアップデートするコンボ ボックス アクションを追加する*

ページアクションの概要

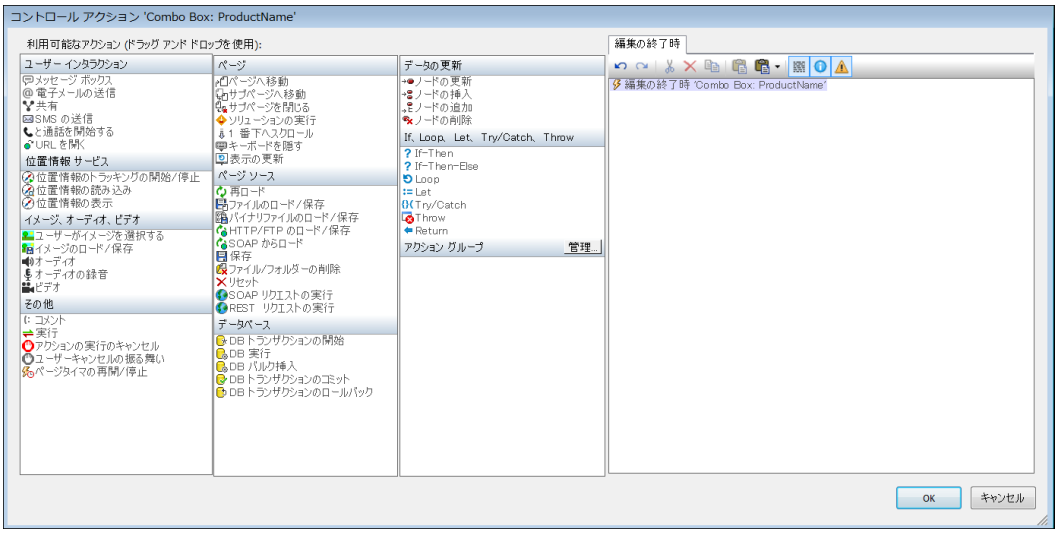
SplashScreens ページのすべてのアクションの概要に移動するには、**[ページ | アクションの概要]** をクリックします。ページ アクション概要ダイアログ(*下のスクリーンショット*) が表示されます。すべてのイベントとそのアクションが現在ページのために定義されている通り表示されます。ページイベントとコントロールイベント、およびこれらの対応するアクションなどが表示されます。下のスクリーンショットのSplashScreens 要素はページを参照します。他の全ての要素はページデザインの異なるコントロールです。現在イベントのアクションは定義されていません。



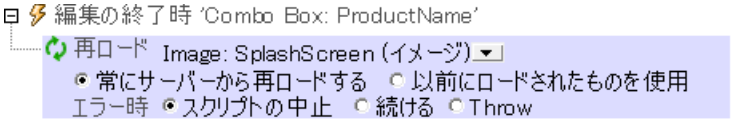
コンボ ボックス コントロールの再ロード アクションの定義

コントロールアクションの定義は2 部から構成されています: (i) アクションをトリガーするコントロールイベントの選択; と(ii) イベントが起こると、実行されるアクションの指定。例では、以下を行います: ユーザーがコンボ ボックスの選択を行うと、イメージを再ロードする。これは、イメージURL が(コンボ ボックスで選択された)新しいProduct ノードの値と再評価されるためです。ですから、コンボ ボックスの編集が完了すると、イメージを再ロードする*再ロードアクション*が必要になります。このコンボ ボックスイベントとアクションの定義は以下の通り行います。

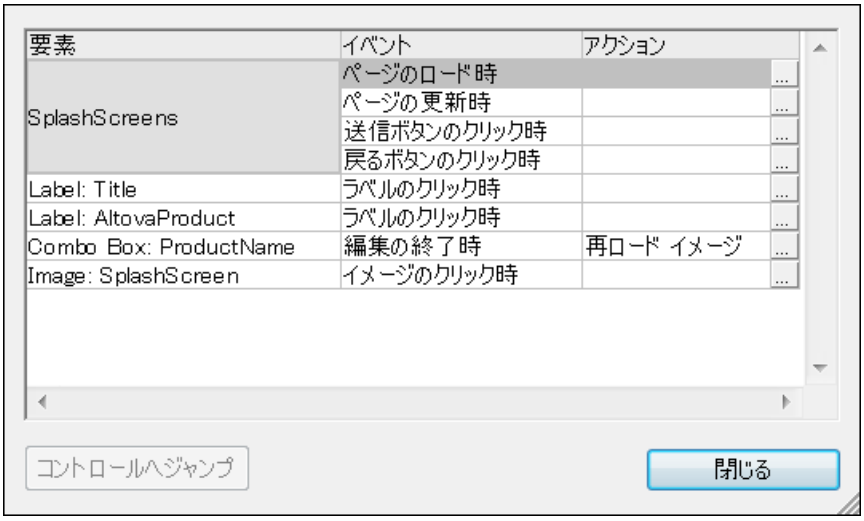
1. **XML:Product** コンボボックスを右クリックして、**[編集の終了時 のコントロールアクション]** を選択する。このオプションによりコンボボックスのコントロールアクションダイアログが開かれます(*下のスクリーンショット参照*)。コンボボックスイベントは1 つしかありません **編集の終了時** (*右側のペイン参照*)。もし、追加イベントが利用可能な場合、右側のペインの追加タブとして表示されます。左側のペインに、グループ分けされたイベントのすべての利用可能なアクションがリストされます。



2. **再ロード** アクションをページソースグループからドラッグし、**編集の終了時** タブ (*下のスクリーンショット*) にドロップします。これによりコンボボックスが編集されると再ロードアクションが実行されることが指定されます。
3. ドロップダウンの (`$XML1` の横の) 矢印をクリックして、`Image: SplashScreen` を選択し、**[OK]** をクリックします。これによりコンボボックスの値が変更されるとイメージコントロールが更新されます。



ページアクション概要ダイアログを見ると、[*再ロード イメージ*] アクションがコンボボックスの編集の終了時 イベントに定義されていることが確認できます。



要素	イベント	アクション
SplashScreens	ページのロード時	
	ページの更新時	
	送信ボタンのクリック時	
	戻るボタンのクリック時	
Label: Title	ラベルのクリック時	
Label: AltovaProduct	ラベルのクリック時	
Combo Box: ProductName	編集の終了時	再ロード イメージ
Image: SplashScreen	イメージのクリック時	

プロジェクトの検証

デザインページが完成したので、プロジェクトを[**プロジェクト | 検証**] で検証してエラーを確認します。検証では、エラーまたは警告の有無を知らせるメッセージが表示されます。

- 警告とは、注意が必要なデザイン問題ですが、致命的ではなく、ソリューションの実行を妨ぐものではありません。
- エラーメッセージとは、一方警告の反対で、致命的である可能性があり、解決されることが奨励されます。

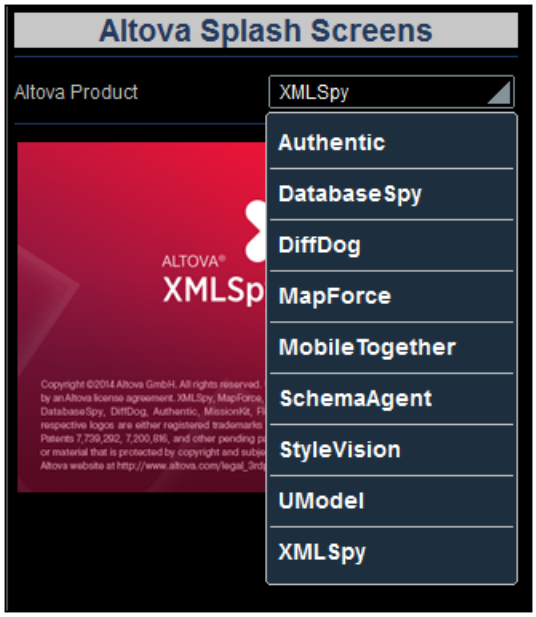
メモ メッセージウィンドウの複数のメッセージは展開することができる詳細を含んでいます。また詳細を提供するクリックすることのできるリンクも含まれています。

シミュレーションの実行

MobileTogether Designer 内で直接プロジェクトワークフローのシミュレーションを実行することができます。シミュレーションデバイスはメインツールバーのプレビュー デバイス コンボ ボックスで現在選択されているデバイスです。異なるデバイスでシミュレーションを確認するためプレビューデバイスを変更することができます。シミュレーションを実行するには **[プロジェクト| ワークフローのシミュレーション]** または **[F5]** を選択します。これによりシミュレーターを開きシミュレーションが開始します。

`Product` のデフォルトの値が `xmlspy` に設定されているため、XMLSpy スプラッシュスクリーンが表示されます。`$XML1` ノード(下のスクリーンショット) を展開すると、右の ページソース ペインに表示されます。

シミュレーションでは、ドロップダウン リストを表示するためコンボ ボックスをクリックします(下のスクリーンショット)。



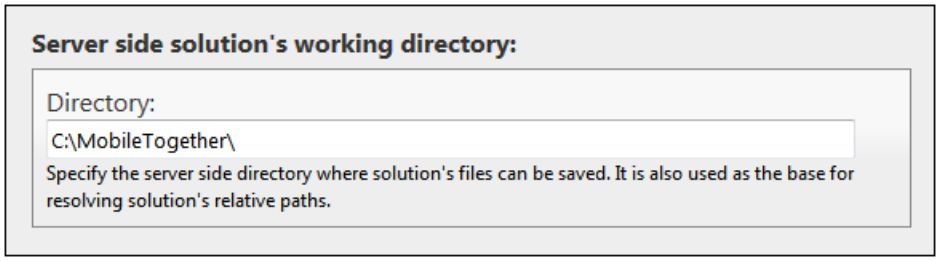
リストから製品を選択するとイメージと`Product` ノードの値が変更されることが確認できます。(**[セット]** を押して選択を確認する必要があります。)

終了すると**[閉じる]** または **[Esc]** を押して、シミュレーターを閉じます。

サーバーシミュレーションの実行

MobileTogether Server の**[プロジェクト | ワークフローのシミュレーションにサーバーを使用]**を使用してサーバーシミュレーションを行うことができます。MobileTogether Server のWeb UI の**[設定 | サーバー側ソリューション作業ディレクトリ]** によりソリューションの作業ディレクトリを設定します。 *下のスクリーンショット参照*)。デザインのすべての相対パスはこの設定で指定されたディレクトリに相対して解決されます。サーバーシミュレーションが正しく作動するためには、レファレンスファイルが保存されているディレクトリのパスを入力します。スクリーンショット(`C:\MobileTogether\`) で表示されているディレクトリの設定は、(デザイン内の相対イメージがイメージファイルとデザインが同じディレクトリに存在するよう指定す

るため) デザインとイメージファイルがフォルダー `C:\MobileTogether\` に存在しなくてはならないことを意味します。



サーバーにデプロイ

プロジェクトの検証とシミュレーションのテストに成功すると、プロジェクトをMobileTogether Server にデプロイすることができます。プロジェクトがサーバーにデプロイされると、MobileTogether Client アプリが作動できるソリューションとなります。プロジェクトとファイルのデプロイの詳細については、プロジェクトのデプロイ および プロジェクト ファイルの場所のセクションを参照してください。

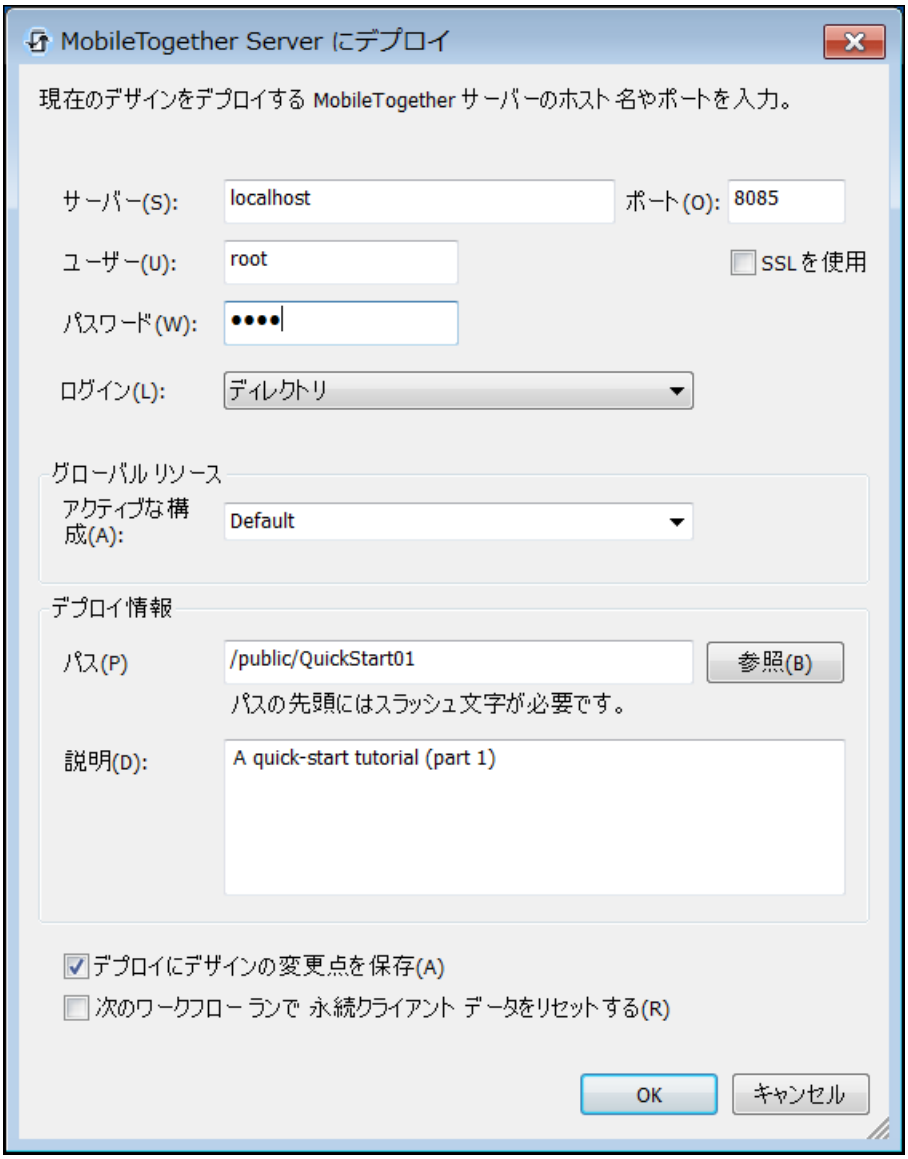
MobileTogether Server にプロジェクトをデプロイするには、*デザイナーからのワークフローの保存の*特権を持つユーザーとしてサーバーにアクセスする必要があります。MobileTogether Server のユーザーのアクセス権に関しては、MobileTogether Server のWeb UI に説明されています。ユーザーの特権の設定に関する情報は MobileTogether Server ユーザーマニュアル を参照してください。

MobileTogether Server にプロジェクトをデプロイ

プロジェクトと関連した(イメージ) ファイルのMobileTogether Server へのデプロイは以下の通り行います:

1. [ファイル] ペイン(*下のスクリーンショット*) 内イメージファイルを右クリックし、表示されるコンテキストメニューの **[ファイルの追加]** を選択します。

2. イメージファイルを参照して、イメージ ファイル リストに追加します。
3. ファイルを右クリックし、 **相対的なパスにする** を選択します。
4. ステップ1 と2 を繰り返し、必要なイメージファイルすべてを追加します。
5. **[ファイル | MobileTogether Server へデプロイ]** を選択すると、MobileTogether Server へデプロイダイアログが表示されます (*下のスクリーンショット*)。



6. MobileTogether Server アドレスと、管理者ポート(デフォルト HTTP ポートは 8085、HTTPS ポートは 8086) を入力します。
7. MobileTogether Server にアクセスするための、ユーザー名とパスワードを入力します。このユーザーは、デプロイを成功させるためデザイナーからワークフローの保存の特権を持つ必要があることに注意してください。ユーザーの特権に関しては MobileTogether Server ユーザーマニュアル を参照してください。
8. デプロイ | パスのフィールドに、サーバーでこのソリューションがデプロイされる名前を入力します。例えば、上部のスクリーンショットで、ワークフローは、**/public** コンテナーに保存され、**QuickStart01** という名前を持ちます。MobileTogether Server では、 **QuickStart01** というエントリのみが表示されます。このエントリに含まれるイメージファイルは異なるファイルとしては表示されません。
9. **[OK]** をクリックして完了します。デプロイに選択されたプロジェクトとファイルは MobileTogether Server に **QuickStart01** という名前のワークフローとしてデプロイされます。

以上です！

4.2 クイックスタート(パート2)

このクイックスタートのパート2 はパート1 の最後からの続きです。パート2 はデータソースとしての外部 XML ファイルの使用に焦点をあてます。コンボ ボックスのための動的なドロップダウン リストをXML ファイル内のデータから生成する方法、ページへの動的リンクの作成する方法、ファイルへデータを戻して保存する方法などをこのパートで学習することができます。

このチュートリアルは以下のセクション別に整理されており、各セクションはMobileTogether デザイン作成の重要な側面を取り扱っています。

- ファイルからデータをロード
- ソースノードの変更
- シミュレーションの実行
- コンボ ボックス エントリにファイルデータを使用
- データ ファイルをデフォルト ファイルとして設定
- Web ページへの動的リンクの作成
- データをファイルに戻して保存

チュートリアル ファイル

このチュートリアルのファイルは(マイ ドキュメント内のMobileTogether フォルダーにあります: **MobileTogetherDesignerExamples\Tutorials\QuickStart**。Tutorials フォルダーはチュートリアルで参照されるスプラッシュ スクリーン イメージも含みます。これらのイメージを保存して、他の場所にあるXML ファイルを希望する場合使用することができます。

- パート1 で作成したファイルを使用して開始します。代わりに、提供されるデザイン ファイル **QuickStart01.mtd** を使用することもできます。
- 完成されるファイルは、**QuickStart02.mtd** に類似したものです。
- XML データ ファイルは**AltovaProducts.xml** です。
- イメージ ファイルは***.bmp** です。

ファイルからデータをロード

このパートでは以下を行います:

- ソリューションページがロードされた際、ソリューションのためのデータがXML ファイルから取得されることを指定します。チュートリアル フォルダーにあるファイルは、**AltovaProducts.xml** と呼ばれ、リストは下部にあります。パート1 で作成されたデータソースに類似した構成を持ちますが、唯一の違いは新しい要素 **Selection** を持つことです。
- データソースのツリー構造を変更して、XML データファイルの異なる構造にマッチさせます。

XML ファイルのリスト、AltovaProducts.xml

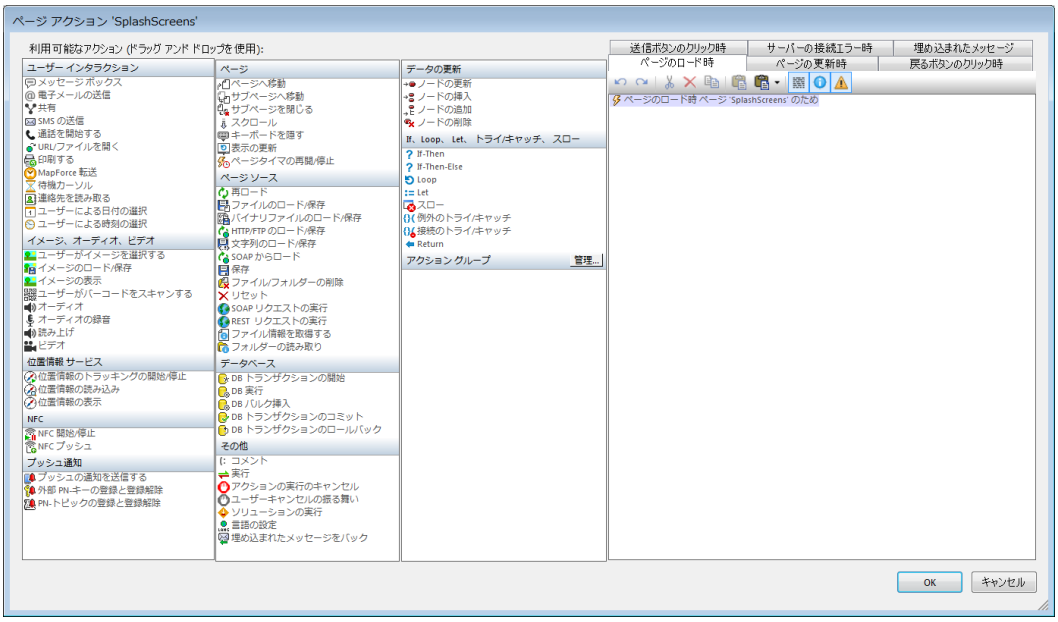
(マイ ドキュメントフォルダー)でMobileTogether フォルダーを探します:
MobileTogetherDesignerExamples\Tutorials\AltovaProducts.xml

```
<?xml Version="1.0" encoding="UTF-8"?>
<Products>
<Selection></Selection>
<Product>XMLSpy</Product>
<Product>MapForce</Product>
<Product>StyleVision</Product>
<Product>MobileTogether</Product>
<Product>DatabaseSpy</Product>
<Product>DiffDog</Product>
<Product>SchemaAgent</Product>
<Product>UModel</Product>
<Product>Authentic</Product>
</Products>
```

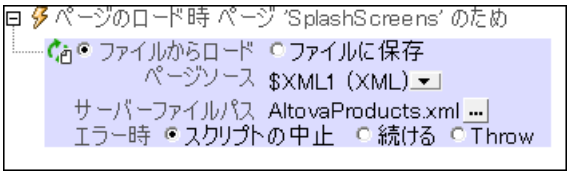
ページのロード時に使用するデータファイルの指定

XML ファイルから抽出されるページデータソースのデータを指定するには、以下を行います:

1. (マイ ドキュメントフォルダー)のMobileTogether フォルダーにある**QuickStart01.mtd** を開きます: **MobileTogetherDesignerExamples\Tutorials**.
2. **[ページ | ページアクション]** をクリックします。ページアクションダイアログが表示されます(*下のスクリーンショット*)。



3. ファイルのロード/*保存* アクションをドラッグして、ページのロード時 イベントのタブにドロップします。
4. ファイルからロードのボタンが選択されていることを確認しページソースが`$XML1` であることを確認してください(*下のスクリーンショット参照*)。
5. **設定** エントリの**[追加ダイアログ]** ボタンをクリックすると、ファイル指定ダイアログが表示されます。
6. 「パスをデザインに対して相対的にする」を選択して、`AltovaProducts.xml` ファイルを参照します。
7. デザインファイルと共にMobileTogether Server にデプロイするか問われます。**[はい]** をクリックします。ページがロードされる際、ファイルはデータファイルとしてセットされデータソース`$XML1` のためにロードされます(*下のスクリーンショット*)。



8. **[OK]** をクリックして終了します。

ページソースのデータ構成の変更

XML データファイルには特別の`Selection` 要素があります。この要素からXML ツリーが、データを得るためには、[ページ ソース] ペインのデータソースのXML ツリーに`Selection` 要素を追加します(*下のスクリーンショットと上のリスト参照*)。 ツリーに `Selection` 要素を`Product` または`Products` を右クリックして追加し、それぞれ**[子の追加]** または**[追加]**、**[要素]** を選択します。 要素をダブルクリックして名前を編集して、要素を`Selection` に名前変更します。

Selection にデフォルトの値を追加しません。この理由は、ページがロードされた際、ページのデータが AltovaProducts.xml より来るようにするためです。これは、このページのページのロード時 イベントのために定義されたアクションです(*上部参照*)。 **Selection** のデフォルトの値を設定すると、このデフォルトの値は、AltovaProducts.xml の**Selection** ノードから得られた値を上書きします。 [ページソース] ペインでデフォルトの値が割り当てられていない場合、ページがロードされると、 **Selection** ノードは空白になります。これは AltovaProducts.xml の**Selection** ノードが空白であるからです (*上部のファイルリストを参照*)。チュートリアルの後半でこのシミュレーションを行います。

ソース ノードの変更

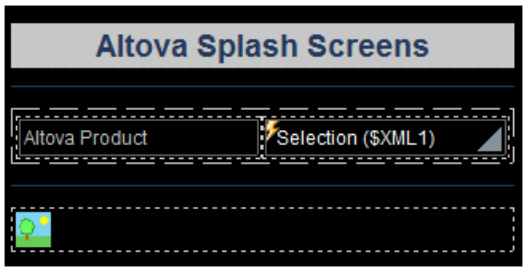
このパートでは以下を行います:

- コンボ ボックスのソース ノードの変更
- イメージのXPath 式の保存

コンボ ボックスのソースノードの変更

各ページコントロールは、データソースの1 つのノードである、**[ソースノード]** または**ページソースリンク)** を持つことができます。リンクは、データソースノードを[ページ ソース] ペインからデザインのコントロールに ドラッグすることで作成できます。 基本的に、ソースノードは、XML ノードからコントロールにデータを送信します。XML ノードのデータがどのようにコントロールに関係するかは、コントロールの種類によります。例えば、: コンボ ボックスのセクションはソースノードを更新します— XML ノード? そしてその値がコンボ ボックスのディスプレイに反映されますが、イメージのソースノードは、イメージのURL を提供します。コントロールをポイントすると、ソースノードの使用方法がポップアップします。例えば、(コンボ ボックスの) XML ノードとして編集する、または、(イメージの) データ元として編集するなど。

コンボ ボックスのソースノード リンクを**Product** から**Selection** に変更します。**Selection** ノードを[ページ ソース] ペインからコンボ ボックスのコントロールへドラッグします(*下のスクリーンショット参照*)。



エンドユーザーのコンボ ボックスの選択を、**Product** 要素ではなく、**Selection** 要素に入れるための手順の理由は以下の通りです:

- 複数の兄弟 **Product** 要素が`AltovaProducts.xml` にはあり、各要素には変更を希望しないデータが含まれています。
- `Product` のページ ソース リンクが、最初の`Product` 要素 (`Product[1]`) である場合、コンボボックスの選択で最初のみが更新されます。
- 最善の解決法はエンドユーザーの選択を別の要素に保管することです。

ページ ソース リンクを**Product** から**Selection** に変更した後、コンボ ボックスの選択は、**Product** ノードではなく、**Selection** ノードを更新します。

イメージ URL の XPath 式の変更

コンボボックスの選択のXML 値が**Selection** ノードに入るため、このノードは、イメージURL を作成するXPath 式に使用される必要があります。デザインで、イメージを選択し、 ([スタイル & プロパティ] ペイン内の) **Image URL** プロパティの**[XPath]** ボタンをクリックします。[XPath/XQuery 条件式の編集] ダイアログが表示されるので、XPath 式を変更し、**Product** を**Selection** と置き換えます。例えば

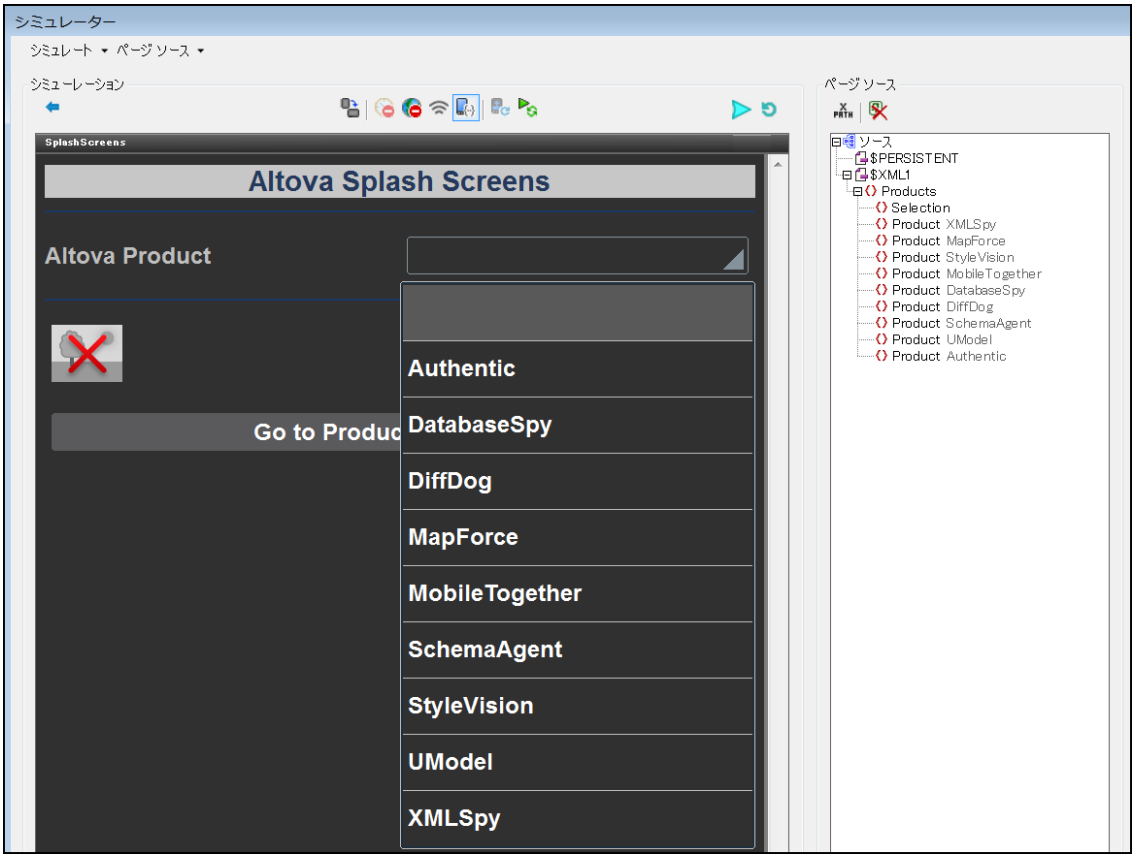
以下を `concat(Product,'bmp')`

以下に変更: `concat(Selection,'bmp')`

このXPath 式は、(**Selection** ノードに保管されている) エンドユーザーのコンボボックスの選択を使用しイメージファイル名を生成します。イメージファイルとデザインファイルは同じフォルダーにあり、XPath 式により生成されるファイル名は、デザインファイルの場所からのイメージへの相対パスです。

シミュレーションの実行

[プロジェクト | ワークフローのシミュレーション] または **[F5]** でシミュレーションを実行することができます。以下が表示されます:



表示	理由
XML データツリー内の **Selection** ノードが空白	値がページ開始時にコードされたファイルの `AltovaProducts.xml` の空白の **Selection** ノードから来るため
コンボボックスが空白	**Selection** ノードが空白のため
ドロップダウンリストに空のエントリが存在する	現在のコンボボックスの選択 (=空白) の結果として空白のエントリが追加されたため。
スプラッシュスクリーンが表示されていない	空白の **Selection** ノードを使用したイメージURL の作成

ドロップダウンリストから製品(例えば `MobileTogether`) を選択すると、その製品のスプラッシュスクリーンが表示されます(下のスクリーンショット)。

理由:

1. コンボボックスの選択 (**`MobileTogether`**) が対応する値 (`mobiletogether`) を**`Selection`** ノードに与えます。
2. **`Selection`** ノードの値は正しくイメージ URL を作成するために使用されます。

今後コンボボックスのドロップダウンリストには空白のエントリは存在しません。これは、ノードが空白ではなく、定義されたコンボボックスのエントリのため、エントリがもう空白ではなくなったからです。

完了すると、**[閉じる]** または **[Esc]** をクリックしてシミュレータを閉じます。

コンボ ボックス エントリにファイル データを使用

このパートでは以下を行います:

- データツリー構造を使用してコンボボックスエントリを生成する
- 変更の効果をテストするためシミュレーションを実行する

XML ファイルのリスト、AltovaProducts.xml

(マイ) ドキュメントフォルダー のMobileTogether フォルダーを探す: **MobileTogetherDesignerExamples\Tutorials\AltovaProducts.xml**.

```
<?xml version="1.0" encoding="UTF-8"?>
<Products>
<Selection></Selection>
<Product>XMLSpy</Product>
<Product>MapForce</Product>
<Product>StyleVision</Product>
<Product>MobileTogether</Product>
<Product>DatabaseSpy</Product>
<Product>DiffDog</Product>
<Product>SchemaAgent</Product>
<Product>UModel</Product>
<Product>Authentic</Product>
</Products>
```

コンボ ボックスエントリの編集

コンボボックスのエントリの編集は以下の通り行います:

1. コンボボックスを選択して、[スタイル & プロパティ] ペインの**コンボ ボックス入力値** のプロパティの追加ダイアログ ボタンをクリックします。コンボボックスの編集ダイアログ(下のスクリーンショット)が表示されます。

2. *条件式の使用*を選択し、[*XML の値とエントリに対して異なるXPath を使用*] を選択します。

3. 上のスクリーンショットに表示されるように、*可視エントリとXML の値*に対するXPath 式を入力します。
4. ダイアログの下部にある[*値を並べ替え*]チェックボックスをチェックして、ディスプレイ時のリストを並べ替えます。
5. **[OK]** をクリックして終了します。

Products ノードはこのページのデフォルトのXPath コンテキスト ノードとして定義されています。 **for** 式のXPath は Products (コンテキスト ノード) のProduct 子ノードに繰り返し適用され、すべての一意かつ明確な値のシーケンスをアルファベット順に返します。この場合の*XML 値*のシーケンスでは、値は一意性をフィルターする前に、値が小文字に変換されます。これらの2 つのシーケンスはドロップダウン リストのエントリで、対応するXML (*可視エントリ*) は、(*XML 値*)の値です。コンボボックスのエントリ作成およびデータのコードにデータソースファイルにツリー構造を使用する利点は、コンボボックスのエントリは、データソース ファイルから動的に生成され、デザインのアイテムのリストのようにコード化されていないからです。従って、新しい製品がファイルに追加されると、ドロップダウン リストのエントリとして自動的に表示されます。

シミュレーションの実行

シミュレーションを実行すると、コンボボックスのエントリがリスト通り実行されます(前のセクション シミュレーションの実行参照)。唯一の違いは、ドロップダウン リストのエントリが**AltovaProducts.xml** のProduct 要素の値であることです(*上のリスト参照*)。ドロップダウン リストのエントリが選択されると、対応する(小文字の) XML 値が**Selection** ノードに入力され、イメージURL が正確に評価されます。

データソースファイルのデータの変更

データソースファイルAltovaProducts.xml (*上のリスト参照*)に以下の2 つの変更を加えます:

- 下のリストに表示される**Selection** ノードに小文字の製品名を追加します。
- ファイルからいくつかのProduct 要素を削除、またはファイルにProduct 要素を追加し、要素の順番を変更します。コンボボックスダイアログ(*上を参照*)、値の並べ替えを選択することにより、選択/選択の解除の効果をテストすることができます。

```
<?xml version="1.0" encoding="UTF-8"?>
<Products>
  <Selection>databasespy</Selection>
  <Product>XMLSpy</Product>
    ...
    ...
  <Product>DatabaseSpy</Product>
</Products>
```

これらの変更を加えた後、ファイルを保存してシミュレーションを実行します。初期のスプラッシュスクリーンは、**Selection** のノード内の製品です。また、コンボボックスのドロップダウン リストにはお空白のエントリがあり、ドロップダウン リストのエントリの数は、XML ファイルの一意のProduct 要素の数と同じです。

データファイルをデフォルトファイルとして設定

このパートでは以下を行います:

- ページデータソースのデフォルトのファイルの指定
- シミュレーションの実行

このセクションのボタン

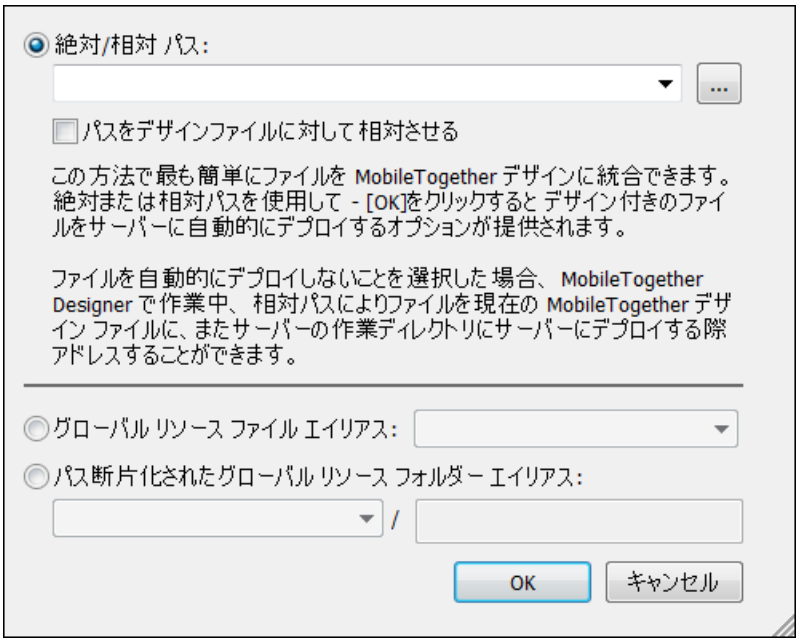
追加ダイアログ

ページ データ ソースのデフォルトのファイルの指定

データソースに入るデータはデータソースのデフォルトファイルを選択することで指定できます。以下を行います:

1. **$XML1** デフォルトファイル(*下のスクリーンショット*) の**[追加ダイアログ]** ボタンをクリックします。

2. *サーバーと絶対 相対 Path*] を選択して、ファイル**AltovaProducts.xml** を参照します。



3. **[OK]** ファイルがデフォルトファイルとして追加され、データはデータソースツリーを作成します。
4. **[ページ | ページアクション]** をクリックしてページ アクション ダイアログを開きます。
5. ページのロード時 タブで、[ファイル エントリからロード] を選択して削除します。これは、`AltovaProducts.xml` ファイルがページの唯一のデータソースのデフォルトのファイルとして指定されているため、 *ファイルからロード アクション* が重複するからです。
6. デフォルトのファイルが使用されているか確認するためシミュレーションを実行します。

Web ページへの動的リンクの作成

このパートでは以下を行います:

- (XPath 式を使用して) Web ページへ動的にリンクするボタンの追加
- シミュレーションの実行

Web ページにリンクするボタンの追加

コンボボックスで選択された製品の説明ページにリンクするボタンを追加します。これには以下を行います:

1. [コントロール] ペインからボタン コントロールをドラッグして、イメージの下にドロップします (*シミュレーターと下のスクリーンショット参照*)。
2. テキスト *製品の説明*を入力します。
3. ボタンを右クリックして、**[ボタンのクリック時 に対するコントロール アクション]** を選択します。
4. 表示されるアクション ダイアログで (*下のスクリーンショット*) [*URL/ ファイル を開く*] アクションを ボタンのクリック時 タブにドラッグし、下のスクリーンショットに表示されるように、クリック時 および 長いクリック時 イベントにドロップします。双方のクリックの型にアクションが指定されていないため、2 つのうちのどちらかのイベントがトリガーされたのちに次の (追加) アクションが実行されるように、*URL を開く*アクションが実行されます。

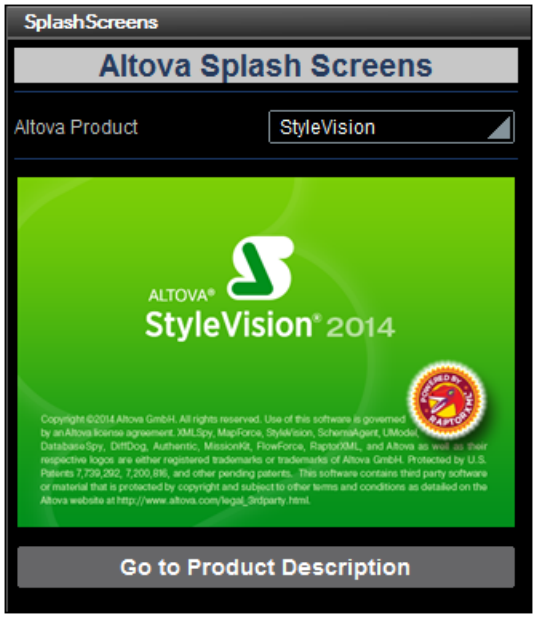
表示されるアクション ダイアログで (*下のスクリーンショット*) [*URL を開く*] アクションを ボタンのクリック時 タブにドラッグします。

5. **[XPath]** ボタンをクリックすると、[XPath/XQuery 条件式の編集] ダイアログが表示されます。以下の XPath 式を入力します: `concat('http://www.altova.com/', Selection, '.html')`

6. **[OK]** をクリックして終了し、ファイルを保存します。

シミュレーションの実行

[F5] または **[プロジェクト | ワークフローのシミュレーション]** をクリックしてシミュレーションを実行します。シミュレーションが開始すると、コンボボックスの製品を選択して **[製品の説明]** ボタンをクリックします(*下のスクリーンショット参照*)。これによりAltova Web サイトの製品の説明ページへ移動します。


SplashScreens
Altova Splash Screens
Altova Product
StyleVision
ALTOVA®
StyleVision® 2014
Go to Product Description

データをファイルに戻して保存

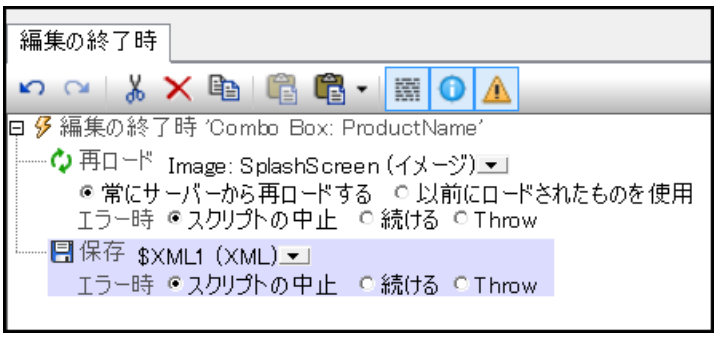
このパートでは以下を行います:

- コントロールアクションを使用して変更されたデータをファイルに戻す。
- *ファイルに保存 アクション*をテストするシミュレーションの実行

コンボ ボックス編集後データをファイルに保存する

コンボ ボックスの編集が行われるとファイルに戻して保存することを指定できます。コンボボックスのソースノードは `Selection` 要素で、コンボボックスの選択はこの要素に保存されます。コンボ ボックスには、`Selection` へのページ ソースリンクが存在するため、コンボ ボックスの選択はこの要素に保存されます。変更が デフォルトファイルの`Selection` 要素に保存されるように指定するには、コンボ ボックスの **編集の終了時** イベントに *保存アクション*を追加します。これは以下の通り行います。

1. コンボボックスを右クリックして、**[編集の終了時 に対するコントロール アクション]** をコンテキストメニューから選択します。
2. これにより、すでにイメージをターゲットにした*再ロード アクション*をもつコンボボックスのコントロールアクションダイアログが表示されます。
3. *保存 アクション*をドラッグして、*再ロード アクション*の下にドロップします(*下のスクリーンショット参照*)。



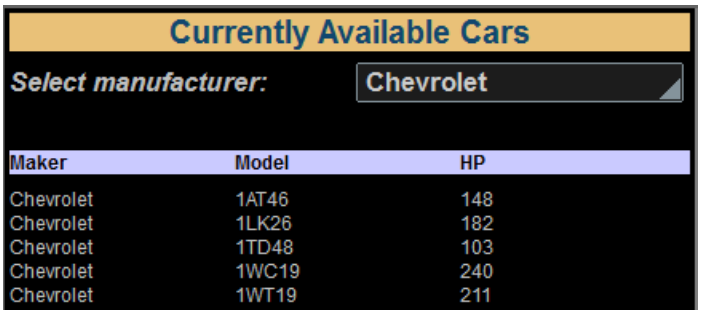
4. **[OK]** をクリックして終了して、ファイルを保存します。
5. デフォルトのファイルに変更が保存されたか確認するには、デフォルト ファイルをエディターで開き、シミュレーションを実行し、コンボ ボックスのエントリを選択し、エディターにデフォルト ファイルを再ロードします。新しいコンボボックスの選択が、デフォルトファイルの`Selection` 要素として表示されます。

メモ　ファイルをデフォルト ファイル以外で保存することを希望する場合、(*保存 アクション*の代わりに) *ファイルに保存アクション*を使用します。Web ページの保存を希望する場合は、*HTTP/FTP に保存アクション*を使用します。(この場合、ユーザーがHTTP URL でページを変更することを許可する認証の詳細を提供する必要があります。)

以上です！

4.3 シンプルなデータベース

このチュートリアルでは以下について説明されています: (i) DB データソースを基にしたシンプルなデザインの作成、および (ii) ユーザーの選択をベースにしたDB 記録のロードと表示。デザインのためのデータはMicrosoft Access データベース内に保管されている車種のデータベースをソースにしています。ソリューション内では、ユーザーは製造業者を選択することができます。自動車の製造業者はテーブル内に表示されています (*下のスクリーンショット参照*)。



チュートリアルのファイル

*マイドキュメント*MobileTogether フォルダー内にこのチュートリアルに必要なファイルは存在しています: **`MobileTogetherDesignerExamples\Tutorials\Databases`**。

- 車種の戻るの記録を含んでいるAccess データベースは自動車の製造業者のリストにより構成されています: **`MyCars.mdb`**
- デザインファイルは、**`SimpleDatabase.mtd`** と同様の結果を表示します。

チュートリアルの構成

このチュートリアルは以下のように構成されています:

- DB データソース
- ユーザーの入力を保管するための永続的なツリー
- ユーザーの選択に基づきDB データをロードする

DB データソース

このセクションでは、新規のMobileTogether デザインが作成されます。Microsoft Access データベース `MyCars.mdb` をデータソースとして使用しています。

下のスクリーンショット内に
DB の構造が表示されています。(DB の最初の記録の一部のみがスクリーンショットに表示されていることにご注意してください)

1	Manufacturer	Model	Fuel	Cylinder	Horsepower	YearFrom	YearTill	Source	Series
2	BMW	550i xDrive Gran Turismo	Gas	8	448	2013	2014	US	5
3	BMW	550i xDrive GranTurismo	Gas	8	400	2010	2012	US	5
4	BMW	640i Convertible	Gas	6	300	2012	2014	US	6
5	BMW	640i Coupe	Gas	6	300	2012	2014	US	6
6	BMW	640i xDrive Coupe	Gas	6	300	2014	2014	US	6
7	BMW	645CI	Gas		325	2004	2005	US	6
8	BMW	645CI CONVERTIBLE	Gas		325	2004	2005	US	6
9	BMW	650CI	Gas	8	360	2006	2010	US	6

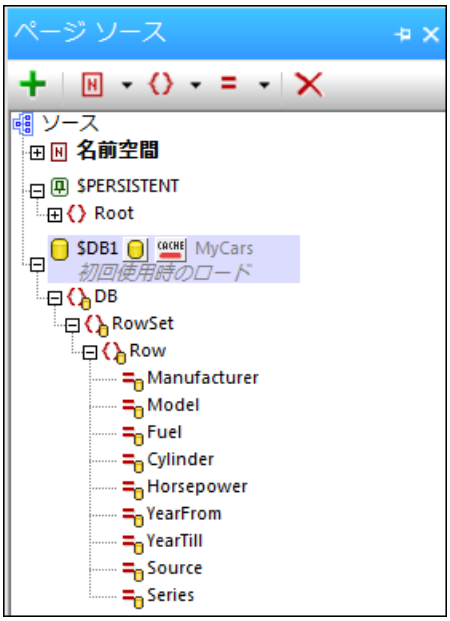
デザイン内で DB データソースをセットアップする

DB データソースのセットアップは、以下のとおりです:

1. 新規のデザインを作成します (**ファイル | 新規作成**)
2. ページソースペイン内の**ソースの追加**アイコンをクリックしてデータソースを作成します。
3. 表示されるページソースの追加ダイアログから、「新規のDB 構造」を選択して、**「次へ」**をクリックします。
4. ダイアログの次のスクリーン内で、デフォルトの設定を保持し、**「完了」**をクリックします。
5. 表示される接続ウィザードダイアログから、*Microsoft Access (ADO)*を選択し、**「次へ」**をクリックします。
6. 次のスクリーンから「既存のデータベースを使用する」を選択して、`MyCars.mdb` DB を参照し、**「接続」**をクリックします。
7. 接続が作成され、データベースオブジェクトの挿入ダイアログが表示されます (*下のスクリーンショット*) 。

`Cars` テーブルをスクリーンショット内のとおり表示されるように選択し、**OK** をクリックします。

`Cars` テーブルは、デザインページのDB データソースとして作成され、ページソースペイン内で表示されます (*下のスクリーンショット*)。



次のセクションでは、デザイン内でコンボボックスが作成され、エンドユーザーが表示される自動車の製造業社名を選択することができます。

ユーザーの入力を保管するための永続的なツリー

エンドユーザーの選択を反映するには、自動車製造業者名がリストされているコンボボックスを使用します。**$PERSISTENT** ツリーは、ノード内にユーザーの選択が保管できるように作成される必要があります。このノードの値は、DB からのデータの選択が後に製造業者の自動車を表示するために使用されます (次を参照：*次のセクション*)。

$PERSISTENT ツリーとコンボボックスについてこのセクションでは説明されています。

$PERSISTENT ツリーを作成する

デザインが作成されると、**$PERSISTENT** ツリーが自動的にページソースペイン内で **Root** と呼ばれるルート(またはドキュメント) と共に作成されます(*下のスクリーンショットを参照*)。ユーザーの選択を保管するには、単一のノードのみが必要で、子要素または子属性要素のみが必要とされます。**Manufacturer** と呼ばれる属性ノードを **Root** 要素に以下のように追加します。**$PERSISTENT** ツリー内から **Root** を選択し、ツールバー内の **属性の追加** アイコンをクリックして、**子属性の追加** を選択します。**Manufacturer** に属性の名前を変更します(*下のスクリーンショットを参照*)。

属性 **Manufacturer** には空白の文字列 (**""**) が開始の値として存在しています (*スクリーンショットを参照*)。これにより、**$PERSISTENT** ツリーがリセットされると、**Manufacturer** 属性が空白の文字列である値を持つようになります。

ユーザーの選択のためにコンボボックスを作成する

ページのデザインが以下のようになるようにデザインします:

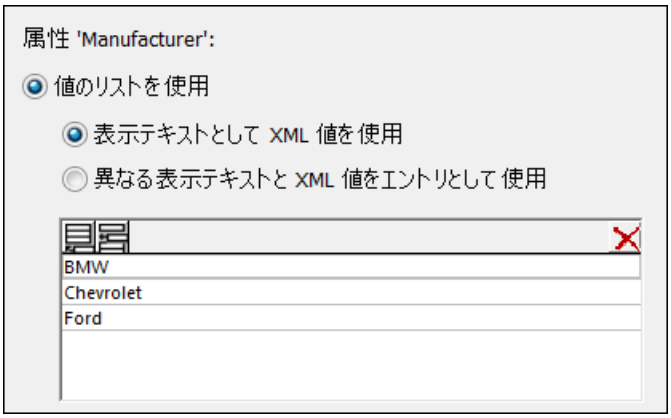
この時点でのデザインは以下を含んでいますデザイン:

- ラベル コントロールにはこのページのタイトルが表示されます。
- 静的テーブルコントロールには2つのセルが含まれています。
- 左側のテーブルセルには、文字 (*Select manufacturer*) を含んだラベル が含まれています。
- 右側のテーブルセルには、自動車製造業者のリストからエンドユーザーが選択するオプションが表示されるコンボボックスが含まれています。

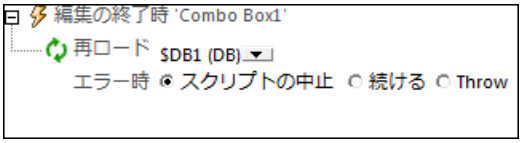
ラベル、テーブル、とコンボボックス コントロールを追加するには、コントロールのペインから対応するコントロールをドラッグします。コントロールのプロパティに関する詳細は、対応するコントロールの詳細を参照してください。

コンボボックスの設定
コンボボックスの設定には3つの重要な点があります:

- コンボボックスが表示する値、および、ユーザーの選択として保管される値。これらの値を定義するには、コンボボックスをダブルクリックします。表示されるコンボボックスの編集ダイアログ内で(下のスクリーンショット)、スクリーンショット内で表示されているように、値を追加します。これらの値は、コンボボックスのドロップダウンリストのアイテム、および、対応するXML 値となります。



- コンボボックスの選択を保管するためのデータソースリンク。**$PERSISTENT/Root/Manufacturer** 属性をページソースペインからコンボボックスにドラッグします。これにより、データソースリンクが構築されます。この結果、エンドユーザーがコンボボックス内のドロップダウンリストのアイテムを選択すると、選択されたアイテムの対応するXML 値が**$PERSISTENT/Root/Manufacturer** ノードに保存されます。
- コンボボックス内の新規のドロップダウンリストアイテムをエンドユーザーが選択すると、**$DB1** ツリーが再ロードされます(これは、DB データソースからロードされたデータがコンボボックスの選択をベースに選択される必要があるからです。コンボボックスの値が変更されると、データの選択も変更される必要があります。コンボボックスの選択をベースにするDB データの選択は、次のセクションで更に詳しく説明されています。**$DB1** ツリーを再ロードするには、再ロードアクションをコンボボックスの編集の完了時イベントに追加します。これはコンボボックスを右クリックして、**編集の完了時のためのコントロールアクション**を選択して行います。表示されるアクションダイアログでは、下のスクリーンショットに示されるように再ロードアクションを追加します。



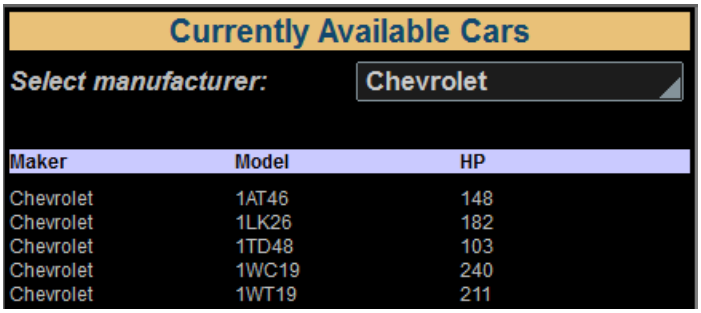
次に、選択された自動車製造業者の自動車名をDB からソリューションにロードし、テーブルに自動車名を表示します。

ユーザーの選択に基づき DB データをロードする

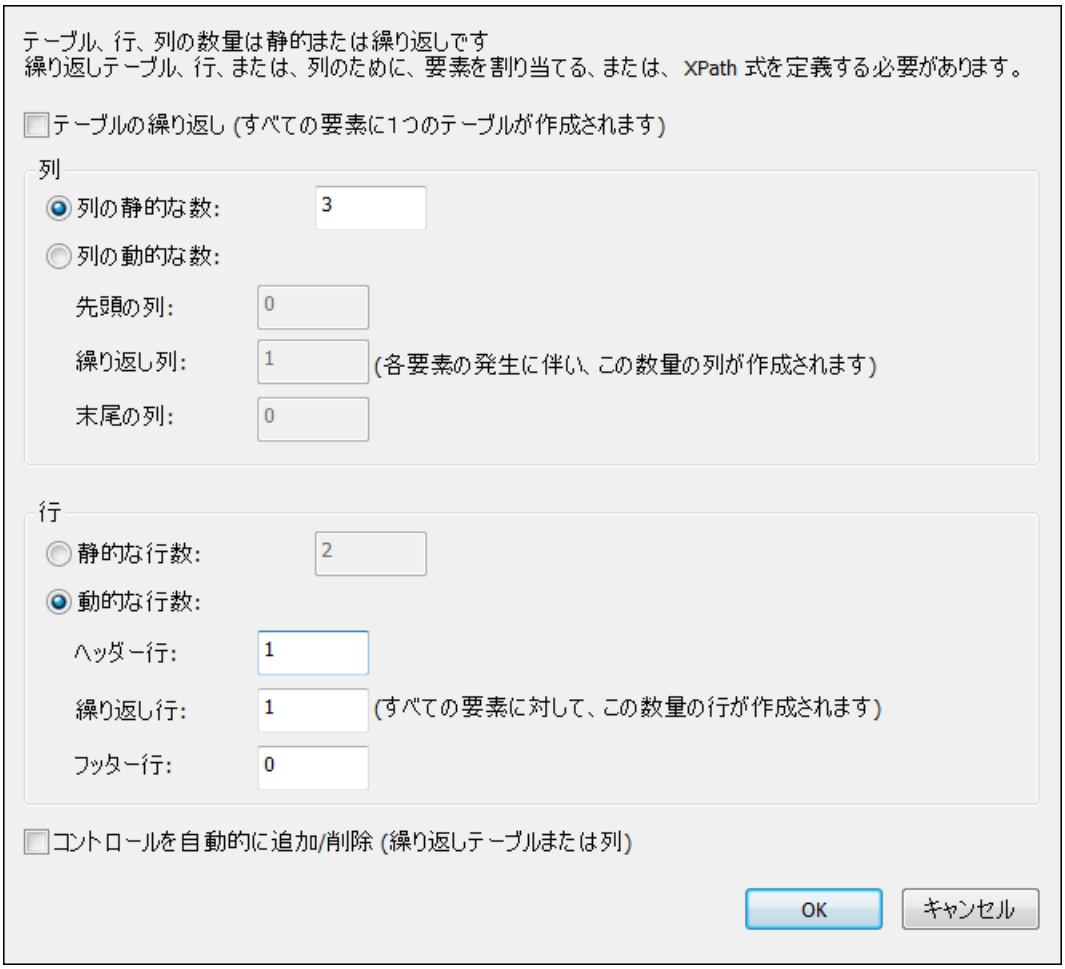
車種のDB レコードはロードされ、エンドユーザーにより選択された自動車製造業者に従い(コンボボックス内で)テーブル内に表示されます。セットアップの方法に関しては、前のセクションを参照してください。このセクションでは、記録を表示するためにテーブルの作成、**$DB1** ツリーにロードするレコードの選択方法、および表示方法についての説明が行われます。

レコードの表示のためのテーブル

DB 内では、各レコード(または行) は異なる車種に対応しています。各テーブル行がDB 行に対応するデザインへ繰り返し行を使用したテーブルを追加することがこれらのレコードを表示するために最適なアプローチです。テーブルには3つの列 (自動車の製造業者、モデル、自動車の馬力) が存在し、また、ヘッダー行も存在します (*下のスクリーンショット参照*)。ユーザーがコンボボックス内の選択を変更すると、新しい自動車製造業者の車種がロードされ、表示されます。



上記のプロパティを持つテーブルを挿入するには、下のスクリーンショット内で表示されるとおりの仕様を持つ、テーブルコントロールを追加します。



テーブル内で以下を行います:

- DB 行へのテーブルの主要繰り返し行。要素 **$DB1/DB/RowSet/Row** をテーブルの繰り返し行 アイコンへドラッグして行います。**$DB1** ツリー内の各 (レコード) 行に対して、テーブル内に対応する行が存在するように指定します。
- ラベル コントロールを、3つそれぞれのヘッダーセルにドラッグアンドドロップして、3つの列ヘッダーに適切なテキストを与えます(*下のスクリーンショット参照*)。
- 3つの列のコンテンツに関しては、それぞれ、ページソースペインから**行** 要素の次の属性ノードをドロップし、以下のラベルコントロールを作成します: **Manufacturer**、**Model**、**Horsepower** (*下のスクリーンショット参照*)。

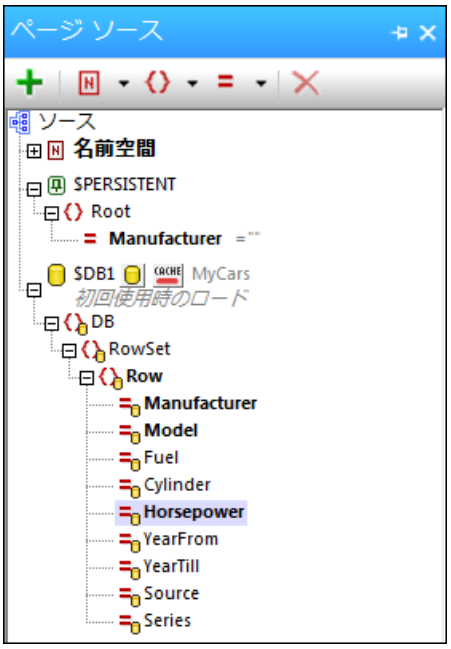
完了すると、デザイン内のテーブルは以下のようになります:

ロードして表示するために DB レコードを選択する

作成されたテーブルは、DB からロードされた全ての記録を表示します。**$DB1** ツリーが現在定義されているように、**DB 内の全ての記録が**ロードされます。**全ての自動車の製造業者**の車種が表示されます。しかしながら、このチュートリアルの目的は、コンボボックス内でユーザーが選択する自動車製造者の車種のみ**ロード**し、表示することです。

選択された自動車製造業者の車種のみをロードするには、**$DB1** ツリー上で**SELECT** ステートメントを作成します。以下を行います:

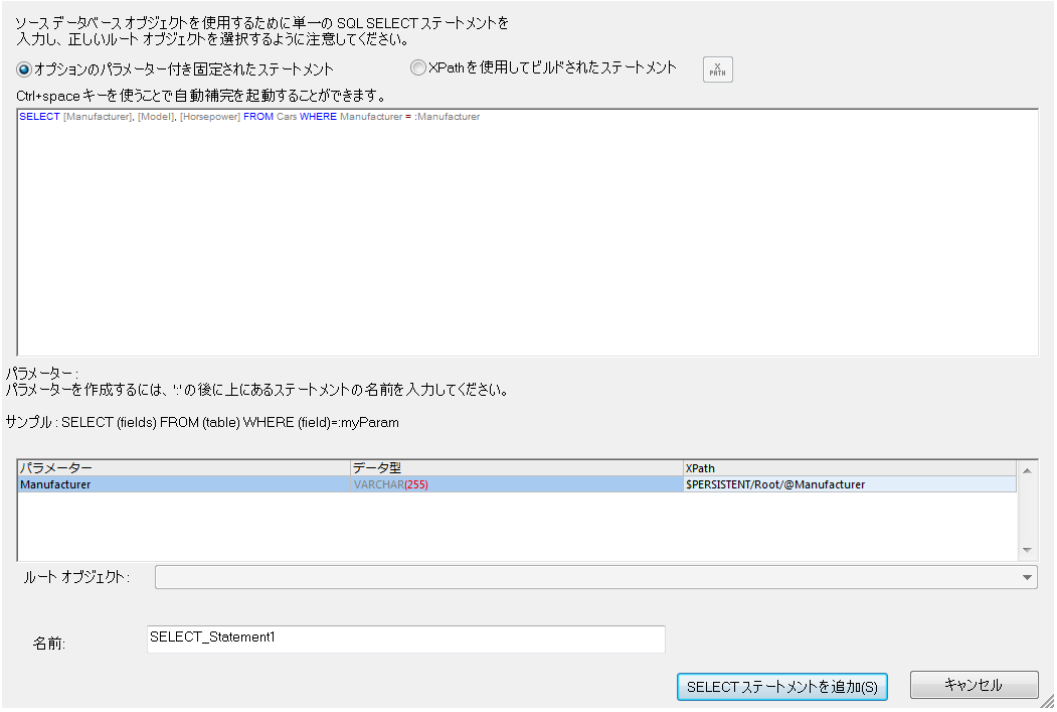
1. **$DB1** ツリーレジェンドの右の DB アイコンをクリックします (*下のスクリーンショット内のカーソルを参照してください*)。



2. 表示されるデータベースオブジェクトの挿入ダイアログ内で、**SELECT ステートメントを追加 編集する**をクリックします。

3. SQL SELECT ステートメント ダイアログに、以下の**SELECT** ステートメントを入力します: **SELECT** [Manufacturer], [Model], [Horsepower] **FROM** Cars **WHERE** Manufacturer = :Manufacturer。このステートメントは、Manufacturer パラメーターにより与えられた値に**Manufacturer** フィールドが一致する箇所のこれらのレコードの**Manufacturer**、**Model**、および**Horsepower** フィールドのみ

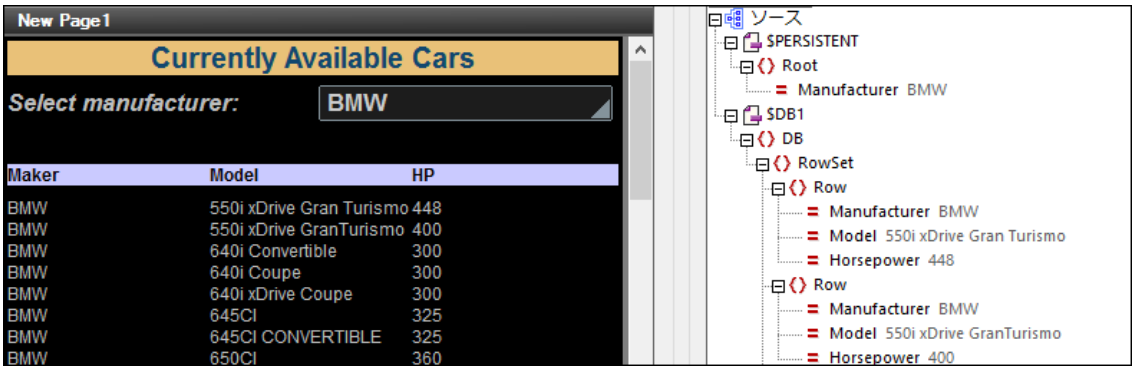
を選択し、ロードします。**SELECT** ステートメントにはパラメーター (**:Manufacturer**) が含まれており、パラメーターの定義のための行がダイアログの下のペインに自動的に追加されます (*スクリーンショットを参照*)。



4. **:Manufacturer** パラメーター値の定義として、次の XPath 式を入力します: **`$PERSISTENT/Root/@Manufacturer`**。これにより、SQL **SELECT** ステートメントがユーザーにより選択される **`$PERSISTENT/Root/@Manufacturer`** ノード内の現在の値に **Manufacturer** フィールドが一致する DB レコードを選択するように設定します。

`$DB1` データソース上で定義されている **SELECT** ステートメントが DB からデータを選択し、**`$DB1`** ツリーに**ロード**することは重要であることに注意してください。作動のからくりは以下のとおりです:

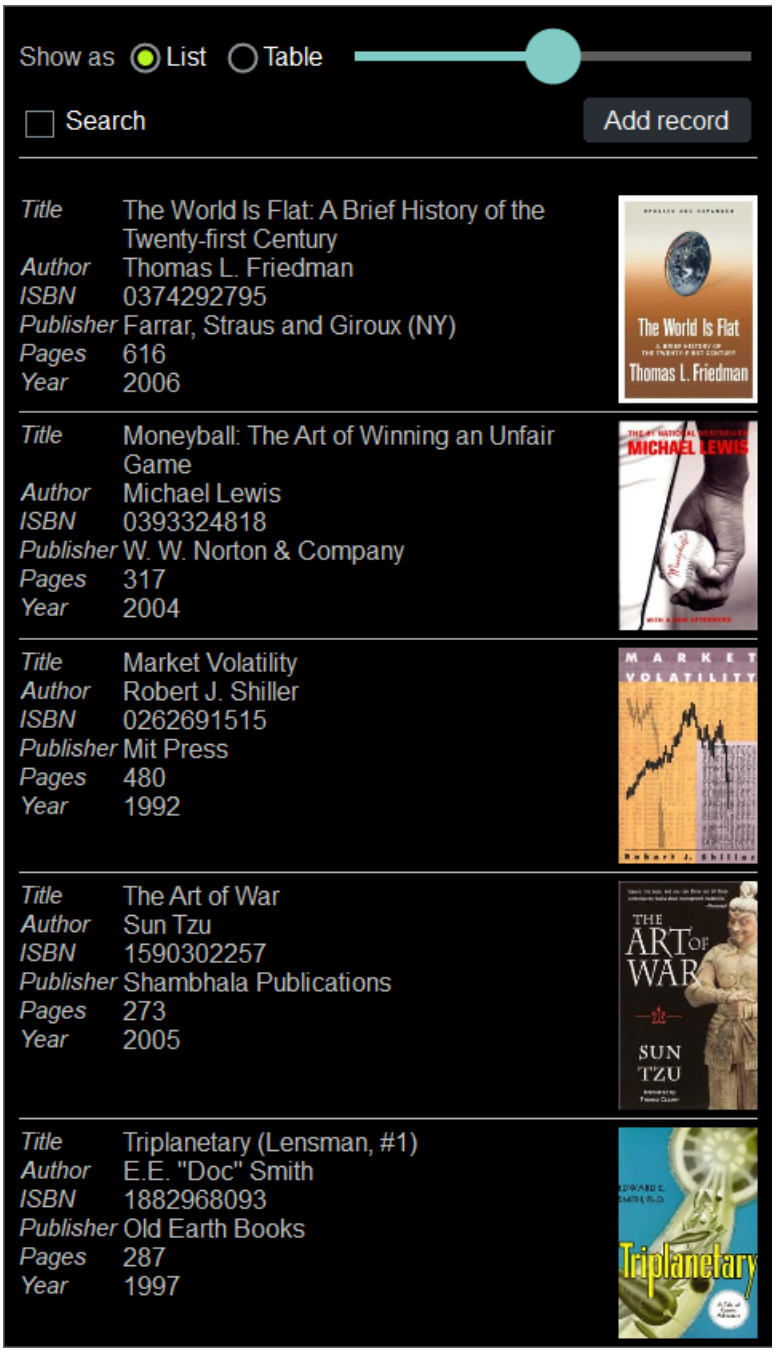
- コンボボックス内の値をユーザーが変更すると、**編集の完了時** イベントの再ロードアクションは、**`$DB1`** ツリーを再ロードします (コンボボックスの定義を参照してください)。
- **`$DB1`** ツリーは、**SELECT** ステートメントを前提にしてロードされます。
- この **SELECT** ステートメントは、新規のユーザーの選択を保管する **`$PERSISTENT/Root/@Manufacturer`** 属性の値としてを持つパラメーターを使用しています。パラメーターにより、ユーザーが選択する自動車製造業者と同じ値を持つ **Manufacturer** フィールド値を持つこれらの DB 行のみが選択されるようになります。
- **`$DB1`** ツリーにロードされる**すべての DB 行**は、テーブルに表示されます。しかし、ユーザーによる選択に対応する行**のみ**が **`$DB1`** ツリーにロードされ、テーブルはユーザーによる選択に対応する車種のみを表示します (*下のスクリーンショット参照*)。


New Page1
Currently Available Cars
Select manufacturer:
BMW
Maker Model HP
BMW 550i xDrive Gran Turismo 448
BMW 550i xDrive GranTurismo 400
BMW 640i Convertible 300
BMW 640i Coupe 300
BMW 640i xDrive Coupe 300
BMW 645CI 325
BMW 645CI CONVERTIBLE 325
BMW 650CI 360
ソース
$PERSISTENT
Root
Manufacturer BMW
$DB1
DB
RowSet
Row
Manufacturer BMW
Model 550i xDrive Gran Turismo
Horsepower 448
Row
Manufacturer BMW
Model 550i xDrive GranTurismo
Horsepower 400

4.4 詳細データベース

このチュートリアルではDB ベースのデザインの主要な機能について説明されています (*下のスクリーンショット*)。SQLite DB は書籍のカタログです。デザインで以下を行うことができます:

- 書籍のカバーのイメージと共に(書籍) レコードを表示することができます。
- (DB 内のカバーイメージの変更を含む) レコードを編集することができます。
- (カバーイメージを含む) 新規のレコードを追加することができます。
- DB を検索し、検索の結果を表示することができます。



チュートリアルのファイル

このチュートリアルに必要なファイルは マイドキュメントMobileTogether フォルダーで見つけることができます: **`MobileTogetherDesignerExamples\Tutorials\Databases`**.

- 書籍のレコードを含むSQLite データベース **`Books.sqlite`**
- デザインファイル: **`Books.mtd`**

これらのファイルをフォルダーにコピーし、ないでデザインファイルを開き、MobileTogether Designer、MobileTogether Designer 内でシミュレーションを実行することができます。

チュートリアルの構造

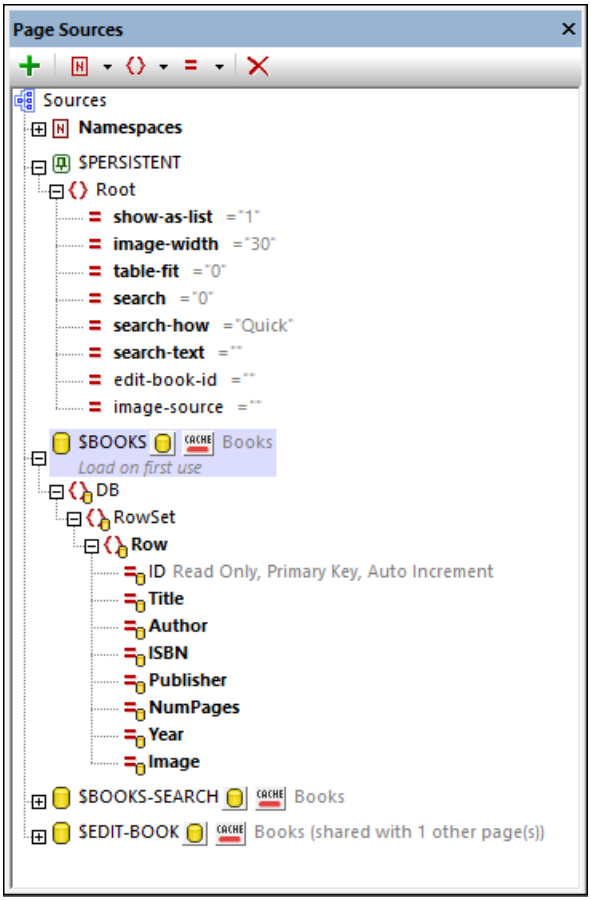
このチュートリアルは次のセクションに整理されています:

- ページソース DB とPersistent
- 可視性の管理
- DB 内のイメージ
- レコードの編集
- レコードの追加

ページソース: DB と Persistent

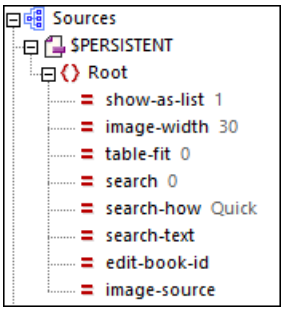
デザインには以下の2つのページが存在します: (i) *Main Page* という名前のトップページ (ii) *Edit Book* という名前のサブページ

- *Main Page* は以下の種類の書籍を表示します (i) フィルター無し、または (ii) 検索用語に従いフィルターされた書籍。*Main Page* 上では、エンドユーザーは以下を行うことができます: (i) 編集のためにレコードをクリックします。(ii) 新規のレコードを追加するためにボタンをクリックします。これらのクリック両方により*Edit Book* サブページにアクセスすることができます。*Main Page* には以下のページソースが4つ存在します (*下のスクリーンショット参照*): **$PERSISTENT**、**$BOOKS**、**$BOOKS-SEARCH**、**$EDIT-BOOK**。
- *Edit Book* サブページには、レコードのフィールドが含まれているテンプレートが含まれています。レコードが追加されると、テンプレートは空になります。*Edit Book* には2つのページソースが存在します: **$PERSISTENT** と **$EDIT-BOOK**。*Main Page* を持つ**$EDIT-BOOK** を共有しています。



$PERSISTENT

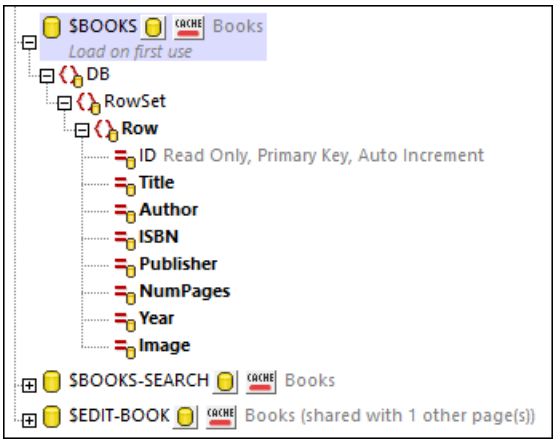
$PERSISTENT ページソースは動的なデータを保管するために使用されます。**Root** という名前の単一の要素が存在しており、複数の属性を持ちます (*下のスクリーンショット参照*)。ユーザーアクションによりその値は決定されます。例えば、属性 **show-as-list** はユーザーが *リスト* または *テーブル* ラジオボタンを選択するにより、**1** または **0** の値を取ります。

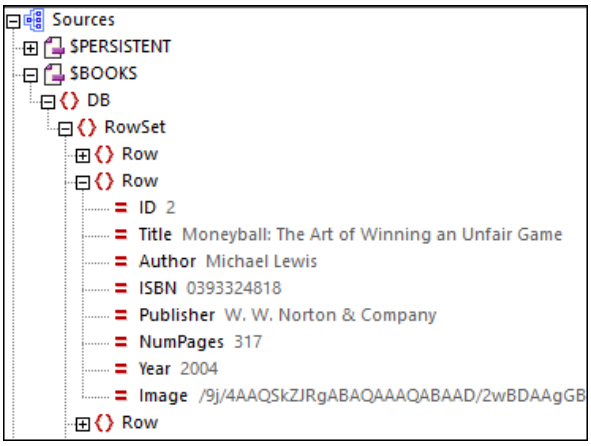


$PERSISTENT ページソースの値は、デザインの異なるアスペクトを条件付きで決定するために使用されます。 例えば、**show-as-list** の値が**1** の場合、書籍のカタログはテーブルとしてではなくリストとして表示されます。

$BOOKS

$BOOKS ページソースは、**Books** テーブルをSQLite DB **Books.sqlite** から選択します。各書籍のレコードは **Row** 要素の属性に対応する各 DB 行のフィールド(または列) と共にページソース内の**Row** 要素に対応します。これは、下のスクリーンショット内のデザイン(左) とシミュレーション(右)で表示されています。

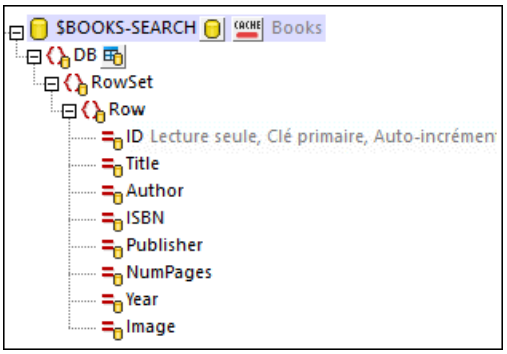


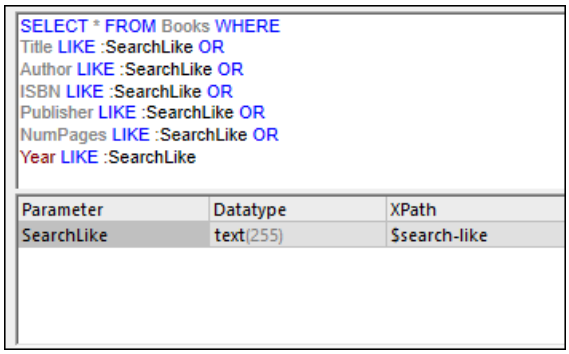


$BOOKS ページソースは書籍のカタログ全体を表示するために使用されます。

$BOOKS-SEARCH

$BOOKS-SEARCH ページソース(*左下のスクリーンショット*) は、SQLite DB **Books.sqlite** から**Books** テーブルをフィルターし、部分的に検索予後に一致するレコード内のフィールドの値を選択します。左下のスクリーンショットでは、ルート要素 **DB**. 上のページソース内でフィルターが定義されていることに注意してください。スクリーンショットで表示されるフィルターのSQL **SELECT** ステートメントは右下に表示されるとおりです。**SELECT** ステートメント内の:**SearchLike** パラメーターは、エンドユーザーにより入力される検索用語に対して解決されます。

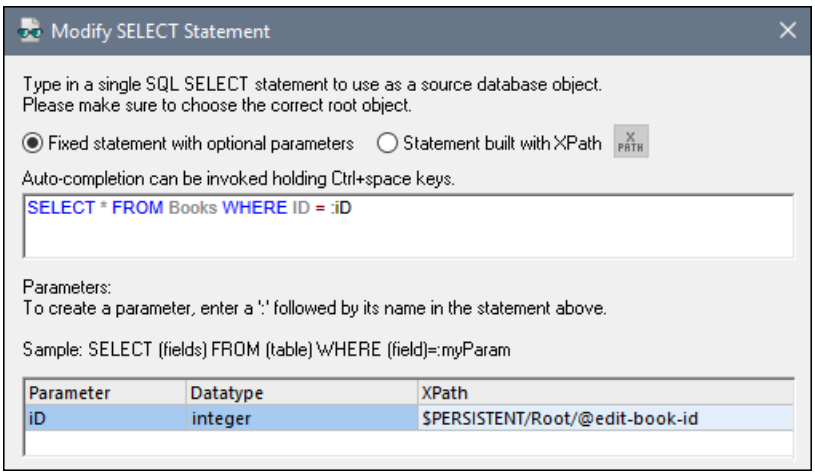




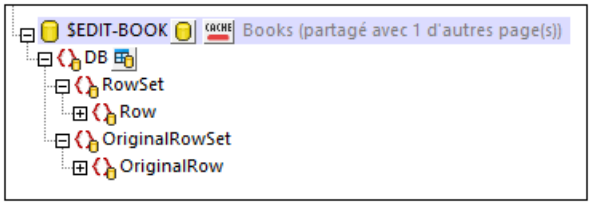
検索中にトップページの表示内で、**$BOOKS-SEARCH** ツリーは、**$BOOKS** ツリーを置き換えます。

$EDIT-BOOK

$EDIT-BOOK ページソースは、**Books.sqlite** から**Books** テーブルをフィルターする**$BOOKS-SEARCH** ページソースに類似しています。フィルターのSQL **SELECT** ステートメントは下のスクリーンショットで表示されています。**SELECT** ステートメント内の**:iD** パラメーターは、編集、または、追加される書籍のレコードのID である ノード**$PERSISTENT/Root/@edit-book-id** 内の値に対して解決されます。



$EDIT-BOOK ページソースは、*Edit Book* サブページ内の編集テンプレートのデータリンクとして使用されます。**RowSet** が現在の(編集済み) データを保持中に元のデータを保つため **$EDIT-BOOK** ページソースは、(コンテキストメニューを使用して作成される) **OriginalRowSet** ノードも含んでいます (*下のスクリーンショット参照*)。


SEDIT-BOOK
Books (partagé avec 1 d'autres page(s))
DB
RowSet
Row
OriginalRowSet
OriginalRow

可視性の管理

Main Page は、テーブルとテーブル列の可視性プロパティを使用して、エンドユーザーアクションに従いテーブルコンポーネントの表示をオンとオフに切り替えることにより、レイアウトを管理します。

セット内の1つのテーブルのユーザーを優先した表示

Main Page のデザインは、上下の3つのテーブルにより構成されています:

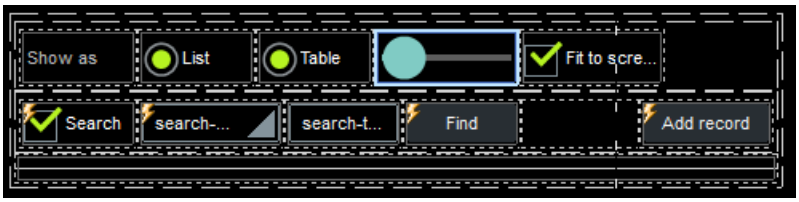
- データの表示を決定するユーザーインタラクションのためのコントロールをホールドする。
- エンドユーザーが *リスト* ラジオボタンを選択すると表示される、書籍カタログをリストとして表示する。
- エンドユーザーが *テーブル* ラジオボタンを選択すると表示される、書籍カタログをテーブルとして表示する。

書籍のカタログはリスト、または、テーブルとして表示されますが、リストとテーブルの両方としては表示されません。これは、次のメカニズムを使用して定義されます:

1. (*リスト* または *テーブル*) ラジオボタンが選択されると、ノード **$PERSISTENT/Root/@show-as-list** は、*List のために* **1** *Table のために* **0** の値を受け取ります。
2. (デザイン内で) 書籍カタログをリストとして表示するテーブルの **Visible** プロパティは次に設定されます: $PERSISTENT/Root/@show-as-list = **1**. ラジオボタンの選択が *リスト* の場合のみテーブル画表示されます。
3. 書籍カタログをテーブルとして表示するテーブルの **Visible** プロパティは次に設定されます: $PERSISTENT/Root/@show-as-list = **0**. ラジオボタンの選択が *テーブル* の場合のみこのテーブルの可視性はオンに切り替えられます。

テーブル列の条件付き表示

Main Page の一番上のテーブルは(*下のスクリーンショット*) は2つの一行からなるテーブルにより構成されます。これら2つのテーブル内での複数の列の表示は、特定のユーザーの選択により条件付けられています。この設定を確認するには、行内をクリックして、そのテーブル列の **Visible** プロパティの XPath 式を確認してください。



これらの列の可視性の条件の一部は、以下のとおりです:

- *リスト* ラジオボタンが選択されている場合、水平スライダーの列が表示されます (*上記で水色で表示されています*) *テーブル* ラジオボタンが選択されている場合、
- *Fit to Screen* チェックボックスの列は、*テーブル* ラジオボタンが選択されている場合、表示され、*リスト* ラジオボタンが選択されている場合表示されません。
- *Search How* コンボボックスの列と、編集フィールドは、*検索* チェックボックスが選択されている場合表示され、選択されていない場合、表示されません。
- チェックボックスが選択されていない場合、*レコードの追加* ボタンは表示され、選択されている場合表示されません。

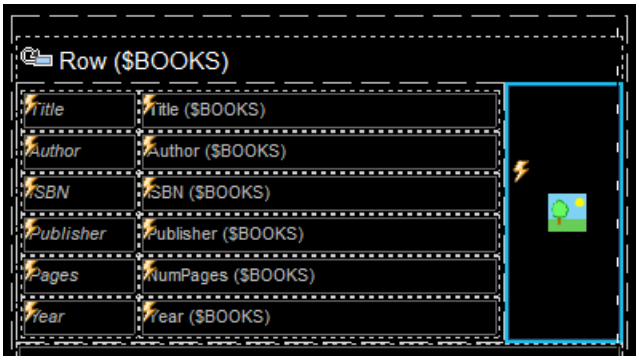
DB 内のイメージ

Base64 エンコード テキスト文字列として DB とページソース ノード内に保管されることに注意してください。

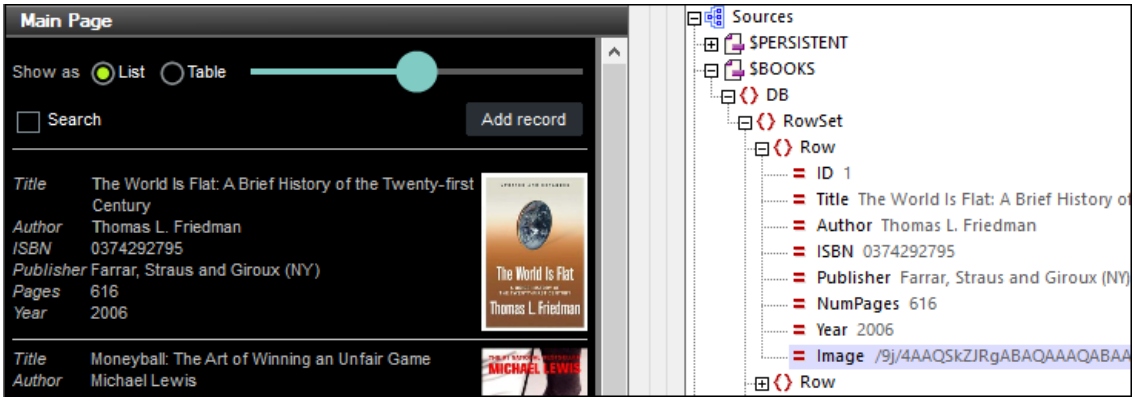
メモ サーバー上のページソースノードをファイルがパススルーしない場合イメージをイメージファイル(PNG or JPEG などの) から直接クライアントにエコードすることもできます。詳細に関しては、 (i) イメージコントロール と(ii) セクション イメージを参照してください。

DB からイメージを表示する方法

Base64 エンコード文字列としてイメージはデータベース保管されるため、表示は簡単です: 書籍カタログのリストの表示内のイメージコントロールは以下を決定します (*下のスクリーンショット参照*) (i) コントロールの `Image Source Type` プロパティの値としての **`base64`** および(ii) Base64 エンコード文字列を含むノード

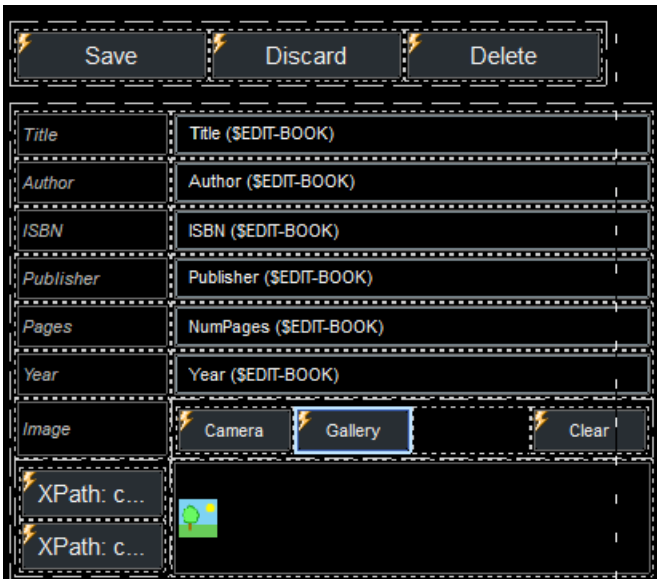


(*下に表示される*) シミュレーションでは、(*ハイライトされている*) Base64 エンコード文字列 はページソースツリー内で表示されています。

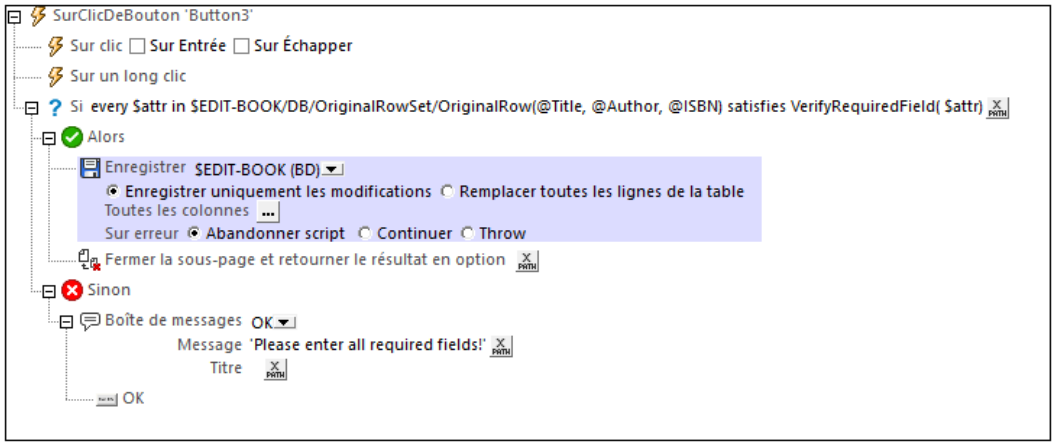


DB にイメージを保存する方法

DB にイメージを保存するには、以下のステップに従ってください。Base64 エンコード文字列としてイメージは DB に保存されなければならないことに注意してください。 *Edit Book* サブページのアクションを検索してこれらのステップに従ってください。

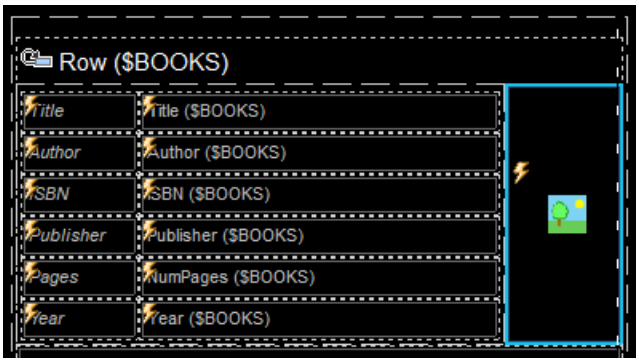


1. デバイスのギャラリーから選択されたイメージ、または、デバイスのカメラにより撮影された写真。ユーザーによるイメージの選択 アクションがこのステップで使用されます。**ギャラリー** と**カメラ** ボタンのアクションを参照してください(*上のスクリーンショット参照*)。
2. イメージコントロールの左側のボタンは`mt-transform-image` 機能を使用して、イメージを逆時計進行方向、または、進行方向に90度回転することができます。
3. Base64 エンコード文字列として`$EDIT-BOOK` ページソースのノードにイメージは保存されます。ユーザーによるイメージの選択 アクションは、自動的にイメージデータをBase64 エンコード文字列として保存します。
4. ページソース ノード内にイメージデータが保存されると、DB にノードが保存されます。デザイン内では、これはページソース(`$EDIT-BOOK`) をDB に保存することにより行うことができます。レコードが編集されている既存のレコードの場合、既存のレコードが上書きされます。 **「保存」** ボタンのアクションを参照してください(*下のスクリーンショット参照*)。

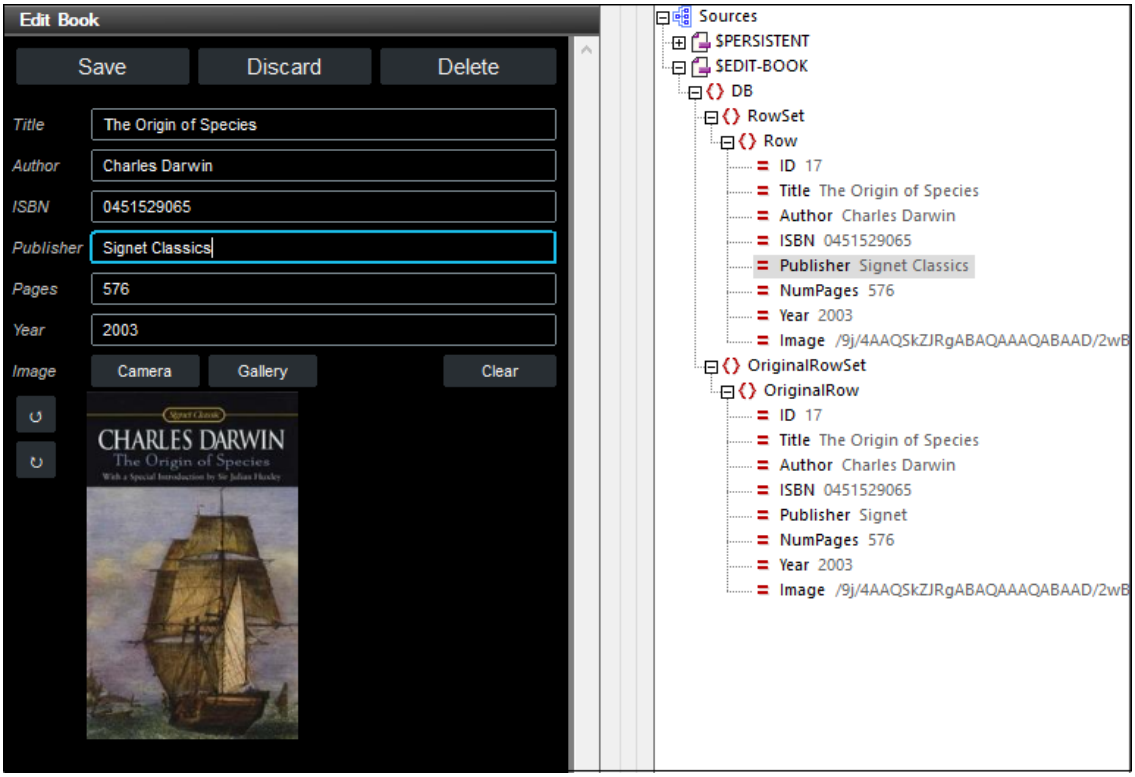


レコードの編集

トップページで表示されるレコードを編集するには、表示フィールドの1つをクリックしてレコードを選択します。各フィールドには、*Edit Book* サブページに基本的に移動する定義済みのアクションのセットが存在します (*下のスクリーンショット参照*)。1つまたは複数のフィールドのアクションを確認してフォローすることができます。



選択された書籍は*Edit Book* サブページの編集テンプレート内で表示されます(*下のシミュレーター内で表示されています*)。



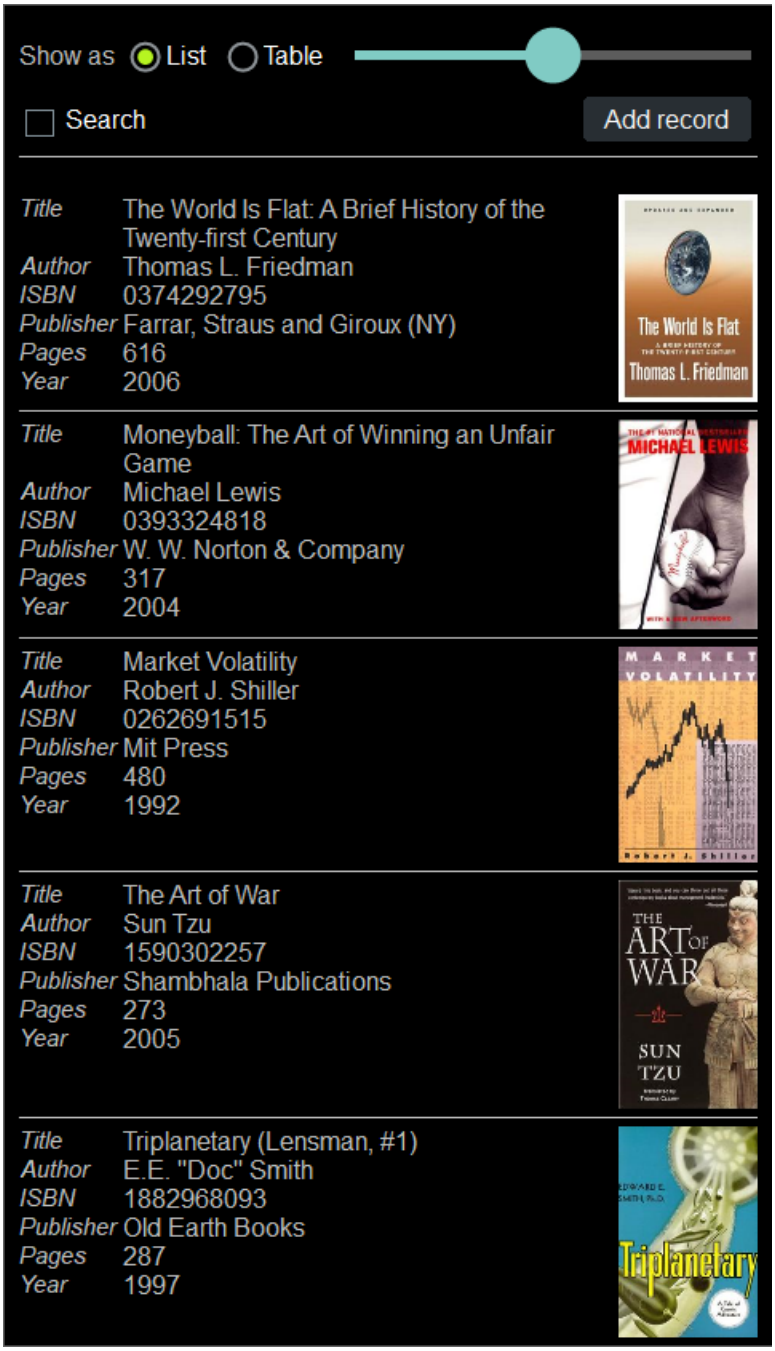
以下の点に注意してください:

- *Edit Book* サブページのページソースは、DB のレコードをフィルターし、選択された書籍のみを表示する **$EDIT-BOOK** です。
- ページソースは**OriginalRowSet** ノードをオリジナルのデータを保つために含みますが、**RowSet** は編集されるデータを保ちます。上のスクリーンショットでは、*Publisher* フィールドは*"Signet"* から*"Signet Classics"* に編集されています。**OriginalRowSet** と**RowSet** の発行者フィールドを比較します。

- **「保存」**ボタンがクリックされると、**RowSet** からのデータがDB に保存されます。 **「保存」**ボタンのアクションを参照してください。
- **「破棄」**ボタンがクリックされると、**RowSet** からのデータが破棄されます。 **「破棄」**ボタンのアクションを参照してください。
- イメージの処理に関しては、セクションDB 内のイメージ内で説明されています。

レコードの追加

レコードを追加するには、トップページの**レコードの追加**ボタンをクリックします (*下のスクリーンショット参照*)。



レコードの追加 ボタンのアクションにより、新規の書籍のレコードの詳細を追加し、選択されたイメージをDB 内に保管することのできる*Edit Book* サブページに移動することができます (*下のスクリーンショット*)。

以下の点に注意してください：

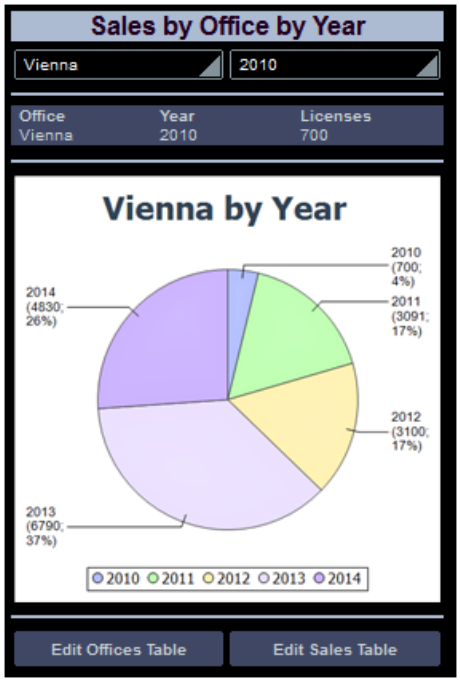
- *Edit Book* サブページのページソースは、DB のレコードをフィルターし、選択された書籍のみを表示する **$EDIT-BOOK** です。
- ノードがページソースのために定義されているためページソースには、**OriginalRowSet** ノードが含まれますが、ノードは空です。 既存のレコードが編集される場合このノードは使用されます。
- **「保存」**ボタンがクリックされると、**RowSet** からのデータがDB に保存されます。 **「保存」**ボタンのアクションを参照してください。
- **「破棄」**ボタンがクリックされると、**RowSet** からのデータが破棄されます。 **「破棄」**ボタンのアクションを参照してください。
- イメージの処理に関しては、セクションDB 内のイメージ内で説明されています。

4.5 データベースとチャート

データベースとチャートチュートリアル(**DBAndCharts.mtd**) はデータベース(DB) とチャートとの作業について説明します:

- テーブルをデータソースとしてセットアップすることにより、DB テーブルデータを表示または編集する方法。
- エンドユーザの選択をベースにしてのDB データの表示方法。
- DB データをベースにするチャートの作成方法。

下部のスクリーンショットは**DBAndCharts.mtd** ソリューションの最初のページを表示しています。エンドユーザーは表示されるセールスのオフィスと年度を選択することができます。デザインのトップにあるコンボボックスでこれらの選択ができます。年度の総売り上げは、コンボボックス下の結果テーブルの*Licenses* 列に表示されます。コンボボックスで新しい年度またはオフィスが選択されると、結果テーブルは新しい選択を自動的に更新し反映します。さらに、選択されたオフィスの円グラフが表示されます。各領域は、年度の売り上げ量とこれまでの総売り上げに対するその年度のパーセンテージを表します。新しいオフィスが選択されると、その特定のオフィスの統計が反映された円グラフが表示されます。



チャートの下には、関連するDB テーブルデータを編集することのできるページに移動する2 つのボタンがあります。

チュートリアル ファイル

このチュートリアルのファイルは*マイ ドキュメント*の MobileTogether フォルダーにあります:
MobileTogetherDesignerExamples\Tutorials\DBAndCharts。

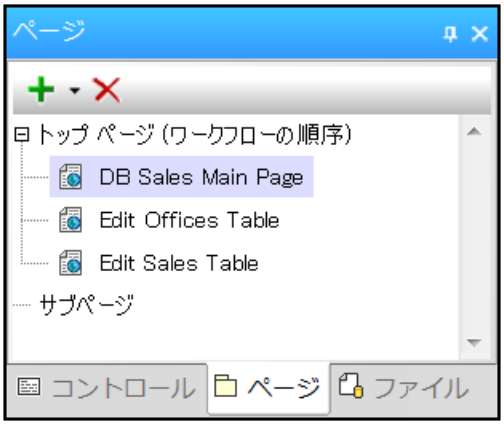
- デザインファイルは**DBAndCharts.mtd**. です。ファイルを開いて、チュートリアルを開き、デザインの作成と機能の説明を読みます。
- MS Access データベース**OfficeSales_DB.mdb** は、デザインのデータソースとして使用されたテーブルを含みます。

プロジェクトの構造

DBAndCharts.mtd を開き **[プロジェクト | 検証]** で正確にAccess DB、**OfficeSales_DB.mdb** に接続できるか検証します。接続エラーが発生した場合、継続する前にエラーを訂正してください。(詳細に関しては メインページのデータソースを参照してください。)

[ページ] ペイン(*下のスクリーンショット*) に表示されるように、プロジェクトは3 つのトップページから構成されています:

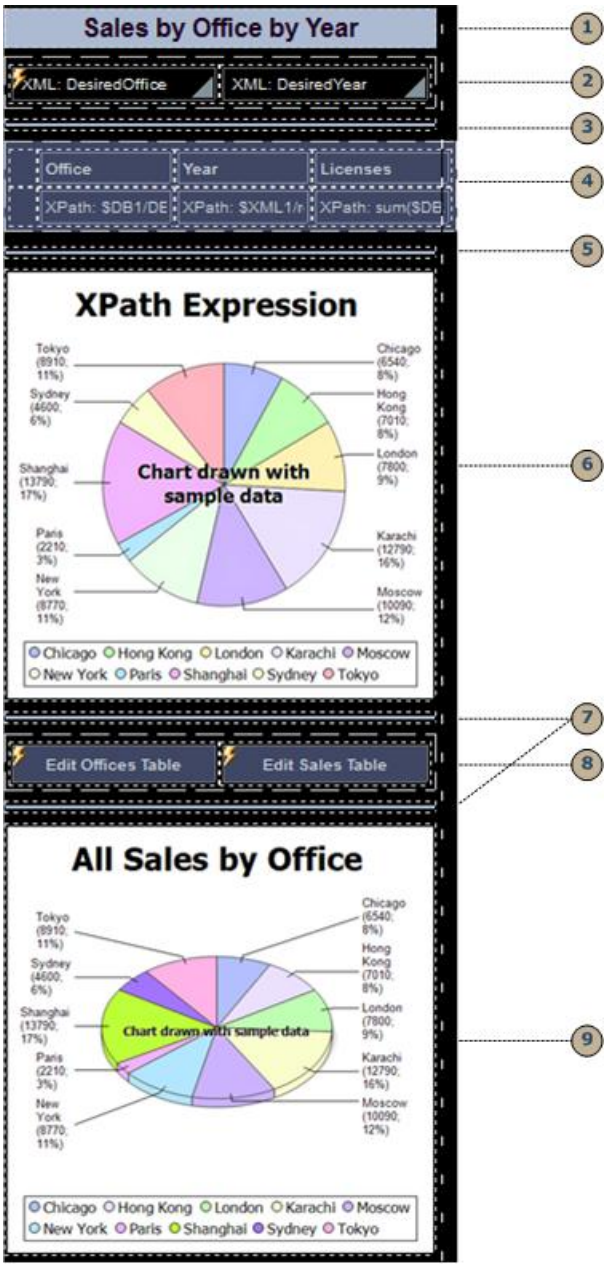
- *DB セールス メイン ページ(Sales Main Page): スタート ページです。* DB データを表示し、他のトップ ページに移動する2つのボタンがあります。
- *オフィス テーブルの編集 (Edit Offices Table): メイン ページのボタンをクリックすることで、このページにたどり着けます。* DB オフィス テーブルの編集を有効化することができます。
- *セールステーブルの編集 (Edit Sales Table): メイン ページのボタンをクリックすることで、このページにたどり着けます。*DB セールス テーブルの編集を有効化することができます。



ソリューションが実行されると、最初のトップページ *DB Sales Main Page* がクライアント アプリにロードされることに注意してください。

メインページ

*DB セールス メイン ページ*のデザインは以下に表示されています。コンポーネントは番号付き吹き出しになっていて、下部に説明があります。



すべてのコンポーネントは、[コントロール] ペインからドラッグして、デザインにドロップするコントロールです。各コントロールは、[スタイル & プロパティ] ペインでプロパティを割り当てられています。データソースのデータに関連付ける必要のあるコントロールは、データソース ノードをコントロールにドラッグすることにより適切なソースノードが割り当てられています。コンボ ボックスとボタンにはお追加してイベントに関連したアクションがあります。アクションは、コントロールを右クリックして、**[に対するコントロール アクション...]** コマンドを選択することによりアクセスされる、アクション ダイアログでコントロールに割り当てられます。

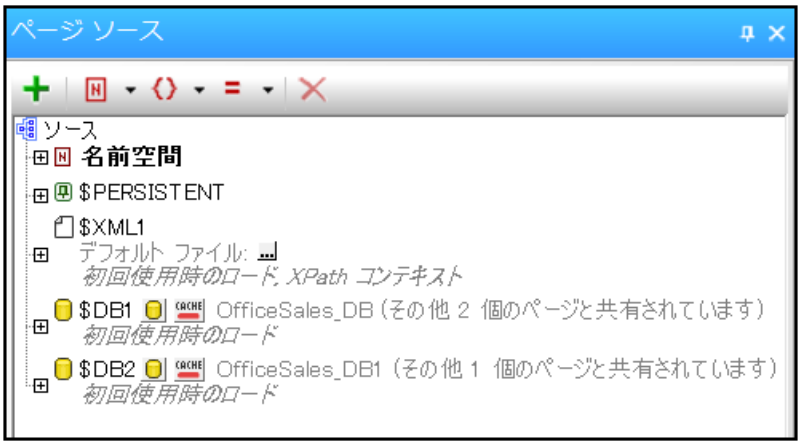
1	タイトルページを表示するラベルコントロール; スタイルプロパティが適用されています

2	Office とYear のエンドユーザーの選択のコンボボックス。*詳細を参照*
3 5 7	レイアウトコンポーネントとしての水平線; スタイルプロパティが適応されます。
4	DB データを含むセル付きテーブルコントロール。*詳細を参照*
6 9	チャート形式でDB データを表示するチャートコントロール。*詳細を参照*
8	*Edit Offices* と*Edit Sales* ページに移動するボタンのクリック時 アクション付きのボタンコントロール。

DB Sales Main Page には、データソースノードをアップデートする、ページのロード時 イベント **[ページ | ページ アクション]** のために定義されたアクションがあります。このアクションは次のセクション*メインページのデータソース*で説明されています。

メインページのデータソース

メインページには次の3 つのデータソースがあります: **$XML1**、**$DB1**、と**$DB2**。これらは[ページ ソース] ペイン(*下のスクリーンショット参照*)で表示され管理されます。



ページのXPath コンテキストはルートノード**$XML1** です。これは、このページのすべてのXPath 式が、をコンテキストノードとして持っていることを意味します。他のツリー(これらのツリーのルートノードである$DB1 と$DB2) で、ノードをみつけるには、各ルートノードのXPath ロケーター パスを開始します。

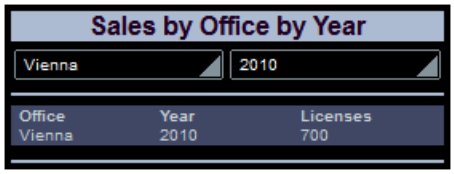
第 1 のデータソース: $XML1

このデータソースは編集可能な空白のXML として作成されています。ルート ノード**$XML1** は、2 つの属性(**DesiredOffice** と**DesiredYear**) を持つルート要素 (**ルート**) を含んでいます。このルート ノード$XML1 は(コンテキスト メニューを介して)ページ2 のXPath コンテキスト ノードとして設定されています。デフォルトのファイルは設定されていないため、ツリーにデータはインポートされません。

このデータソース($XML1) はユーザーのコンボ ボックスの選択 を保つために作成されています:

- DesiredOffice 属性はエンドユーザーのOffice の選択を保ちます。
- DesiredYear 属性はエンドユーザーの Year の選択を保ちます。

コンボ ボックスで選択されたデータを保つために、2 つの属性ノードがコンボ ボックスにページ ソース リンクとして関連づけられています。2 つのページ ソース リンクは各自、属性ノードを各コンボ ボックスにドラッグすることで作成されます (*下部のシミュレーターのスクリーンショット参照*)。



各ノードは、(コンテキスト メニュー コマンドにより**[ページロード時に存在することを確認 (XPath 値)]**)ページロード時に初期の値を与えられています。これは、ノードの値が関連したコンボボックスに表示されるため、コンボボックスに初期の値が必要だからです(*上のスクリーンショット参照*)。XPath 式に与えられる初期の値は次の通りです:

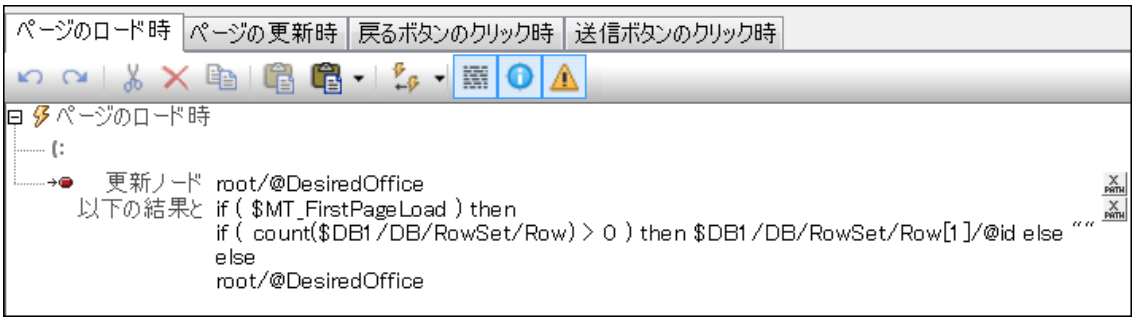
- `@DesiredOffice: if (count($DB1/DB/rowset/Row) > 0) then $DB1/DB/RowSet/Row[1]/@id else ""`

  `$DB1`, に1 つ以上の記録がある場合、最初の記録の`@id` の値を`@DesiredOffice` の値にセットします。記録がない場合、`@DesiredOffice` の値を空白の文字列に設定します。

- `@DesiredYear: min(distinct-values($DB2/DB/RowSet/Row[@Office=$XML1/root/@DesiredOffice]/@Year))`

  `$DB2` では、`@DesiredOffice` で選択されたすべての記録を選択して、記録から一意の年度を集め、最低限の数値を持つ年度を選択します。

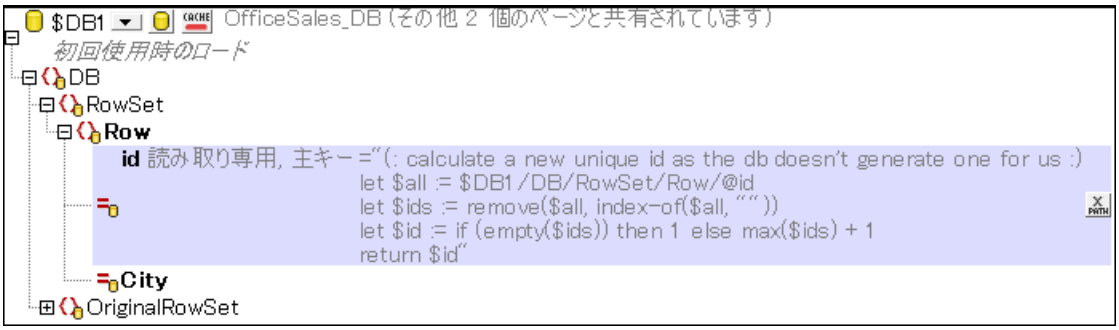
更に、メインページがロードされる都度、**`@DesiredOffice`** ノードが正確にフィルされるように指定します。これは、メインページの **ページのロード時** イベントのノードの更新アクション**[ページ | ページ アクション]** により行われます。



アクションは**`@DesiredOffice`** ノードをアップデートします。これが、ページの最初のロードである場合、最初のオフィスの ID は**`@DesiredOffice`** のコンテンツとしてパスされます。それ以外の場合、値は**`@DesiredOffice`** 内の既存の値となります。結果は、実行中、**`@DesiredOffice`** の値は変更されず、ページが最初にロードされる都度、値は初期化されます。

第 2 のデータソース: `$DB1`

第 2 のデータソース(**`$DB1`**)は、MS Access データベース**`OfficeSales_DB.mdb`** 内の`Offices` テーブルです。このデータソースのデータは、DB の`Office` テーブルから得られます。



Offices のテーブルには、自身は、DB テーブル内の行と関連している要素の Row 属性のツリーとして表示されている列が 2 つあります (**id** と **City**)。id 列は、主キーで値は変更できないため、この列は編集できません。しかし、新しい行のために、id 値を作成する必要があります。これを、XQuery 式を作成し、作成されたすべての新しい行のために id 値を生成し自動化します。XQuery 式をコンテキスト メニュー コマンド **[ページロード時に存在することを確認 (XPath 値)]** を使用して挿入します:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```

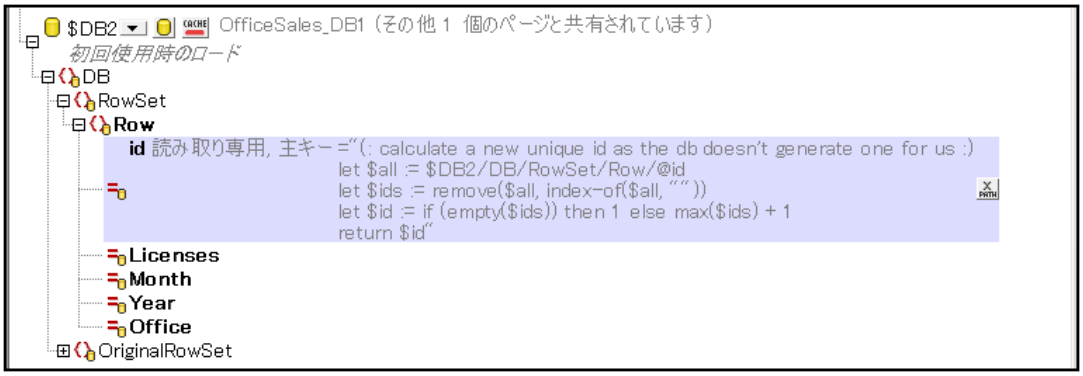
id 値は、オフィスの一意の ID 番号ですが、**City** 値は、オフィスがある都市の名前です。($XML1/root/@DesiredOffice ノードを介して) **id** はオフィスを一意的に識別するために使用されますが、エンドユーザーのオフィスを識別するために都市の名前を使用するため重要です。

データソース内のノードが編集される必要がある場合、(コンテキストメニューを介して) **OriginalRowSet** ノードを作成する必要があります。これは、OriginalRowSet がオリジナルのデータを保ち、**RowSet** が現在の(編集された)データを保つために必須です。

2 つのデータのセット(オリジナルおよび編集された)は必須です。これは、MobileTogether Designer は、何か新しくアップデートされているか、または削除されたかを識別し、正しいタイミングで必要な変更を加えるために確認します。また、XQuery **let** ステートメントを使用して新しい主キーを作成するためにも必要です。データベースが更新されると、更新されたデータは、新しいオリジナルデータとなり、OriginalRowSet ノードに入力されます。

第 3 のデータソース: $DB2

第 3 のデータソース(**$DB1**) は、MS Access データベース **OfficeSales_DB.mdb** 内の Sales テーブルです。このデータソースのデータは、DB の Sales テーブルから得られます。



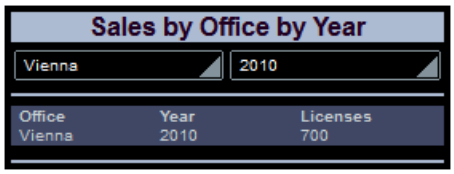
Sales テーブルの各行には5つの列 (**id** 、**Licenses**、**Month**、**Year**、および**Office**) があります。テーブルの列は、**Row** 要素の属性に対応します。id 属性はXQuery 式を持ち、すべての作成された行にid 値を生成します。XQuery 式をコンテキスト メニュー コマンド **[ページロード時に存在することを確認 (XPath 値)]** を使用して挿入します:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```

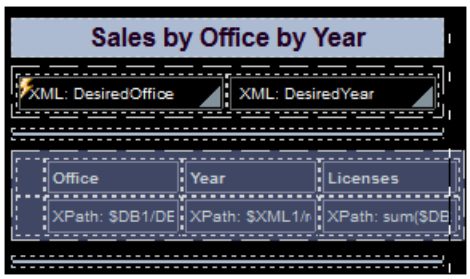
データソース内のノードが編集される必要がある場合、(コンテキストメニューを介して) **OriginalRowSet** ノードを作成する必要があります。これは、OriginalRowSet がオリジナルのデータを保ち、**RowSet** が現在の(編集された) データを保つために必須です。

コンボ ボックス

ページトップの2 つのコンボ ボックスは、エンドユーザーの選択を受け入れるため、また、これらの選択を基にしたデータを表示するために使用されます。下のスクリーンショットは、ソリューションが実行される時のコンボ ボックスを表示しています。表形式のレポートでは、コンボ ボックスはコンボ ボックスの選択をベースにしています。



次のスクリーンショットでは、デザインのコンボ ボックスを表示しています。コンボ ボックスはテーブルの個別セルにレイアウトするために使用されます。



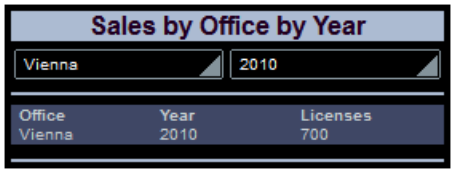
`DesiredOffice` コンボ ボックス

以下が設定されます:

- ソースノード リンクがコンボ ボックスと`$XML1/root/@DesiredOffice` ノードの間にノードをコンボ ボックスにドラッグすることにより作成されました。これによりコンボ ボックスの選択がノードにパスされ、ノードの値がコンボ ボックスにパスされます。
- コンボ ボックスのドロップダウン リストのアイテムは、([スタイル & プロパティ] ペイン内の) コンボ ボックス入力値 プロパティの **[追加ダイアログ]** ボタンからアクセスできる、コンボ ボックスの編集ダイアログで定義されます(*下のスクリーンショット*)。

ドロップダウン リストの値は **$DB1/DB/RowSet/Row/@City** ノード(すなわち、都市の名前)から取得されることに気づいてください。$XML1/root/@DesiredOffice ノードに入る値は、(ソースノード リンクのため) **$DB1/DB/RowSet/Row/@id** ノードから取得されます。*値を並べ替え*チェック ボックスが選択されているため、ドロップ ダウン リストのアイテムは並べ替えられます。

- ノードの更新 アクションが**編集の終了時** イベントにセットされています。コンボ ボックスを右クリックして、**[編集の終了時 に対するコントロール アクション]** を選択しアクションの定義を表示します。更新されるノードは、**root/@DesiredYear** です。更新された値は XPath 式により提供されます: min(distinct-values($DB2/DB/RowSet/Row[@Office=$XML1/root/@DesiredOffice]/@Year))。この式は、コンボ ボックスで選択されたすべての記録を選択して、記録から一意の年度を集め、最低限の数値を持つ年度を選択します。



ですから、最初のコンボ ボックスで Vienna が選択された場合 (上部のスクリーンショット)、@Office='Vienna' を持つ **$DB2** 内の全ての記録が検索され、記録が作成された一意の年度別のシーケンスが作成されます。最小の数値を持つ? この場合 **2010** ? がノードにパスされます? この場合 $XML1/root/@DesiredOffice。このノードは 2 番目のコンボ ボックス(@DesiredYear コンボ ボックス) のソースノードであるため、このコンボ ボックスは最小年度の値を表示します ? この場合 **2010** ?。

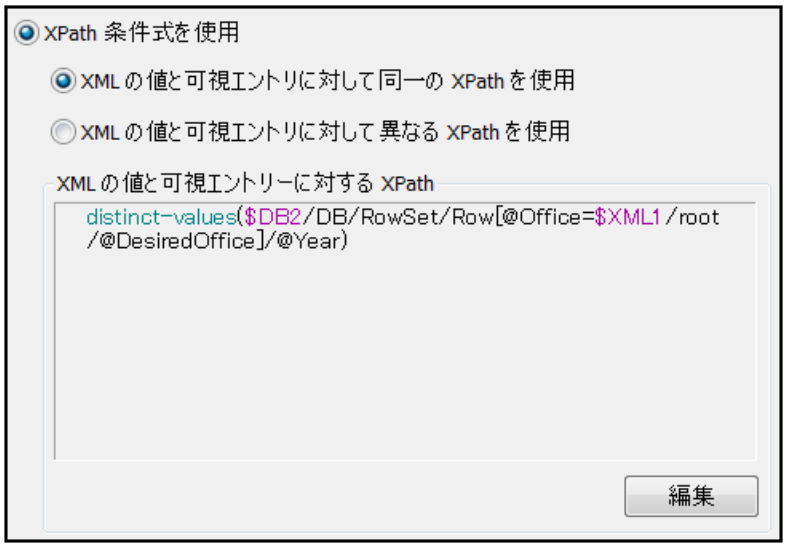
DesiredYear コンボ ボックス

以下が設定されます:

- ソースノードがコンボ ボックスと $XML1/root/@DesiredYear ノードの間にノードをコンボ ボックスにドラッグすることにより作成されました。これによりコンボ ボックスの選択がノードにパスされ、ノードの値がコンボ ボックスにパ

スされます。

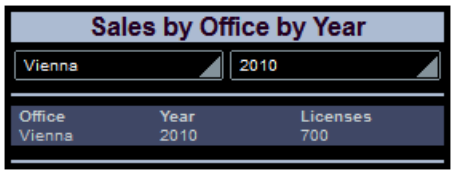
- コンボ ボックスのドロップダウン リストのアイテムは、([スタイル & プロパティ] ペイン内の) コンボ ボックス入力値 プロパティの **[追加ダイアログ]** ボタンからアクセスできる、コンボ ボックスの編集ダイアログで定義されます(*下のスクリーンショット*)。



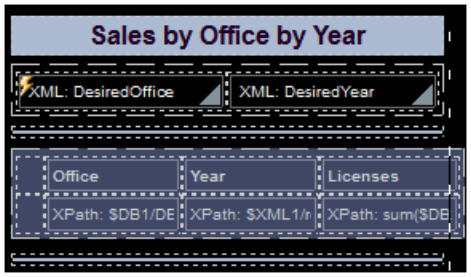
ドロップダウン リストの値はXML ノードと同じことに気づいてください。これらは、選択されたオフィスが記録されたセールスを持つ一意の年度のシーケンスです。*値を並べ替え*チェック ボックスが選択されているため、ドロップ ダウン リストのアイテムは並べ替えられます。

表形式レポート

表形式のレポートは以下の2 つのコンボ ボックスにテーブルで表示されています。エンドユーザーがレポートに必要なオフィスと年度を選択すると、表形式のレポートでその年度の総売り上 (ライセンスの数) が表示されます。下のスクリーンショットは実行されたソリューションを表示します。



次のスクリーンショットは、デザイン内の表形式レポートを表示します。テーブルは1 列目は余白として使用される2 つの行と4 つの列から構成されています。残りの6 つのセルはテキストの値を持つラベルを含み、直接入力またはXPath 式により計算されます。[スタイル & プロパティ] ペイン内の各ラベルの テキスト プロパティを参照してください。



XPath 式は以下の通りです:

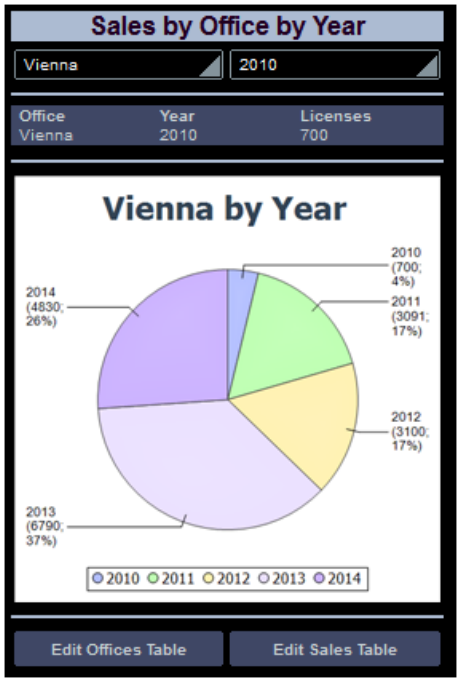
- *希望するオフィス (DesiredOffice):* は **$DB1** から取得されます。これは、**@id** がコンボ ボックスの選択の **id** の値と同じ値の **行** の**@City** の値です。
  $DB1/DB/RowSet/Row[@id=$XML1/root/@DesiredOffice]/@City

- *希望する年度 (DesiredYear):* は **$XML1** から取得されます。これは、**@DesiredYear** の値です。年度はエンドユーザーによりコンボ ボックスから、または、そのオフィスの一意の売り上げの最小値を持つ年度です。
  $XML1/root/@DesiredYear

- *販売されたライセンス (Licenses Sold):* **$DB2** はから取得されます。これはコンボ ボックスの選択と同じ値を持つ**@Office** と**@Year** 属性付きの**行** 要素のすべての**@Licenses** 値をまとめます。(**$DB2** の **@Office** 値 はオフィスのID 値であり、都市の名前ではないことに注意してください。)
  sum($DB2/DB/RowSet/Row[@Office= $XML1/root/@DesiredOffice][@Year= $XML1/root/@DesiredYear]/@Licenses)

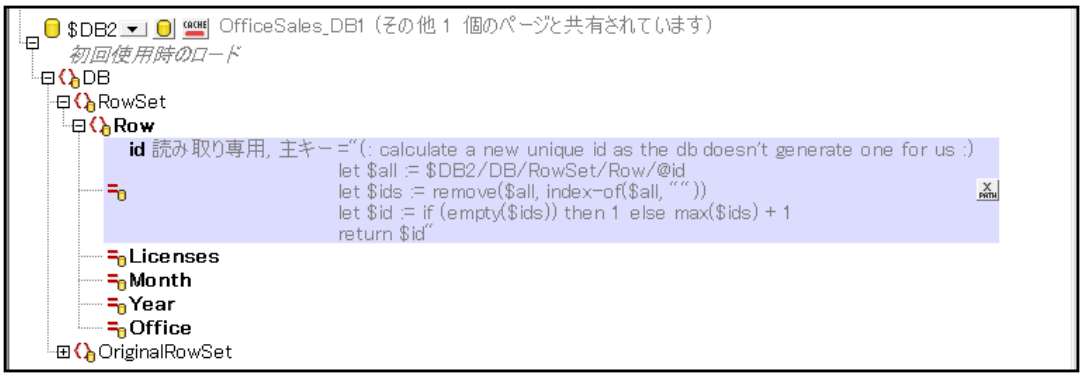
チャート

デザインには2 つのチャートがあります。最初のチャートはコンボボックスで選択されたオフィスの全ての売り上げの内訳を表示しています (*下のスクリーンショットのシミュレーター参照*)。



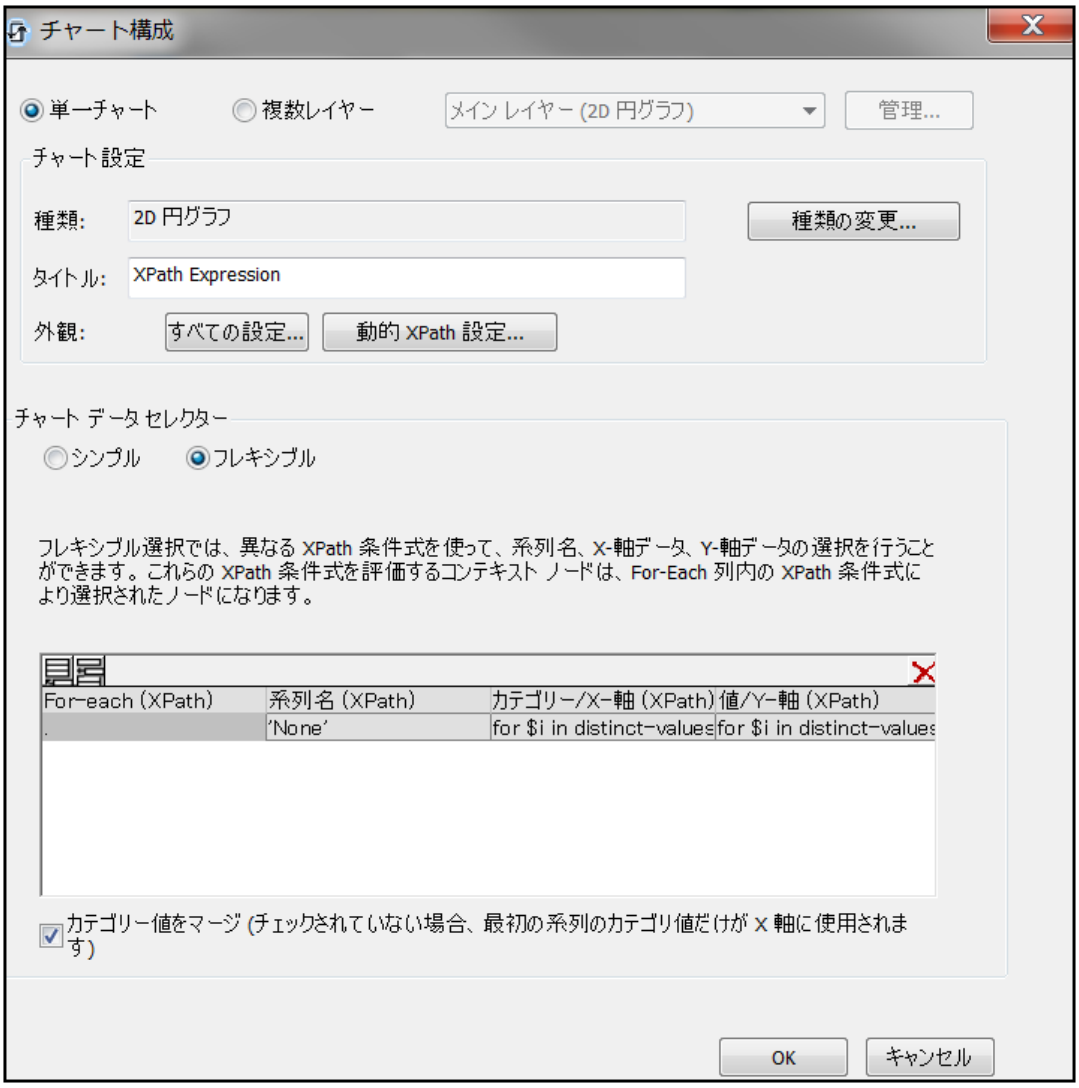
XPath コンテキスト ノード

メインのチャートの定義は、X軸 とY軸で何が発生するかです。これらは XPath 式で定義されており、チャートの XPath コンテキストを正確に選択することは重要です。コンテキスト ノードに関しては、X軸 とY軸で使用されるノードセットの直接の親を選択することが最善策です。Sales データテーブルからデータを使用するため、**$DB2** ツリーをチャート作成に使用します(*下のスクリーンショット*)。両軸のノードセット**行**要素から構成されるため、**RowSet** をXPath のコンテキスト ノードとして選択します。**RowSet** ノードをチャートにドラッグして行います。ノードは太字で表示され、ソースノードがあることを表示します。



チャートの軸の定義

チャートの軸を定義します。チャートをダブルクリックまたは、([スタイル & プロパティ] ペイン内の) チャート設定 プロパティの **[追加ダイアログ]** ボタンをクリックして、チャートのチャート構成ダイアログを開きます (*下のスクリーンショット*)。チャートの種類が円グラフであることに注意してください。



円グラフには 2 つのシリーズ(X軸 とY軸)が必要です。フレキシブル オプションは 2 つのシリーズの軸を定義するときに理想的です。*For-Each* 設定に関しては、現在のノード(`RowSet`) を選択します。2 つの軸のため以下のXPath 式を定義します:

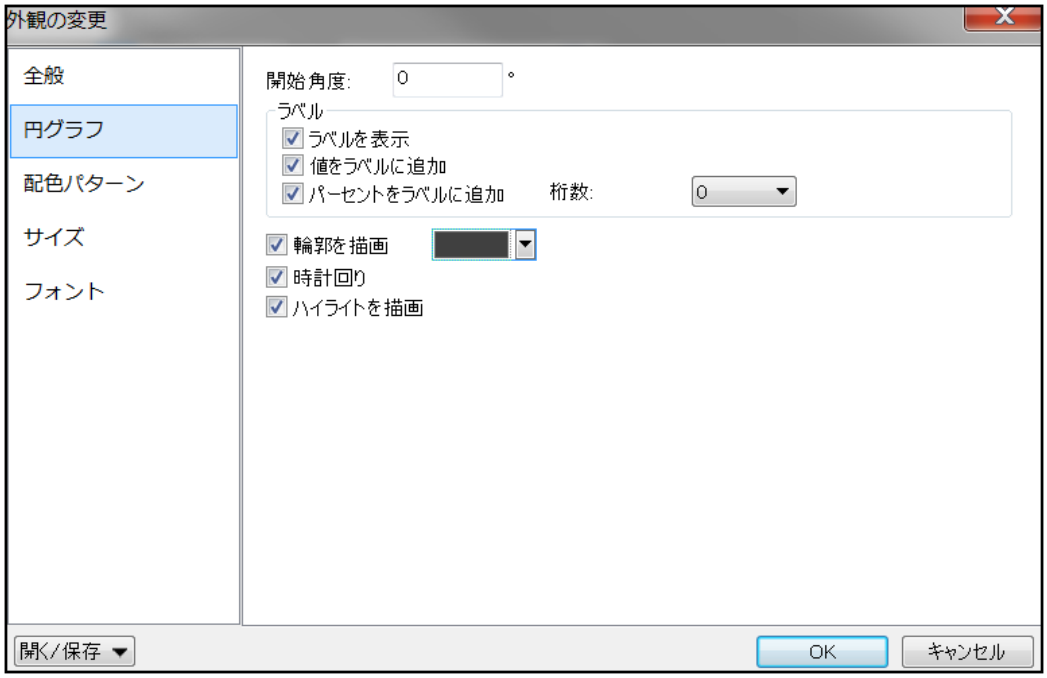
- *X-軸*: 選択されたオフィスの記録されたセールスの一意の年度のシーケンスを作成する。
  `for $i in distinct-values(Row[@Office=$XML1/root/@DesiredOffice]/@Year) return $i`
- *Y-軸*: 選択されたオフィスと各一意の年度のセールスをまとめる(`@Licenses` 属性に保管されます)。
  `for $i in distinct-values(Row[@Office=$XML1/root/@DesiredOffice]/@Year) return`

```
sum(Row[@Office= $XML1/root/@DesiredOffice][@Year=$i]/@Licenses)
```

追加定義

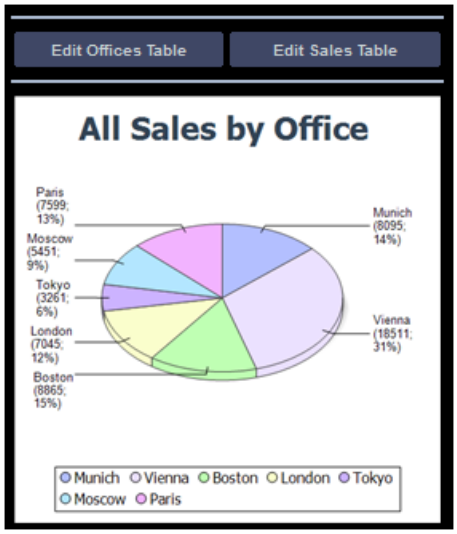
更に、以下が設定されます:

- チャート構成ダイアログで、**[動的 XPath 設定]** をクリックして、XPath 式を使用してタイトルを設定します。これによりタイトルに選択されたすべてのオフィスが表示されます。
- チャート構成ダイアログで、**[すべての設定]** をクリックすると、外観の変更ダイアログが表示されます。[*円グラフ*] を選択して、[*値をラベルに追加とパーセントをラベルに追加*] を選択します。



第 2 のチャート

第 2 のチャートは第一のチャートに類似していますが、3D 円グラフです(*下のスクリーンショット*)。総売り上げの一部として各オフィスの売り上げが表示されています。



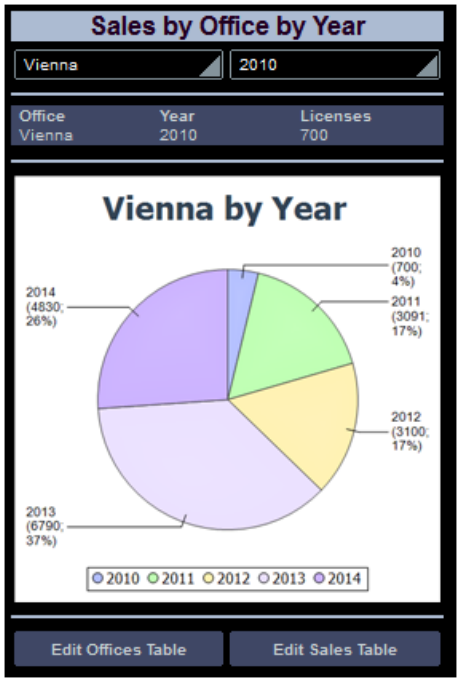
XPath 式は以下の通りです:

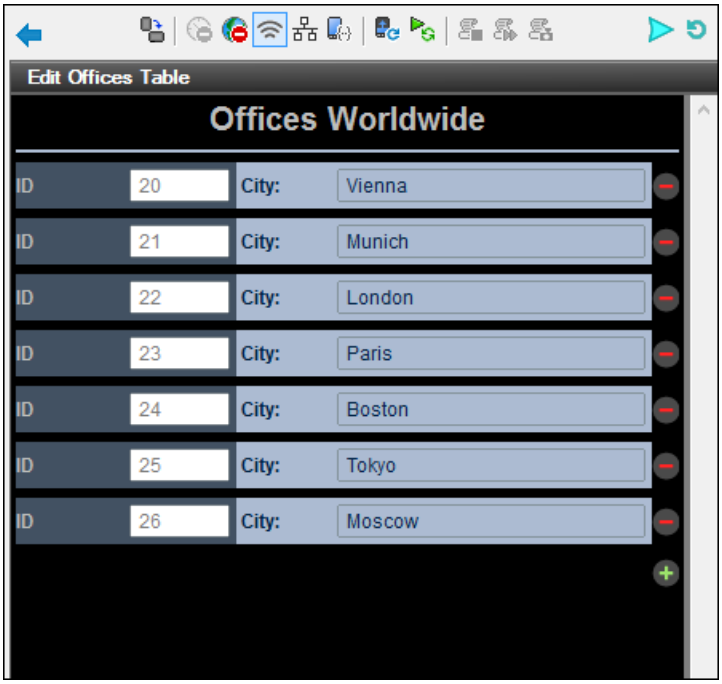
- *X-軸*: **$DB1**. から取得した都市の名前を持つ(IDではない) オフィスの名前のシーケンスを作成する。
  for $i in distinct-values(Row/@Office) return $DB1/DB/RowSet/Row[@id=$i]/@City

- *Y-軸*: (`@Licenses` 属性に保管される) 選択されたオフィスの売り上げをまとめる。
  for $i in distinct-values(Row/@Office) return sum(Row[@Office=$i]/@Licenses)

オフィス テーブルの編集

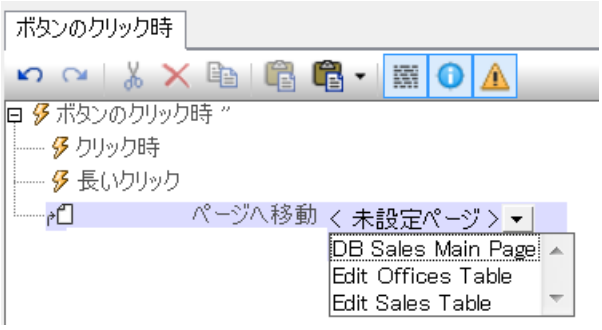
オフィス テーブルの編集 (Edit Offices Table) が別のトップページに作成されます。ソリューションが実行されると、このページにメインページからアクセスすることができます (*下部左スクリーンショット参照*)。**[オフィステーブルの編集(Edit Offices Table)]** ボタンをクリックすると、オフィス テーブルの編集をロードします (*下部右スクリーンショット参照*)。オフィス テーブルは7 つの行があり、各行には編集不可能な ID 列、編集可能な都市の列、および削除コントロールがあります (*下部右スクリーンショット参照*)。更に、行の追加コントロールが最後の行の下にあり、**[送信]** ボタンがオフィス テーブルの編集バーにあり、前のページ (この場合メインページ) に戻る **[戻る]** ボタンがあります。





デザインの **[編集]** ボタン (*下部第 1 スクリーンショット参照*) には、**ボタンのクリック時** イベントに対して、ページへ移動ア

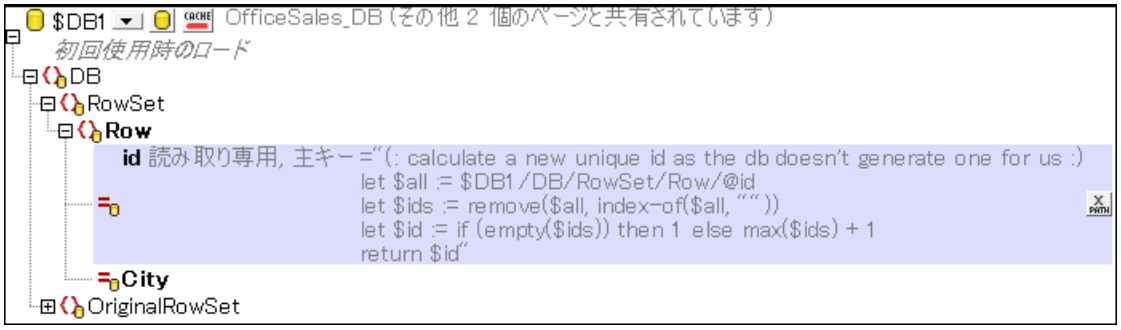
クションが割り当てられています。(ボタンを右クリックして **[ボタンのクリック時 に対するコントロール アクション]** を選択する)。これらのページへ移動アクションは対応するターゲット ページをロードします (*下部第 2 スクリーンショット参照*)。



編集可能なオフィステーブルの作成

DB のオフィステーブルには、データソース **$DB1** に表示された構成があります(*下のスクリーンショット*)。**@id** 属性が主キーなため、変更することはできません。これは、新しい記録が追加されると、エンドユーザーはソリューションを介して @id の値を入力できないことを意味します。@id の値は、自動的に XQuery 式を使用して生成される必要があります。 XQuery 式はコンテキスト メニュー コマンド **[ページロード時に存在することを確認 (XPath 値)]** を使用して挿入されます:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```



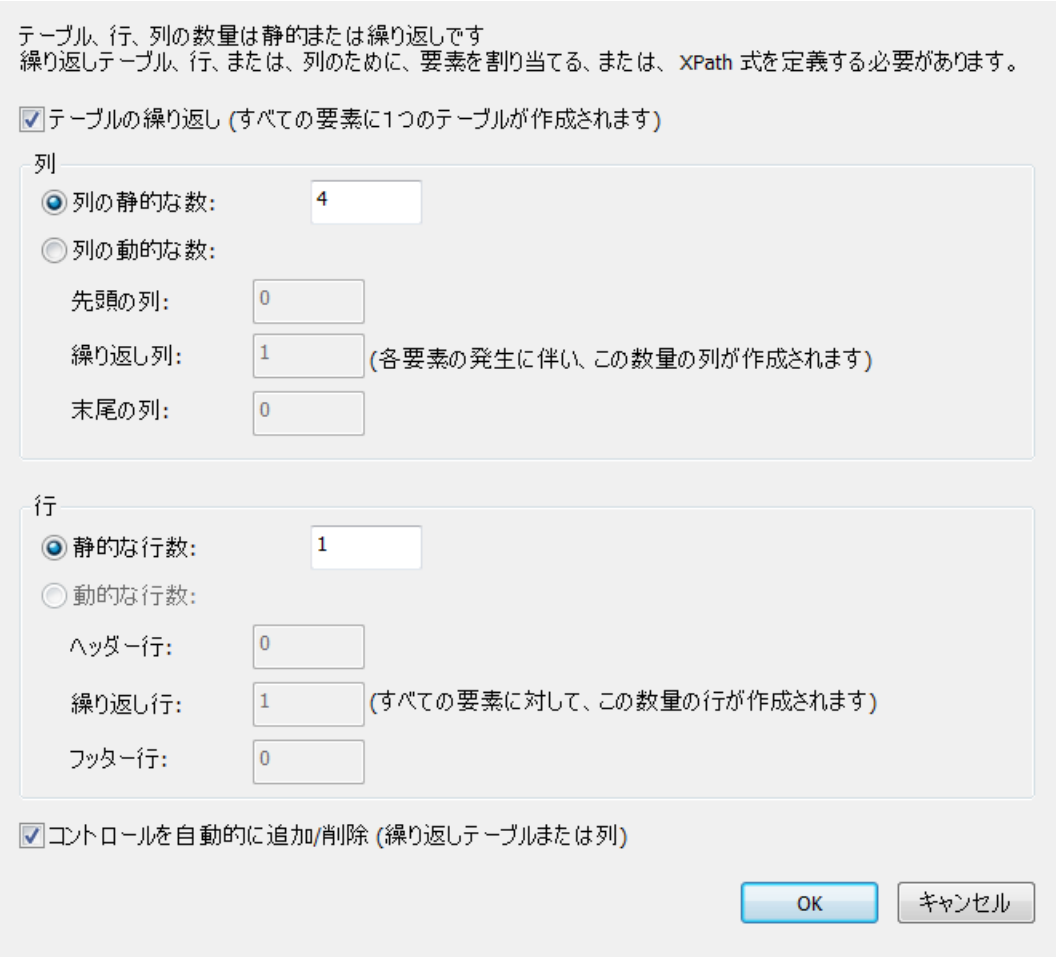
デザインでは以下を行います:

すること...	方法
すべての (Office) 行を表示する	Office 行を繰り返し要素とする繰り返しテーブルを追加する。
行の追加および削除のコントロールを含む	テーブルを追加する時、削除 追加コントロールの自動的な包括を有効化する。
@City の値の編集を有効化する	@City にソースノードを持つフィールドの編集コントロールを追加する。
DB に変更を保存する	*保存 アクションをページの* 送信ボタンのクリック時 イベントに追加する。

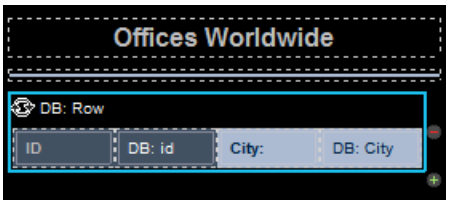
	$DB1 を右クリックして、**元の行セットの作成**に切り替えます。
メインページに戻る	*ページに移動アクションをページの戻るボタンのクリック時 イベントに追加する。*

追加 /削除コントロールを持つ繰り返しテーブルの追加

[コントロール] ペインから[テーブル] コントロールをドラッグして、デザインにドロップすると、新しい[テーブル] コントロールダイアログが表示されます(*下のスクリーンショット*)。

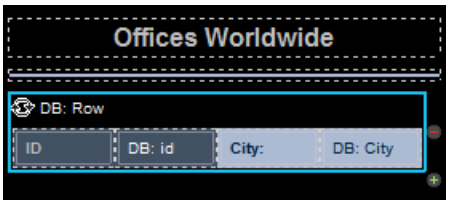


テーブルの繰り返しを指定します。行数 (4) と列数 (1) を入力して、*自動追加 削除コントロール チェックボックス*を選択して、**[OK]** をクリックします。下のスクリーンショットに表示されるように、行の最初の3 つのセルにラベルが追加されます。$DB1 の@id ノードへのソースノード リンクが3 番目のラベル(DB:id) のために作成されます。



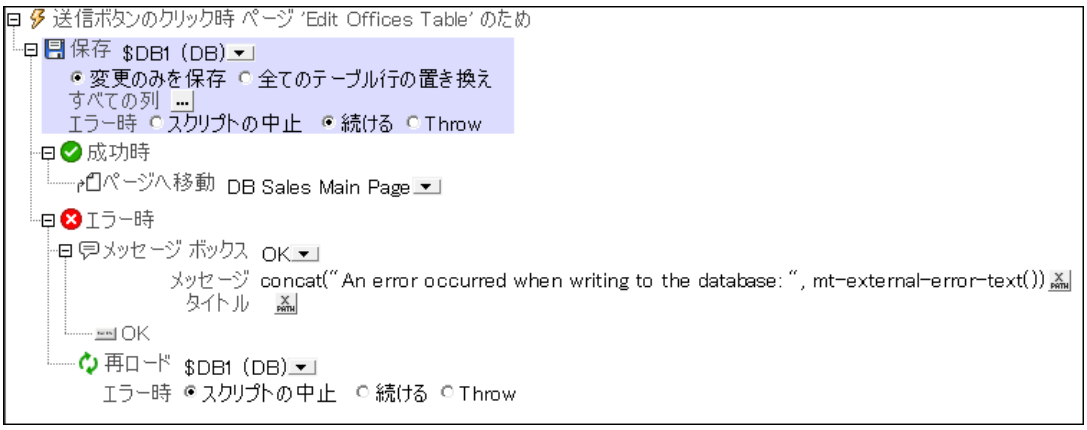
- $DB1 内の @City の編集を有効化する

  A4番目のセルにフィールド コントロールが追加され、ソースノード リンクが作成され、$DB1 の@City ノードが (DB:City) のために作成されます。エンドユーザーが@City 値を編集できるように、このセルでフィールドの編集を使用します; その他すべてのセルには[ラベル] コントロールがあります。



- ページアクション: [保存] して[ページへ移動]

  [ページ | ページ アクション] クリックしてページ アクション ダイアログを開く(*下のスクリーンショット*)。



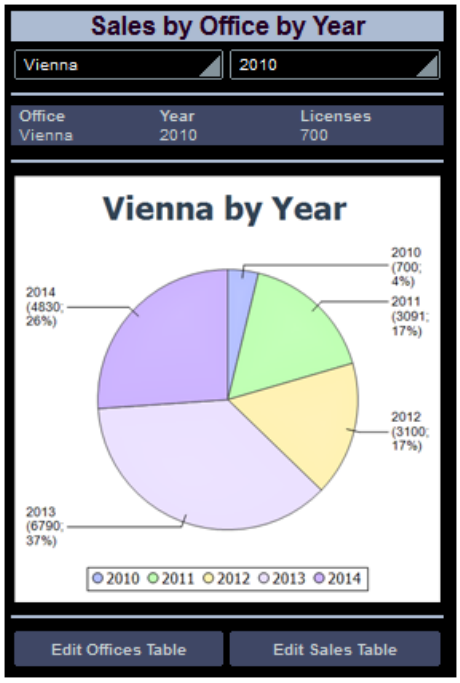
以下のイベントでアクションは定義されます:

- **送信ボタンのクリック時**: DB ($DB1) の全ての列を保存して、メインページに移動する。DB に記録が保存されていない場合、修正されていないデータと共にDB が再ロードされるため、再ロードアクションの追加が奨励されます。(*上のスクリーンショットを参照*)。
- **戻るボタンのクリック時**: メインページに移動する。

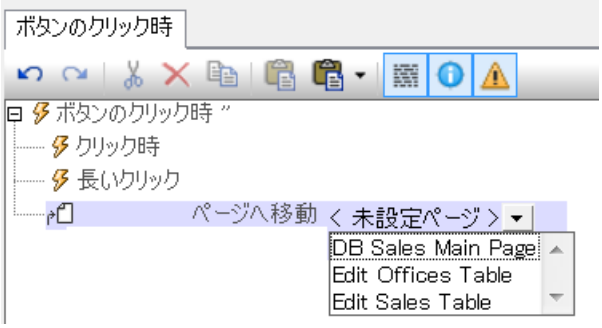
データソースのツリーは、RowSet 要素のコピーである OriginalRowSet 要素を含む必要があります。RowSet 要素 が編集できるように、オリジナルのデータは、OriginalRowSet 要素に保存されます。OriginalRowSet 要素は、新しいデータが保存された際に新しい値と共に更新されます。

セールステーブルの編集

セールステーブルの編集 (Edit Sales Table) がオフィステーブルの編集 (Edit Offices Table) 同様、別のトップページに作成されます。ソリューションが実行されると、このページはメインページからアクセスすることができます (下部左側*スクリーンショット*)。 **[セールステーブルの編集 (Edit Sales Table)]** ボタンをクリックすると、セールステーブルをロードします (下部右側*スクリーンショット*)。セールステーブルには、複数の行があり、各行には編集できない (セールスアイテム) ID 列、編集できるオフィス、月、年度、ライセンス列があります (下部右側*スクリーンショット参照*)。更に、行の追加コントロールが最後の行の下にあり、**[送信]** ボタンがオフィステーブルの編集バーにあり、前のページ (この場合メインページ) に戻る **[戻る]** ボタンがあります。



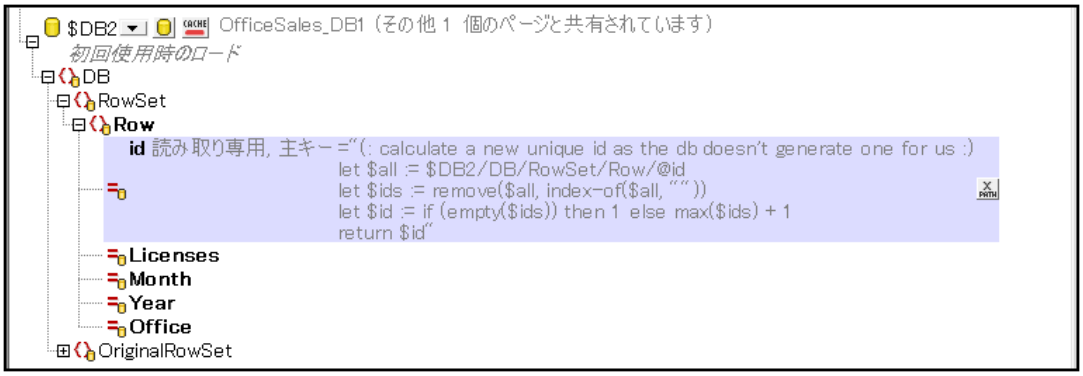
デザインの**[編集]** ボタン (*下部第 1 スクリーンショット参照*) には、**ボタンのクリック時** イベントに対して、ページへ移動アクションが割り当てられています。(ボタンを右クリックして **[ボタンのクリック時 に対するコントロール アクション]** を選択する)。これらのページへ移動アクションは対応するターゲット ページをロードします (*下の第 2 スクリーンショット参照*)。



編集可能なセールス テーブルの作成

DB のセールステーブルには、データソリー **$DB2** に表示された構成があります(*下のスクリーンショット*)。**@id** 属性が主キーなため、変更することはできません。これは、新しい記録が追加されると、エンドユーザーはソリューションを介して @id の値を入力できないことを意味します。@id の値は、自動的に XQuery 式を使用して生成される必要があります。
XQuery 式はコンテキスト メニュー コマンド **[ページロード時に存在することを確認 (XPath 値)]** を使用して挿入されます:

```
let $all := $DB2/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```



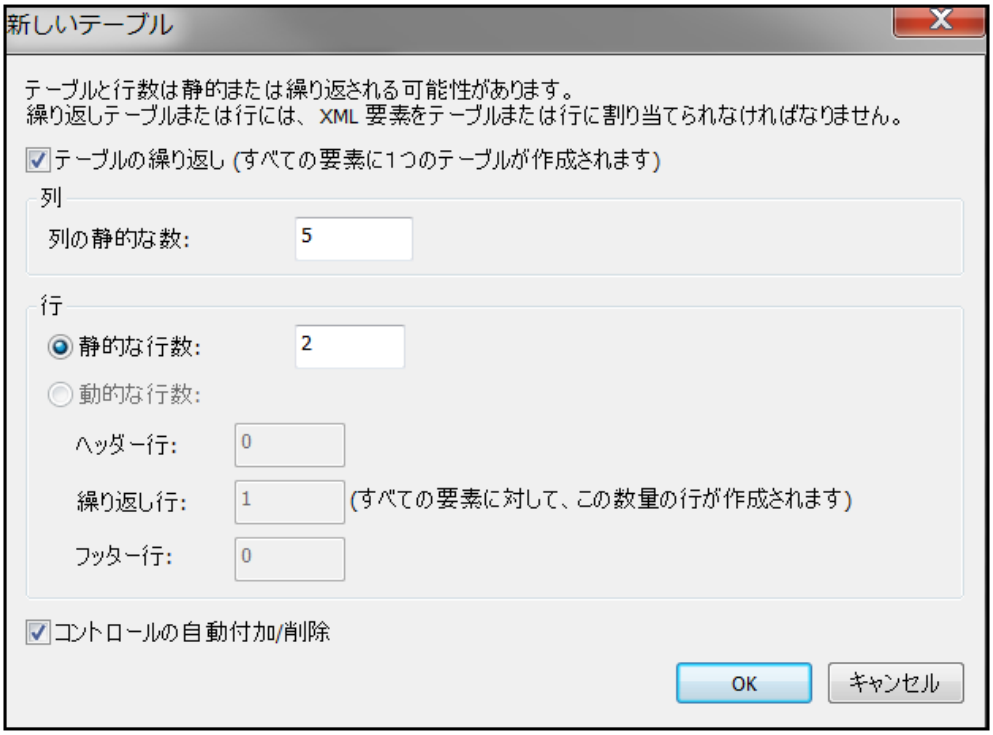
デザインでは以下を行います:

すること...	方法
すべての(Sales) 行を表示する	Sales 行を繰り返し要素とする繰り返しテーブルを追加する。
行の追加および削除のコントロールを含む	テーブルを追加する時、削除 追加コントロールの自動的な包括を有効化する。
編集可能な値の編集を有効化する	コンボ ボックスを追加して、ページソースリンクを持つ編フィールドを編集する。
DB に変更を保存する	*保存 アクションをページの戻るボタンのクリック時 イベントに追加する。* $DB2 を右クリックして、**元の行セットの作成 (Create OriginalRowSet)** に切り替えます。

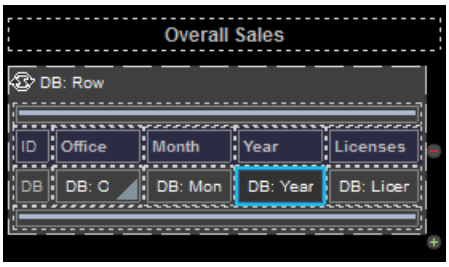
メインページに戻る	*ページに移動アクション*をページの戻るボタンのクリック時 イベントに追加する。

- 追加 /削除コントロールを持つ繰り返しテーブルの追加

[コントロール] ペインから[テーブル] コントロールをドラッグして、デザインにドロップすると、新しい[テーブル] コントロールダイアログが表示されます(*下のスクリーンショット*)。



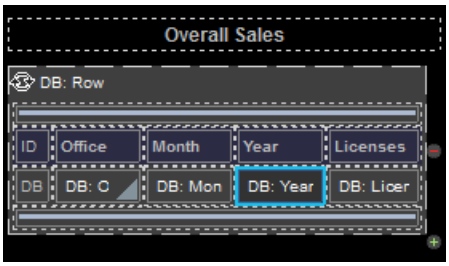
テーブルの繰り返しを指定します。行数 (5) と列数 (2) を入力して、*自動追加 削除コントロール チェックボックス*を選択して、**[OK]** をクリックします。最初の行のセルにヘッダーへラベルを追加します。2番目の行の最初のセルの編集不可能な@id 値のラベルを追加します。 $DB2 の@id ノードへのソースノードリンクがこのラベル(DB:id) のために作成されます。



- 編集可能なノードの編集の有効化

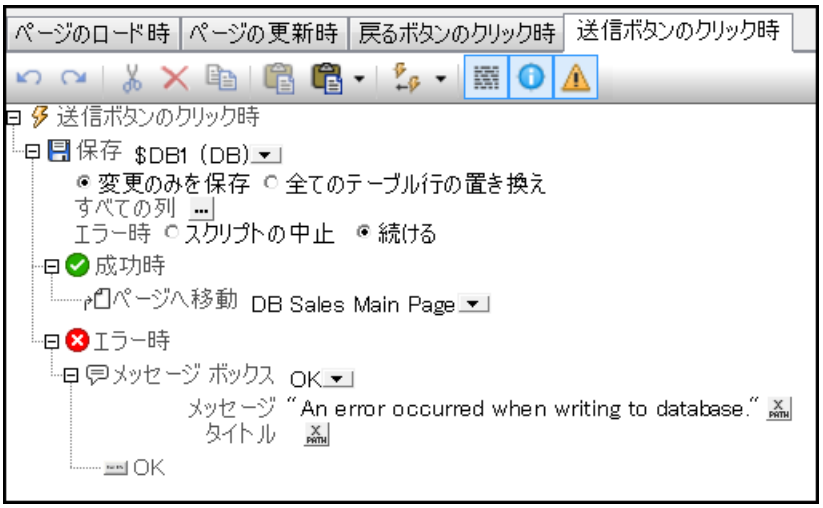
オフィスのために(@Office へのページソースリンクと共に) コンボボックスが追加され、対応するノードへのソースノード

リング付きの月、年度、ライセンスセルの編集フィールドが追加されます。



- ページアクション: [保存] して[ページへ移動]

[ページ | ページ アクション] クリックしてページ アクションダイアログを開く(*下のスクリーンショット*)。



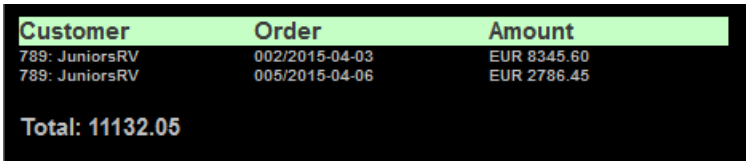
以下のイベントでアクションは定義されます:

- **送信ボタンのクリック時**: DB ($DB1) の全ての列を保存して、メインページに移動する。DB に記録が保存されていない場合、修正されていないデータと共にDB が再ロードされるため、再ロードアクションの追加が奨励されます。(*上のスクリーンショットを参照*)。
- **戻るボタンのクリック時**: メインページに移動する。

データソースのツリーは、`RowSet` 要素のコピーである `OriginalRowSet` 要素を含む必要があります。`RowSet` 要素 が編集できるように、オリジナルのデータは、`OriginalRowSet` 要素に保存されます。`OriginalRowSet` 要素は、新しいデータがに保存された際に新しい値と共に更新されます。

4.6 サブページと可視性

このチュートリアルは、トップページからサブページの開きかた、および`Visible` プロパティを使用してデータ構造の表示のフィルターの方法を説明します。また、動的テーブル、アクショングループ、更新ノードアクション、XPath 関数内の小数点などの使用方法も説明します。トップページ (*最初のスクリーンショット*) は、現在データベースに保存されているすべてのカスタマーを表示しています。エンドユーザーがカスタマーの詳細 (name、city、etc) をクリックすると、カスタマーの現在のオーダーを表示するサブページが表示されます (*2 番目のスクリーンショット*)。



デザインのデータは、以下の2 つのデータソースに保管されています: カスタマーデータを保管するデータソースおよびオーダーの詳細を保管するデータソース。2 つのデータソースには、カスタマーデータをオーダーの詳細とつなぐ共通のカスタマーコード列があります。このチュートリアルでは、XML ファイルを使用しますが、カスタマーコード列が主キーとして使用されているデータベースをデータソースとして使用することもできます。

チュートリアルファイル

このチュートリアルのファイルは(*マイ ドキュメント*内のMobileTogether フォルダーにあります: **`MobileTogetherDesignerExamples\Tutorials\SubPagesAndVisibility`**。

- カスタマーデータを含むXML データファイルは、 **`Customers.xml`** です。
- オーダーデータを含むXML データファイルは、**`Orders.xml`** です。
- 完了時のデザインファイルは、以下に類似します: **`SubPagesAndVisibility.mtd`** 。

チュートリアルの構造

このチュートリアルは以下のセクションに整理されています:

- デザイン構成
- データソースリスト
- トップページ: データソース
- トップページ: カスタマーテーブル
- トップページ: アクショングループ サブページへ移動
- トップページ: すべてのオーダー アクションを表示する
- サブページ: データソース
- サブページ: オーダーテーブル
- サブページ: 可視性プロパティ
- サブページ:XPath 内の総額

- シミュレーションおよびテスト

デザイン構成

デザインファイルは、トップページ(名前の付けられたカスタマー) および サブページ(名前の付けられたオーダー) 含みます。トップページ(*カスタマー: 下のスクリーンショットの最初のページ*) は、データベースに現在保管されているすべてのカスタマーを表示します。エンドユーザーがトップページのカスタマーの詳細 をクリックすると、サブページ(*下のスクリーンショットの2番目のオーダー ページ*) が開かれます。このページは、カスタマーの現在のオーダーを表示します。 トップページは、サブページ、データベース内の現在のオーダー、すべてのカスタマーのオーダーを表示するオプションも表示します。

このデザインの主要なメカニズムは、トップページで選択された(サブページ内の) 特定のカスタマーのオーダーを表示することです。これは、現在のオーダーすべてを表示するテーブルのプロパティと共にアーカイブされています。 プロパティは、どの要素が可視されるかを指定し、また、表示フィルターとして振る舞います。XPath 式を介して、選択されたカスタマーのそのオーダーのみが可視できるように指定します。このフィルターされたテーブルは、選択されたカスタマーのオーダーのみを含むカスタマー固有のテーブルの作成の簡単および効果的な代替となります。

デザインステップ

デザインは、以下で説明されるとおり作成されます。 (*スクリーンショットは、完成されたデザインのシミュレーションを表示しています。*)

トップページ: カスタマー

- トップページおよび2 つのデータソースの作成: **$XML1** および**$CUSTOMERS**
- $CUSTOMERS のカスタマー データを含む動的テーブルの作成 。テーブルの各行は、XML データソース $CUSTOMERS 内のカスタマーに対応します。
- サブページ名前の付けられたオーダーの作成
- アクショングループの作成 には以下を行います: (i) ユーザーがクリックするカスタマー ノードに関するデータを使用して、$XML1 内のノードを更新する。(ii) オーダーと名前の付けられたサブページへ移動する。
- カスタマーデータを含む各ラベルへのアクショングループの割り当て この結果、カスタマーデータがクリックされると、アクショングループが実行されます。
- すべての オーダーの作成を表示するラベル。この ラベル アクション(すべての オーダーの表示)は、他のサブページへ移動 アクションとは対照的に、選択された単一のカスタマーのオーダーを表示します。

サブページ: オーダー

- サブページのために3 つのデータソースを作成する(トップページと共有された) **$XML1** 、(トップページと共有された) **$CUSTOMERS** 、および(トップページと共有された) **$ORDERS**。
- データファイル内のオーダーの詳細を表示する動的テーブルを作成する(*下のスクリーンショットは、テーブル(i) 選択されたカスタマー ,および(ii) すべての カスタマーのオーダーを表示します*)。テーブルの各行は、XML データソース$ORDERS 内のオーダーに対応します。
- 繰り返し行グループのテーブルの可視性プロパティのセットアップ を行い、以下の項目を表示します(i) トップページで選択されたカスタマーのみ、または(ii) すべての カスタマー。
- 総額を生成するためのXPath 式の作成 (i) 選択されたカスタマーのすべての オーダー、または(ii) 現在のすべてのオーダー。

Customer	Order	Amount
789: JuniorsRV	002/2015-04-03	EUR 8345.60
789: JuniorsRV	005/2015-04-06	EUR 2786.45

Total: 11132.05

Customer	Order	Amount
456: HiDeHo	001/2015-04-03	EUR 4906.38
789: JuniorsRV	002/2015-04-03	EUR 8345.60
123: New Fashion	003/2015-04-04	EUR 5645.20
123: New Fashion	004/2015-04-05	EUR 3805.68
789: JuniorsRV	005/2015-04-06	EUR 2786.45
456: HiDeHo	006/2015-04-07	EUR 7460.50

Total: 32949.81

データソース リスト

デザインには、3 つのXML データソースがあります:

- **$XML1** は、デザイン内に直接作成されたツリーです。目的は、サブページ内にカスタマーのオーダーの詳細を表示するためのエンドユーザーの選択を保管することです。$XML1 データソースは、デザインの両方のページで共有されています。
- **$CUSTOMERS** は、3人のカスタマーの詳細を含んでいます。XML ツリー構造およびカスタマー データは、XML ファイルCustomers.xml からインポートされます。$CUSTOMERS データソースは、トップページとサブページ両方で使用されます(下のリストを参照)。
- **$ORDERS** は、Customers.xml の3 人のカスタマーの6 つのオーダーに詳細を含みます。XML ツリー構造 および オーダーの詳細は、XML ファイルOrders.xml からインポートされます(下のリストを参照)。$ORDERS データソース内のオーダーは、オーダーサブページのテーブル内に表示されます。

XML データソースの リスト、Customers.xml

(マイ ドキュメント フォルダーのMobileTogether フォルダー内の検索のMobileTogether フォルダー内の検索 : **MobileTogetherDesignerExamples\Tutorials\Customers.xml**.

```
<?xml version="1.0" encoding="UTF-8"?>
<Customers>
  <Customer code="123">
    <Name>New Fashion</Name>
    <AddressLine01>56 Tromer Street</AddressLine01>
    <AddressLine02></AddressLine02>
    <City>Stockholm</City>
    <ZipCode>1000</ZipCode>
    <Country>Sweden</Country>
    <Email>contact01@newfashion.dummy</Email>
    <Phone/>
  </Customer>
  <Customer code="456">
    <Name>HideHo</Name>
    <AddressLine01>7 Norsk Street</AddressLine01>
    <AddressLine02></AddressLine02>
    <City>Oslo</City>
    <ZipCode>7065</ZipCode>
    <Country>Norway</Country>
    <Email>contact02@hideho.dummy</Email>
    <Phone/>
  </Customer>
  <Customer code="789">
    <Name>JuniorsRV</Name>
    <AddressLine01>81 Bjork Street</AddressLine01>
    <AddressLine02></AddressLine02>
    <City>Copenhagen</City>
    <ZipCode>4538</ZipCode>
    <Country>Denmark</Country>
    <Email>contact03@juniorsrus.dummy</Email>
    <Phone/>
  </Customer>
</Customers>
```

XML データソースの リスト、order.xml

(マイ ドキュメント フォルダーのMobileTogether フォルダー内の検索: **MobileTogetherDesignerExamples\Tutorials\AltovaProducts.xml**.

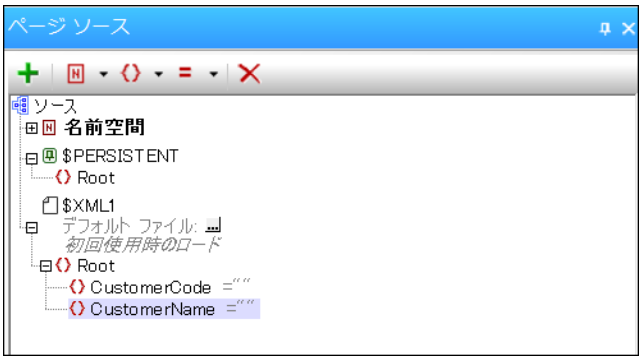
```
<?xml version="1.0" encoding="UTF-8"?>
<Orders>
```

```
<Order number="001">
  <CustomerCode>456</CustomerCode>
  <OrderDate>2015-04-03</OrderDate>
  <OrderAmount>490638</OrderAmount>
  <Currency>EUR</Currency>
</Order>
<Order number="002">
  <CustomerCode>789</CustomerCode>
  <OrderDate>2015-04-03</OrderDate>
  <OrderAmount>834560</OrderAmount>
  <Currency>EUR</Currency>
</Order>
<Order number="003">
  <CustomerCode>123</CustomerCode>
  <OrderDate>2015-04-04</OrderDate>
  <OrderAmount>564520</OrderAmount>
  <Currency>EUR</Currency>
</Order>
<Order number="004">
  <CustomerCode>123</CustomerCode>
  <OrderDate>2015-04-05</OrderDate>
  <OrderAmount>380568</OrderAmount>
  <Currency>EUR</Currency>
</Order>
<Order number="005">
  <CustomerCode>789</CustomerCode>
  <OrderDate>2015-04-06</OrderDate>
  <OrderAmount>278645</OrderAmount>
  <Currency>EUR</Currency>
</Order>
<Order number="006">
  <CustomerCode>456</CustomerCode>
  <OrderDate>2015-04-07</OrderDate>
  <OrderAmount>746050</OrderAmount>
  <Currency>EUR</Currency>
</Order>
</Orders>
```

トップページ: データソース

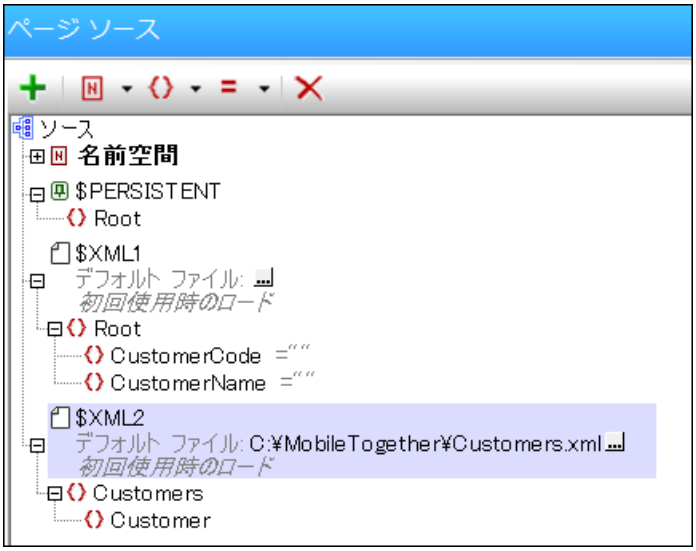
このセクションでは、トップページおよびそのデータソースを作成します。以下を行ってください:

1. **[ファイル| 新規作成]** コマンドで、新しいデザインファイルを作成します。
2. ファイルを希望する名前で保存します。
3. [ページ] ペインで、トップページ(デフォルトで作成される。) のページ名をダブルクリックして、編集をし `Customers` に名前変更します。
4. [ページソース] ペインで、**ソースの追加** アイコンをクリックして、*新しい空白のXML* を選択することにより新しい空白のXML ソースを追加します。2番目のスクリーンでデフォルトの選択を維持します。`$XML1` という名前の、データソースが作成されます (*下のスクリーンショット参照*)。
5. このデータソースの構造を手動で下のスクリーンショットのように作成します。これは、(`$XML1` から始まる) ツリー内のノードを右クリックし、**子の追加**、コンテキストメニューの **追加** および **挿入**のコマンドを使用して行います。
6. **`CustomerCode`** ノードを右クリックして、コマンド **ページのロード時に存在することを確認(固定値)** を選択します。ノードのために表示される値ボックス内で、値を入力せず **[Enter]** を押します。**`CustomerName`** ノードに対しても同様の手順を行います。これにより、ページのロード時に`$XML1` ツリー内の2つのノードが空白で作成されることを保証されます。

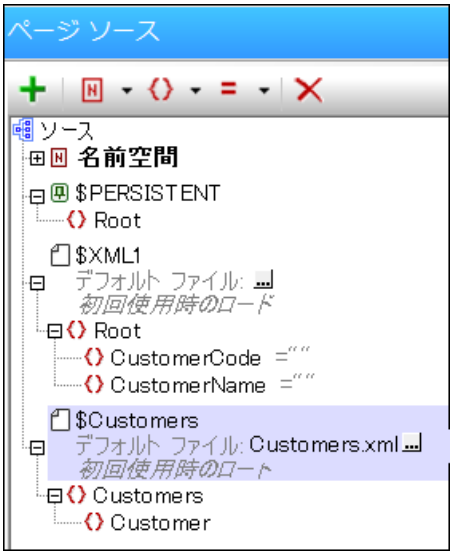


ランタイムでは、このページこのページに表示されるカスタマーのリストからエンドユーザーにより選択されたカスタマーのデータ(コードおよび名前) と共に`$XML1` ツリーのノードが更新されるようにします。

7. **ソースの追加** アイコンをクリックすることにより2番目のデータソースを作成します。*ファイルからインポートされた新しいXML または HTML 構造*を選択します。 ファイル`Customers.xml` を検索して、**[開く]** をクリックします。ファイルをデプロイするか問うプロンプトが表示されますので、**[はい]** を選択します。`$XML2` という名前のデータソースが作成されます(*下のスクリーンショット参照*)。



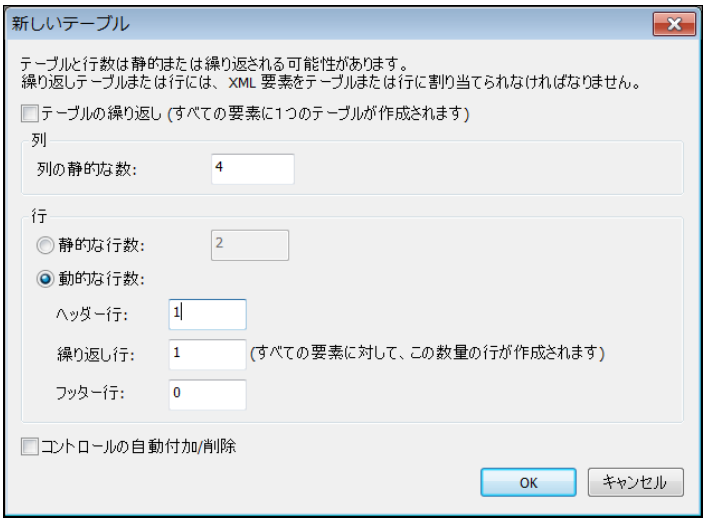
8. ルートノード $XML2 ダブルクリックして、名前を $CUSTOMERS に編集します(*下のスクリーンショット参照*)。
9. $CUSTOMERS デフォルトのファイルの **追加 ダイアログ** ボタンをクリックします。表示されるダイアログから、相対パスのチェックボックスを選択して、デザインへのファイルのパスを相対にします (*下のスクリーンショット参照*)。



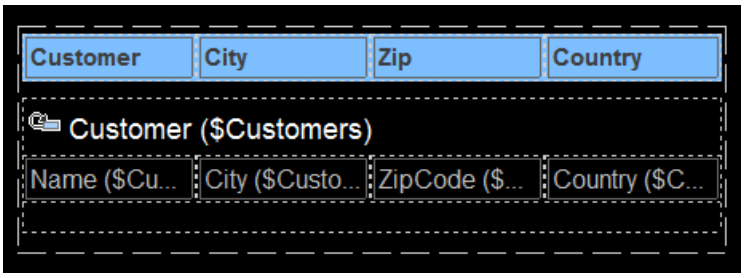
トップページ: カスタマーテーブル

XML データソース**Customers.xml** に現在保管されているすべてのカスタマーの詳細を表示するテーブルを作成します。以下の通りテーブルを作成してください。

1. [コントロール] ペインから[テーブル] コントロールをドラッグして、デザインにドロップします。
2. 表示される新しいテーブルダイアログ内で、(*下のスクリーンショット参照*) テーブルを動的な行数を選択して動的 テーブル として作成します。行います。これにより、データソース内の行要素に対応した数の行と共にテーブルが作成されます。テーブルには、4 つの列と1 つのヘッダー行が作成されるように指定します(*下のスクリーンショット参照*)。 **[OK]** をクリックしてテーブルを作成します。



3. [ページソース] ペインから Customer 要素 を、デザイン内のテーブルの**繰り返し行** アイコンにドロップします。各 customer 要素 は、テーブルの行に対応します。また、 customer 要素は、テーブルのXPath コンテキストノードとなります。
4. [コントロール] ペインから[ラベル] コントロールをドラッグして、ヘッダー行の最初の列にドロップします。ラベルのテキストとして、カスタマーを入力します (*下のスクリーンショット参照*)。同様に以下を含む他の列のためにもヘッダーを作成します: *City*、*Zip*、および*Country*。
5. (各 ラベルの選択中に**[Ctrl]** を押して) 4 つの ラベルすべて を選択します。そして、([スタイル & プロパティ] ペインを使用して) ラベル フォーマットを適用します。
6. [コントロール] ペインから、[ラベル] コントロールをドラッグして、テーブル ボディ行の最初の列にドロップします。[ページソース] ペインから、$CUSTOMERS データソースのCustomer/Name 要素をドラッグして、ラベルにドロップします (*下のスクリーンショット参照*)。このラベルは、カスタマーのName 要素のコンテンツを表示します。
7. 他のテーブル列のコンテンツを同様に作成します: City、ZipCode、および Country 要素 を対応するラベルにドラッグします(*下のスクリーンショット参照*)。



8. すべての4 つの ラベルを(各 ラベルの選択中に**[Ctrl]** を押し)選択して、([スタイル & プロパティ] ペインを使用して) ラベルフォーマットを適用します。

トップページ: アクショングループ、サブページへ移動

前のセクションでは、個別のテーブル行内で各カスタマーの詳細を表示するテーブルが作成されました。カスタマーの現在のオーダーを表示するには、エンドユーザーは、カスタマーの詳細 (名前 または 都市名など) をクリックすることにより、詳細が表示されます。方法としては、選択されたカスタマーのみの現在のオーダーをフィルターするサブページを作成することにより行います。エンドユーザーが トップページ内のカスタマーを選択すると、共有される $XML1 データソースを介して、サブページに伝達されます。 サブページ内で選択されたカスタマー データは、テーブル行 グループの **Visible** プロパティ内で、オーダーをフィルターするために使用されます。 ですから、以下を行う必要があります。 カスタマー 詳細 ラベルをクリックすることにより エンドユーザーは選択を行います:

- クリックされたラベルの カスタマー 詳細を (トップページ および サブページで共有される) $XML1 ツリー に伝達します。
- 選択されたカスタマーのオーダーを表示するサブページへ移動する。

これらは、カスタマー行内のラベルがクリックされると実行されるアクションです。同じアクションのシーケンスが各ラベルのために実行される必要があり、アクションのシーケンス を共通のアクショングループに保存し、この アクショングループを、各 ラベルの ラベルのクリック時 イベントに割り当てることができます。アクションのシーケンスを定義する前に、サブページへ移動 アクションが ターゲットにするサブページを作成する必要があります。

サブページの作成

[ページ] ペイン内のアイテムをクリックして、**サブページの追加** を選択します。をクリックします。追加されたサブページをオーダーに名前変更します (*下のスクリーンショット参照*)。サブページの名前をダブルクリックして編集することができます。

アクショングループの作成

次の アクションから構成されるアクショングループを作成します:

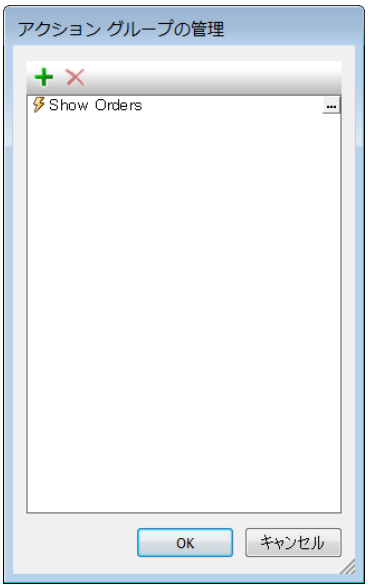
- 2 つの ノードの更新 アクションは、$XML1 データソースの 2 つのノードをアップデートします。
- サブページへ移動 アクションは、オーダー をサブページへ移動します。

アクショングループを作成後、2つの テーブル ボディ行 ラベルの ラベルのクリック時 イベントに割り当てることができます。 ランタイムでは、アクション シーケンスは、エンドユーザーがカスタマーの詳細のラベルをクリックした時に実行されます。

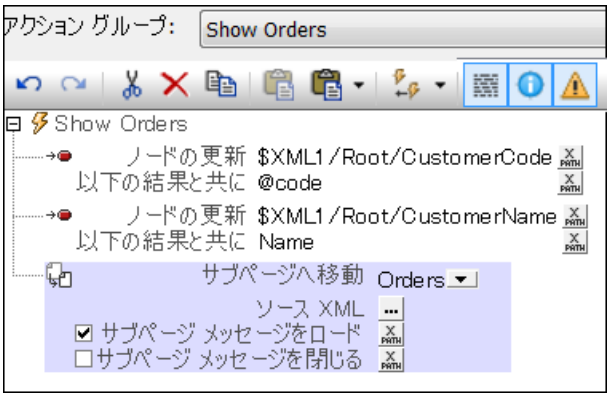
アクショングループを以下の通り作成します:

1. 4 つの テーブル ボディ行 ラベルを右クリックして、**ラベルクリックの時のためのコントロールアクション**を選択します。
2. 表示される アクション ダイアログ 内の左側のペイン内の、アクショングループの **管理** ボタンをクリックします。
3. 表示される管理 グループ アクション ダイアログ内の (*下のスクリーンショット*) メニューバーの、**グループの追加** アイ

コをクリックします。オーダーを表示するため追加されたグループの名前を変更します。名前を編集するために名前をダブルクリックします。(*下のスクリーンショット*)。



4. オーダーの表示の**追加 ダイアログ** ボタンをクリックして、アクショングループ ダイアログを表示します (*下のスクリーンショット*)。
5. 2 つの ノードの更新 アクション とサブページへ移動 アクションを下のスクリーンショットに表示されているとおり定義します。XPath 式 を表示されているとおり入力してください。サブページへ移動 アクションに関しては、コンボボックスのドロップダウンリストからを選択してください。



6. 完了すると**[OK]** をクリックします。

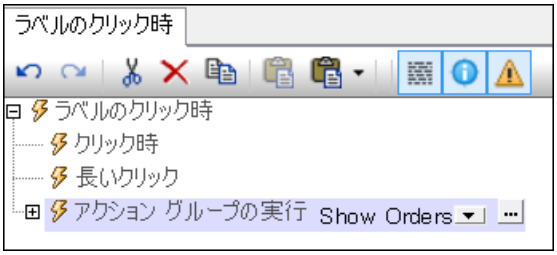
次の アップデートが設定されました (i) `$XML1//CustomerCode` 要素は選択されたカスタマー (`Customers.xml` から) の`Customer/@code` 値 とともに更新されます。および(ii) `$XML1//CustomerName` 要素は、選択されたカスタマー(`Customers.xml` から) の`Customer/Name` 値 とともに更新されます。

アクショングループをイベントに割り当てる

ラベルがクリックされるとオーダーの表示アクショングループが表示されるように指定する必要があります。以下のとおり行います .

1. テーブル ボディ行の最初の列の名前ラベルを右クリックして、**ラベルクリックの時のためのコントロールアクション** を選択します。
2. 表示される アクション ダイアログ からオーダーの表示アクショングループを、下のスクリーンショットに表示されるようにド

ラッグして、長いクリックイベントの下にドロップします。 **[OK]** をクリックして終了します。



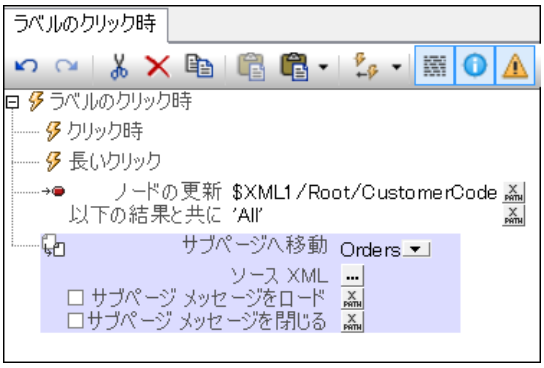
3. テーブル ボディ行の他の3 つのラベルそれぞれにステップ2 を繰り返します。これにより、アクションのオーダーの表示シーケンスが、行の4 つの ラベルのいずれかがクリックされた時に実行されます。

トップページ: すべてのオーダー アクションを表示する

前のセクションでは、エンドユーザーがカスタマーのテーブル内のカスタマーをクリックすると行われるアクションのシーケンスを作成しました。このようなイベントでは、その顧客のオーダーがサブページで表示されます。(オーダー テーブルの可視プロパティを使用して実行する予定です。) このセクションでは、エンドユーザーが、`Orders.xml` データベース内の (すべての カスタマーの) すべてのオーダーを閲覧する場合、クリックすることのできるラベルを作成します。

すべての オーダー ラベルの表示の追加は以下のとおり行います:

1. [コントロール] ペインから [ラベル] コントロールをドラッグして、カスタマー テーブルにドロップします。すべてのオーダーを表示するとラベルのテキストとして入力します (*下のスクリーンショット参照*)。
2. [スタイル & プロパティ] ペインからのプロパティと共にラベルをフォーマットします。
3. [アクション] ダイアログ をコントロールの コンテキスト メニューコマンド **コントロール アクション ラベルのクリック時 アクション** 使用して開き、ラベルのクリック時 イベントのためのアクションのシーケンスを追加します (*下のスクリーンショット参照*)。



`$XML1/Root/CustomerCode` 要素は、エンドユーザーがすべての *オーダー* ラベルをクリックすると、`'All'` をアップデートするために定義されていることに注意してください。

4. **[OK]** をクリックして終了します。

サブページ: データソース

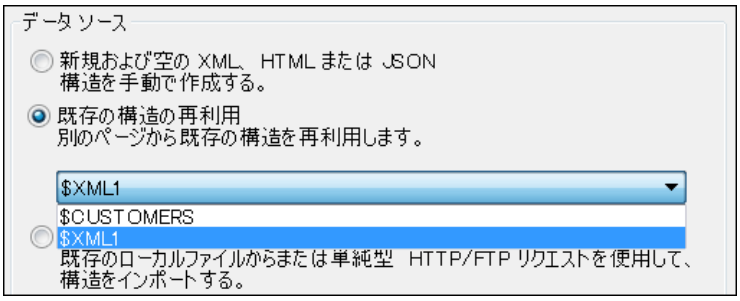
サブページへ移動 アクションを定義する際にこのステップはで必須であったため、サブページはすでに作成されています。 サブページはオーダーという名前で、次のデータソースを有することになります:

- **$ORDERS** は、構造 および `Orders.xml` のコンテンツを有します。 このデータソースは、`Orders.xml` 内で含まれるオーダーを表示するために必要です。
- **$XML1**: このデータソースは、トップページと共有されています。 このデータソースは、エンドユーザーが閲覧を必要とすることを示す情報を含むため、サブページで必要とされます。特に、トップページでユーザーが選択したカスタマーのコードを含んでいるためこのデータソースは重要です。
- **$CUSTOMERS**: このデータソースは、トップページで使用されているものと同じで、トップページと共有されています。 このデータソースは、カスタマーの名前などのカスタマー情報を取得するためにサブページで使用されています。

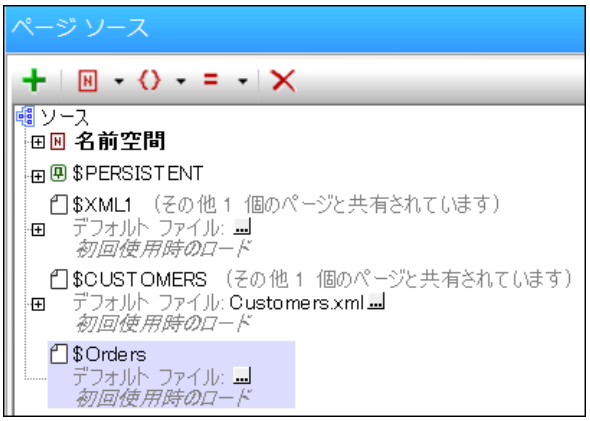
データソースの追加

3 つのデータソースの追加は以下のとおり行います:

1. [ページソース] ペイン内の **ソースの追加** アイコンをクリックして、*既存の構造の再利用* を選択します(*下のスクリーンショット参照*)。
2. オプションのコンボボックス(*下のスクリーンショット*) から、`$XML1` を選択して、**[OK]** をクリックします。



   `$XML1` ソースが追加されます。名前の横には、他のページと共有されていることを知らせるメッセージが表示されます。 トップページ内の `$XML1` の構造 および コンテンツは既に存在します。
3. 共有される第2番目のデータソース `$CUSTOMERS` を同様に追加します。構造、コンテンツ、および デフォルトファイルは、トップページ内に作成されることに注意してください。
4. [ページソース] ペイン内の **ソースの追加** アイコンをクリックして、*ファイルからインポートされた新しい XML または HTML 構造* を選択します。 ファイル `Orders.xml` を検索して、**[開く]** をクリックします。ファイルをデプロイするか問われると、**[はい]** を選択します。 `$XML2` という名前のデータソースが作成されます。
5. ルート ノード `$XML2` をダブルクリックして、名前 を `$ORDERS` に編集します(*下のスクリーンショット参照*)。
6. `$ORDERS` デフォルトファイルの **[追加ダイアログ]** ボタンをクリックします。表示されるダイアログから、ファイル パスをデザインに相対させるために相対パスのチェックボックスを選択し、**[OK]** をクリックします (*下のスクリーンショット参照*)。

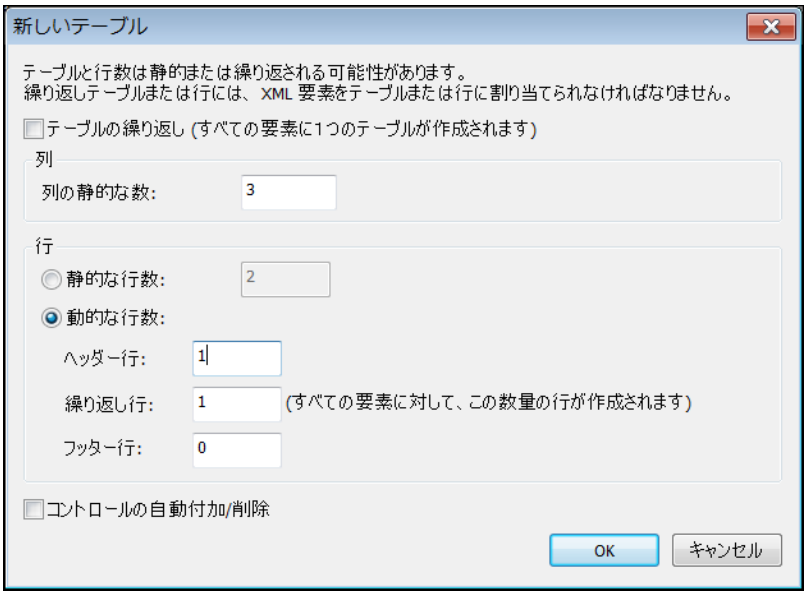


データソースが追加されると、サブページのデザインを作成する準備が整いました。

サブページ: オーダー テーブル

エンドユーザーにより、トップページ上で選択されたカスタマーのオーダーを表示するテーブルを作成します。オーダーは、XML データソース`Orders.xml` に保管されます。オーダー テーブルを以下のとおり作成します。

1. [コントロール] ペインから[テーブル] コントロールをドラッグしてデザインにドロップします。
2. 表示される新しいテーブル ダイアログは、テーブルを動的 テーブルとして作成します(*下のスクリーンショット参照*)。これは、*動的な行数* を選択することで行います。これにより、データソース内の対応する行要素が必要なだけテーブルに作成されます。テーブルに3 つの列 および1つのヘッダー行が作成されるように指定します(*下のスクリーンショット参照*)。 **[OK]** をクリックして、テーブルを作成します。



3. [ページ ソース] ペインから`Order` 要素 をデザイン内のテーブルの**繰り返し行** アイコンにドラッグします。各 `order` 要素は、テーブルの行に対応します。`order` 要素は、テーブルのXPath コンテキスト ノードになります。
4. [コントロール] ペインから[ラベル] コントロールをドラッグして、ヘッダー行の最初の列にドロップします。ラベルのテキストにカスタマーと入力します (*下のスクリーンショット参照*)。同様に以下の他の列のためにもヘッダー を作成します: *オーダー* および数量 .
5. (各ラベルを選択中に**[Ctrl]** を押す) ことで、3 つすべての ヘッダー ラベルを選択し、[スタイル & プロパティ] ペイン内でラベルフォーマットを適用します。
6. [コントロール] ペインから[ラベル] コントロール をドラッグして、テーブル ボディ行の3 つの列 にそれぞれドロップします。
7. (各ラベルを選択中に**[Ctrl]** を押す) ことで、3 つすべての テーブル ボディ行 ラベルを選択し、[スタイル & プロパティ] ペイン内でラベルフォーマットを適用します。

テーブル ボディ行のラベルは、テーブルセルに入力されました。XPath 式 を使用して指定するテキストは次のセクションで説明されています。

ラベルテキストのための XPath 式 の作成

テーブルの出力は以下に表示されています。異なる列のコンテンツに注意してください。

Customer	Order	Amount
789: JuniorsRV	002/2015-04-03	EUR 8345.60
789: JuniorsRV	005/2015-04-06	EUR 2786.45

ラベルのテキストのためのXPath 式を作成するには、最初にラベルを選択します。[スタイル & プロパティ] ペイン内から、ラベルのText プロパティを選択して、ペインのメニューバー内のXPath アイコンをクリックします。 表示されるXPath ダイアログに、XPath 式 対応するを入力します。XPath コンテキスト ノードは、対応するド$ORDERS/Orders/Order 要素であることに注意してください。

カスタマー列のため

```
if ($XML1/Root/CustomerCode != 'ALL')
then concat(CustomerCode, ': ', $XML1/Root/CustomerName)
else concat(CustomerCode, ': ', for $i in CustomerCode return $CUSTOMERS/Customers/
Customer[@code=$i][1]/Name)
```

- 選択されたカスタマーのオーダーのテーブルのためには、カスタマーの名前は、$XML1 ツリーより取得されます。
- すべてのオーダーを表示するテーブルのためには、カスタマーの名前は、$ORDERS ツリー内のカスタマーコードをキーとして使用し$CUSTOMERS ツリーより取得されます。（カスタマーコードは両方のツリーに存在します。）

オーダー列のため

```
concat(@number, '/', OrderDate)
```

数量列のため

```
concat(Currency, ' ', OrderAmount)
```

オーダー テーブルの作成の完了

各列のコンテンツを定義した後、[スタイル & プロパティ] ペインからプロパティを使用してラベルをフォーマットします。テーブル 行 グループの可視プロパティを指定して、トップページで選択されたカスタマーのみがテーブルに表示されるようにします。 可視プロパティに関しては、次のセクションで説明されています。

サブページ: 可視性プロパティ

オーダーサブページ内に作成されたオーダーテーブル は、データソースOrders.xml 内の、各 Order 要素のために行を生成する 動的 テーブルです。Order 要素は、データファイル内で発生する順序で表示されています。しかし、Order 要素 が表示される順序を管理することもできます。これは、テーブル行 グループの**Visible** プロパティを使用して行います。プロパティは、表示するOrder 要素を選択するXPath 式を取ります。

Visible プロパティのXPath 式 を設定するには、デザイン内の繰り返し行を選択してください。そして、[スタイル & プロパティ] ペイン内で、テーブル行グループのプロパティに移動し、**Visible** プロパティの**XPath** アイコンをクリックします。表示される[XPath/XQuery 式の編集] ダイアログ に次のXPath を入力します:

```
if ($XML1/Root/CustomerCode != 'All') then CustomerCode=$XML1/Root/CustomerCode else
CustomerCode
```

このXPath 式 は以下のように作動します:

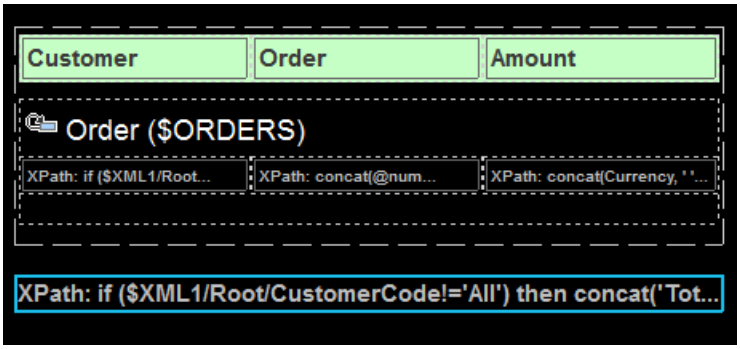
1. 式のif 句は、要素 $XML1/Root/CustomerCode が文字列 Allを含むかをテストします。
2. 要素 $XML1/Root/CustomerCode が文字列 Allを**含まない場合、**$XML1/Root/CustomerCode 要素のコンテンツに等しいCustomerCode 要素 コンテンツを持つすべてのOrder 要素のOrderAmount 要素はが選択されます。これは、エンドユーザーにより選択されたカスタマーのOrder 要素の数量になります。カスタマーのCustomerCode は、$XML1 データソースに保管されています (*トップページ: アクショングループ: サブページへ移動*を参照)。
3. 要素 $XML1/Root/CustomerCode 文字列 Allを**含む**場合、子 CustomerCode 要素を持つすべてのOrder 要素 が選択されます。実質上、これはデータファイル内のすべてのOrder 要素です。

メモ **Visible** プロパティを使用する利点は、選択された要素のみを含むテーブルを生成する他の方法よりも、簡単かつ効果的な代替であることです。

サブページ:XPath 内の総額

デザインを完了するには、総額を表示するラベルを表示されたオーダーに追加します。以下のとおり行います:

1. [コントロール] ペインから [ラベル] コントロールをドラッグしてオーダーテーブルにドロップします (*下のスクリーンショット参照*)。
2. [スタイル & プロパティ] ペイン内のコントロールの Text プロパティのコントロールの **XPath** アイコンをクリックします。
3. 表示される [XPath/XQuery 式の編集] ダイアログに、総額を計算する XPath 式 (*式は下に与えられています*) を入力します。そして、**[OK]** をクリックします。



総額を計算する XPath 式

以下の 2 つのイベントの総額を計算します: (i) 選択されたカスタマーのオーダー、および (ii) すべてのオーダー。これは、次の XPath 式を使用して行うことができます:

```
if ($XML1/Root/CustomerCode!='All')
then concat('Total: ', xs:decimal(sum ($ORDERS//Order[CustomerCode=$XML1/Root/CustomerCode]/
OrderAmount)))
else concat('Total: ', xs:decimal(sum ($ORDERS//OrderAmount)))
```

この XPath 式は、以下のとおり作動します:

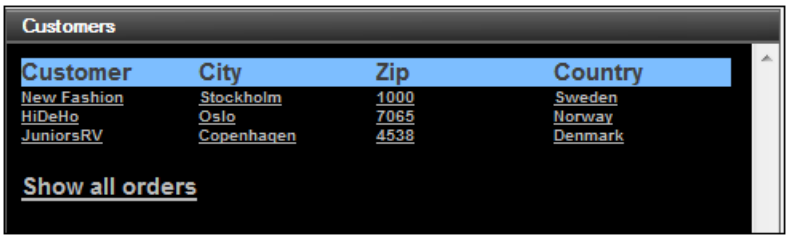
1. 式の if 句は、要素 $XML1/Root/CustomerCode が文字列 All を含むかをテストします。
2. 要素 $XML1/Root/CustomerCode が文字列 All を**含まない場合**、$XML1/Root/CustomerCode 要素のコンテンツに等しい CustomerCode 要素コンテンツを持つすべての Order 要素の OrderAmount 要素はが選択されます。これは、エンドユーザーにより選択されたカスタマーの Order 要素の数量になります。カスタマーの CustomerCode は、$XML1 データソースに保管されています (*トップページ: アクショングループ: サブページへ移動*を参照)。
3. 要素 $XML1/Root/CustomerCode が文字列 All を**含む場合**、すべての OrderAmount 要素が選択されます。

選択された OrderAmount 要素は、XPath の sum() 関数を使用して集計されます。sum() 関数は、xs:double 型を使用するため、xs:double 数値を返します。数量は通貨が必要とする 2 つの小数点とともに表示されます。ですから、xs:decimal 型コンバーターを使用して、xs:double に四捨五入し、2 つの小数点を使用します。

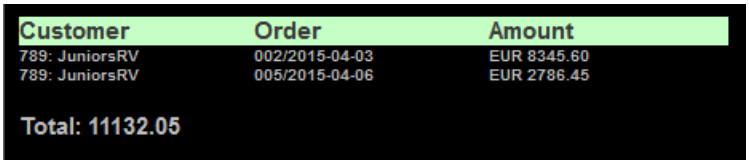
シミュレーションとテスト

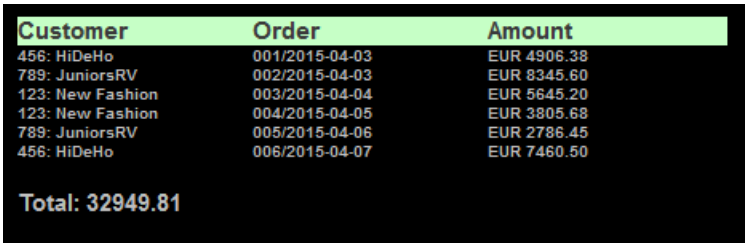
デザインが完了すると (**[F5]** を押して) シミュレーションを実行して、デザインをテストします。下のスクリーンショットは、MobileTogether Designer シミュレーター内の トップページ および サブページ を表示しています。

トップページ: カスタマー



サブページ: (選択されたカスタマー および すべてのカスタマーのための) オーダー



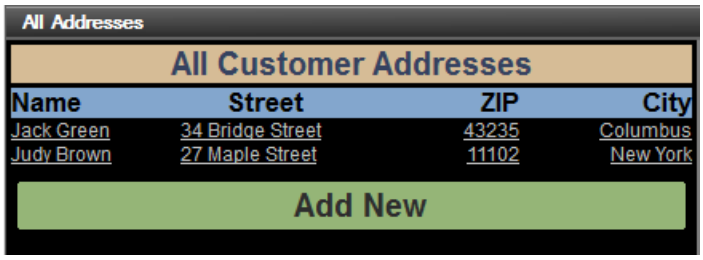


これだけです！

4.7 レコードの追加と編集

このチュートリアルの目的は以下のとおりです:

- 顧客の記録をカスタマー データベースに追加する
- データベース内のレコードを編集する



デザインのプラン

デザインは以下のとおり構成されています:

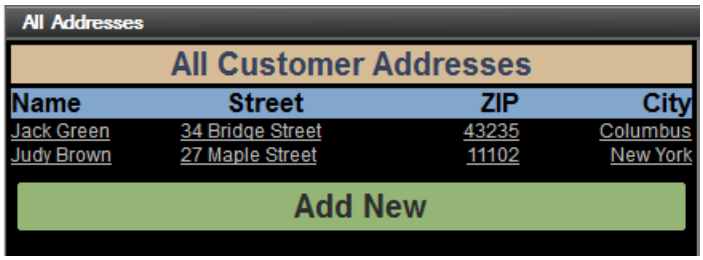
- XML データソースが使用されます。`$PERSISTENT` ツリーは、カスタマー データベースを保管するために使用されます。`$EDIT` ツリーは、現在編集されている1つのレコード(`Customer` 要素) を保管するために使用されます。
- `$EDIT/Customer` 要素は、 **「新規を追加」**ボタンがクリックされると作成されます。
- **「保存」**ボタンがクリックされると、`$EDIT/Customer` 要素が`$PERSISTENT` ツリーの最後の要素として追加されます。レコードは、(記録が後に新規として追加、または、編集のために使用できるようにコードされ) `$EDIT` ツリーから削除されます。 .
- カスタマー データベースは、`$PERSISTENT` ツリー内で`Customer` 要素を表示するテーブルとして表示されます (*上のスクリーンショットを参照してください*).
- カスタマー データベース内のレコードのフィールドがクリックされると、編集を実行する`$EDIT` ツリー内に記録がロードされます。
- 記録がこのように編集された後、「保存」ボタンがクリックされると、レコードは、`$PERSISTENT` ツリーの元の場所に保存されます。

チュートリアルファイル

完成したチュートリアルファイルは *マイドキュメント*MobileTogether フォルダー:
`MobileTogetherDesignerExamples\Tutorials\AddEditRecords\AddAndEditRecords.mtd`
内にあります。このチュートリアルを読みながら、このファイルを開きデザインの設定をチェックしてください。

デザインページ

ソリューションが開始すると、「*All Addresses*」という名前のメインページが表示されます。このページには以下が含まれます: (i) (`$PERSISTENT` ツリー内に含まれる)顧客のデータベース内の全ての記録を表示するテーブル、および(ii) ユーザーが顧客のデータベースに新規のレコードを追加することのできる **「新規を追加」**ボタン。

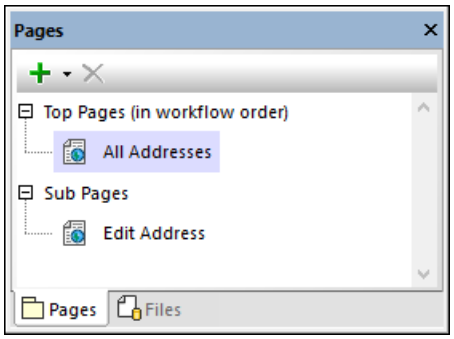


「新規を追加」ボタンがユーザーによりクリックされると、ボタンのアクションにより「*Edit Address*」という名前のサブページに移動します(*下のスクリーンショットを参照*)。このページでユーザーは新規の顧客の情報を入力することができます。サブページはワークフローが2つの個別のページ間を移動するために必要とされます。

新規の顧客のデータが*Edit Address* ページ内に入力されると、ユーザーは、**「保存」**をクリックすることができます。または、ユーザーは、**「キャンセル」**をクリックして中断することができます。どちらかのボタンがクリックされると、ワークフローはサブページを終了し、メインページに戻ります。これらのボタンがクリックされた時に実行されるアクションに関しては、新規のレコード追加するを参照してください。

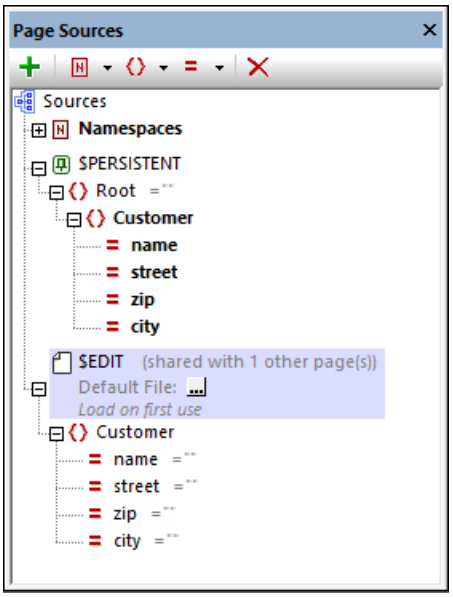
ページの作成

トップページとサブページは、ページペイン内で作成されます（*下のスクリーンショットを参照*)。



データソース

カスタマーデータベースのレコードは、**$PERSISTENT** ツリー内に保管されています。新規のレコードが既存のレコードに追加され、または、編集される場合、このレコードは、**$EDIT** ツリー内にコードされます。両方のツリーで、各レコードは、**Customer** 要素内に保管されます。 ツリーはページソースペインのツールバーアイコンを使用して手動で作成、または、ツリー内のノードのコンテキストメニューを使用して作成されます (*下のスクリーンショットを参照*) 。

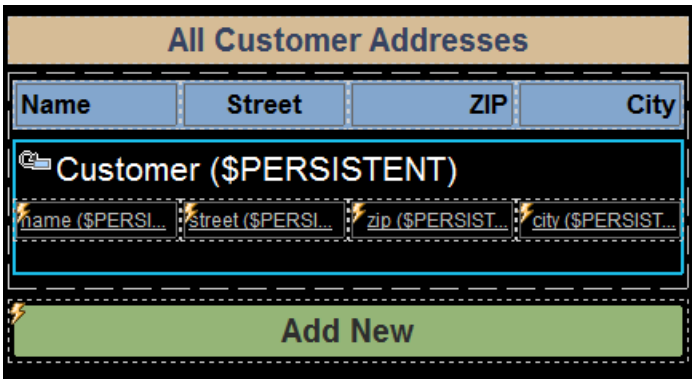


以下の点に注意してください:

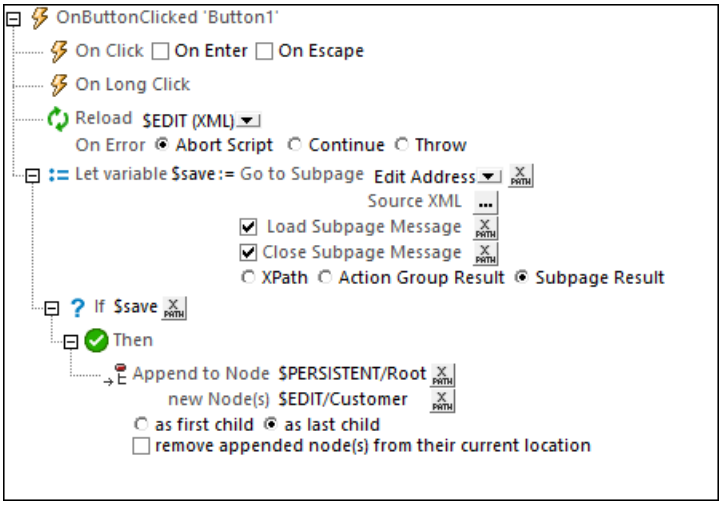
- 両方のツリー内で、**Customer** 要素は、単一の顧客の記録に対応します。
- 各顧客のデータは、顧客の**Customer** 要素の属性内に保管されます。
- **$EDIT** ツリーは、デザインの両方のページにより共有されます。これは、ツリー内のデータが両方のページに対して使用できることを意味します。
- **$EDIT** ツリー内で、これらの属性のそれぞれは、ツリーがロードされた時に空白の文字列の固定値を持つように設定されます。この設定 **「ロード時に存在を確認する」** は、各属性のコンテキストメニュー内で使用することができます(属性を右クリックすることにより実行することができます)。この切り替えには以下を意味します: **「新規を追加」** ボタンがクリックされると、空白の属性の値を持つこのツリーを (再) ロードする必要があります。新規の顧客の詳細は、ですから、空白の顧客の記録に追加することができます。
- **Customer** 要素とその子属性が太字で表示されている理由は、これらがデザイン内のページソースリンクとして作成されているからです。これらは、メインページ上のテーブルの列内の顧客のデータベースを表示しています (*下のスクリーンショットを参照*)。

新規レコードを追加する

ユーザーがソリューションを開くと、カスタマーデータベースは空白です。レコードを追加するには、**新規を追加**」ボタンをクリックします (*下のスクリーンショットを参照*)。

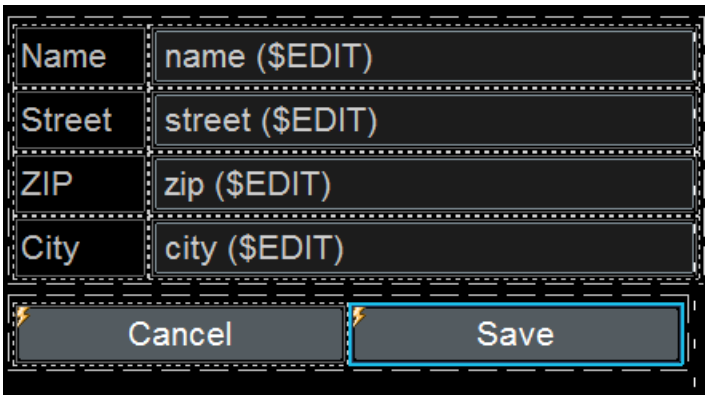


新規を追加」ボタンには、下のスクリーンショット内で表示される2つのボタンのクリック時 アクション(再ロード とLet) が存在します。



これらの2つのアクションは以下のステップのシーケンスで実行されます:

1. 再ロード アクションは、`$EDIT` ツリーを再ロードします。このツリーのノードは、空白の文字列の固定値と共にコードされるように定義されているため、顧客の記録の全てのフィールドは空白です。
2. Let アクションはサブページ *Edit Address* に移動し、結果を取得する `$save` という名前の変数を作成します (*下のスクリーンショットを参照*)。

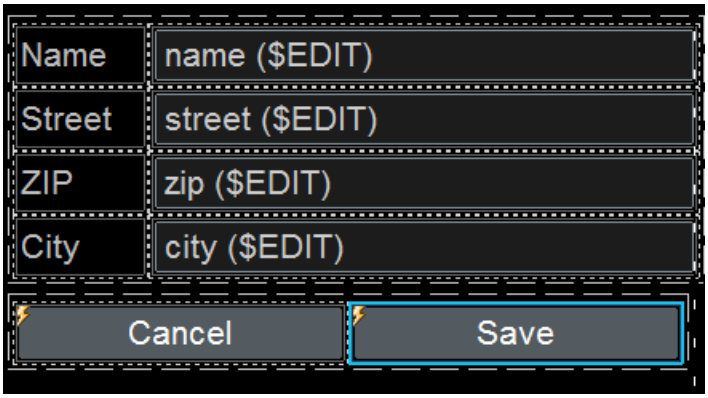


サブページの結果は、ページの **「保存」** ボタンがクリックされると返されます。 **「保存」** ボタンは、サブページを閉じるアクションを実行し、`Customer` ノード。である結果を返します。その後、このノードは、`$save` 変数に保管されます。

3. If-Then アクションは、次に`$save` 変数が存在するかをチェックします。
4. `$save` 変数が存在すると、アクションの`Then` 句が実行されます。これにより、(ノードの追加 アクションを使用して) `$PERSISTENT/Root` 要素の最後の子ノードとして`$EDIT/Customer` 要素が追加されます。このようにして、*Edit Address* に追加内に新しい顧客のデータが追加されると、サブページが保存されます。顧客の記録全体は、`$PERSISTENT` ツリー内の顧客のデータベースの最後の記録として追加されます。
5. **「キャンセル」** ボタンは、結果を返すことなくサブページを閉じるアクションを実行します。顧客のデータベースを変更することなくメインページに戻ることができます。

新規レコードデータを入力する

Edit Address (サブ) ページは、下で説明されている心気の顧客のデータを入力するために使用されます (*下のスクリーンショットを参照*) :



- 各顧客のデータ編集フィールドは、`$EDIT` ツリー内のノードであるページソースリンクに関連付けられています。
- **「保存」**ボタンがクリックされると、`$EDIT/Customer` 要素全体が`$PERSISTENT/Root` 要素の最後の子として追加されます。このメカニズムについては、前のセクション新規レコードを追加するで説明されています。
- **「キャンセル」**ボタンは、結果を返すことなくサブページを閉じます。**「キャンセル」**ボタンのサブページを閉じるアクションを使用して定義することができます。

全てのレコードを表示する

顧客のデータベースの記録は、動的な行が存在するテーブル内で表示されている`$PERSISTENT` ツリー内で表示されます (*下のスクリーンショットを参照*)。

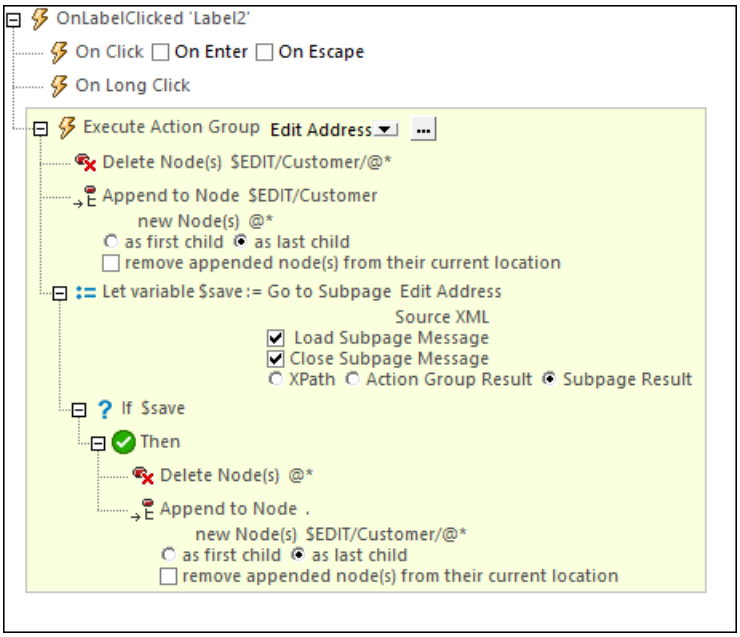
テーブルは以下のように定義されます:

- テーブルには、動的な行の外に存在するシンプルなヘッダー行があります。
- 動的な行は、`$PERSISTENT/Root/Customer` データソース要素にリンクされている繰り返し行です。ソリューション内の結果、新規の行が各 `Customer` 要素のために作成されます。
- 各テーブル列のセルには、それぞれ`Customer` 要素: `name`、`street`、`zip`、と`city` の異なる属性ノードにリンクされているラベルコントロールが含まれています。
- 各ラベルは、ラベルのクリック時 イベントのために定義されている同じアクションのセットを有しています (*下のスクリーンショットを参照*)。これらのアクションは、各顧客の記録が個別に編集でき、顧客のデータベースに保存できるようになっています (*既存のレコードの編集のセクションを参照してください*)。アクションのセットは、4つ全てのラベルと同じのため、4つのラベルのために再利用する事のできる単一のアクショングループ内で定義されています。

住所を編集するアクショングループ

各レベルの ラベルのクリック時 イベントのために追加されるアクショングループは、下のスクリーンショット内で表示され、説明され提案す。

メモ　アクショングループ内に追加されたノードは、`$PERSISTENT/Root/Customer` ノードです。これはアクショングループ内の全てのXPath 式のコンテキストノードです。

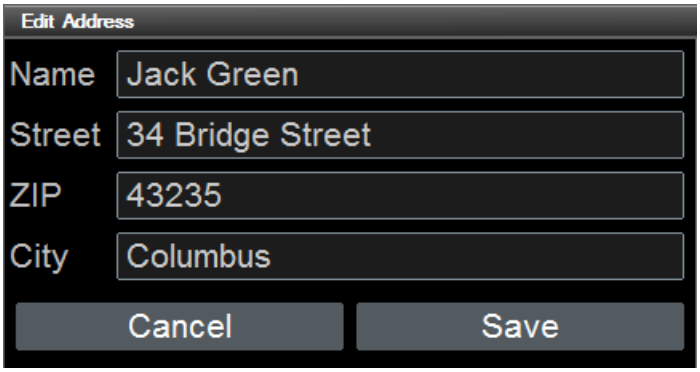


アクショングループ内のアクションは以下を行います:

- ノードの削除 アクションは、`$EDIT/Customer` を削除します。このデータノードには、現在編集されている顧客の記録が含まれています。
- `$EDIT/Customer` ノードの削除された属性ノードは、編集する記録の属性ノードと置き換えられます。これらのノードは、現在のコンテキストノード(`$PERSISTENT/Root/Customer`)の属性ノードです。この置換えは、ノードの追加 アクションを使用する事におこなうことができます。
- Let アクションは、サブページ*Edit Address* に移動する`$save` という変数を作成し、結果を取得します (*下のスクリーンショットを参照*) 。サブページの結果は、ページの **「保存」**ボタンがクリックされると返されます。 **「保存」**ボタンは サブページを閉じる を実行し、`Customer` ノードである結果を返します。このノードは、`$save` 変数に保管されます。
- If-Then アクションは、`$save` 変数が存在するかをチェックします。
- `$save` 変数が存在する場合、アクションの`Then` 句が実行されます。これにより、顧客のデータベースの現在の`Customer` 要素の属性ノードが削除され、(ノードの追加 アクションを使用して) `$EDIT/Customer` 要素からの属性ノードが顧客のデータベースの顧客のデータに追加されます。
- **「キャンセル」** ボタンがクリックされると、結果を返すことなくサブページを閉じるが実行されます。顧客のデータベースを変更することなくメインページへ戻る事ができます。

既存のレコードを編集する

顧客の記録内のフィールドをユーザーがクリックすると、ソリューションは、ユーザーが特定の記録を編集することのできる*Edit Address* サブページに移動します(*下のスクリーンショットを参照*)。



- **「保存」**をクリックすると、顧客のデータベースに編集された記録が保存され、ソリューションはメインページに移動します。
- ユーザーが「キャンセル」をクリックすると、元の記録は変更されず、ソリューションはメインページに戻ります。

これら2つのボタンのボタンのクリック時 アクションは、新規のレコード追加するのセクションで説明されています。サブページは以下の場合呼び出されます (i) 新規の記録が追加された場合、または、(ii)記録が変更された場合。両方のデータへの変更は、`$EDIT` ツリー内に保存されます。ボタンアクションは両方の場合とも同じです。

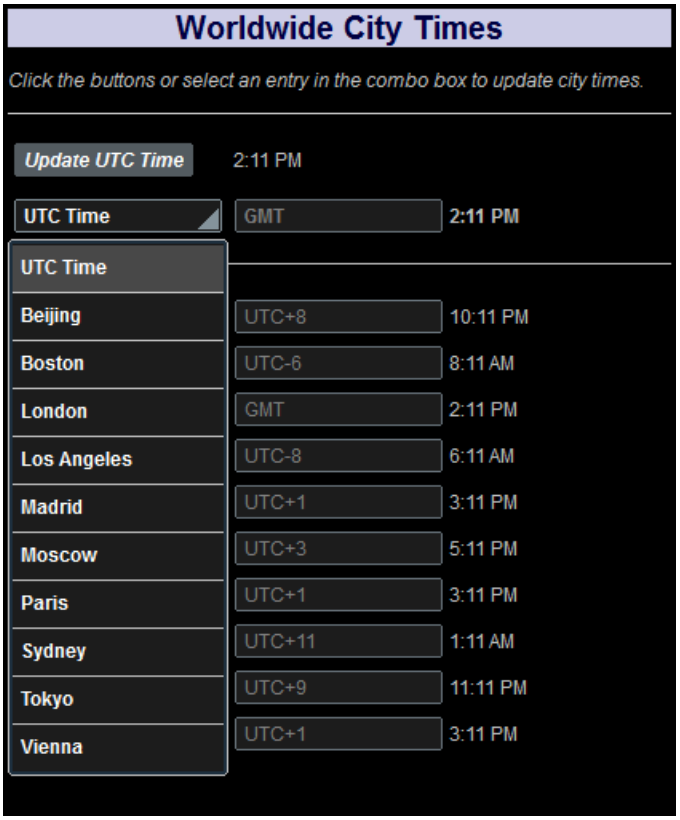
顧客のデータベース内の古い顧客のデータと変更されたデータの置換えに関しては、関連するアクションを実行するアクショングループの説明を参照してください。

4.8 SOAP リクエスト

このチュートリアルは、SOAP によるデータを使用し構築されたデザイン(`CityTimesViaSOAP.mtd`)の作成方法を説明します。デザインはWSDL ファイル(`TimeService.wsdl`)からSOAP リクエストを生成します。リクエストはWeb サービス(`http://www.nanonull.com/TimeService`)に送信され、サービスのSOAP レスポンスが使用されデザインのXML ツリー内のノードを更新します。

Web サービスは以下を提供します (i) 現在のUTC 時刻、(ii) 特定のタイムゾーン内の現在の時刻。タイムゾーンは関連するSOAP リクエストのパラメーターとして提供されます。デザインの目的はインターフェイスの以下の項目を更新することです (i) UTC 時刻、(ii) 選択された都市の時刻。UTC 時刻に関しては、(パラメーター無しの) SOAP リクエストがWeb サービスに送信されます。レスポンスはXML ノードを更新するために使用されます。都市の時刻に関しては、都市のタイムゾーンがSOAP リクエストのパラメーターとして送信されます。ノードはレスポンスにより更新されるため、また更新されたノードは特定のコントロールのページソースリンクのため、更新された時刻はすぐにソリューション内に表示されます。

インターフェイスの外観は下のスクリーンショットのようになります。スクリーンの下の部分は選択された都市のリストを含んでいます。*UTC 時刻の更新*ボタンまたは*<City>* ボタンをクリックすることにより、対応する時刻を更新します (*スクリーンショットを参照*)。コンボボックス内の都市を選択することにより、表示される都市の時刻を更新します。ページが更新されると、都市の時刻も自動的に更新されます。このメカニズムはページの更新のセクションで説明されています。



チュートリアルファイル

マイドキュメントMobileTogether フォルダーにこのチュートリアルで使用されるファイルを検索することができます: **`MobileTogetherDesignerExamples\Tutorials\SOAPRequests`**。

- `CityTimes.xml`: これは、都市とタイムゾーンのリストを含んだXML データファイルです。デザインによりリクエスト

されたデータの構築のために使用されます。

- **`TimeService.wsdl`**: これは、SOAP リクエストによりWeb サービスが生成されるためのWSDL ファイルです。
- **`CityTimesViaSOAP.mtd`**: これは完成されたMobileTogether デザインファイルです。このファイルを開き、このチュートリアルの説明を読む際に参照してください。 MobileTogether Designer シミュレーションは**[F5]** を押すことで行うことができます。

デザイン内のファイルパスは相対的です、またXML とWSDL データファイルはサーバーにデプロイされていません。ですから、これら3 つファイルをフォルダーにコピーすると、MobileTogether Designer 内で正確にシミュレーションを行うことができます。

チュートリアルの構造

このチュートリアルは以下のセクションに整理されています:

- XML ページソースは、デザインの構造とデータのために使用されるXML データソースについて説明しています。
- デザインコンポーネントは、デザインの異なるコントロールとアクションについて説明しています。
- ページの更新 は、ページ更新のためのアクションにより定義されている、表示される値がどのように自動的に更新されるかについて説明しています。

XML ページソース

1つのXML ページソース(**$XML1**) が、デザインのために必要なデータを保管し構築するために必要です。以下から構成されるXML ファイル**CityTimes.xml** を使用します (*下のリスト参照*):

- ユーザーがボタンを押すと、SOAP リクエストを介してUTC 時刻 と共にアップデートされる**UTC** 要素 (*下のリスト参照*)。このノードはユーザーにUTC 時刻を表示するために使用されます。
- 自動的なページの更新の間で秒単位の時間を保留するための**RefreshTime** 要素(*下のリスト参照*)。ユーザーはこのノードの値を選択することができます。
- ユーザーが選択する都市の詳細(**Name**、**TimeZone**、および**Time**) を含む**SelectCity** 要素(*下のリスト参照*)。ユーザーがコンボボックスのドロップダウンリストから選択した**SelectCity** 要素の**Name** 子要素が都市名と共に更新されます。コンボボックスの選択が行われると、 **SelectCity/TimeZone** 要素が**Cities** データベースより更新されます。 **SelectCity/TimeZone** 要素は、ユーザーが都市を選択すると送信されるWeb サービスのSOAP リクエストに対するレスポンスにより更新されます。
- **Cities** 要素 は、選択された都市の詳細(**Name**、**TimeZone**、と**Time**) (*下のリスト参照*) を含むデータベースです。(希望する場合は都市を追加することもできます)。Web サービスは都市のタイムゾーンを選択された都市の現在の時刻を返すためにアクセスします。データベースはですから、タイムゾーンの情報を含む必要があります。 **City** 要素の**Time** 子要素は、現在の時刻を保管するために使用され、SOAP リクエストのレスポンス内のWeb サービスより取得されます。上のUTC 時刻では、ユーザーがボタンを押すと、またはコンボボックス内の都市を選択すると、SOAP リクエストが送信されます。

ドキュメントの構造を表示するXML ページデータソースCityTimes.xml の短いバージョン

```
<CityTime xmlns="http://www.Nanonull.com/TimeService/">
        <UTC>12:00 AM</UTC>
        <RefreshTime>60</RefreshTime>
        <SelectCity>
                <City>
                        <Name>UTC Time</Name>
                        <TimeZone>GMT</TimeZone>
                        <Time>12:00 AM</Time>
                </City>
        </SelectCity>
        <Cities>
                <City>
                        <Name>Beijing</Name>
                        <TimeZone>UTC+8</TimeZone>
                        <Time>12:00 AM</Time>
                </City>
                ...
        </Cities>
```

XML ページデータソースのフルリストCityTimes.xml

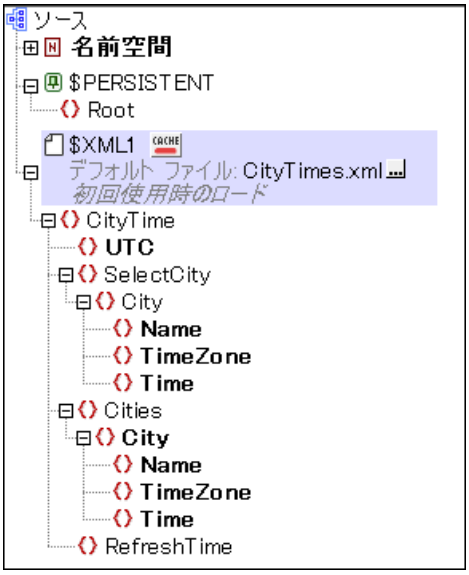
*マイドキュメントフォルダー*のMobileTogether フォルダー内: **MobileTogetherDesignerExamples\Tutorials\AltovaProducts.xml**.

```
<?xml version="1.0" encoding="UTF-8"?>
<CityTime xmlns="http://www.Nanonull.com/TimeService/">
        <UTC>12:00 AM</UTC>
        <SelectCity>
                <City>
                        <Name>UTC time</Name>
```

```
                <TimeZone>GMT</TimeZone>
                <Time>12:00 AM</Time>
            </City>
        </SelectCity>
        <Cities>
            <City>
                <Name>Beijing</Name>
                <TimeZone>UTC+8</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Boston</Name>
                <TimeZone>UTC-6</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>London</Name>
                <TimeZone>GMT</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Los Angeles</Name>
                <TimeZone>UTC-8</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Madrid</Name>
                <TimeZone>UTC+1</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Moscow</Name>
                <TimeZone>UTC+3</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Paris</Name>
                <TimeZone>UTC+1</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Sydney</Name>
                <TimeZone>UTC+11</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Tokyo</Name>
                <TimeZone>UTC+9</TimeZone>
                <Time>12:00 AM</Time>
            </City>
            <City>
                <Name>Vienna</Name>
                <TimeZone>UTC+1</TimeZone>
                <Time>12:00 AM</Time>
            </City>
        </Cities>
    </CityTime>
```

XML ページソースの追加:

1. ページソースペイン内の**[ページソースの追加]** アイコンをクリックします(上のスクリーンショット参照)。ファイルからインポートされた新規 XML、HTML または JSON 構造 を選択します。次のスクリーンで ページソース設定を確認し、(デフォルトを変更することなく) **[完了]** をクリックします。



2. 開くダイアログが表示されます。`CityTimes.xml` ファイルを参照します (ロケーションを確認して) **[開く]** をクリックします。`$XML1` ツリーが作成されます。このツリーは、`CityTimes.xml` ファイルがデフォルトファイルとして設定されており、XML ツリーはデフォルトのファイルの構造を持ちます。

SOAP レスポンスと XML ツリー内のノードの名前空間

この特別な Web サービスからの SOAP レスポンス内のノードにはプレフィックスはなく、以下の名前空間に属します: `http://www.Nanonull.com/TimeService/`。ですから、SOAP レスポンスノードに正確にターゲットするには、デザインの XPath 式は この名前空間に対する XPath デフォルト名前空間 に設定されている必要があります(下のスクリーンショット)。

この名前空間を XPath デフォルトの名前空間として設定することは、デザインの XPath 式内の**全てのプレフィックス無しの**ノードがこの名前空間内に存在すると考えられます。XML ツリーのノードにプレフィックスがない場合 (XML ツリーの場合のように)、これらのツリーノードがプレフィックス無しに XPath 式に入力された場合、これらのノードは、XPath デフォルトの名前空間と考えられます: `http://www.Nanonull.com/TimeService/`。

このため、名前空間 `http://www.Nanonull.com/TimeService/` を ノード XML ツリーにも割り当てます。上の XML リストでも、ドキュメントのルート要素が名前空間に割り当てられていることに気が付くでしょう

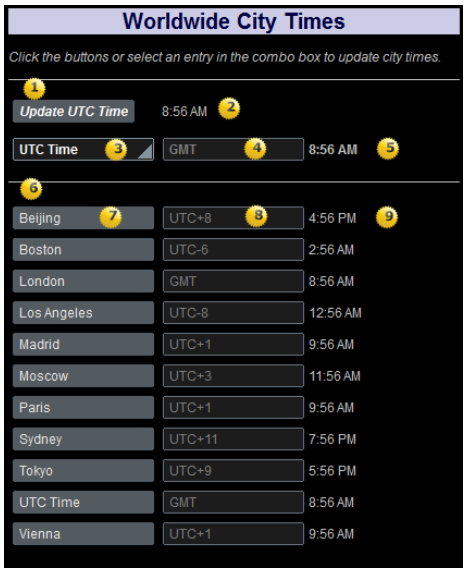
```
<CityTime xmlns="http://www.Nanonull.com/TimeService/">
```

- この名前空間がXML ドキュメント全体のスコープであるため、また子孫要素の他の名前空間マッピングにより上書きされていないため、名前空間は ドキュメント全体に適用されます。
- ドキュメントの名前空間宣言にはプレフィックスが無く、この名前空間がXML ドキュメントのデフォルトの名前空間になります。この結果、プレフィックス無しのノードがこの名前空間内に存在します。

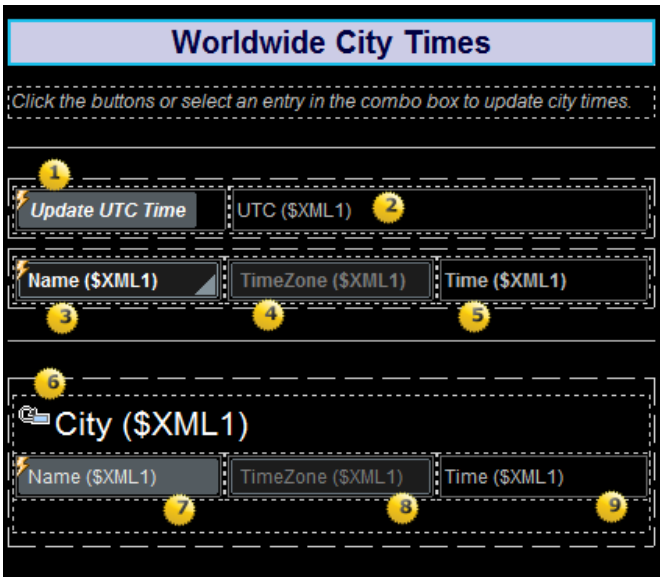
メモ XML ドキュメントがSOAP レスポンスノードの名前空間以外にある場合はプレフィックスがあります。XML ドキュメント内でドキュメントの名前空間をプレフィックスと共に宣言することが奨励されます。デザイン内で、同じ `prefix:namespace` 値がデザインの名前空間のコレクション内に入力されていることを確認してください。デザインのXPath 式内では、XML ツリーノードを参照する際宣言されたプレフィックスを使用する必要があります。代わりに、XPath 式内では、**`*:NodeName`** のような星印プレフィックスを持つノードをアドレスすることができます。これにより、ノードの持つ名前空間に関わらず、ローカル名 **`NodeName`** を持つ全てのノードをマッチすることができます。

デザインコンポーネント

以下のスクリーンショット内では、デザインコンポーネントには番号が付けられ、コールアウトのセクションで更に詳しく説明されています。左側のスクリーンショットは、シミュレーションを、右のスクリーンショットはデザインを表示しています。コールアウトをクリックして、対応するデザインコンポーネントの詳細を確認してください。



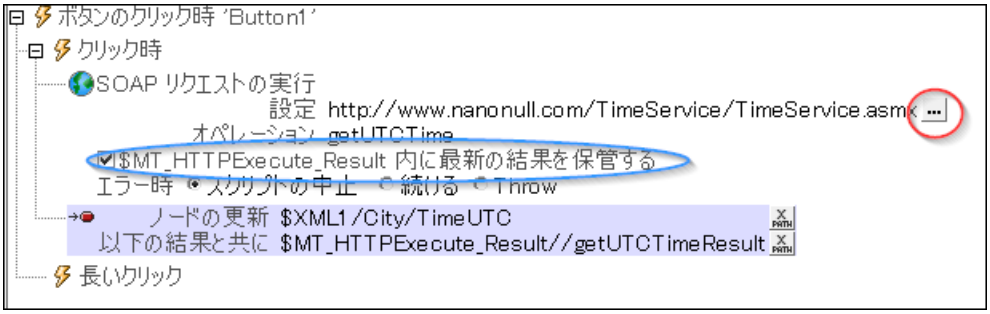
ランタイムソリューションのシミュレーション



ページデザイン

1: UTC 時刻を更新するためのボタン

UTC 時刻の更新 ボタンがランタイムでクリックされた場合、**ボタンのクリック時** イベントは2つのアクションをトリガーします (*下のスクリーンショット参照*)。最初に、 SOAP リクエストの実行 アクションがSOAP リクエストをUTC 時刻のためにWeb サーバーに送信します。Web サービスからのSOAP レスポンスは、`$MT_HTTPExecute_Result` 変数に保管されています (*下のスクリーンショットで青い枠に囲まれています*)。次に、 ノードの更新 アクションはUTC 時刻を使用して `$XML1/CityTime/UTC` ノードを更新します。このノードのコンテンツは即にラベルに表示されます (*下のコールアウト2 を参照*)。



Web サービスは、現在のUTC 時刻を取得するオペレーション(`getUTCTime`) を提供します。SOAP リクエストがどのように定義されるかを確認するには、SOAP リクエストの実行アクションの**[編集]**(上のスクリーンショットで赤い枠で囲まれていますをクリックします) 表示されるSOAP リクエストダイアログ内のプレビューペインにSOAP リクエストのテキストが表示されています。

- 2: UTC 時刻を表示するラベル

  このラベルはページソースノード **$XML1/CityTime/UTC** (ラベルのページソースリンク) と関連しています。このノードからのデータがラベル内に表示されます。ノード **$XML1/CityTime/UTC** が*UTC 時刻の更新* ボタンがクリックされる際に更新されるため (*上のコールアウト1 参照*) 更新されたUTC 時刻 がこのラベル内に表示されます。

- 3: 都市を選択するためのコンボボックス

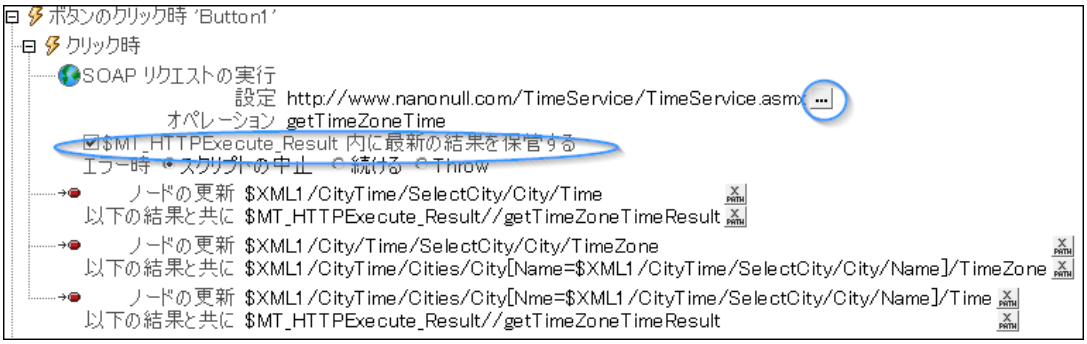
  コンボボックスの目的は、以下のとおりです:

  1. XML ページソースの **Cities** 要素にリストされる都市の名前を表示するため
  2. ユーザーが都市を選択すると、その都市の現在の時刻をSOAP リクエストに送信するため
  3. ユーザーの選択により影響を受けるすべての **//Time** と **//Timezone** ノードを更新するため(*下のスクリーンショット内の更新アクション を参照*)

  コンボボックスのドロップダウンリスト内のアイテムの選択
  デザイン内でコンボボックスコントロールをダブルクリックして、[コンボボックスの編集] ダイアログを表示します。コンボボックスドロップダウンリストのアイテムは、XML ページソースの **Cities** 要素内の都市名です。これらの都市名は、XPath 式 **$XML1/CityTime/Cities/City/Name** と一緒に選択されます。これらの都市名のXML 値は、都市名のテキスト(コンボボックスのドロップダウンリストの表示されるエントリ) と同じになるように設定されています。

  ユーザーが都市をコンボボックス内から選択すると、(コンボボックスエントリと同じ)選択のXML 値がノード **$XML1/CityTime/SelectCity/City/Name** にパスされます。 これは、コンボボックスとこのXML ツリーノード間にページソースリンクを作成し、ページソースペインからツリーノードをコントロールにドラッグアンドドロップすることにより、アーカイブされます)。

  都市の現在の時刻を取得するためにSOAP リクエストを定義する
  コンボボックスのアクションダイアログを開くためにコンボボックスの左上のコントロールアクションシンボルをダブルクリックします。(*下のスクリーンショット*)。 SOAP リクエストの実行 アクションはコンボボックスの **[編集の完了時]** イベントのために定義されています。Web サービスは、指定されたタイムゾーンの現在の時刻を取得する (**getTimeZoneTimeResult**) オペレーションを提供します。時刻がリクエストされるタイムゾーンは、SOAP リクエストのパラメーターとして送信されます。SOAP リクエストがどのように定義されたか確認するには、SOAP リクエストの実行アクションの実行の **[編集]** ボタン(*下のスクリーンショット出赤い枠でかこまれています*)をクリックします。SOAP リクエストダイアログが表示されます。



SOAP リクエストダイアログ内の、プレビューペインにSOAP リクエストのテキストが表示されます。タイムゾーンパラメーターは、パラメーターペインに表示されます。 パラメーターの**XPath** ボタンをクリックして、**m:timezone** パラメーターの値を選択するXPath 式を確認します:

```
for $i in $XML1/CityTime/SelectCity/City/Name return $XML1//Cities/City/
TimeZone[../Name=$i]
```

XPath 式 は、最初にユーザーがコンボボックス内から選択した都市名を選択し、値を式の**$i** 変数内に保管します。式は、次に(XML ページソースの**Cities** 要素から) **$i** 内の値にマッチする**Name** 要素を持つ都市の**Timezone** 要素を選択します。このようにして、**m:timezone** パラメーターのSOAP リクエストとしてユーザーが選択した都市のタイムゾーンが設定されます。このリクエストを受け、 Web サービスは、リクエストされたタイムゾーンの現在の時刻を返します。

SOAP レスポンスを変数内に保管する

Web サービスからのSOAP レスポンスは、**$MT_HTTPExecute_Result** 変数に保管されており、(*上のスクリーンショット出青い枠でかこまれています*)。XML ドキュメントであるSOAP レスポンス全体が変数に保管されていることに注意してください。タイムゾーンの時刻を含むノードを選択するためにSOAP レスポンスの構造を知る必要があります。下のサンプルの場合、次のXPath 式 がSOAP レスポンス内に保管されたタイムゾーンの時刻を検索します:

```
$MT_HTTPExecute_Result//getTimeZoneTimeResult
```

メモ **getTimeZoneTimeResult** ノードは、SOAP レスポンス内でプレフィックスが無く、**http://www.Nanonull.com/TimeService/** 名前空間内にあります。デザインのXPath デフォルトの名前空間は、ですからの名前空間に変更されます。これが行われない場合、SOAP レスポンス内のタイムゾーンの時刻が次のXPath 式でアクセスすることができます:他の名前空間内で要素ノードを検索する $MT_HTTPExecute_Result//*:getTimeZoneTimeResult。要素ノードを名前空間内で検索するgetTimeZoneTimeResult。以下も参照: SOAP レスポンス内のノードの名前空間とXML ツリー。

タイムゾーンの時刻付きのノードの更新

ノードの更新 アクションは、2つのXML ツリー ノードを受け取ったタイムゾーンの時刻を使用して更新します: (i) **$XML1/CityTime/SelectCity/City/Time** と(ii) **$XML1/CityTime/Cities/City[Name=$XML1/CityTime/SelectCity/City/Name]/Time**。ハイライトされた部分は、2番目の式で、ユーザーが選択した都市の名前に一致する**Cities** データベース内の都市がアップデートされるように指定します。これらの更新されたノードのコンテンツは、ページソースリンクを介して、すぐに表示されます (*下の コールアウト5 と9 を参照*)。タイムゾーンの時刻の値は、**$MT_HTTPExecute_Result** 変数を使用したSOAP レスポンスから取得されます。

選択された都市のタイムゾーンの表示

ノードの更新 アクションは、**$XML1/CityTime/SelectCity/City/TimeZone** ノードを更新するために使用されます。ノードが更新されるために使用する値は、次の式で選択されるノードのコンテンツです: **$XML1/CityTime/Cities/City[Name=$XML1/CityTime/SelectCity/City/Name]/Timezone**。この式は、**Cities** データベース内でユーザーが選択した都市の名前に一致する名前を持つ都市の**TimeZone** 要素を選択します。更新されたノードのコンテンツは、ページソースリンクを介して編集フィールドに表示されます (*下のコールアウト4 を参照*)。

4: ユーザーが選択下都市のタイムゾーンを表示する編集フィールド

この編集フィールドはページソースノード**$XML1/CityTime/SelectCity/City/TimeZone** (編集フィールド

の ページソースリンク) と関連しています。ですから、ユーザーがコンボボックス内から都市を選択すると、その都市のタイムゾーンが編集フィールド内で表示されます。一連のアクションは、以下のとおりです: ユーザーが都市を選択すると、**SelectCity//TimeZone** ノードが更新されます (コンボボックス内の更新アクションのため。上のコールアウト3 を参照)。そして、**SelectCity//TimeZone** ノードが編集フィールドのページソースリンクであるため、編集フィールドは自動的に**SelectCity//TimeZone** ノードの更新した値を表示します。

5: ユーザーが選択した都市の現在の時刻を表示するラベル

このラベルはページソースノード**$XML1/CityTime/SelectCity/City/Time** (ラベルの ページソースリンク) と関連しています。ユーザーが都市をコンボボックス内から選択すると、(i) SOAP リクエストがその都市のタイムゾーンの現在の時間を取得するために送信されます、そして (ii) そのタイムゾーン内の現在の時刻と共に**SelectCity//Time** ノードが更新されます (コンボボックスの更新アクションのため。上の コールアウト3 参照)。ラベルは、更新された**SelectCity//Time** ノードのページソースリンクのため更新された時間を自動的に表示します。

6: 都市のデータベースを表示するテーブル

XML ページソースの**Cities** 要素内の都市は、**City** 要素内でそれぞれ定義されています。**City** 要素はですから、3つのカラムと動的な行のテーブル内の繰り返し行として作成されます。それぞれの都市が行に表示されます。列はそれぞれ、各都市の名前、タイムゾーン、と時刻を表示します。列で使用されるコントロールは、それぞれ、ボタン (ページソースリンクとしての都市の**Name** 要素)、編集フィールド、(ページソースリンクとしての都市の**Timezone** 要素) とラベルです (ページソースリンクとしての都市の**Time** 要素)。 下のコールアウト7、8 と9 参照。

7: 都市の時刻を更新するボタン

ボタンは、**$XML1/CityTime/Cities/City/Name** に対するページソースリンクとして都市名を表示します。ランタイムでは、都市のボタンがクリックされると、SOAP リクエストが都市の(タイムゾーン) 時刻を取得するために送信されます (下のスクリーンショット参照)。リクエストされた**m:timezone** パラメーターの値は、都市の**TimeZone** 要素から取得されます。コンテキストノードが**City** のため、都市のタイムゾーンを取得するXPath 式は、以下の通りです: **TimeZone**. SOAP レスポンスは、**$MT_HTTPExecute_Result** 変数内に保管されます。次に、ノードの更新 アクションは、タイムゾーンの時刻付きの**$XML1/CityTime/Cities/City/Name** ノードを更新します。この更新ノードの内容は即ラベルに表示されます (下の コールアウト9 参照)。

8: データベース内の都市のタイムゾーンを表示するフィールドを編集する

編集フィールドはXML ノード**$XML1/CityTime/Cities/City/TimeZone** (編集フィールドの ページソースリンク) と関連付けられます。このノードの内容は変更されません。

- 9: 都市の現在の時刻を表示するラベル

  ラベルはXML ノード**$XML1/CityTime/Cities/City/Time** (ラベルの*ページソースリンク*) と関連付けられます。このノードからのデータは、ユーザーが対応する**City** ボタンをクリックするとラベル内に表示されます (上の*コールアウト7*)。これは、 (i) ボタンにはこのノードを更新するアクションがあり (*上のコールアウト7 参照*)、 (ii) このノードがラベルの*ページソースリンク*であるからです。

ページアクション

ページアクションを確認するには、ページ内を右クリックして**[ページアクション]** を選択します。表示されるダイアログ内でページのロード時イベントのための3つのアクションが定義されています。これらのアクションは、ページがロードされる際に更新されます。これらのアクションは最初のページの表示のためのデータを提供します。

以下のアクションが定義されています:

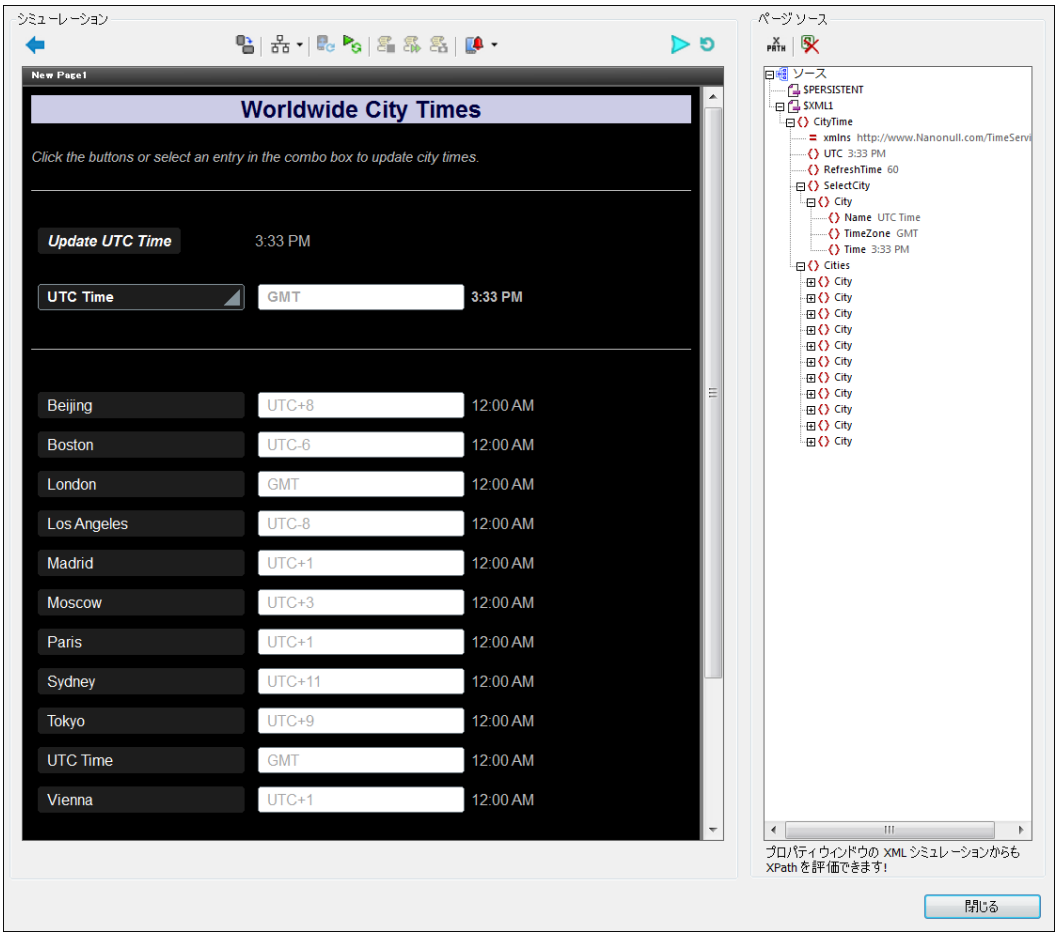
- SOAP *リクエストの実行*: アクションは、Web サービスからのUTC 時刻をリクエストし、レスポンスを**$MT_HTTPExecute_Result** 変数に保管します。リクエストは、UTC 時刻 ボタン(*コールアウト1*) と同じように定義されます。
- *ノードの更新*: UTC 時刻と共にノード**$XML1/CityTime/UTC** をアップデートします。ノードがUTC 時刻ラベルの*ページソースリンク*であるため、(*コールアウト2*)、現在のUTC 時刻と共にラベルは初期化されます。
- *ノードの更新*: UTC 時刻と共にノ**$XML1/CityTime/SelectCity/City/Time** ードを更新します。選択された都市 (**SelectCity//Name**) の初期の値は、UTC 時刻 (*XML ファイル参照*) であり、現在のUTC 時刻と共に**SelectCity//Time** ノードを初期化します。

ページの更新

上記で紹介されたデザインコンポーネントは、(ボタンをクリックまたはコンボボックス内で都市を選択することにより) イベントがトリガーされるとSOAP リクエストを実行します。これは、表示されるUTC 時刻と都市の時刻は手動で更新されない限りすぐに不正確な時刻を表示することになります。しかしながら、継続的に表示される時刻を更新する方法があります。これは、ページの更新時 イベントを使用すると行うことができます。

表示される都市の時刻を自動的に更新するには、以下のメカニズムを使用します。

- XML ファイル内のノードは、更新の間の一定の時間を、秒の単位で保留するように定義されています: **`$XML1/CityTime/RefreshTime`**
- ページの更新時 イベントはページで以下が行われると更新されるように定義されています: (i) ユーザーがデバイスの **[更新]** ボタン (*下のスクリーンショット参照*) を押した場合、および (ii) **`X`** が **`$XML1/CityTime/RefreshTime`** ノード内での数値である場合、毎 **`X`** 秒後。 各ページの更新の都度、表示内での都市の時刻が更新されるように一連のアクションが定義されています。
- ラジオボタンによりユーザーはページの更新を `10/20/30/45/60` 秒の間隔で行うことができます。

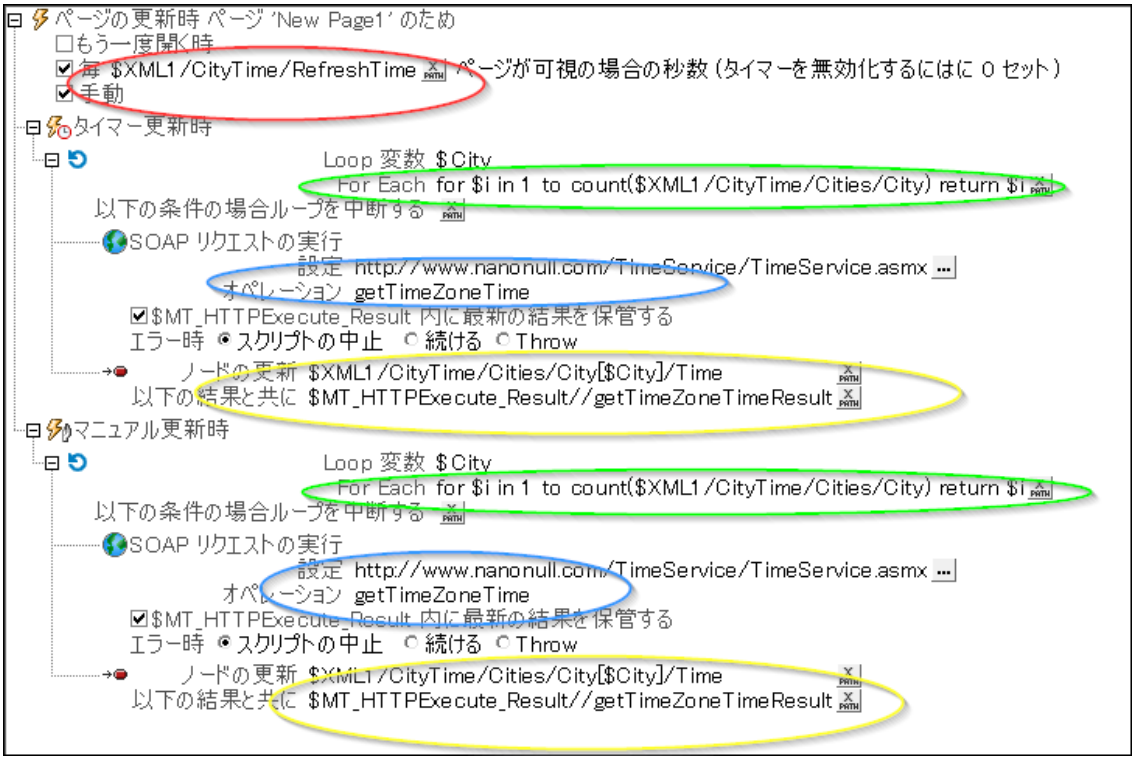


このメカニズムの主要な点は以下に説明されています .

ページの更新時イベントとそのアクション

3つのメソッドを使用してページの更新を定義することができます (*下のスクリーンショット参照*)。サンプルでは以下の2つの方法を用いてページを更新します:

- タイマーは、毎 **x** 秒です。秒数は、**$XML1/CityTime/RefreshTime** 要素のコンテントとして定義されています。(*下のスクリーンショットでは、赤いサークルで囲まれています*)。
- 手動で、ユーザーが**[更新]** ボタンを押すと更新されます (*上のスクリーンショット参照*)。



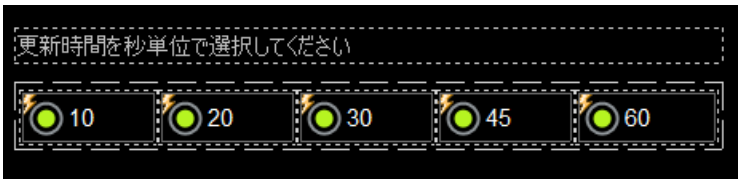
ディスプレイ内で各都市の現在の時刻を更新するには、以下を行います:

- ループの作成 は、各都市を反復します (定義は緑のサークルで囲まれています)。ループは整数のシーケンスを返します。各整数は、その**City** ノードのインデックスであることにより、**City** ノードに繋がっています。ループ内でこれらのノードを更新しないため、直接**City** のノードを反復することはしません。また ノードが反復されている間は更新を行うことはできません。
- ループ内では、それぞれの都市のために以下が実行されます (i) (青いサークルで囲まれた) その都市の **TimeZoneTime** を取得するためにSOAP リクエストが実行されます、そして (ii) (黄色いサークルで囲まれた) その都市のタイムゾーン内の現在の時刻を使用してその都市の**Time** ノードを更新します。

Loop アクション は、双方の方の更新に同様に使用することができ、データベース内の都市の現在の時刻を更新します。

ユーザーによる更新の間隔の選択の有効化

(都市の時刻が更新される) ページの更新の間隔をユーザーが行うことを有効化するために、5 つセットのラジオボタンを作成しました (*下のスクリーンショット参照*)。



ラジオボタンには、以下の設定があります:

- **テキスト** プロパティの値と**チェックされた値** プロパティの値は、共に更新の間隔を設定します: `10/20/30/45/60` 秒。**テキスト** プロパティの値は、ラジオボタンの近くの値を表示します (*上のスクリーンショット参照*)。**チェックされた値** プロパティの値は、ラジオボタンが選択された時に使用されるXML データ値です。
- 5つすべてのラジオボタンには、**`$XML1/CityTime/RefreshTime`** 要素へのページソースリンクがあります。これは、それぞれ相互的に排他的なセットを作成し、選択されたラジオボタンのチェックされた値は**`RefreshTime`** 要素のコンテンツとなります。
- 各ボタンには、**編集の終了時** イベントのために定義されたページタイマーの再開/停止アクションがあります。これは、**ページの更新時** イベント内で定義されたページタイマーを新しい更新の間隔を使用して再開するために必要です (*上を参照*) 。タイマーは、**`$XML1/CityTime/RefreshTime`** 要素から更新の間隔を取得することに注意してください (*上を参照*)。 (ノードへのページソースリンクのため) ラジオボタンの選択は、この要素ノードを更新します。

ページの更新のテストのシミュレーションの実行

MobileTogether Designer 内でのシミュレーションを実行するために**[F5]** を押します。**`$XML1/CityTime/RefreshTime`** ノードから取得された値を使用して、ページ更新のためのタイマーが開始されます。元のデータツリー内では 60 (秒) です。

- ラジオボタンの1つを選択すると、ボタンのチェックした値は**`$XML1/CityTime/RefreshTime`** ノードノードにパスされ、(ボタンイベントのページタイマーの再開/停止 アクションで定義されたとおり) ユーザーにより選択されたページ更新時間を使用して、タイマーが再開されます。
- また、都市の時刻を手動で更新するために、シミュレーターの右上にある**[更新]** ボタンをクリックすることができます (このトピックの最初のスクリーンショットを参照してください)。

4.9 位置情報の共有

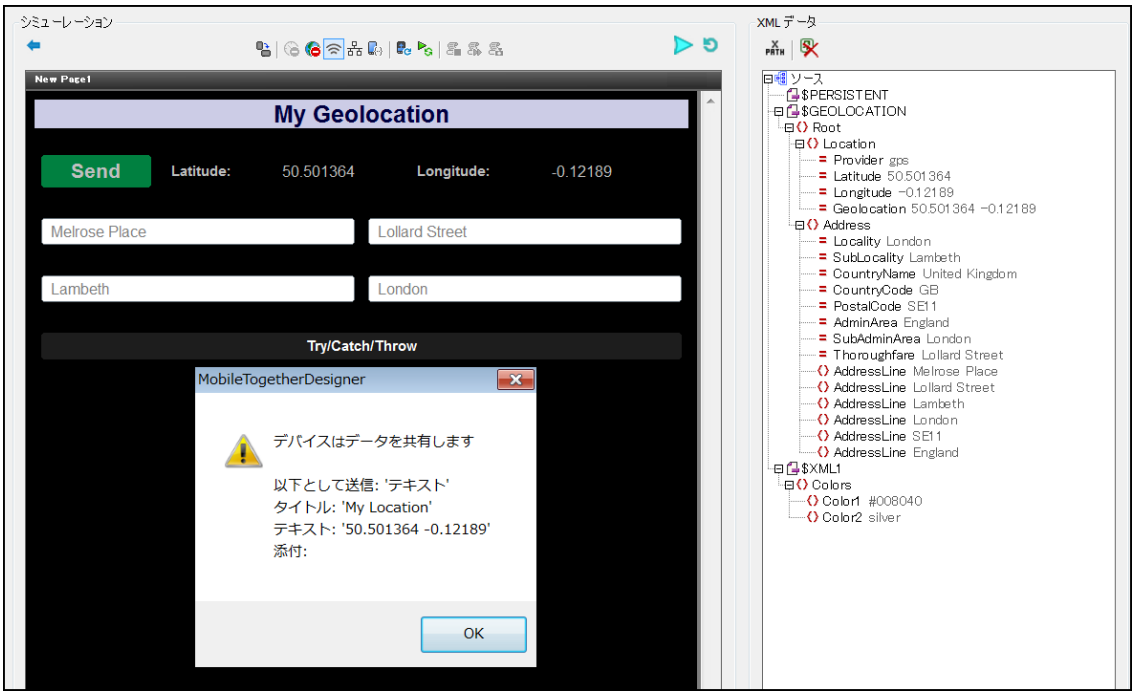
このチュートリアルは以下について説明します:

- モバイルデバイスの現在の位置情報データを読み込み、このデータをデザインの **$MT_GEOLOCATION** ツリーにこのデータを書き込む方法。
- モバイルデバイス内で位置情報データを表示するために、**$MT_GEOLOCATION** ツリーにアクセスする方法。
- デバイスのメッセージとシリアルネットワーキングアプリを使用して連絡先と位置情報データを共有する方法。
- エラー発生時に例外をスローして、この例外を表示する方法。

ソリューションが表示する項目と表示しない項目

下のスクリーンショットはMobileTogether Designer 内のデザインのシミュレーションを表示しています。デザインの機能は2つのボタンによりアクセスすることができます:

- **Send:** デバイスの位置情報のトラッキングを開始し、**$MT_GEOLOCATION** ツリーに位置情報データを書き込みます。また、主な位置情報アイテムをソリューションに書き込み、デバイスの共有メニューを開きます。
- **Try/Catch/Throw:** 位置情報の座標が米国外の場合、警告メッセージを表示します。



チュートリアルファイル

このチュートリアルのファイルは(*マイ ドキュメント*の MobileTogether フォルダーにあります:
MobileTogetherDesignerExamples\Tutorials\Geolocations。

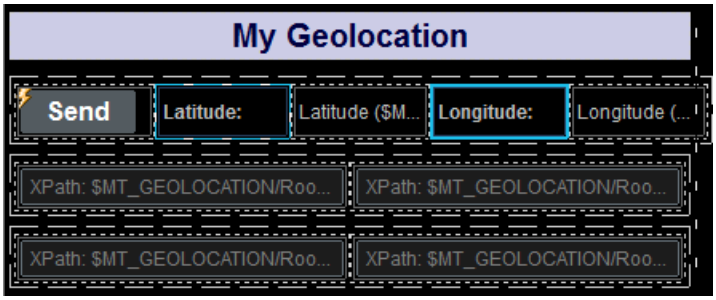
- **SharingGeolocations.mtd**: これは、完成されたMobileTogether デザインファイルです。このファイルを開き、このチュートリアルを実行中に参照してください。MobileTogether Designer 内のシミュレーションは **[F5]** を押すことにより行うことができます。

- **`LondonLocations.xml`**: これは、ロンドンのロケーションの位置情報データを含むXML データファイルです。シミュレーションは、デスクトップで行われるため、このファイルをモバイルデバイスの位置情報データのために使用します。

デザインファイル内のパスを使用することができ、XML データファイルはサーバーにデプロイされていません。ですから、これら2つのファイルをフォルダーにコピーすることにより、MobileTogether Designer 内で正確にシミュレーションを行うことができます。

位置情報の読み込みと共有

デザインの上の部分は(*下のスクリーンショット*) デバイスの位置情報データを表示しています。



- 一番上の行は、**[Send]** ボタンと4つのラベルにより構成されています。青い枠によりハイライトされている2つのラベルは、ラベルテキスト(`Latitude:` と`Longitude:`)のための静的なテキストです。他の2つのラベルには、**`$MT_GEOLOCATION`** ツリーの経度と緯度ノードへのページソースリンクがあります: **`$GEOLOCATION/Root/Location/@Latitude`** と**`$GEOLOCATION/Root/Location/@Longitude`**。この結果、ノードが更新されると、これら2つのラベルも更新されます。
- 第2および第3番目の行は、合計4つのを編集フィールド含んでいます。これらのコントロールはそれぞれページソースリンクで、**`$MT_GEOLOCATION`** ツリーの最初の4つの**`AddressLine`** ノードです。: **`$GEOLOCATION/Root/Address/AddressLine`**。4つすべての編集フィールドは、ですから、対応するノードが更新されると、更新されます。
- 残るは**[Send]** ボタンです。ボタンの *ボタンのクリック時* アクションは、以下に必要なすべてのアクションを定義します: (i) モバイルデバイスの位置情報を取得し表示する、(ii) デバイスのアプリを使用して位置情報を共有する。**[Send]** ボタンの アクションは以下で定義されています。

メモ 位置情報のトラッキングの開始/停止 アクション、または、位置情報データの読み取りアクションがデザインに追加されると、**`$MT_GEOLOCATION`** ツリーは自動的にページのデータソースに追加されます。

[Send] ボタン アクション

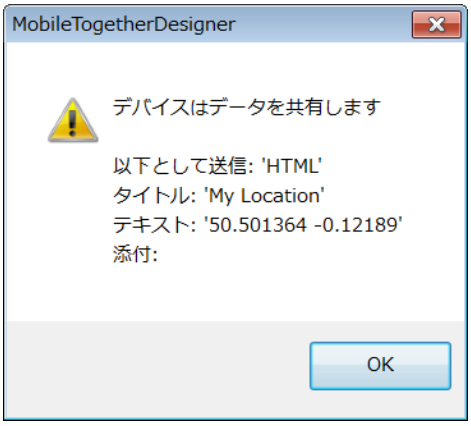
ボタンの アクションダイアログを開くために、**[Send]** ボタンの左上のアクションダイアログアイコンをダブルクリックします(*上のスクリーンショット参照*) (*下のスクリーンショット*)。

次のアクションは**ボタンのクリック時** イベントのために定義されています。これは、ボタンがクリックされると、ボタンのために定義されている全てのアクションが、1つずつ順番に実行されることを意味します。

- 位置情報のトラッキングの開始 は、デバイスのトラッキングを開始します。XML ファイルがシミュレーションファイルとして使用されるように定義されています。このファイルは**`LondonLocations.xml`** と呼ばれ、MTD ファイルと同

じフォルダーにあります。MobileTogether Designer を作動しているデスクトップでシミュレーションを実行中、を実際の位置情報データを使用することができないため、このファイルがモバイルデバイスの位置情報の代替として使用されます。

- 位置情報の読み込み アクションは、トラッキングにより与えられる位置情報データを取り込み、**$MT_GEOLOCATION** ツリーの XML フォーマットに構成します。アクションの定義内では、モバイルデバイスの **Location** 要素と **Address** 要素の両方からのデータが **$MT_GEOLOCATION** ツリーに書き込まれるように指定しています。シミュレーションの場合、**Location** と **Address** データが、**LondonLocations.xml** ファイルから取り込まれます。**$MT_GEOLOCATION** ツリーは、新しい位置情報データと共に更新され、ソリューション内の全てのラベルと編集フィールドが更新されたデータを自動的に表示します。これは、コントロールと更新されたノードの間にページソースリンクが存在するからです。
- 共有 アクション は、*My Location* というタイトルのテキストメッセージを作成します。メッセージの内容は、(緯度と経度の連結の値である) **$GEOLOCATION/Root/Location/@Geolocation** 属性の値です。実際の場合の実行では、共有アクションによりモバイルデバイスの共有メニューが開かれ、エンドユーザーがモバイルデバイスの共有アプリを使用して現在の位置情報を送信することができます。シミュレーションでは、タイトルとメッセージの内容を含むメッセージボックスが表示されます (*下のスクリーンショット参照*)。



更に詳しい情報に関しては、それぞれのアクションの詳細を参照してください。

Try/ Catch/ Throw 例外の使用

米国外の場所の位置情報の座標の場合、警告を表示するためにトライ/キャッチとスローアクションを使用しました。これらのアクションは**[Try/Catch/Throw]** ボタンがクリックされると実行されます (下のスクリーンショット参照)。

Try/Catch/Throw アクション

デザイン内の**[Try/Catch/Throw]** ボタンの左上にあるアクションダイアログアイコンをダブルクリックして、ボタンのアクションダイアログを開きます (下のスクリーンショット)。

アクションは以下のように定義されています:

1. トライ/キャッチ アクションが追加されました。
2. 例外メッセージを保留する変数 **`$Not-USA-Warning`** が設定されました。
3. トライ パートは、位置情報がUSA でないかをテストする条件をセットアップします。この条件は、スロー アクションのXPath 式内で指定されています。条件がtrue の場合、例外がスローされます。例外メッセージは、**`$Not-USA-Warning`** 変数に保管されます。条件がfalse の場合、例外はスローされません。空のシーケンスが生成され、**`$Not-USA-Warning`** 変数内には何も保管されません。XPath 式の詳細は下で説明されています。
4. トライ/キャッチ アクションの キャッチ パートは、例外がスローされた場合のみ処理されます。(これは、テストされた上記の条件がtrue を評価した場合のみです)。キャッチ パート内では、**`$Not-USA-Warning`** 変数のコンテンツを表示するメッセージボックスを表示します。

アクションの XPath 式

スローアクションのXPath 式は、以下のとおりです:

```
if ($MT_GEOLOCATION/Root/Address/@CountryName != 'USA')
then (concat( 'Warning: Device location is outside the US: ', $MT_GEOLOCATION/Root/Address/
```

```
@CountryName))
else ()
```

この式は以下のように動作します:

- **if** 句は、**$MT_GEOLOCATION/Root/Address/@CountryName** ノードの値が 'USA' でないかをチェックします。
- **then** 句は、国名が USA' **でない**場合、処理されます。この句は文字列を生成します。
- **else** 句は、国名が USA' **である**場合、処理されます。この句は空のシーケンスを生成します。

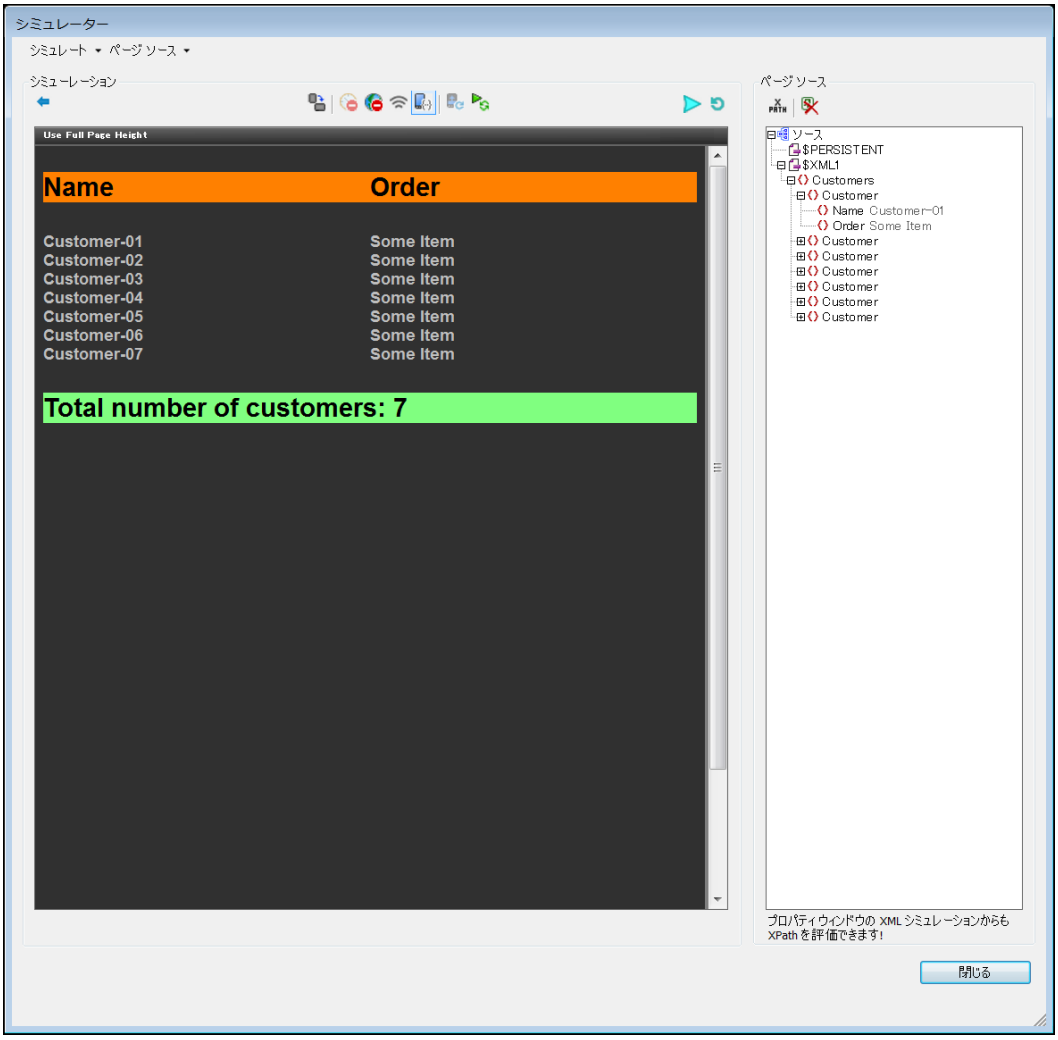
位置情報の国が 'USA' **でない**場合、条件は true で、式はこの **then** 句により生成された文字列を評価します。この結果は空のシーケンスではないため、例外がスローされ、生成された文字列は トライ/キャッチ 変数 $Not-USA-Warning 内に保管されます。

位置情報の国が 'USA' **である**場合、条件は false で、(**else** 句により生成された) 式は空のシーケンスを評価します。結果が空のシーケンスであるため、例外はスローされません。ですから、トライ/キャッチ アクション の キャッチ パートは実行されません。

4.10 スクロールすることのできるテーブル

このチュートリアルではスクロールすることのできるテーブルについて説明されています。デザインファイル (`ScrollableTables.mtd`) は、いかに表示されている2つのトップページを含んでいます:

- スクリーンの下に表示される内容をページの最後に指定するスクロールすることのできるテーブルの設定 (*下のスクリーンショット参照*)
- スクリーンの高さのパーセンテージである高さで表示されているテーブル。



チュートリアルファイル

このチュートリアルのファイルは(*マイ ドキュメント* 内のMobileTogether フォルダーにあります: `MobileTogetherDesignerExamples\Tutorials\ScrollableTables`。

- `ScrollableTables.mtd`: これは、完成されたMobileTogether デザインファイルです。チュートリアルを実行中にこのファイルを開いてください。MobileTogether Designer 内でシミュレーションを行うには、**[F5]** を押します。

- **ScrollableTables-01.xml**: これは簡単かつ短い7人の顧客の記録のデータベースを含むXML ファイルです。ファイルの構成は上のスクリーンショットの **$XML1** ツリー内で確認することができます。
- **ScrollableTables-02.xml**: **ScrollableTables-01.xml** の長いバージョンです。29 の記録が含まれています。

このデザインファイルのパスは相対的であり、XML ファイルはサーバーにデプロイされていません。ですから、これら3つのファイルをフォルダーにコピーすることにより、MobileTogether Designer 内で、正確にシミュレーションを実行することができます。

チュートリアルの構造

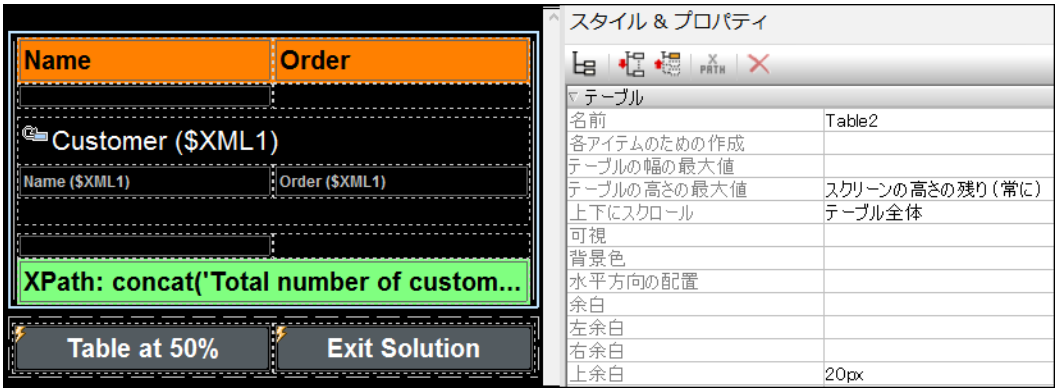
このチュートリアルは以下のセクションに整理されています:

- スクリーン全体の高さで表示されるテーブル では、ページがスクリーン全体で表示されるように高さを自動調整する設定について説明されています。
- 特定の高さを持つテーブル では、テーブルの高さの特定の率を表示するテーブルの作成方法について説明されています。表示するテーブルの行の高さがテーブルで設定された高さよりも高い場合、テーブルをスクロールすることができます。

スクリーン全体の高さで表示されるテーブル

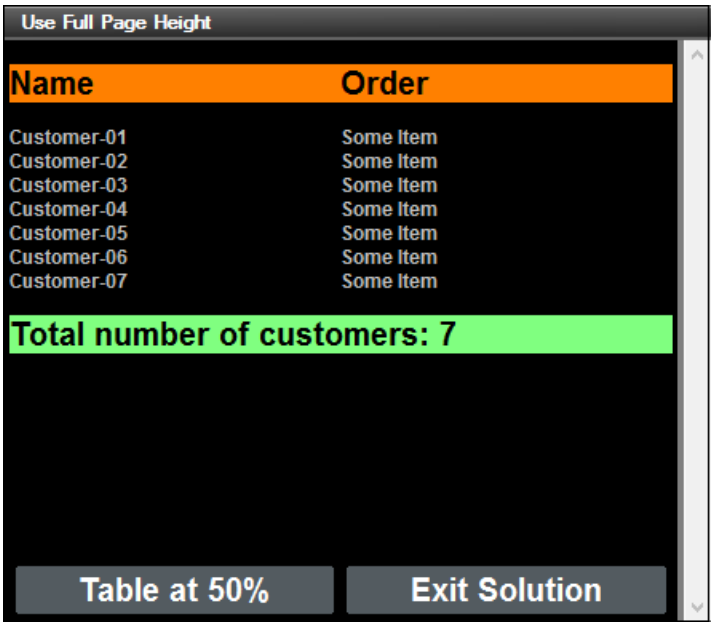
デザインの最初のページ *Use Full Page Height* には、`$XML1/Customers/Customer` 要素が繰り返し要素として使用されている動的 テーブルが含まれています (*下のスクリーンショット参照*)。これは各 `Customer` 要素がテーブル行グループとして作成されており、自身の行として表示されていることを意味します。テーブルには(オレンジ色の背景の) ヘッダーおよび(緑の背景色の) フッターがあります(*下のスクリーンショット参照*)。テーブルのためのXML データはXML ファイル `ScrollableTables-01.xml` から取得されています。

テーブルの下に2つのボタンが作成されています: 次のページに移動するボタン、ソリューションを終了するボタン。



テーブルの高さに関わらず、つまり、テーブルにスクリーンの下に届くまでの行が存在しない場合でも、スクリーンの下に2つのボタンが常に表示されるデザインを作成します。以下を使用して設定します:

- テーブルの テーブルの高さの最大値 プロパティをスクリーンの高さにリセットに設定します (*常に*)。これにより、追加スペースがテーブルの下に追加され、ページの最後のコンポーネントがスクリーンの一番下に表示されます。
- テーブルの 上下にスクロール プロパティをテーブル全体に設定します。これにより、フッターがテーブルのボディに含まれます (*下のスクリーンショット参照*)。それ以外の場合、フッターはページのその他のコンテンツの上に表示され、テーブルの最後の行とフッターの間にギャップが生じる可能性があります。



以下の点に注意してください: テーブルの テーブルの高さの最大値 プロパティがスクリーンの高さのリセット(*常に*)に設定されて

いる場合、テーブルの高さは自動的に調整されページのコンポーネントがスクリーン全体に表示されます。

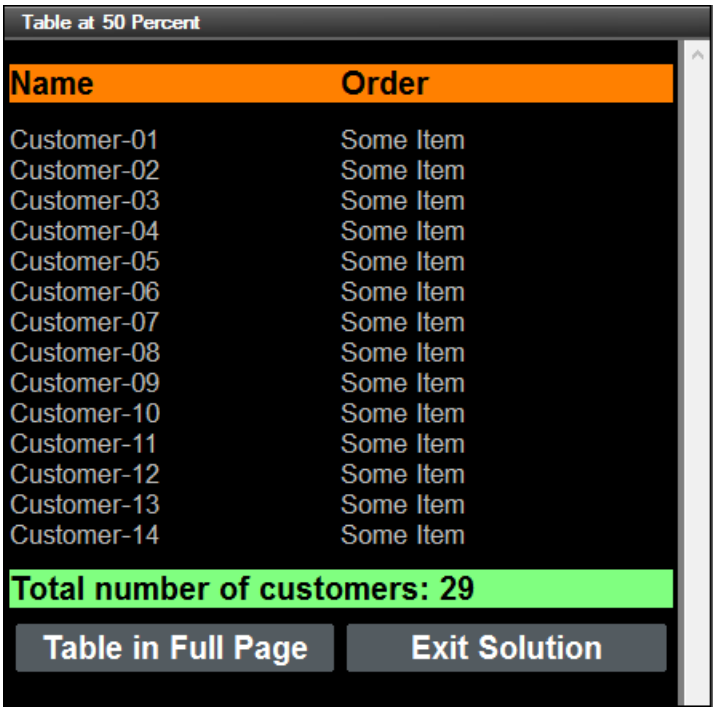
多種の可能性をテストするためにプロパティの値を変更することができます。スクロールすることのできるテーブルプロパティの詳細に関しては、テーブルプロパティを参照してください。

特定の高さを持つテーブル

デザインの第2ページの50 パーセントでのテーブルには、繰り返し要素として使用されている**$XML2/Customers/Customer** 要素により作成された動的な行が存在するテーブルが含まれています(*下のスクリーンショット参照*)。このテーブルは、前のページで作成されたテーブルに類似しています。違いは、前のページで作成されたページは、自動的に高さを調節するため、ページのコンポーネントすべてがスクリーン全体に表示され、このテーブルは、スクリーンの高さの50% に設定されている点です(*テーブルの テーブルの高さの最大値プロパティを参照してください*)。

テーブルの高さの最大値の50% 設定の結果、テーブルの下に作成された2つのボタンは、出力内で、50% テーブルの下に表示されます(*下のスクリーンショット参照*)。テーブルの上下にスクロールプロパティをヘッダーとフッターを除く行に設定することができます。これにより、行の内容がスクロールされても、ヘッダーとフッターが固定された高さのテーブル内で固定することができます(*下のスクリーンショット参照*)。

メモ MobileTogether Designer シミュレーター内では、スクロールホイールを使用して上下にスクロールし、クリックアンドドラッグを使用して左右にスクロールします。



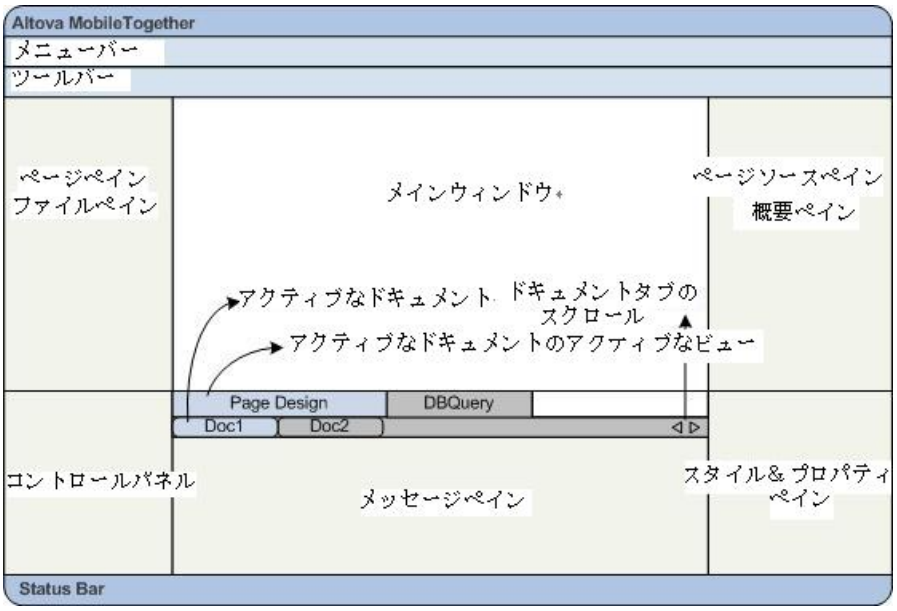
多種の可能性をテストするためにプロパティの値を変更することができます。スクロールすることのできるテーブルプロパティの詳細に関しては、テーブルプロパティを参照してください。

チャプター 5

ユーザー インターフェイス

5 ユーザー インターフェイス

グラフィカル ユーザー インターフェイス(GUI) は(ページ デザイン、DB クエリビュータブ)から構成される) メイン ウィンドウと複数のペインから構成されています(下のスクリーンショット参照)。デフォルトでは、ペインはメイン ウィンドウの周囲にあり、GUI 周辺を移動、縮小化、また非表示に設定することができます。



このセクションのサブセクションは以下にリストされたペインについて説明しています:

- [ページ] ペイン
- [ファイル] ペイン
- [コントロール] ペイン
- [ページ ソース] ペイン
- [概要] ペイン
- [スタイル & プロパティ] ペイン
- [メッセージ] ペイン

ペインの表示 /非表示

[表示] メニューでオンとオフに切り替えることによりペインは、表示または非表示できます。表示されたペインはタイトル バーを右クリックして、**[隠す]** コマンドを選択することにより非表示になります。

ペインのフロートとドッキング

各ペインは GUI から自由にフロートまたは、GUI に(固定ドッキング) することができます。フロートするペインは、最後にドッキングされた場所にドッキングします。ペインは他のウィンドウ内でタブとしてドッキングすることができます。

次の方法を使用してウィンドウをフロートまたはドッキングすることができます:

- ペインのタイトルバーを右クリックして、必要なコマンド **[フロート]** または **[ドッキング]** を選択します。
- ペインのタイトルバーを右クリックします。ドッキングされている場合は、ペインはフロートします。フロートしている場合、ペインは最後にドッキングされた場所にドッキングします。
- (タイトルバーをハンドルとして)ペインをドラッグし、フロートさせます。フロートしているペイン(のタイトルバー) をドッキングしたい場所にドラッグします。 2 組の矢印が表示されます。外側の 4 つ矢印のセットは、アプリケーションのウィンドウに相対して(GUI の上下左右の端に合わせて) ドッキングを有効化にします。内側の矢印のセットは、現在カーソルが置かれているウィンドウに相対して、ドッキングを有効化にします。ドラッグされたペインを内側の矢印のセット(またはペインのタイトルバー) の中央のボタンにドロップすると、ドラッグされたウィンドウのタブされたペインとしてドキングします。

タブされたペインをフロートするには、タブをダブルクリックします。タブされたペインをタブグループから移動する場合はタブをドラッグしてください。

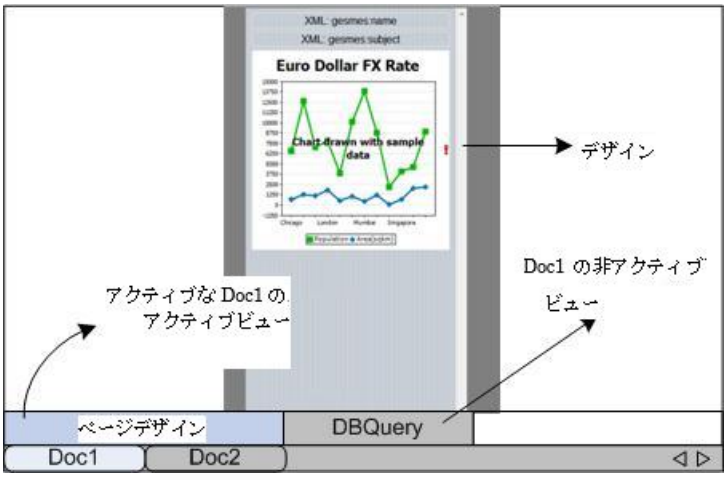
ペインを自動的に隠す

自動的に隠す機能は、ドックされたペインをアプリケーションウィンドウの端に並ぶボタンに縮小化します。これにより、メインウィンドウと他のペインにスペースができます。縮小化されたペインをスクロールすると、そのペインが表示されます。

ペインを自動的に隠して、復元するには、ペインのタイトルバーの押しピンアイコンをクリックします(または、タイトルバーを右クリックして **[自動的に隠す]** を選択します)。

5.1 メイン ウィンドウ

メイン ウィンドウ(*下のスクリーンショット*) で、MobileTogether Designer プロジェクトのページをデザインし、テーブルデータを閲覧するため直接データベースをクエリします。メイン ウィンドウは、一度に 1 つのみアクティブ化できる、次の 2 つのビューから構成されています: ページ デザイン と DB クエリ。



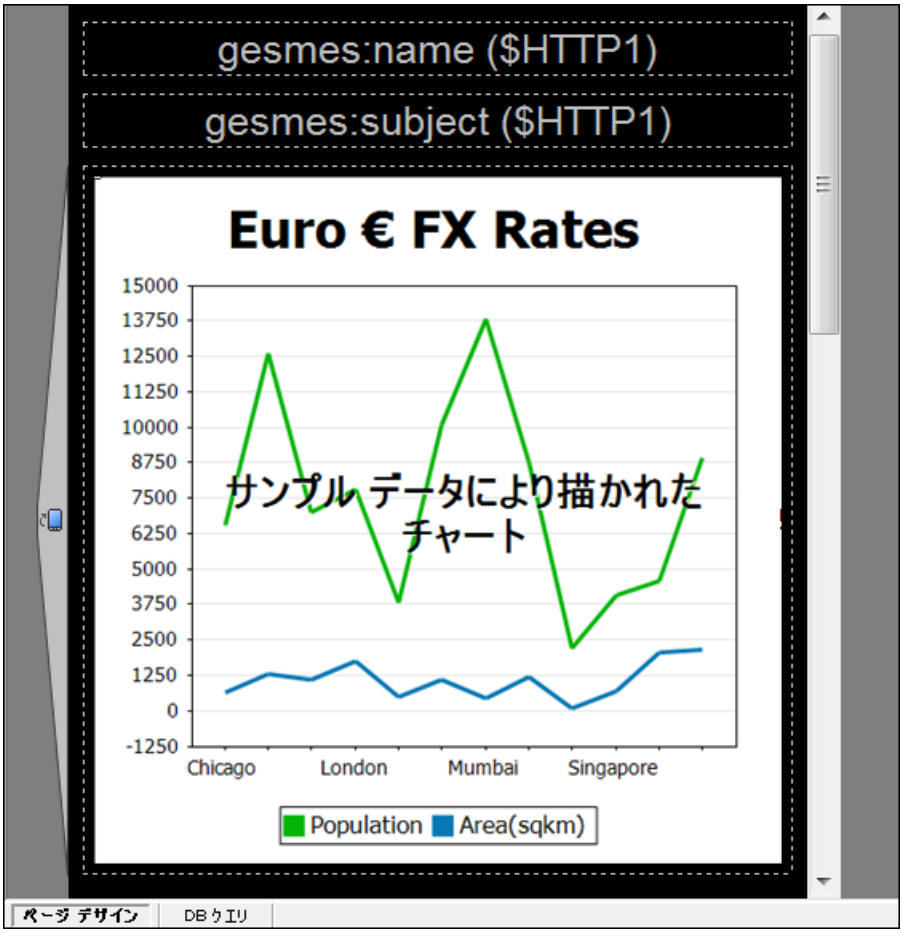
メイン ウィンドウの MobileTogether デザイン (.mtd) ファイル

以下の点に注意してください:

- MobileTogether デザイン(`*.mtd`) ファイルはいくつでも同時に開くことができます。開かれているドキュメント間の切り替えをして編集することができます。
- 開かれた各ドキュメントは、自身のウィンドウとタブを持ち、メイン ウィンドウの下に名前が表示されます。
- 複数のファイルが開かれている場合、ドキュメント タブ バーのスペースが足りない場合、見えない場合があります。ドキュメントタブは以下の方法で表示できます: (i) ドキュメント タブ バーの右横のスクロール ボタンを使用する、または、(ii) ウィンドウ メニューの下のリストから必要なドキュメントを選択します。
- **[Ctrl+Tab]** または **[Ctrl+F6]** を使用してシーケンスの開かれているファイルをアクティブ化できます。
- ファイル タブを右クリックして、[**印刷**] と[**閉じる**] などのファイルコマンドを含むコンテキストメニューを開きます。
- メイン ウィンドウのコンポーネントにカーソルをポイントすると、コンポーネントの機能についての更なる説明を表示するポップアップが表示されます。

ページ デザイン

[**ページ デザイン ビュー**] (略して**ページ ビュー**) は、モバイルデバイスにデプロイされるページがデザインされるビューです (*下のスクリーンショット参照*)。



ページをデザインするには、コントロールを[コントロール] ペインからデザインにドラッグ アンド ドロップし、[プロパティ] ペインでコントロールの設定を指定します。ページにコントロールをドラッグすると、ドロップできる可能性のある場所が矢印で表示されます。コントロールの設定は、デザインのコントロールを[プロパティ] ペインから選択して編集することにより後で編集することができます。デザインのコントロールは選択し**[削除]** を押すことにより削除できます。

ページビューの設定

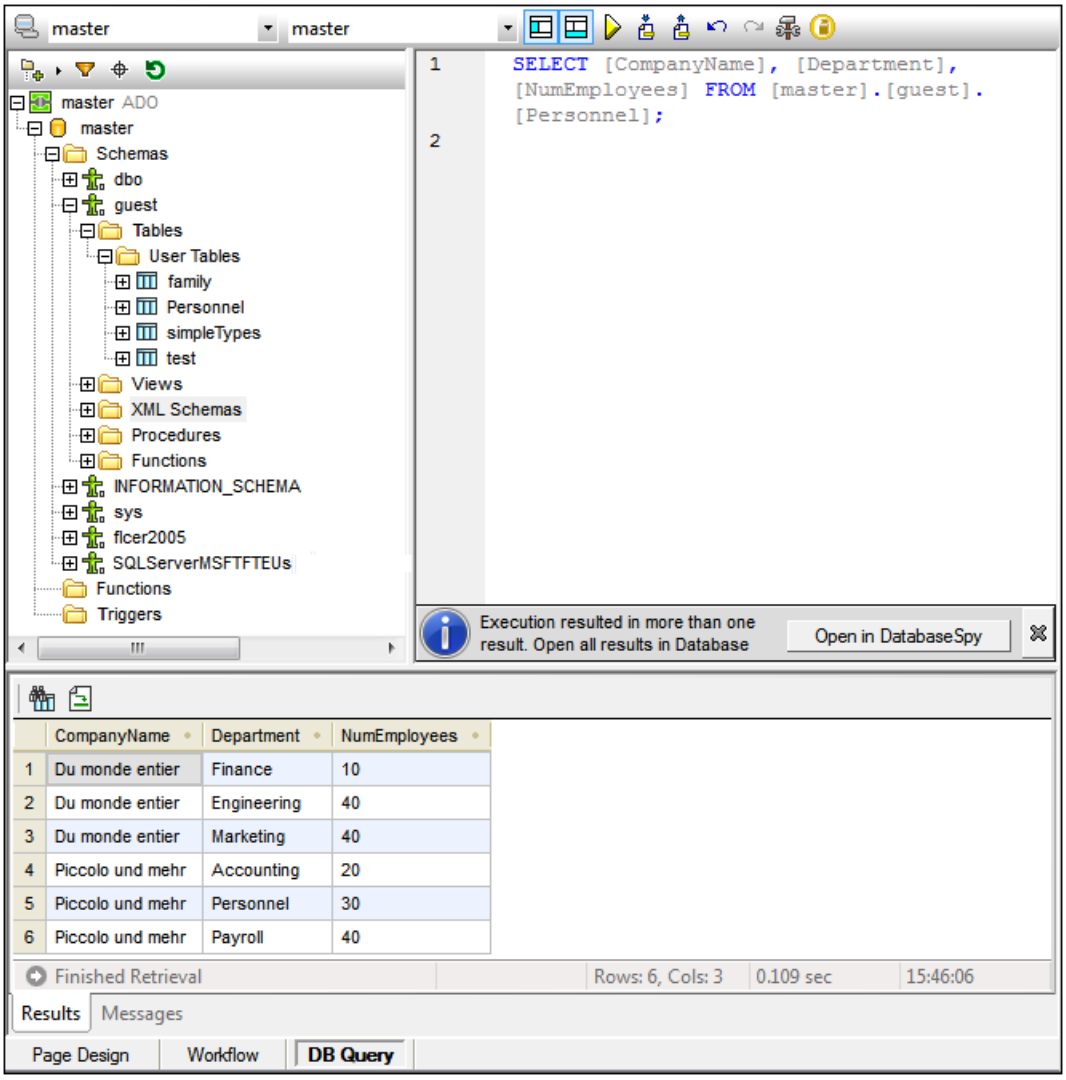
下のスクリーンショットには、メイン ツール バーのページ ビュー設定が表示されています。

- *デバイス選択*: デザインを特定のデバイスでプレビューするため、このコンボ ボックスから、多種のモバイル デバイスが選択することができます。
- *縦長 /横長 プレビューの切り替え*: 横長と縦長のデザイン プレビューを切り替えます。

- *拡大率のレベル:* 拡大率レベルを10% から100% の10 のステップで選択することができるコンボ ボックス。ズーム レベルは[**表示**メニュー] からも変更できます。
- *すべてのページビューを同じ拡大率とプレビュー デバイスにロック:* すべての開かれているドキュメントが現在選択されているデバイスと拡大率にロックされます。

DB クエリ

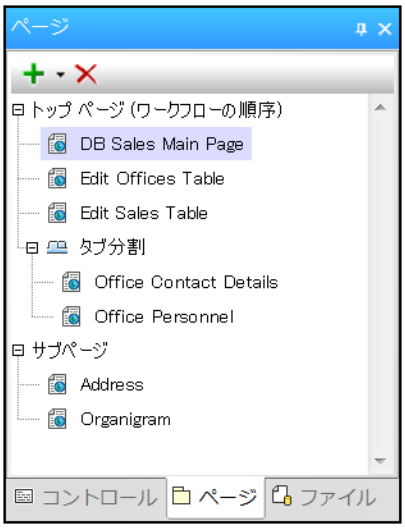
DB クエリビュー(*下のスクリーンショット*) により、MobileTogether Designer GUI 内のメジャーなデータベースに直接クエリを行うことができます。データベースはアクティブ ドキュメントまたは外部データベース内で参照されるデータソースとなる可能性があります。各 DB クエリペインは現在アクティブなデザインに関連付けられています。単一デザインは複数のデータベースの接続を持つことができます。 MobileTogether Designer で複数のデザインを開くことができます。クエリとアクションはDB クエリビューで定義されており、クエリビューはMobileTogether Designer タブから独立していて、`.mtd` ファイルの一部として保存されません。



詳細はデータベース クエリを参照してください。

5.2 ページ ペイン

[ページ] ペインは、新しいページをプロジェクトに追加し、プロジェクトの全てのページのツリーを表示します(*下のスクリーンショット参照*)。ページのデフォルトのロケーションを確認するにはグラフィカル ユーザー インターフェイス(GUI) に移動します。



ページを表示するには、[ページ] ペインのエントリをクリックしてください。

トップページ、サブページ、タブされたページ

プロジェクトページには3 つの種類があります:

- **トップページ**: トップページはワークフローを構成するページのシーケンスです。ソリューションが開始されると、最初から最後のページに、[ページ] ペインのリスト順に進みます。リストのページの位置をページの新しいポジションにドラッグして変更することができます。タブされたページもワークフロー シーケンスの一部ですが、サブページは異なります。
- **サブページ**: サブページはプロジェクトのワークフローを構成するページのシーケンスの一部ではありません。トップページ(またはタブされたページ) 内のコントロールから呼び出されるモジュールと類似しています。例えば、トップページのボタン コントロールの ボタンのクリック時 イベントは サブ ページへ移動 アクションを使用して、特定のサブページに移動し、トップページに戻る(または他のページに移動する)ことができます。
- **タブされたページ(またはタブ分割)**: タブされたページ(またはタブ分割)はページを含むタブ付きページです。例えば、下のスクリーンショットは、タブされたページ(タブ分割プロセス) は*Office Contact 詳細* および*Office Personnel* のページを含む2 つのタブを持つと定義されています。タブされたページはプロジェクトのワークフローを構成するページ シーケンスの一部です。

ページの追加、名前変更、および削除

以下のテーブルにリストされた方法以外にも、コンテキスト メニュー コマンドを他のタスクに使用することができます(更に下を参照)。

すること...	方法 ...
ページの追加	パネルのツールバーの**[ページの追加(Add Page)]** アイコンをクリックします。ドロップダウンメニューから、**[トップページの追加]**、**[タブ分割の追加]**、または **[サブページの追加]** を選択します。[ページ] ペインに*Page X* または *Process Tab Split* という名前の新しいページを追加すると、空の新しいページがメインウィンドウに表示されます。
ページの名前変更	[ページ] ペインの名前をダブルクリックして、名前を編集します。
ページの削除	削除したいページを選択して、ペインのツールバーの**[削除]** アイコンをクリックまたは**[削除]** を押す。

コンテキスト メニュー

[ページ] ペインのコンテキスト メニューにより、選択されたアイテムの前にページを挿入 **[挿入]** することができ、またトップページのシーケンスとサブページのリストにページを追加 **[追加]** することができます。子ページはタブされたページにタブされたページ アイテムのコンテキスト メニューを介して、タブされたページに追加されます。

すること...	方法 ...
トップ ページ、サブ ページ、タブされたページ、またはタブされたページの子ページの前にページを追加する。	ページを右クリックして、**[ページの挿入]** を選択する。
トップ ページ リストの最後にトップ ページを追加する	アイテムを右クリックして、**[トップ ページの追加]** を選択する。
サブ ページ リストの最後にサブ ページを追加する。	アイテムを右クリックして、**[サブ ページの追加]** を選択する。
タブされたページの子ページ リストの最後に子ページを追加する。	タブされたページを右クリックして、**[ページを子として追加]** を選択する。

5.3 ファイル ペイン

[ファイル] ペイン(*下のスクリーンショット*) は、サーバーにデプロイすることのできるプロジェクトのファイルをリストします。デフォルト ファイルまたはデータソースなどの、ファイルの一部は、これらのファイルがプロジェクトに関連付けられると、デプロイ可能なファイルのリストに自動的に追加されます。しかし、ファイルをデプロイ可能なファイルに([ファイル] ペイン内を右クリックして取得することのできる) コンテキスト メニューを介して直接追加することもできます。デプロイ可能なファイルはイメージ ファイルまたは、XML ファイルなどの、その他のファイルで、次の2 つの見出しの下に整理されています: *イメージファイル とその他のファイル*(*下のスクリーンショット参照*)。

各ファイルの横にはチェックボックスがあります。ファイルをデプロイするには、このチェックボックスをチェックします。デプロイを取り消すにはチェックボックスのチェックを解除します。

次の情報とアクションが [ファイル] ペインで使用することができます:

- ペインのコンテキスト メニューのコマンドを介してファイルを追加または削除します (*下を参照*)。
- ファイルのコンテキストメニュー内の対応するコマンドを使用してファイル パスを相対または絶対にします。
- ファイルがページ全体で使用された回数がファイル名の横に表示されます。
- プロジェクトをサーバーにデプロイするには、各ファイル名の前のチェックボックスをチェックして、MobileTogether Server にファイルをデプロイします。デプロイを取り消す場合は、ボックスのチェックを解除します。

デプロイされたファイルについて
デプロイされたファイルは、サーバーに保存された読み取り専用ファイルです。デプロイはデフォルトの XML ファイル、イメージファイル、データ読み取り専用に使用される他のデータファイルには理想的です。データファイルが書き込まれサーバーに保存される場合、デプロイせずに、MTD が正しく参照する、サーバーの場所に保存します (詳細はプロジェクト ファイルの場所 を参照してください)。ソリューションがデプロイされると、デプロイされるファイルがサーバーに送信されます。(詳細は プロジェクトのデプロイ を参照してください。) ファイルがデプロイされると、そのデザインに保管されます。その後の他のデザインを使用してのファイルのデプロイは、前のファイルのデプロイに影響しません。

+ ファイルが読み取り専用として使用されている場合のみデプロイしてください。
+ 何らかの理由で、クライアントがファイルの URL にアクセスできない場合、デプロイしてください。
+ クライアントが同じ変更無しの状態にアクセスすることを希望する場合、デプロイしてください。
+ ファイルのロードの速度を速くしたい場合、デプロイしてください。
- ファイルへの書き込みが必要な場合、デプロイしないでください。
- サーバでの大きなファイルのサイズが問題の場合、デプロイしないでください。
- ファイルが継続的に変更され、ソリューションが最新バージョンである必要がある場合、デプロイしないでください。
- デザインが他に送信される場合、デプロイしないでください。この場合、データ ファイルをデザインに埋め込むことを考えてください。

[ファイル] ペイン内のファイルへのファイルの追加と削除

すること...	方法 ...

ファイルの追加	[ファイル] ペインを右クリックして、表示されるコンテキストメニューから **[ファイルの追加]** をクリックし、希望するファイルを参照します。追加されたファイルは、ファイルの種類に応じて、*イメージファイル*または*その他のファイル*のリストに表示されます。
ファイルの削除	削除したいファイルを右クリックすると、表示されるコンテキストメニューから **[ファイルの削除]** をクリックします。

デプロイ可能なファイルをファイル ペインの外部からプロジェクトに追加する

[ファイル] ペインの外部からファイルを以下の方法でデプロイすることができます:

- ファイルがページ ソースとして、またはイメージ ファイルとしてページデザインに追加されると、ファイルを MobileTogether Server にデプロイするかを問うダイアログが表示されます。**[はい]** をクリックすると、チェックボックスがチェックされたファイルが [ファイル] ペインに追加されます。**[いいえ]** をクリックすると、チェックボックスがチェックされていないファイルが [ファイル] ペインに追加されます。[ファイル] ペインのファイルの選択 解除チェックボックスを後から選択することもできます(*上のスクリーンショット参照*)。
- [ページ ソース] ペインのデータソースのルートノードのコンテキスト ノード。

関連項目

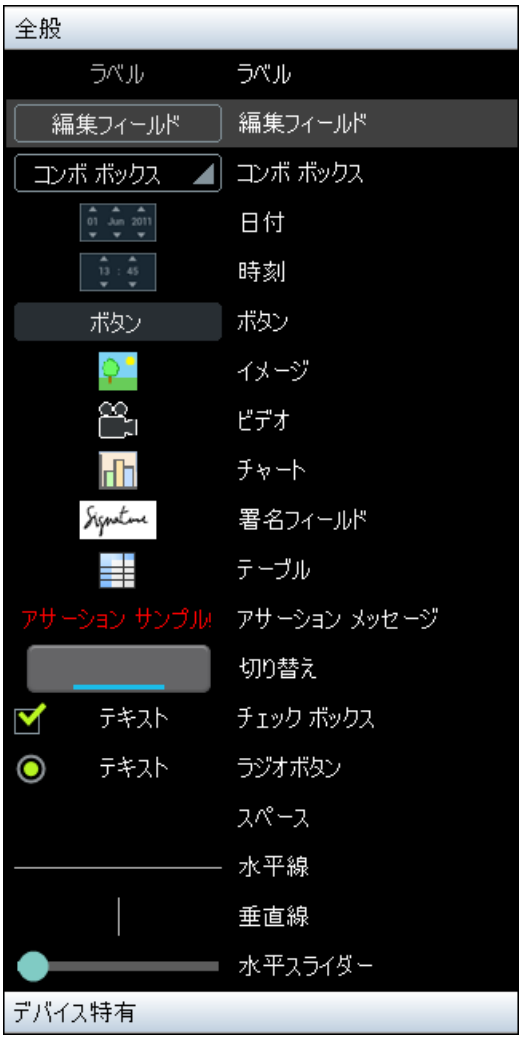
MobileTogether Server にデプロイ
プロジェクト ファイルの場所
プロジェクトのデプロイ
サーバー上のデータストレージ

5.4 コントロール ペイン

[コントロール] ペインは、ページデザインまたはワークフローに追加することができ、すべてのコントロールを表示します。コントロールパネルのデフォルトのロケーションを確認するには、グラフィカル ユーザー インターフェイス(GUI) に移動してください。

ページ デザイン コントロール

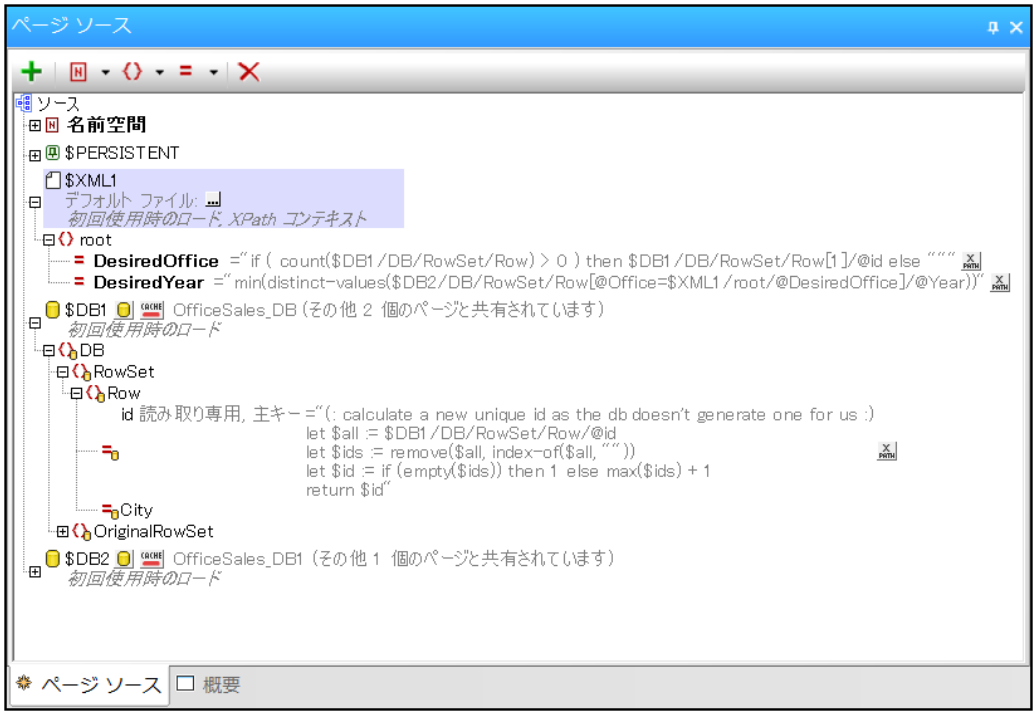
ページデザインコントロールは、ページ デザイン タブがメインウィンドウで選択されていると使用可能です。[コントロール] ペインの外観とコントロールは、現在選択されているデバイスに対応しています。例えば、下のスクリーンショットで表示されている[コントロール] ペインは、Android LG Optimus 7 (左) とiPhone 6 Plus (右) です。



ペインのコントロールは*一般* セクションと*デバイス特有*のセクションに整理されています。コントロールをページ デザインに追加するには、ページ デザイン内の希望する場所にコントロールをドラッグ アンド ドロップします。

5.5 ページソース ペイン

[ページ ソース] ペイン(*下のスクリーンショット*) でページ データソースが管理されます。データソースは、(i) デザインで使用されたデータツリーの構成と(ii) データツリーに保管されているデータの両方を与えられています。



ペインには以下のメイン機能があります:

- [ページ] ペインで現在選択されているページの全てのデータソースを表示する。
- アクティブなプロジェクトのために宣言されているすべての名前空間を表示する。
- ツールバーの**[ソースの追加]** アイコンを介して、ページへデータソースの追加を有効化する。
- ツールバーの**[名前空間の追加]** アイコンを介して、プロジェクトへの名前空間の追加を有効化する。
- データセットのデフォルトのファイルの設定を有効化する。データファイルは、データソースのノードへ入るデータを提供します。
- 選択されたノードに対応して、ツリーへの要素および属性の追加を有効化する。
- 要素と属性に固定された値またはXPathにより生成された値をページのロード時に追加することを有効化します。
- デフォルトのXPath コンテキスト(*上のスクリーンショットのハイライトされたノード*) ノードの設定を有効化する。選択されたノードは、ページで定義されている、すべてのXPath 式のコンテキスト ノードとなります。
- ページ デザインのコントロールに関連したノードを有効化する。これは、ノードをコントロールにドラッグすることにより行います。(ページソースリンクと呼ばれる)関連したノードは、太字で表示されます。データソースツリー内のこの様なノードにマウスをポイントすると、デザイン内の関連したコントロールに関する情報がポップアップされます。ページソースリンクに関連したコントロールにはコントロールの左上にアイコンが表示されます。アイコンにマウスをポイントすると関連したページソースリンクの情報が表示されます。
- ペインのアイテム(名前空間、ツリー、要素、および属性) の削除を有効化する。

ページソースとの作業方法の詳細に関しては、セクション[ページ データソース]を参照してください。

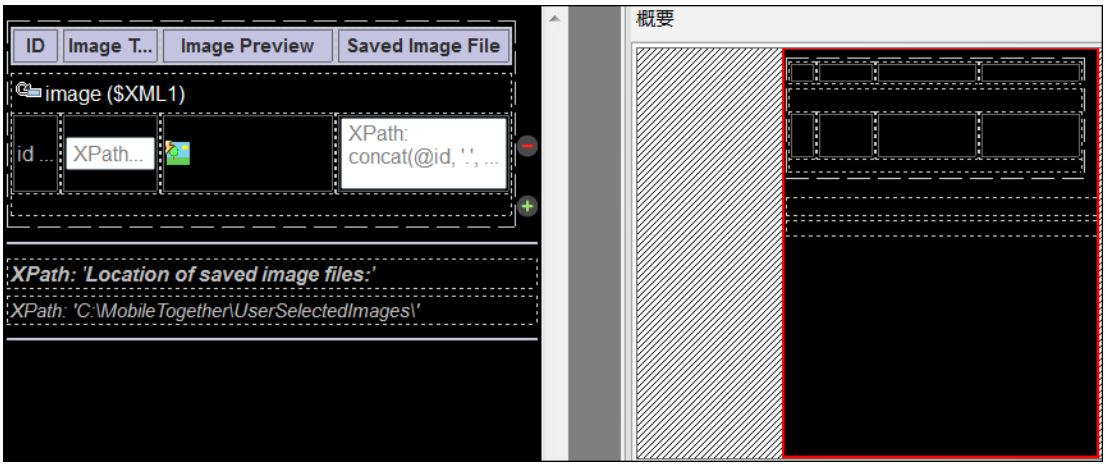
手動によるツリー構造の作成

要素と属性は(ルート ノードを含む)ツリー構造のノードに対応して追加することができます。データソースのノードを選択して、適切なツールバーコマンドをクリックします (*下部のツールバーのスクリーンショット参照*)。一時的な要素と属性は、計算のためにデータを保つ、または他の理由のためにファイルに保存されません。一時的ノードのデータは保存されません。

アイコン	コマンド	方法 ...
+	**ソースの追加**	ページソースの追加ダイアログを表示します。ルート ノードが追加されたデータソースのために作成されます。ルートノードには子要素 1 つだけ追加できます。
	名前空間の追加	名前空間宣言を*名前空間* エントリの下に挿入または追加します。デフォルトのプレフィックスを必要に応じて編集して、名前空間を入力します。
	要素の追加	選択されたノードに関連した子要素を挿入、追加します。
=	**属性の追加**	選択されたノードに関連した子要素を挿入、追加します。
×	**削除**	選択されたノードを削除します。

5.6 概要ペイン

[概要]ペイン(*下のスクリーンショット*) は[ページ デザイン ビュー]の小さいバージョンを表示します。[概要]ペインのデフォルトの場所を確認するには、[グラフィカル ユーザー インターフェイス(GUI)]に移動します。



上のスクリーンショットの[概要]ペインでは、デザインの範囲は黒のエリアで示されています。赤枠の四角はビューポートです。これは、[ページ デザイン ビュー]で、現在見える、デザインの部分を示していて、デザインの表示されている部分のディメンションに相当します (*上と下のスクリーンショットを比較してください*)。デザイン内のコンポーネントが選択されていると、選択されたコンポーネントが[概要]ペイン内で表示されます(*スクリーンショット参照*)。デザインの一部がビューポートで表示されていない場合、[概要]ペインの赤い四角をつかみ移動させることにより、デザインの必要な部分がビューポートに表示されます。上のスクリーンショットは、ビューポート内のデザイン全体を表示していますが、下のスクリーンショットでは、デザインの一部がビューポートの外部にあります。これらの部分は赤枠の四角をドラッグすることにより移動することができます。

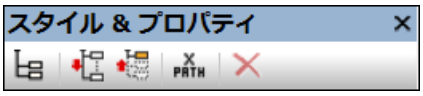
[概要]ペインは、大きなダイアグラムを閲覧する場合とても役に立ちます。

5.7 スタイル ＆プロパティペイン

[スタイル ＆ プロパティ] ペイン(*下のスクリーンショット*) では、現在選択されている[ページ] コントロール、ページ、プロジェクトのプロパティを表示して、1 つの場所で便利に編集することができます。[スタイル ＆ プロパティ] ペインは次のサブペインに分けることができます: (i) デザインで現在選択されている[ページ] コントロールのプロパティのための *コントロール* (ii) 現在のページのプロパティのための ページ と(iii) 現在のプロジェクトのプロパティのための*プロジェクト*。テーブルコントロールとそのパート(セル、列、行) は個別のペインを与えられています (*下のスクリーンショット参照*)。テーブルとテーブル パートのサブペインはデザインでテーブルの一部分が選択されている場合表示されます。

[スタイル ＆ プロパティ] ペインツールバー

スタイル ＆ プロパティツールバー (*下のスクリーンショット*) で使用可能なコマンドは以下のテーブルにリストされています:



アイコン	コマンド	方法
	空でない者の表示	すべてのプロパティと値が定義されているプロパティの表示を切り替えます。デフォルトの値を持つプロパティは空と見なされます。
	すべて展開	すべてのサブペインを展開します。
	すべて縮退	すべてのサブペインを縮退します。
	XPath の編集	XPath 式を必要とするプロパティが選択されると有効化されます。[XPath/XQuery 条件式の編集] ダイアログを表示します。
	リセット	プロパティをリセットして、デフォルトの値を空にします。

プロパティの値の入力と編集

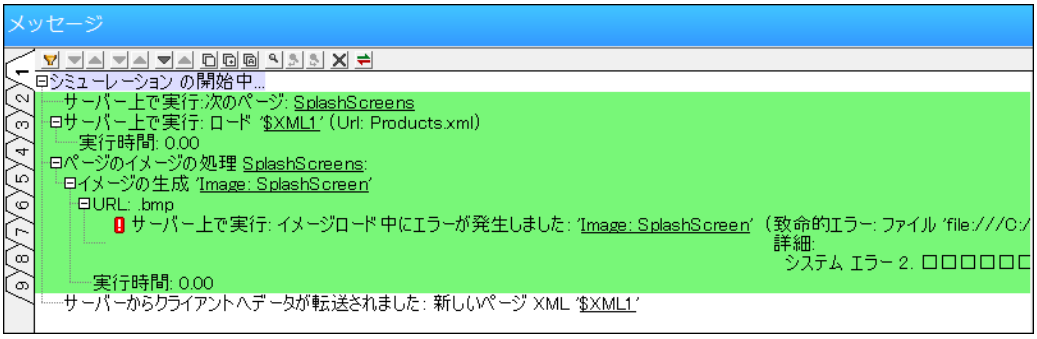
プロパティの値はプロパティのエントリフィールドの種類によって入力されます:

エントリフィールドの種類	方法 ...
テキスト フィールド	ダブルクリックして入力または、プロパティの値を入力または編集します。
コンボ ボックス	ドロップダウン リストから値を選択します。
カラー パレット	カラーパレットをポップアウトして色の選択を有効化します。
XPath の編集	XPath/XQuery 式の編集ダイアログをポップアップして、XPath/XQuery 式の入力を有効化します。式はプロパティの値を提供します。
追加ダイアログ	関連したダイアログをクリックして開きます。これには、例えば、アクションダイアログまたはファイルを開くダイアログなどがあります。

5.8 メッセージ ペイン

[メッセージ] ペイン(*下のスクリーンショット*) は次のコンテキストのメッセージを表示します:

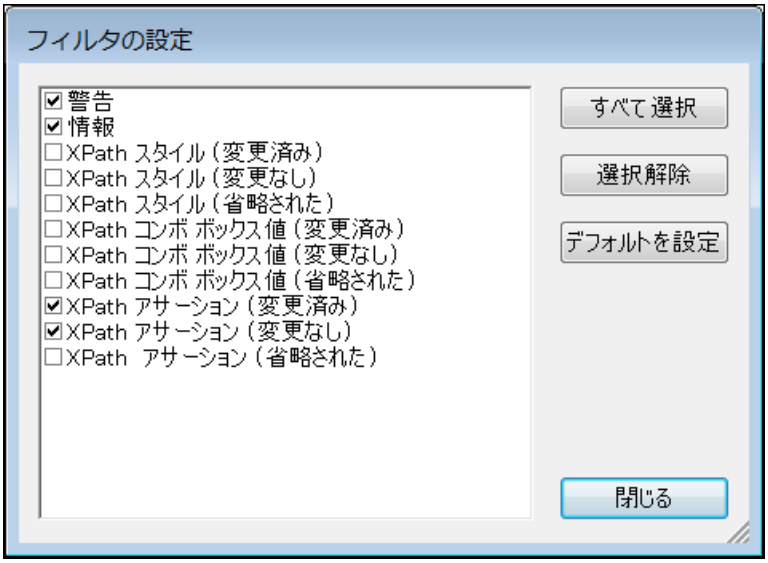
- 現在アクティブなプロジェクトの検証の結果を報告します。プロジェクトの全てのページが含まれます。エラーが報告された場合、エラーを生成したコンポーネントをポイントするリンクをエラーメッセージは含みます。
- シミュレーション実行中、ワークフローの詳細と進行状況のステップバイステップのレポートが提供されます。



[メッセージ] ペインのツールバーは以下のアクションを有効化するコマンドを含みます: メッセージのフィルター、メッセージのナビゲート、メッセージのコピー、メッセージの検索、サーバーおよびクライアントログメッセージの背景の色の設定、現在のタブのクリア。9 つのメッセージタブがあります。ですから、新しいタブでシミュレーションを実行しながら、タブ内で検証の結果を保持することができます。

メッセージのフィルター

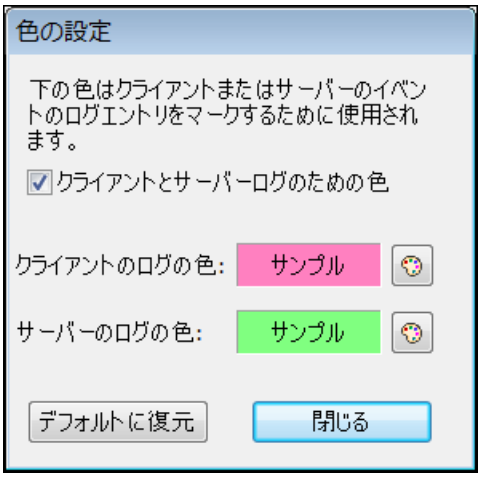
メッセージペイン内に表示されるメッセージを指定することができます。これを行うには、メッセージペイン内のツールバーの**[フィルター]** ボタンをクリックします (*上のスクリーンショット*)。これによりフィルターの設定ダイアログ(*下のスクリーンショット*)が表示されます。表示するメッセージの種類を選択し、**[閉じる]** をクリックします。この機能は、例えば、メッセージが多すぎる場合に1つの種類のメッセージに注目する場合に役に立ちます。



色の設定

シミュレーション中に表示されるメッセージに、サーバーとクライアント上で行われるアクション別に異なる色を設定することができます。見分けのつきやすい色を設定すると、メッセージペイン内のワークフローを簡単にフォローすることができます。この機能はデバッグにとても役に立ちます。カスタム化された色を設定するには、メッセージペインのツールバー内の**[色]** ボタンをクリッ

クします (*上のスクリーンショット*)。これにより色を設定することのできる色設定ダイアログ(*下のスクリーンショット*) が表示されます。



チャプター 6

プロジェクト

6 プロジェクト

このセクションでは、プロジェクトレベルで相対するオプションについて説明されています:

- クライアントとサーバー間のインタラクション
- プロジェクト ファイルのロケーション
- プロジェクトのデプロイ
- プロジェクト プロパティ
- ローカライズ
- 名前空間
- グローバルリソース
- パフォーマンス

(C) 2018 Altova GmbH

6.1 クライアントとサーバー間のインタラクション

このトピックでは、クライアントデバイスとMobileTogether Server 間のインタラクションのレベルを決定する機能と設定について説明されています。

MobileTogether Client アプリまたは MobileTogether AppStore App

まず、プロジェクトが**MobileTogether Client アプリ**のためのソリューションとして、MobileTogether Server に配布されるか、または**MobileTogether AppStore App** のためのソリューションとして配布されるかを決定します:

- エンドユーザーはアプリストアからMobileTogether Client アプリをダウンロードすることができます。エンドユーザーデバイスでは、MobileTogether Client は、1つまたは複数のMobileTogether Server にアクセスできるように構成されています。セキュリティの設定により、サーバーに匿名、または、パスワードを使用するユーザーログインによりアクセスすることができます。エンドユーザーは、サーバーの構成とアクセスの詳細に関する情報が必要です。エンドユーザーがサーバー上のフォルダーへのアクセスを許可されると、フォルダーのソリューションとしてデプロイされている、MobileTogether プロジェクトにエンドユーザーはアクセスすることができます。MobileTogether Server の管理者によりアクセスに関する権限は管理されています。詳細に関しては、次を参照してください：MobileTogether Server ドキュメント。
- MobileTogether AppStore App は、他方、スサーバー上に存在する単一のソリューションに特化したタンドアロンのアプリです。AppStore App は、アプリストアからダウンロードされ、エンドユーザーのデバイスから直接開始することができます。MobileTogether Client が、この種類のアプリを実行するためにインストールされている必要はありません。しかしながら、ソリューションにアクセスするために、適切な MobileTogether Server への接続が必要になります。アプリには、サーバー上のソリューションと接続するためのキーが含まれています。サーバーとのインタラクションは、ですから、サーバーアクセス設定の値により異なります。詳細に関しては、次を参照してください：AppStore App。

プロジェクトのサーバーアクセス設定

プロジェクトの *サーバーアクセス* 設定は、ソリューションの実行中にサーバーアクセスのレベルを指定します。3つのオプションから選択することができます:「常に」、「オンデマンド」、と「無し」。適切なオプションをサーバー上のリソースの種類に従い、ソリューションの必要に従い選択します。「無し」オプションが選択されると、1度サーバーへの接続が作成されると、サーバーにアクセスすることはできません。設定の詳細に関しては、次を参照してください：プロジェクトプロパティ。

MobileTogether Client のための匿名 ログイン

MobileTogether Client アプリが、MobileTogether Server に接続されると、エンドユーザーは、サーバーに識別されているユーザー、または、匿名でログインすることができます。ユーザーとしてログインするには、MobileTogether Server により識別されているユーザー名とパスワードが使用される必要があります。または、サーバー管理者は、MobileTogether Server を構成して、匿名のログインに対してフォルダーへのアクセスを許可することができます。詳細に関しては、次を参照してください：MobileTogether Server ドキュメント。

6.2 プロジェクトファイルの場所

`.mtd` 拡張子を持つプロジェクトファイル(いわゆるデザインまたはMTD ファイル) は、サーバーにデプロイされます。MobileTogether Client アプリによりアクセスされるソリューションのファイルです。プロジェクトファイルがサーバーにデプロイされると、MobileTogether Server データベースに保管され、サーバーはファイルを名前で参照します。プロジェクトファイルは、データを読み込み、さらに、データを書き込むことのできる、(XML ファイルおよびイメージファイルなどの)他のファイルを使用します。これらの関連したファイルは以下の場所に保管することができます:

プロジェクトファイルのサーバーへのデプロイ

- これらのデータファイルはサーバー上では読み取り専用です。
- ファイルはサーバーのデータベースに保管されます。データファイルへのアクセスがプロジェクト内部で行われることが利点です。
- ファイルはデザイン内で相対または絶対パスを使用して参照されることができます。ファイルがデプロイされると、デザイン内でファイルパス(相対または絶対) を構築する文字列がサーバーデータベース上で内部ファイルの参照として使用されます。
- デプロイに関する詳細はプロジェクトのデプロイ、[ファイル] ペイン、MobileTogether Server へのデプロイを参照してください。

プロジェクトファイルへの埋め込み

- これはXML データファイル(通常、データソースのデフォルトファイル)に適用されます 。
- 利点は、XML データとイメージがプロジェクトファイルと共に運ばれるため、プロジェクト内で正確にアクセスすることができます。ですから、これらのファイルのためのサーバーアクセスは必要ありません。
- デプロイされたファイルに関しては、埋め込まれたXML ファイルは読み取り専用です。
- 不利な点は、プロジェクトファイルのサイズが増えることです。
- ファイルを埋め込むには、[ページソース] ペインのルートノードを右クリックして、**[デザインファイルにXML を埋め込む]** を選択します。ユーザーがソリューションを開始またはソリューションをサーバにデプロイする時、ファイルが自動的に再埋め込みされるかどうかを選択できます。

サーバー上でのディレクトリ

- サーバー上でのディレクトリで保管することのできるファイルの種類。ファイルは読み取り専用です。
- しかし、正確に以下の構成を設定する必要があります (i) 追加された時のファイルの場所 、および (ii) MobileTogether Server の*サーバー側の作業ディレクトリ*の設定。
- ファイルが相対パスにより参照されている場合、相対パスは作業ディレクトリに相対して解決されます。
- ファイルが絶対パスで参照されている場合、ファイルを含んでいるディレクトリは作業ディレクトリの子孫ディレクトリである必要があります。例えば
  参照されたファイルの絶対パスが次の場合: `C:\Altova\MobileTogether\Test\First.xml`
  そして、作業ディレクトリが次に設定されている場合: `C:\Altova\MobileTogether`
  XML ファイルは正確にアクセスすることができます。
  作業ディレクトリが次に設定されている場合は、正確にアクセスすることができません: `C:\Altova\MobileTogether\files` または`C:\Altova\MTD`
- サーバ上のディレクトリを使用してデータファイルを保存する利点は、ソリューションによりアクセスされるデータが常に最新であるという点です。

インターネットによりアクセスすることのできるURL

- インターネットでサーバがアクセスすることのできるURL に保管されているすべての種類のファイル。
- ファイルがデータソースとして追加された場合、データソースのプロパティを指定するセキュリティの認証を設定すること

ができます。

- インターネットのロケーションを使用する利点は以下のとおりです: (i) ソリューションによりアクセスされるデータが常に最新であること、および (ii) ソリューションがポータブルであること。

関連項目

MobileTogether Server へデプロイ
[ファイル] ペイン
プロジェクトのデプロイ
サーバー上のデータストレージ

6.3 プロジェクトのデプロイ

MobileTogether Designer でプロジェクトのデザインを完成させると、プロジェクト(またはデザイン) を1 つ以上の MobileTogether Serverにデプロイすることができます。MobileTogether Server にデプロイするには、ターゲットの MobileTogether Server が作動しているコンピューターへのHTTP 接続が必要です。プロジェクトがデプロイされると、MobileTogether Client アプリケーションがモバイルデバイスでアクセスすることのできる、MobileTogether ソリューションとして使用することができます。

デプロイとアクセスコントロール

使用することのできるデプロイオプションにより、ソリューションへのアクセスを管理する柔軟性を得ることができます。アクセスコントロールには2 つのレベルがあります。

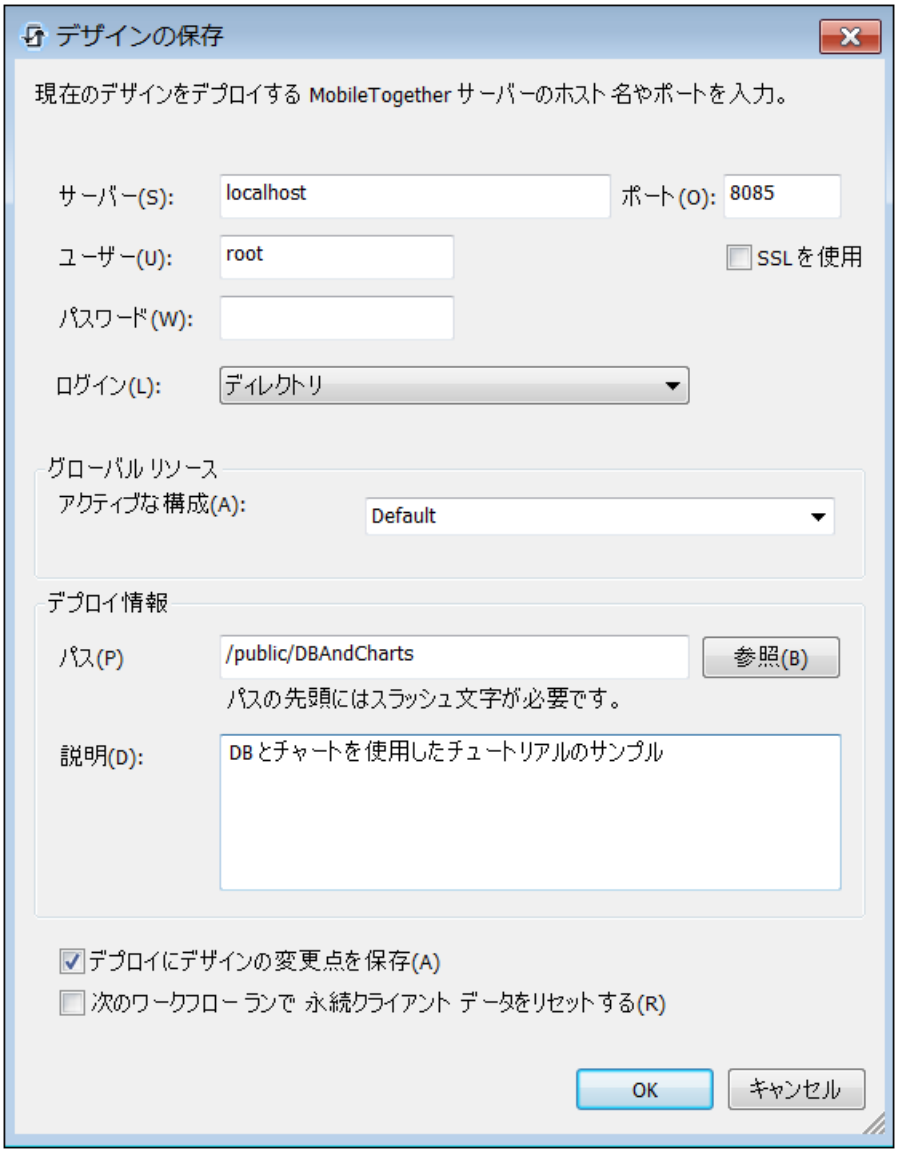
最初のレベルでは、アクセスは、許可されるサーバーアクセス(内部/外部) の種類によるサーバーレベルで管理することができます:

- ファイアウォールの背後での社内サーバーへのデプロイは、自動的にアクセスを例えば、社員などの、内部ユーザーに制限します。
- 外部アクセスサーバーへのデプロイは、例えば、カスタマーまたは会社のクライアントなどの、外部エンドユーザの MobileTogether ソリューションへのアクセスを許可します。

第 2 のレベルでは、各サーバーに対して、そのサーバー上にあるソリューションにアクセスすることのできるユーザーを定義することができます。アクセスは適切なユーザー名とパスワードを持つクライアントのみに与えられます。サーバーのユーザーと特権は MobileTogether Server の設定で定義されます。ユーザー、ロール、ユーザーの特権の設定方法に関する詳細は「MobileTogether Server ユーザーマニュアル」を参照してください。

プロジェクトのデプロイ方法

プロジェクトは **[ファイル] サーバーへデプロイ]** コマンドを使用してデプロイすることができます。このコマンドは、サーバー接続の詳細を指定し、また、サーバーがSSL 通信を使用するかを指定するMobileTogether Server へデプロイダイアログを表示します (*下のスクリーンショット*)。



デプロイとは?
[ファイル] [サーバーへデプロイ] コマンドを利用して、次のファイルをプロジェクトにデプロイすることができます:

- .mtd 拡張子を持つプロジェクトファイル (いわゆるデザイン または MTD ファイル) は、サーバーにデプロイされます。MobileTogether Client アプリによりアクセスされるソリューションのファイルです。
- [ファイル] ペインのデプロイ可能なすべてのファイルのチェックボックスのチェックが選択されています。これらのファイルは通常イメージファイルとデータソースのデフォルトのファイルです。

デプロイされたすべてのファイルはサーバーに保存され、ソリューションファイルにより自動的に正確にアクセスすることができます。これにより、ファイル パスが正確であるか確認する必要がなく、とても役に立ちます。しかし、これらのファイルは読み取り専

用であることに注意してください。ですから、書き込みが必要なファイルはデプロイすることができないため、サーバーに手動で保存される必要があります。書き込み可能なファイルにアクセスできるように、サーバとデザインファイルは正確に構成される必要があります。この点に関する詳細はプロジェクト ファイルの場所 | サーバー上のディレクトリのセクションを参照してください。

メモ オーディオ/ビデオファイルは、MobileTogether Designer プロジェクトのサーバーメカニズムにデプロイを使用して MobileTogether Server にデプロイすることはで**きません**。しかしながら、URL を使用してストリームすることはできませんが、サーバーに手動でオーディオ/ビデオファイルをコピーすることができます。MobileTogether Server に存在する、オーディオ/ビデオファイルをストリームするには、以下を行います : (i) バイナリのロード アクションを使用して、バイナリのオーディオ/ビデオデータをデータソースノードにロードします。(ii) バイナリの保存 アクションを使用して、このノード内のデータをクライアント上のファイルに保存します。(iii) オーディオ/ビデオの再生アクション を使用して、クライアントデバイス上に保存されたファイルを再生します。または、オーディオ/ビデオファイルを MobileTogether Server に保存する代わりにWeb サーバーに保存し、URL を使用して、Web サーバーからオーディオ/ビデオファイルをストリームします。

デプロイされたファイルとプロジェクトファイルの場所

デプロイされたファイルは読み取り専用です。([ファイル] ペイン内で) デプロイがマークされているファイルに保存アクションが定義されている場合、ファイルはデプロイ時読み取り専用であり、書き込むことができないためデザインは無効になります。デプロイされたファイルはデザイン内に保存され、デザインから読み取ることができます。

デプロイされていないファイルは、デザインで正確に参照されているサーバーの場所に保管される必要があります。ファイルに正確な参照を作成するには、以下を正確に構成する必要があります (i) 追加時のファイルの場所 および (ii) MobileTogether Server の*サーバー側のソリューションの作業ディレクトリ*の設定。詳細はプロジェクト ファイルの場所 | サーバー上のディレクトリのセクションを参照してください。

クライアントデバイス上のサーバー設定を更新する

クライアントデバイスでソリューションを実行するには、サーバーのアクセス設定は、そのデバイスで構成される必要があります。サーバー設定が変更された場合、例えば、 MobileTogether Server が、異なるIP アドレスを有する他のマシンに移動された場合、クライアントデバイス上のサーバー設定は必要に応じて変更されなければなりません。MobileTogether 関数 `mt-server-config-url` は、以下に類似した、新しいサーバー設定を含むURL を生成します: `mobiletogether://mt/change-settings?settings=<json encoded settings>` 。このURL は電子メールリンクとして MobileTogether Client デバイスに送信されることができます。リンクがタップされると、クライアント上のサーバー設定は、自動的に更新されます。

URL に含まれている、JSON-エンコードサーバー設定 `mt-server-config-url` 関数 (ここで説明されている) の引数により提供されます。この関数の使用方法の例は、`MobileTogetherExamples/SimpleApps` フォルダー MobileTogether Designer インストール内にあるサンプルソリューション **`ClientConfiguration.mtd`** を参照してください。

メモ サーバー設定の更新を行うリンクは Gmail 内および他の電子メールアプリケーション内では、作動しませんが、AquaMail、K9、および MailWise などのよく使用されるクライアントでは作動します。AquaMail および K9 内でテスト済みでこれらのアプリケーションでは正常に作動します。

関連項目

MobileTogether Server へデプロイ、サーバーへデプロイの詳細 (*上のスクリーンショット*)
プロジェクトファイルの場所
[ファイル] ペイン
サーバー上のデータストレージ

6.4 プロジェクトプロパティ

プロジェクトのプロパティは[スタイル & プロパティ] ペインで定義され、以下で説明されています。マウスをプロパティ名にポイントすると、プロパティの説明を表示するポップアップが表示されます。

▷ ページ	
▽ プロジェクト	
サーバー アクセス	常に
クライアントがサーバーからの応答待ちする際のタイムアウト	15
データを取得する際のサーバーのタイムアウト	10
オーディオ アクション	
スピーチアクションへのテキスト	
NFC アクション	
ワークフローの終了をユーザーに確認する	true
ワークフロー終了のメッセージ	このソリューションを終了しますか
他のソリューションに切り替える	このソリューションをキャンセルする
ワークフロー アイコン	
ブラウザー設定	
他のプロジェクト設定	

サーバーアクセス

このオプションはソリューション実行中のサーバーのアクセスレベルを指定します。デフォルトは常にです。

- **常に**: ソリューションの実行にはサーバへの接続が必要であり、ソリューションの実行中は常にサーバーにアクセスされています。
- **オンデマンド**: MobileTogether Client アプリは自身のソリューションを実行します。サーバーにはサーバーとのデータの交換が必要な場合だけ接続します。ソリューションを実行するには、アプリは内部の `$PERSISTENT` ツリー、他の永続データ または埋め込みデータを使用します。XPath 関数 **`mt-has-serveraccess`** を使用して、サーバー接続を確認して、適切に保存するアクションを使用します。例えば、接続が存在しない場合、データはクライアントに永続データとして保存されることができます。サーバーへの接続が確立されると、データはデータベースおよびまたはサーバー上のファイルに保存することができます。
- **なし**: MobileTogether Client アプリは、インターネットへ空のデータを必要することなく自身でソリューションを実行します。

タイムアウト: クライアントがサーバーからの応答待ちする際のタイムアウト

クライアントがサーバーからの応答待ちする際の待ち時間。値は整数の秒数で、入力またはコンボ ボックスのドロップダウン リストから選択されます。 デフォルトの値は 15 秒です。タイムアウトが過ぎると、クライアントにメッセージが表示されます。

タイムアウト: データを取得する際のサーバーのタイムアウト

サーバーの外部のソース(DB または URL など) からデータを取得する際のサーバーの待ち時間。値は整数の秒数で、入力またはコンボ ボックスのドロップダウン リストから選択されます。 デフォルトの値は 10 秒です。タイムアウトが過ぎると、クライアントにメッセージが表示されます。例外は、ロード アクションの エラー時の設定が `Continue` となっている場合です。 この場合、`Continue` 設定のエラー時 ツリー内で定義されているアクションが実行されます。

ライトのテーマを強制

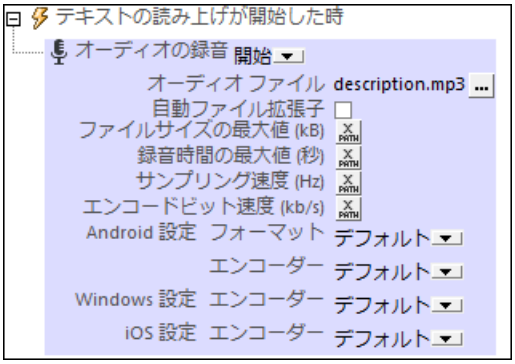
ライトのテーマはライトの背景に表示される暗いテキストを指します。このプロパティの値を true に設定し、プロジェクト全体のためにライトのテーマを調整することができます。false (デフォルトの値) に値を設定すると、クライアントデバイスの現在のテーマが使用されます。

オーディオアクション

オーディオアクションはプロジェクト全体のためにグローバルに定義することができます。このアクションは次の3つのイベントで使用することができます: **オーディオの開始時**、**オーディオのエラー時** および **オーディオの完了時**。これらのイベントのために定義されるアクションはプロジェクト内の**すべてのオーディオ再生イベントに適用することができます**。プロジェクトのオーディオイベントの定義を含む**オーディオダイアログ**ボタンのプロパティをクリックすることで、それぞれのイベントのために、左側のアクションペインからアクションをイベントタブにドラッグアンドドロップして行われるアクションを定義することができます。詳細に関しては、*オーディオ(再生) 機能に関する説明*を参照してください。

読み上げアクション

プロパティの追加ダイアログボタンをクリックすると、プロジェクトアクションダイアログの読み上げアクションが表示されます (*下のスクリーンショットを参照してください*).



次の読み上げイベントを使用することができます:

- 読み上げの開始時: このペインで指定されるアクションは、読み上げアクションが開始されるとシーケンスで実行されます。例えば、上のスクリーンショットで表示されるように オーディオの記録アクション は、ファイル内の読み上げの再生を記録するために開始することができます。
- 読み上げのエラー時: テキストが見つからないなどの、読み上げのエラーが発生すると、実行されるアクションです。
- 読み上げの完了時: 読み上げの再生が完了すると実行されるアクションです。例えば、このイベントのための読み上げアクションを指定することにより他の読み上げの再生を開始することができます。

NFC アクション

2つの NFC に関連したイベントのために定義されたアクションを有効化します:

- **NDEF メッセージのプッシュの完了時** は、(NFC プッシュを使用して) NFC データの転送が完了すると実行されるアクションを指定します。
- **NFC タグの検出時** は、NFC タグが検知されると実行される(追加) アクションを指定します。

プロパティの **「追加ダイアログ」** ボタンをクリックして、2つのイベントの定義に移動してください。詳細に関しては、NFC に関連したイベントを参照してください。

プッシュ通知 アクション (プッシュの通知の受信時)

デザイン時で、プッシュ通知が受信された時に実行されるアクションを指定する **プッシュの通知の受信時** イベントタブを開きます。イベントにアクションが追加されると、`$MT_PUSHNOTIFICATION` ページソースが自動的にデザインに追加されます。

プッシュ通知 (PN) がデバイスで受信されると、*受信先でソリューションが既に実行されている場合の*設定により2つのうちの1 つの選択肢が実行されます:

- 受信されるソリューションの `$MT_PUSHNOTIFICATION` ページソースが通知無しに、プッシュ通知ペイロードと共に更新され、**プッシュの通知の受信時** イベントタブ内のアクションが実行されます。これらはプッシュ通知を表示することなく直接行われます。
- プッシュ通知の表示。ユーザーがPN (または、PN 内のボタン) をタップすると、次の状態が発生します: (i) 実行されていない場合、開始するソリューションが開始されます。(ii) ソリューションの `$MT_PUSHNOTIFICATION` ページソースがプッシュ通知ペイロードからのデータと共に更新されます。(iii) **プッシュの通知の受信時** イベントタブ内のアクションが実行されます。

詳細に関しては、プッシュ通知 を参照してください。

ワークフローの終了をユーザーに確認する

ブール値設定で、ユーザーにソリューション(ワークフロー) の終了をユーザーに確認します。`true` または `false` をコンボボックスから選択します。デフォルトの値は `true` です。もしの場合、`true` 場合、テキストは次のプロパティの値として定義され、ソリューションが終了する前にワークフローの終了メッセージが表示されます。ユーザーが、ソリューションの最初のページで **戻る** ボタンを押すことを問われることが通常考えられます。ユーザーは、**送信** ボタンが押されると、または、キャッシュアクションの実行が処理されると終了の確認を問われません。

ワークフローの終了メッセージ

(ワークフロー) ソリューションの終了を確認するメッセージが表示されます。メッセージは、前のプロパティの*ワークフローの終了をユーザーに確認する*が `true` と設定されている場合のみ表示されます。デフォルトのメッセージは ソリューションを終了しますか?

他のソリューションに切り替え時

ソリューションの実行中、エンドユーザーは、他のソリューションに切り替えることができます。これが発生すると、他のソリューションへの切り替えの設定は、元のソリューションが継続的に背景で実行されるかキャンセルされるかを決定しま

す。ソリューションが背景で実行され、中断されると、それ以上は実行されません。例えば、タイマーが実行されず、位置情報が使用されません。ソリューションが再開されると、ページ更新時 イベントの再開オプションが実行されます。設定のオプションは、以下のとおりです:

- *このソリューションをキャンセルする* デフォルトの値。ソリューションがキャンセルされます。保存されていないデータはすべて削除されます。
- *継続的に実行*: ソリューションが背景で継続的に実行されます。アイコンはデバイスの実行タブで使用することができます。元のソリューションに切り替えるには、エンドユーザーは実行タブ内のソリューションのアイコンをクリックします。

メモ このプロパティをテストするには、ソリューションはサーバーにデプロイされている必要があり、そこから実行することができます。

メモ ソリューションのキャンセルまたは縮小化を指定する他の方法であるソリューションの実行 アクションも参照してください。

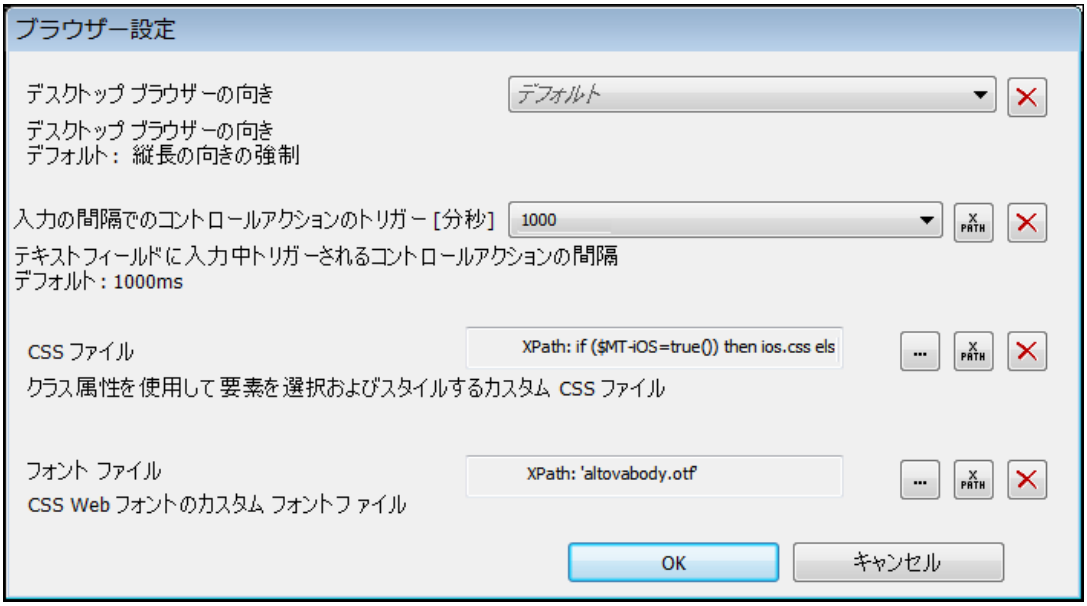
メモ Web クライアントは、保留されているソリューションをサポートしません。アクティブなソリューションのみがサポートされます。

ワークフローアイコン

プロパティの [追加ダイアログ] ボタンをクリックして、PNG イメージファイルをプロジェクトのアイコンとして使用できる参照ダイアログを表示します。デフォルトでは、クライアントアプリ上の MobileTogether のアイコンが使用されます。

ブラウザーの設定

ブラウザー設定 プロパティの追加ダイアログ ボタンをクリックすることにより、ブラウザー設定ダイアログ (*下のスクリーンショット*) が表示され、モバイルデバイスのブラウザーに関連した特定の設定を定義することができます。これらの設定は以下で説明されています。

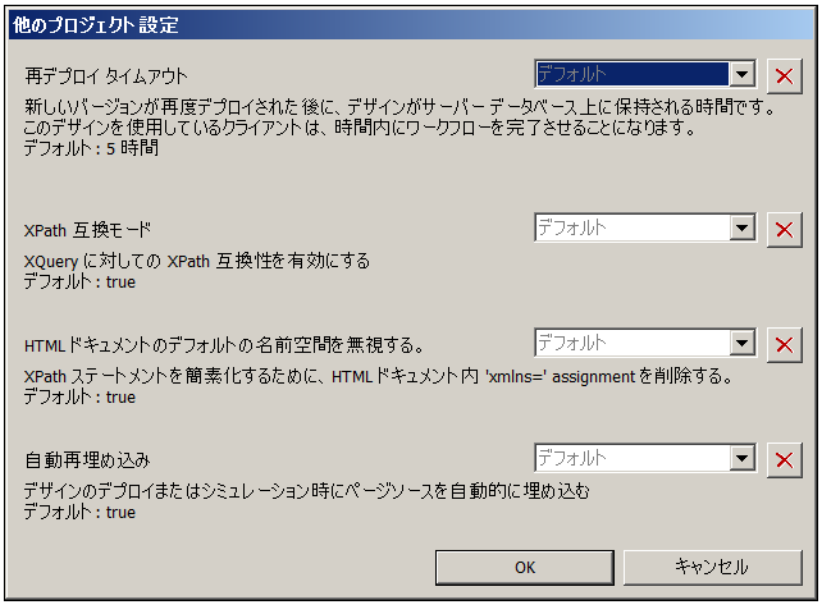


以下の設定を定義することができます:

- *デスクトップ ブラウザーの向き* コンボボックスのオプションにより、ブラウザーの向きを選択することができます: *縦向きと横向きの強制*。デフォルトは、*縦向きの強制*です。
- *タイプ間隔のコントロール アクションのトリガー*: (モバイルデバイス上のMobileTogether Client アプリとは対照的に) この設定はWeb クライアントのみに適用することができます。Web クライアントで編集されるソリューションページの更新は、サーバーで処理するために送信される必要があるため、更新されたデータが何時サーバーに送信されるかを指定することは役に立ちます。ここで選択される値は、更新されたデータが送信された後の時間の間隔です。サーバーは、すべての影響されるページのコンポーネントが更新されるように、更新されたデータを処理し、返します。この設定のデフォルトは`1000` ミ秒 です。(コンボボックスのドロップダウンリストから無効化を選択することにより)設定が無効化され、エンドユーザーがページ内をクリックするなどして、コントロールアクションがトリガーされたます。XPath 式は、文字列`'disabled'`またはMobileTogether Designer により、ミ秒の数値で読み込まれた数値である必要がある設定の値を取得するために使用することができます。
- *CSS ファイル*: デザイン内のコントロールのクラスセレクターに割り当てられた外部 CSS プロパティを評価するために読み込まれるCSS ファイルの指定。デザイン内の各コントロールは、そのコントロールに特有の クラス名を定義するブラウザー `CSS` クラスと称されるプロパティを持ちます。これらのクラスセレクターのCSS プロパティは、サーバーにデプロイされている、外部 CSS ファイル内で定義されます。クラスルールを検索するCSS ファイルはこの(*CSS File ファイル*) 設定で指定されています。ファイルパスまたはグローバル リソース エイリアスを使用してCSS ファイルを選択することができます。XPath 式を使用して、ファイルパスを生成することもできます。異なるクライアントの代替 CSS ファイルは、`if...then...else` 等のXPath の条件構文を用いて指定することができます。(*上のスクリーンショット参照*)。外部 CSS ファイル内で定義されたCSS ルールは、コントロールのプロパティで作成された定義よりも低い優先順位が与えられる点に注意してください。
- *フォント ファイル*: システムフォントに追加して、デザイン内で埋め込むフォントファイルを指定します。フォントファイルを参照、グローバルリソース内で検索、またはXPath 式によりファイルパスを生成することができます。デザイン内でこの方法で埋め込まれたフォントはCSS の`font-family` プロパティを使用して参照されることができます。
- *ブラウザーウィンドウ/タブを閉じる際にユーザーに確認する* ページから移動するかユーザーに問うメッセージボックスが表示されます。変更が保存されていない場合、メッセージはユーザーが移動をクリックすると、ユーザーに変更が保存されていないことを伝えます。ユーザーは更に、追加ダイアログを作成することを避けるかを決定することができます。

他のプロジェクト設定

他のプロジェクト設定のプロパティの[追加ダイアログ] ボタンをクリックすることで、他のプロジェクト設定ダイアログ(*下のスクリーンショット*) が表示されます。ユーザーがシミュレーションを開始またはサーバーにソリューションをデプロイする時、ファイルが自動的に再埋め込みされるかどうか選択できます。



以下の設定を定義することができます:

- *再デプロイタイムアウト*: 新しいソリューションのバージョンがデプロイされた後に、前のバージョンがサーバーに保持される時間です。この時間のオーバーラップにより、古いソリューションを使用しているクライアントは作業を完了することができます。デフォルトは5 時間です。
- *XPath 互換モード*: `true` に設定されている場合、XQuery はXPath で無効なものを解決して、これらのを含むXQuery ステートメントがXPath と互換性を持つようにして、XPath が許可されている場所で使用できるようにします。現在 XQuery のみ、しかしXPath では許可されていない、エンティティと文字参照のみに関連しています。*XPath 互換性モード*が`true` に設定されている場合、XQuery エンドリと文字参照はXPath でテキストとして読み込まれ、解決されません。この設定のデフォルトの値は`true` です。
- *HTML ドキュメントのデフォルトの名前空間を無視する*: XML ドキュメント内では1 つの名前空間だけが許可されていて、HTML ドキュメントのデフォルトの名前空間を無視しないことはXML データソースを読み込む際エラーを引き起こす可能性があります。デフォルトは`true` で、HTML のデフォルトの名前空間は無視されます。
- *自動再埋め込み*: 埋め込み はプロジェクト(デザイン) ファイル内のデータソースを埋め込むことを意味します。*自動再埋め込み*が有効化(`true`) されていると、デプロイ時またはシミュレーション時に、ページデータソースは再埋め込みされ、最新のデータソースファイルが埋め込まれ、データが最新であることが保証できます。デフォルトは`true` です。

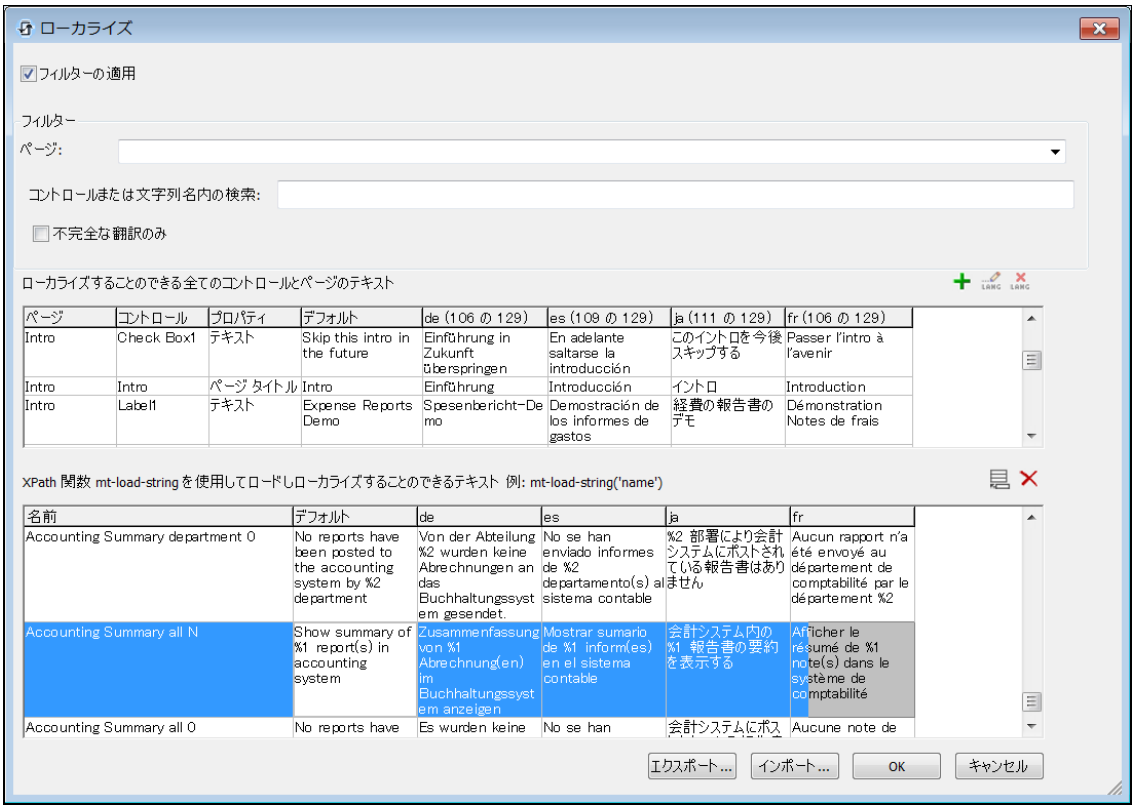
- *トップレベルの余白*: トップレベルのコントロールは、デザイン内に存在するコントロールを指します。すなわち、テーブル内に存在しないコントロールを指します。トップレベルコントロールのためにこのオプション内で設定する余白は、デフォルトのデバイス固有の余白をオーバーライドします。 プロジェクトの各ページのために余白を設

定することはレイアウトをよりよく管理するために重要です。例えば、Android デバイス上で、現在設定されているデフォルトの余白は 9px ですが、プロジェクトページで他の余白を使用する場合、これらのページプロパティを余白に対して使用し、余白を調整します。トップレベルの余白のコントロールプロパティはデフォルトで4辺全てのマージンを指定された値に設定します。上下左右の余白をそれぞれ設定することもできます。マージンの設定が空白の場合、デバイス特有のデフォルトの値が使用されます。(**メモ** Android 上のラベル コントロールは **0px** の余白が設定されています。この設定を変更するためには、トップレベルの余白の設定 (この設定) を変更、または、ラベルコントロールの下の余白を変更します。)

6.5 ローカライズ

ソリューションはデフォルトの言語で作成されます。しかし、ソリューション内で使用されるテキスト文字列は複数の言語にローカライズ(翻訳)されることができます。ソリューションがモバイルデバイスで実行されると、モバイルデバイスと同じ言語にソリューションの言語が自動的に選択されます。ソリューションがモバイルデバイスの言語にローカライズされていない場合、ソリューションのデフォルトの言語が使用されます。(*下のスクリーンショット参照*)。

新しく追加される各言語に列を追加し、この列にローカライズされた文字列を定義することで、ローカライズされた文字列はローカライズダイアログに定義されています(*下のスクリーンショット*)。詳細に関しては、メニューコマンド**[プロジェクト] ローカライズ**を参照してください。更に、名前の付けられたテキスト文字列もローカライズすることができ、その後、XPath 式 **`mt-load-string(NameOfString')`** 内の`mt-load-string` 拡張関数を使用してデザインの任意の場所で参照することができます。



ローカライズされたソリューションは、**[プロジェクト] シミュレーションする言語**コマンドを利用してシミュレーションする言語を選択して、シミュレーションを実行しテストすることができます。

6.6 名前空間

名前空間はノードを正確に認識するため、また、式を使用してノードの位置を正確に決めるために重要です。[ページ ソース] ペイン(*下のスクリーンショット*) の*名前空間* アイテムには、ページ デザイン ビューで現在アクティブなページに関わりなく、プロジェクトで宣言されたすべての名前空間が含まれています。

名前空間を宣言するには次の2 通りの方法があります:

- *データのインポート時に自動的に宣言する* 外部 XML ファイルがページ ソースとして追加された場合、ソースの名前空間は自動的にデザインにインポートされ、プロジェクト全体のスコープのために宣言されます。[ページ ソース] ペイン(*上のスクリーンショット参照*) の名前空間アイテムの下に表示されます。デザイン内にあいまいさを引き起こさない場合、名前空間プレフィックスは、自動的にオリジナルのプレフィックスに一致するように設定されています。名前空間の宣言に割り当てられたプレフィックスは、ノード名に使用されており、ページソース内のノードの場所を決定するための XPath 式に使用される必要があります。
- *ユーザーによる定義*: [ページ ソース] ペイン(*上のスクリーンショット*) のツールバーの**[名前空間]** アイコンをクリックして、名前空間を追加することができます。自身の名前空間をプロジェクトに追加することにより、1 人以上のユーザーにより宣言されるノードを作成することができます。同じローカル名を持つノードのあいまいさをなくすためにこれは有用です。

名前空間を削除するには、ペインのツールバーの**[削除]**を選択します。

メモ 名前空間のプレフィックスは、デザインのプロセス中いつでも、[ページ ソース] ペイン内の名前空間プレフィックスをダブルクリックして編集することにより、名前変更することができます。デザインの XPath 式内の古いプレフィックスへのすべての参照は、新しいプレフィックスに変更されます。

メモ XPath のデフォルトの名前空間 (`xpath-default-ns=''`) は、拡張関数やユーザー定義関数を含むすべての XPath/XQuery 関数に使用されています。

6.7 グローバル リソース

MobileTogether のグローバルリソースにより、異なるデータベースサーバ名、ファイルの場所、または他の構成特有のパラメーター構成を、簡単に開発から生産システムに切り替えることができます。そして、各モバイルソリューション別に設定を行うことができます。グローバルリソースにより、デザインやテストを素早く行うことができ、結果時間を節約することができます。

MobileTogether Designer でモバイル ソリューションを開発するにあたり、グローバル リソースダイアログは各ファイル、データベース接続、またはユーザーに定義された名前付きディレクトリを識別し、リソースをソリューションで参照するために使用します。グローバルリソースにより、異なる構成をセットアップし、モバイル ソリューション モードを異なるグローバル リソースを選択することで簡単に切り替えることができます。

MobileTogether Server にモバイル ソリューションをデプロイすると、この構成をソリューションと共にアップロードすることにより、デプロイ後にも、ソリューションが特定のサーバーにアクセスすることをコントロールできます。管理者はモバイル ソリューションのリソース構成を素早く行うことができ、モバイル ソリューションをテストモードから生産環境にクリック1 つで切り替えることができます。

グローバル リソースに関する詳細は Altova グローバル リソースのセクションを参照してください。

6.8 パフォーマンス

MobileTogether Server とMobileTogether Client アプリが共にどのように作動するか、MobileTogether Server を使用して、データ集中処理を行い多量のデータを保存するか、どのようにMobileTogether Designer のプロジェクト デザインでパフォーマンスの最適化のための設定をするかを理解することによりMobileTogether ソリューションのパフォーマンスを最適化することができます。

- デザイン ファイルにXML を埋め込む
- XQuery 3.1 を使用したデータ クエリ
- サーバー上のデータ ストレージ
- クライアント上の永続的データ ストレージ

デザイン ファイルに XML を埋め込む

ソリューションが外部 XML データソースを参照する代わりに、XML データはデザイン ファイルに直接埋め込まれることができます。このオプションは、コンボ ボックスの選択のリストまたは静的データなどの、クライアント側が必要とする小さいデータセットに最適です。このデータは、クライアントにデザイン ファイル全体の一部として送信され、アプリを実行するたびに、クライアント側で即時使用することができます。ですから、クライアントとサーバー間の追加データの送信は必要ありません。

デザインファイル内にデータソースを埋め込むには、データソースを右クリックして **[デザインファイルに XML を埋め込む]** を選択します。

XQuery 3.1 を使用したデータ クエリ

パワフルな XQuery 3.1 言語を使用し式を書くことにより、サーバーとクライアント間で送信されるデータの量を大幅に減らすことができます。データベース ビュー、クエリまたは外部データ ソースへの Web サービス コールにより、モバイルアプリケーションのための生データを取得することができます。これはよく重複またはモバイルアプリケーション内で想定されたデータ表示ではない理想的ではないデータ構成が含まれている可能性があります。XQuery の FLWOR 式によりデータを簡単に再構成、再グループ化することにより、サーバーからクライアントへの最も効率のいいデータ送信およびクライアント アプリケーションで最も役に立つプレゼンテーションを保証します。

この新しいバージョンの XQuery には、JSON フォーマットのマップ、配列およびデータへのサポートなどを含む、複数の新機能が搭載されています。

[XPath/XQuery 式の編集] ダイアログを使用して XQuery 式を作成および確認することができます。

サーバー上のデータ ストレージ

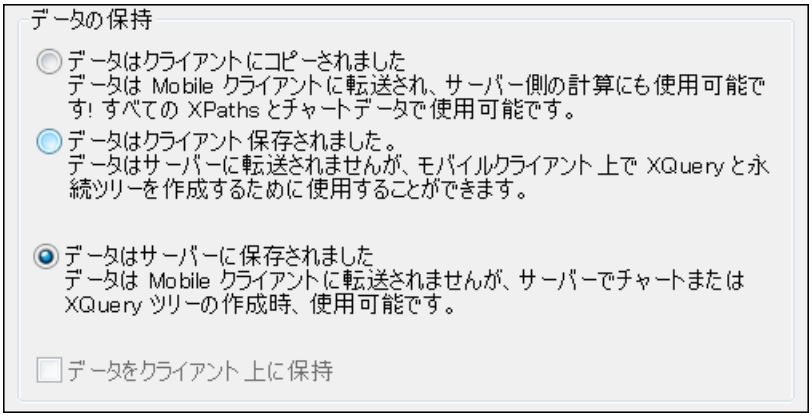
ソリューションのデータが処理されるスピードは、特定の種類のデータをサーバー上に保管することで強化することができます:

- イメージであるチャートやグラフを生成するために使用されるデータはクライアントに送信される必要はなく、イメージのみが送信される必要があります。ですから、チャートやグラフを生成するために使用されたデータは、サーバに保管することができ、送信される必要はありません。

パワフルな [データをサーバー上に保存] の設定

モバイル ソリューションのパフォーマンスを改善する、モバイル データ ネットワーク上でのデータの送信量を減らすために、MobileTogether では、どのデータをクライアント デバイスに送信し、どのデータをサーバーに保持するかを選択することができます。例えば、特定のデータセットはグラフの表示のみに必要な場合、そのデータはサーバー上に保管することができます。グラフのイメージはサーバーでレンダーされ、基になるデータをモバイル ネットワークで送信することなくクライアントに送信されます。大きなデータセットに関しては、大幅なパフォーマンスの向上につながります。

(クライアントにデータを送信せず、サーバー上にデータを保管する) この設定は、データ ソースを追加する際に、データ ソースのために行われます。ページ ソースの追加ダイアログのデータの保持ペイン (下のスクリーンショット) で設定することができます。また、[ページ ソース] ペインのルートノードのコンテキストメニューコマンドでも使用できます。



キャッシュ

MobileTogether Designer とMobileTogether Server の設定を使用して、すべてのデータ ソースのキャッシュの振る舞いを指定することができます。これは、サーバーが既に使用できるデータを持つモバイル アプリからリクエストを受けるため、MobileTogether のスピードを大幅に向上します。キャッシュの作成には2 つの主な理由があります: (i) ページ データソースがレポートをゆっくり作成する場合 (例えば大きなデータベースの場合) および(ii) データ ソースが頻繁に変更されない場合。上記の場合、データがサーバからではなくデータ キャッシュより取得された方が、ソリューションの実行は速くなります。キャッシュを最新状態に保つため、キャッシュが作成される際にキャッシュの更新頻度を設定することができます。MobileTogether Designer でキャッシュが定義されると、異なるデザインのデータ ソースにより使用されることができ、基になるデータ構成に互換性を持たすことができます。

Altova MapForce とFlowForce Server で最初に開発されたキャッシュ機能同様、MobileTogether は、有効期限や更新頻度などの通常のキャッシュ パラメーター以上の機能を搭載しています。手動で次のキャッシュを設定することができ、また、(データベースおよびWeb サービスのための) 複数のクエリパラメーターの一意の組み合わせが自動的にキャッシュされるかなどを定義することができます。データをリクエストしているクライアントはすぐにキャッシュからデータを取得すること

ができますが、サーバーからの場合、キャッシュが経過してからのみデータを取得することになります。MobileTogether は、デザイナーが指定する間隔通りに自動的にクエリを実行し、これは単純なキャッシュ機能を超えるものです。パラメーター付きクエリの場合、デザイナーはキャッシュされるパラメーターの一意の組み合わせを指定することがで、サーバーは、指示に従います。

MobileTogether Designer で、データソースのための新しいキャッシュを定義することができます。[ページソース] ペインのデータソースを右クリックして、**[キャッシュ設定]** を選択し、キャッシュのプロパティを指定します。

データソースがキャッシュを持つと定義されている場合、キャッシュされたデータがソリューションを実行する際に使用されます。ソリューションがサーバにデプロイされると、キャッシュを使用することができます。

クライアント側の永続データ ストレージ

ユーザーにより入力されたデータや頻繁に変更されないデータに関しては、各デバイスにこれらのデータを永続的に保存することを選択できます。これにより、サーバーとクライアント間で送信されるデータの量を減らし、パフォーマンス スピードを向上することができます。また、サーバーとクライアント間の往復時間を (同じユーザーの数時間間隔のあいた異なるセッション間の時間でさえも) 減らすことにより、パフォーマンスを強化します。これは、クライアントデバイスにデータが永続的に保存されているからです。永続データは以下の方法で定義することができます:

- *デフォルトの永続ツリー*: デフォルトでは、$PERSISTENT ツリーがデザイン中のすべてのページのために定義されています。すべての $PERSISTENT ツリー データはクライアント上に保管されています。データは静的または動的であることができます。ツリー内のノードがエンドユーザー入力を受け入れるコントロールに関連付けられている場合、ツリーのノード内のデータは、エンドユーザーにより編集されることができます。
- *永続にすることのできるツリー*: [*ページソース*] *ペイン*内の永続ではないツリーのルートノードを右クリックします。表示されるコンテキストメニュー内から**[クライアント上の永続データ]** を選択します。そのツリーは永続に設定されます。ツリー内のデータは、クライアントに保管され、ソリューションが開始するとロードされます。
- *オンデマンドのサーバーアクセス*: この設定は、[*スタイル & プロパティ*] *ペイン*で定義することができます。クライアントデバイスおよびサーバー間の接続を指定し、必要に応じて行うことができます。これは、ソリューションが、クライアント上で永続データを使用すること、またはデータがソリューションに埋め込まれることを意味します。サーバーへの接続は、デザインに必要とされる詳細に合わせて行われます。例えば、デザインがデータをサーバー上のデータベースに保存すると指定することがあります。このアプローチは、データベースとの作業のパフォーマンスを向上するために役に立ちます。

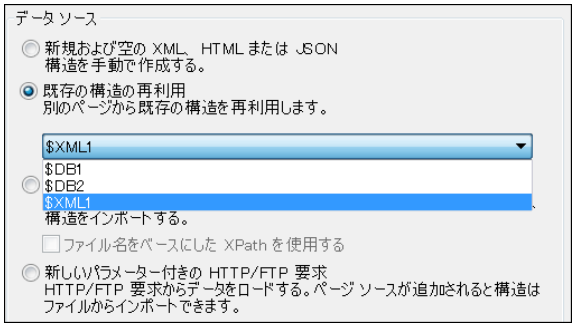
チャプター 7

データソース

7 データソース

各ページはページデータソース(またはページソース) の組み合わせを持つことができます。これらのソース(*下部左側スクリーンショット参照*) はページのコントロールで使用することができ、そのページのみで使用することができます。しかし、複数ページ間で単一のページ ソースは再利用することができます。

- [ページ ソース] ペインのソースの追加コマンドで新しいページ ソースを追加することができます。新しいソースの追加方法はデータソースの追加のセクションに説明されています。
- ページは他のページのページ ソースを再利用することができます。再利用するページソースの指定のオプションは新しいデータソースが追加された場合です(*下部右側スクリーンショット*)。



以下の点に注意してください:

- **ページソース**は[ページ ソース] ペインに一度に1 つずつ追加します(*下のスクリーンショット*) 。
- XML、HTML、JSON、XQuery、またはDB データソースであることのできる**データソース**からページソースを追加します。データソースは、HTTP、FTP、REST、SOAP またはプロジェクトの他のページソースからアクセスすることができます。詳細に関してはデータソースの種類 のセクションを参照してください。
- 各ページは、複数の異なる種類のデータソースをベースにするページソースを持つことができます。例えば、上のスクリーンショットでは、ページはXML ベースのソース1 つDB をベースにするページソース2 つ持っています。
- ページソースは編集可能または編集不可能(読み取り専用)である場合があります。プロパティはページソースが追加された時に指定され、ルート ノードのコンテキスト メニューの切り替えコマンド**[読み取り専用]** で切り替えることができます。

このセクションの構成

このセクションの構成は以下の通りです:

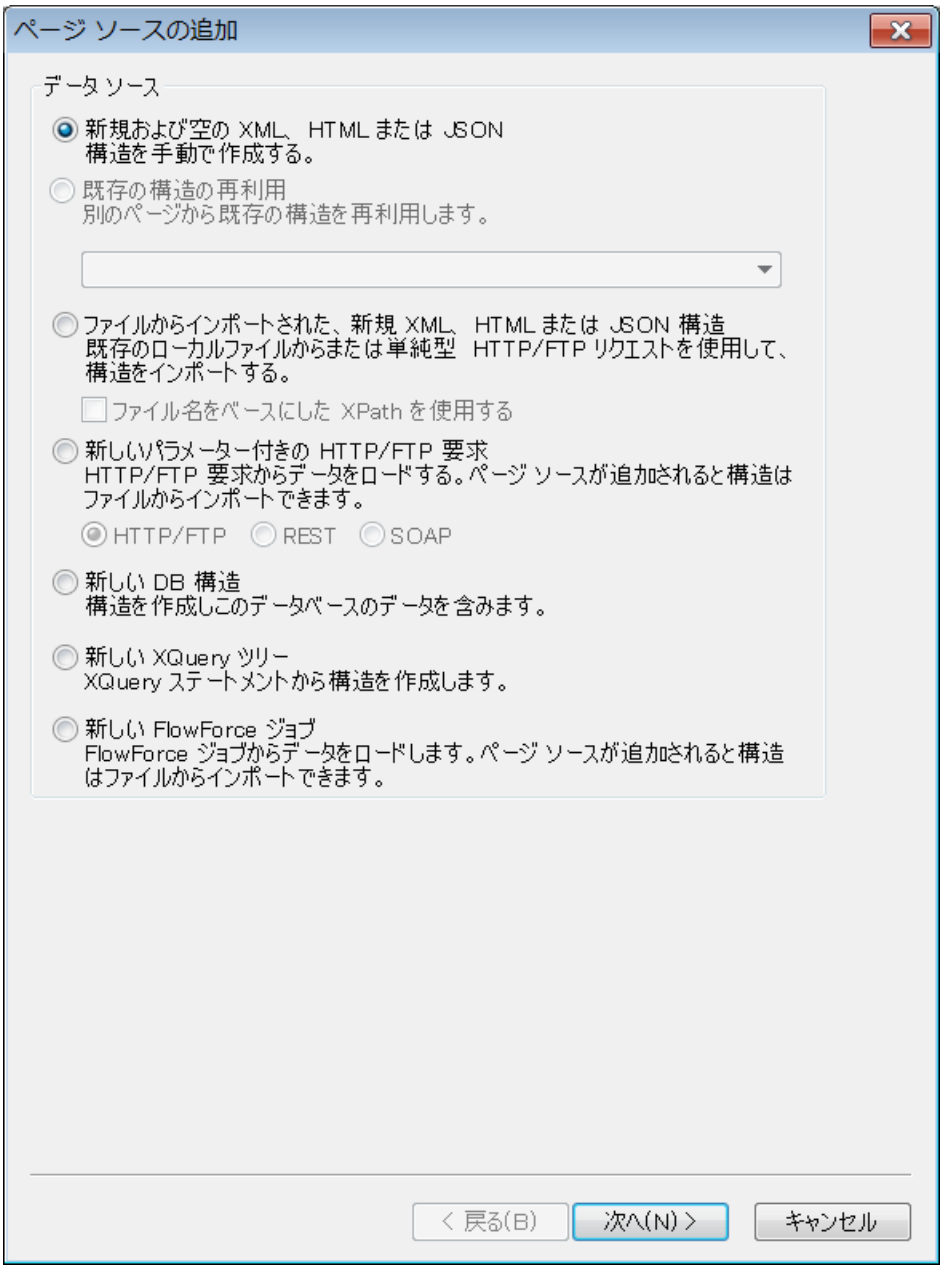
- ページデータソースの追加: ページ ソースの追加方法の説明をします

- データソースの種類: ページに追加することのできる多種のデータソースをリストしその説明をします。
- ページソースオプション: ページソースのプロパティについて説明します。
- HTTP/FTP、REST および SOAP リクエスト: これらのリクエストの作成方法を説明します。
- ルート ノード: ルートノードとユーと要素の概念について説明し、これらがどの様にデータソースのツリーに使用されるか説明します。このセクションは、ルートノードの名前に関する規則についても説明します。
- ページ ソース ツリー: ソースツリーの構成とページ内でのツリー ノードのデータについて説明します。
- プロジェクト内の名前空間: プロジェクト内での名前空間の重要性と名前空間の使用方法について説明します。
- キャッシュ: サーバーワーク上でのデータのキャッシュの仕組みについて説明します。
- コンテキスト メニュー: [ページ ソース] ペインのコンテキスト メニュー コマンドについて説明します。

7.1 ページデータソースの追加

ページにデータソースを追加するには、[ページ] ペインでページを選択して、次の手順を踏みます。

[ページソース] ペインのツールバーの**[ソースの追加]** アイコンをクリックし、ページソースの追加ダイアログを表示します(下のスクリーンショット)。追加するデータソースの種類を選択して、(以下リストされる) このデータソースのプロパティを指定し、**[次へ]** をクリックします。異なる種類のデータソースに関しては、データソースの種類の次セクションで説明されています。



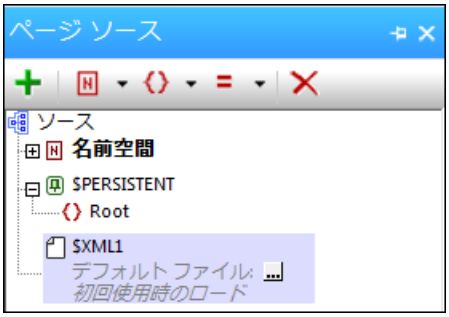
7.2 データソースの種類

ページソースとして追加 することのできるデータソースの種類は下にリストされています:

- 新しい空の XML、HTML または JSON

  新しい空のXML、HTML またはJSON ページソースが[ページソースの追加] ダイアログに追加される場合、XML ページソースとして作成されます。[ページソースの追加] ダイアログの次のスクリーン内で、データソースがXML、HTML またはJSON ファイルであるかを指定します。**$XML** という名前を持つルートノードが新しいページソースのために作成されます (*下のスクリーンショット参照*)。データソースがXML またはHTML として指定されている場合、ツリー内に他のノードまたは名前空間は作成されません。データソースがJSON として指定されている場合、 **json** と呼ばれるルート要素がルートノードの下に作成されます。以下の方法のどちらかまたは両方を使用してルートノードの下にXML ドキュメント構造を作成することができます:

  - (ツールバーコマンドを使用して) ルートノードに要素および属性ノードを追加する。詳細は、ツリー構造のセクションを参照してください。
  - XML、HTML、またはJSON ファイルから構造を、ルート ノードのコンテキスト メニューの [インポート] コマンドを使用してインポートします。

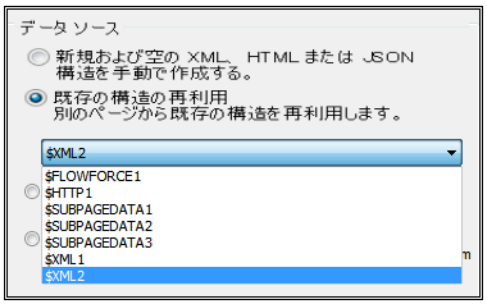


(デフォルト ファイルの割り当てを含む) ツリーへデータを追加する方法は、ツリー データのセクションに説明されています。

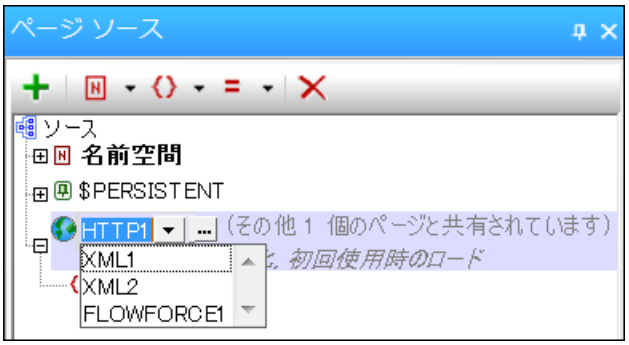
メモ　データソースのデータ型をルートノードのコンテキストメニューコマンド**データの型**を使用して変更することができます。

- 既存の構造を再利用する

  このオプションは、プロジェクトの他のページで既に作成された構造を再利用したい場合役に立ちます。このオプションは、プロジェクトの他のページで既に作成された構造を再利用したい場合役に立ちます。再利用可能なページ ソースは、プロジェクトの同じページではなく *他のページに存在しなくてはならず*、ページはトップ ページまたはサブ ページであることができます。ページソースがプロジェクトの他のページに存在する場合のみ*再利用オプション*を有効化することができます。



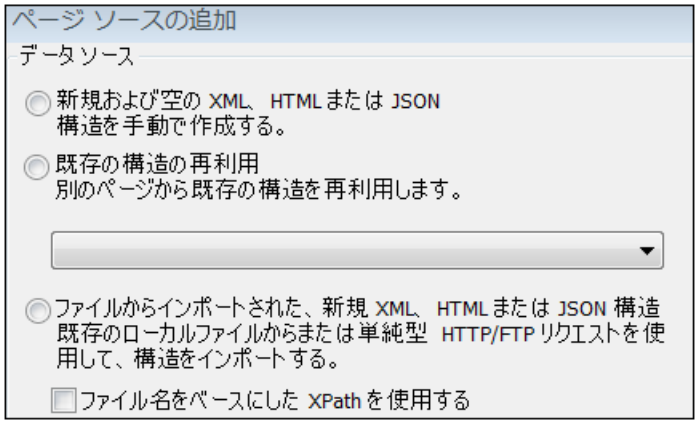
使用可能なページソースは、オプションのコンボボックスのドロップダウンリストのルートノードの名前の横にリストされます(*上のスクリーンショット参照*)。再利用したいページソースを選択して、**[OK]** をクリックします。同じ名前のルートノードと再利用されたページソースの構造が作成されます(*下のスクリーンショット参照*)。データソースが共有されているページ数がリストされます(*下のスクリーンショット参照*)。マウスをツリーのノードの名前にポイントするとページを共有する名前がリストされます。ルートノードの名前の横のコンボボックスから他の再利用可能なページソースを選択することにより、他のページソースのデータ構造に変更することができます(*下のスクリーンショット参照*)。



(デフォルトファイルの割り当ても含む)ツリーにデータを追加する方法は、ツリー構造のセクションで説明されています。ツリーの変更方法は、ツリー構造のセクションに説明されています。

- ファイルからインポートされた新規の XML、HTML または JSON/JSON-5 構造

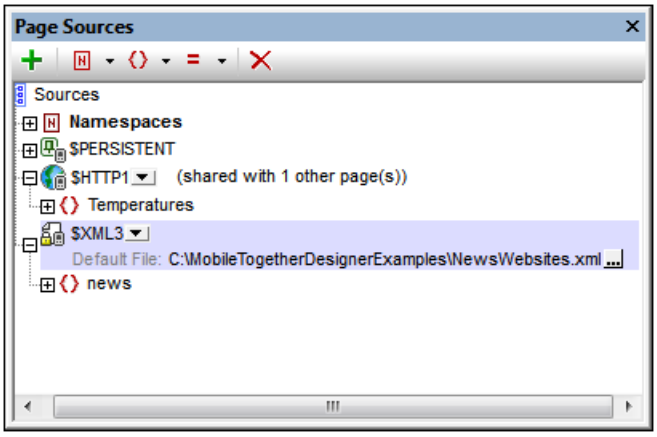
  このオプションが選択された場合、(*下のスクリーンショットを参照してください*)、**[次へ]** をクリックすると選択されたデータソースの使用方法のオプションを設定することのできる[ページソースの追加] ダイアログが表示されます。



[ページソースの追加] ダイアログの次のスクリーンでデータソースがXML、HTML またはJSON ファイルであるかを指定します。ダイアログの**[完了]** ボタンをクリックすると、その後の結果は、*XPath-ベースのファイル名を使用* オプションが選択されているか否かにより異なります (*上のスクリーンショット参照*) 。

- チェックボックスがチェックされている場合、[XPath/XQuery 式の編集] ダイアログが表示されます。ここで必要とするファイルURL を生成するためのXPath 式を構築します。
- チェックボックスがチェックされていない場合、XML、HTML またはJSON ファイルを選択し、ページソースの構造を与えるダイアログが表示されます。ファイルを参照、ファイルURL を使用、またはグローバルリソースを使用することができます。

XML/HTML/JSON ファイルの構造は、ページソースの構造としてインポートされます(*下のスクリーンショット参照*)。XML およびJSON ファイル構造は、**$XML** ルートノードの下に表示されています。HTML ファイル構造は、**$HTML** ルートノードの下に表示されています。HTML またはJSON データソースの構造は、XML ツリー構造としてインポートされます。インポートされたJSON 構造は、**json** という名前のルート要素を持ちます。XML/HTML/JSON (データソース) ファイルは、自動的にページソースのデフォルトのファイルとして定義されます。これは、ファイルからのデータが新しいページソースのノードのデータとして使用されていることを意味します。ファイルがURL と共に選択されている場合、ファイルを取得するために、HTTP またはFTP プロトコールを使用することができます。デフォルトファイルのパスをXPath 式と共に指定することができます。これによりファイルパスの動的構成を許可することができます。例えば、他のページソース内のノードコンテンツをベースにしたパスの場合などがあげられます。



ファイルURL を変更するには、URL エントリをダブルクリックするか、エントリの右にある**[追加ダイアログ]** アイコンをクリックします。他のプロジェクトページからの再利用可能な構造がある場合、ルート ノードの名前の横のコンボ ボックス

により、再利用可能な構造を選択することができます (*上のスクリーンショット参照*)。ツリー構造の変更方法はツリー構造のセクションで説明されています。

HTML の取得は修正パーサを使用して行います。この結果、インポートされたHTML 構造が、不足している要素のため、無効なデータオブジェクトモデルを持つ場合、(HTML 5 仕様に従い)、これらの不足している要素は、[ページソース] ペインのデータソースツリーに追加されます。例:

```
<table>
   <tr/>
   <tr/>
</table>
```

以下に修正されます:

```
<table>
   <tbody>
      <tr/>
      <tr/>
   </tbody>
</table>
```

新しいパラメーターつきの HTTP/ FTP リクエスト

データを含むデータソースは、HTTP またはFTP リクエストとその後追加された構造によりにより追加することができます。 HTTP/FTP、REST またはSOAP を使用してデータソースが取得されるか指定するオプションを選択します (*下のスクリーンショット参照*)。HTTP/FTP またはREST を選択した場合、[ページソースの追加] ダイアログの次のスクリーン内で、データソースがXML、HTML またはJSON ファイルであるかを指定します。後に、対応する設定ダイアログ(HTTP/FTP またはREST)内で選択を変更することもできます。(SOAP を選択した場合、データソースは、XML として解析される必要があります。このオプションは自動的に設定され変更することはできません)。

[ページソースの追加] ダイアログの**[完了]** ボタンをクリックすると、(HTTP/FTP リクエストのための) [Web アクセス設定の編集] ダイアログ、(REST リクエストのための) [RESTful API リクエスト] ダイアログ、または(SOAP リクエストのための) [WSDL ファイル選択] ダイアログが表示されます。これらのリクエストの設定の指定に関する詳細は、HTTP/FTP、REST、およびSOAP リクエストのセクションを参照してください。

リクエストの実行に成功すると、ページソースが(ルートノード として)追加され、データソースからデータがロードされます。ツリー構造は、しかしながら作成されません。ツリー構造はインポートまたは手動で作成することができます。ツリー構造の作成方法はツリー構造のセクションを参照してください。

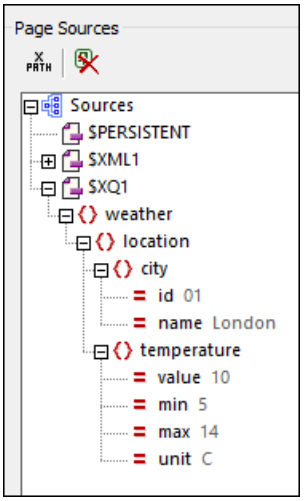
新しい DB 構造

このオプションにより、構造を作成しデータベースからデータを追加することができます。このオプションを選択し、[OK] をクリックすると、データベース(DB) 接続ウィザードが表示されます。DB に接続した後、データソース構造とデータコンテンツのためのインポートするテーブル データを選択することができます。ルート ノード名の横のデータベース名にノードが挿入されます。ルート ノードの下にはDB 構造があります。詳細はデータベースを参照してください。構造があります。DB の例はデータベースとチャートのチュートリアルを参照してください。

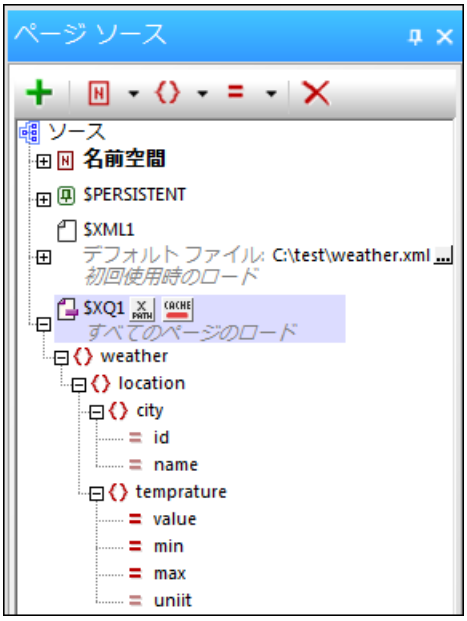
- 新しい XQuery ツリー

このオプションを選択し、**[OK]** をクリックすると、[XPath/XQuery 条件式の編集] ダイアログが開かれます。必要なデータを検索するXQuery ステートメントを入力して、**[OK]** をクリックします。XQuery ステートメントにより取得されたデータの構造を持つデータソースが作成されます。XQuery ステートメント内で指定された構造を持つ **$XQ** という名前のルートノードを持つページデータソースが作成されます。このルートノードを右クリックして、**データのロード**コマンドを選択し、必要に応じて、オプション「最初の使用時」または「全てのページ」をあらかじめ設定します。例えば、次のXQuery ステートメントは、シミュレーションのスクリーンショット内で表示されているツリーを表示しています。

```
element weather {
element location{
element city {attribute id{"01"}, attribute name{"London"}},
element temperature {attribute value{"10"}, attribute min{"5"}, attribute max{"14"}, attribute unit{"C"}}
}
}
```



$XQ ツリーからノードをデザイン内で使用するには、XPath 式を使用してノードを見つけることができます。(例は以下のとおりです: `$XQ1/weather/location/city/@name`)。または、ページソースペイン内で、XQuery ステートメントにより作成されるツリーの構造に一致する一時的なツリーを作成することができます(下のスクリーンショットを参照してください)。そして、ツリーからノードをデザインにドラッグすることができます。ツリーへの**データのロード**と作成は、XQuery データソースのデータのロードオプション(最初の使用時、全てのページ、手動)により異なることに注意してください。

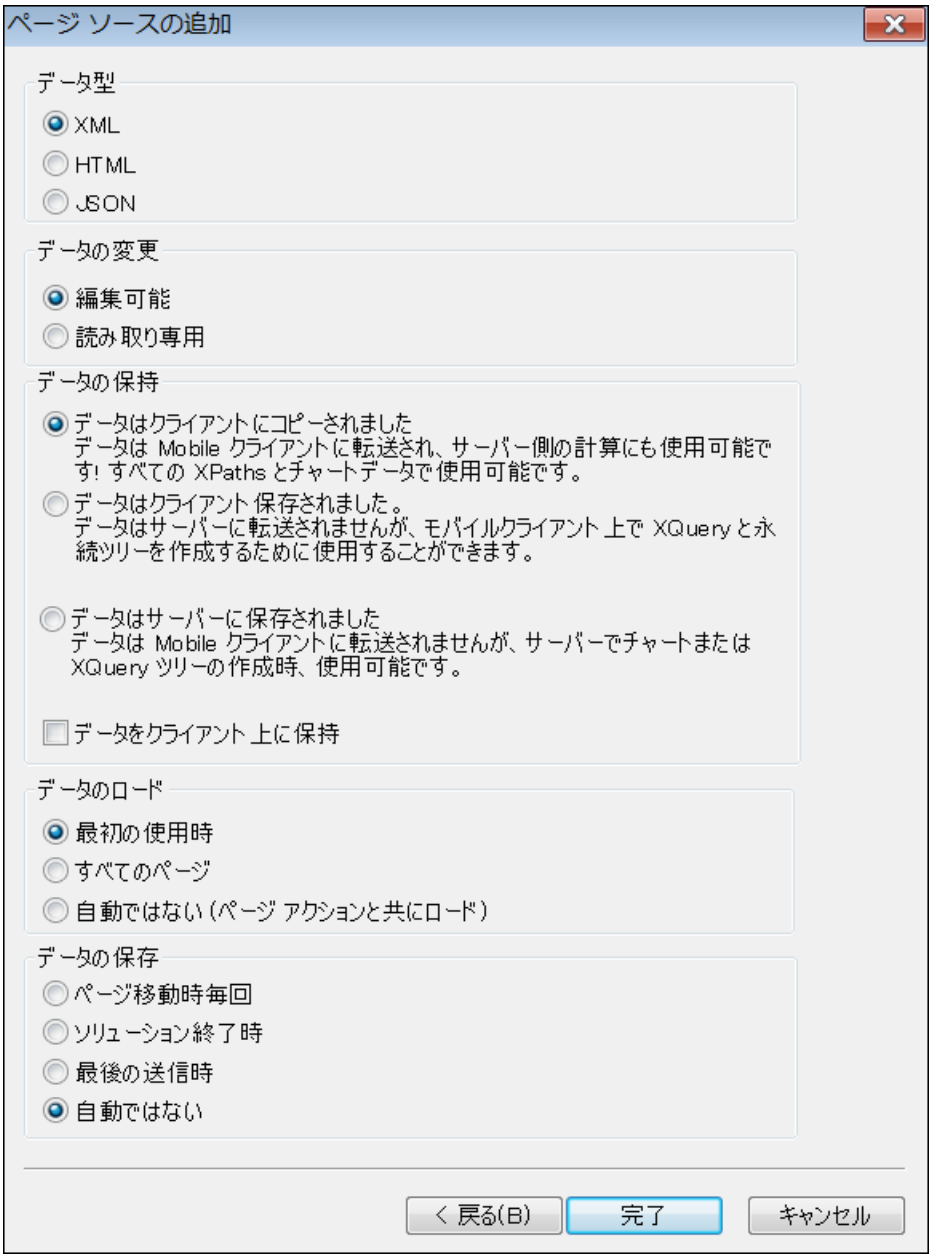


- 新しい FlowForce ジョブ

  このオプションを選択し、**[OK]** をクリックすると、FlowForce 設定の編集ダイアログが開かれます。FlowForce サーバーに接続するため FlowForce の設定を入力して、ジョブを指定します。**[OK]** をクリックすると、FlowForce ジョブを実行して取得されたデータを含む新しいデータソースが作成されます。データソースの構造は後で変更することができます。ツリー構造の変更方法はツリー構造のセクションで説明されています。Altova の FlowForce 製品に関しての詳細は Altova Web サイトの FlowForce のページを参照してください。

7.3 ページソースオプション

追加するデータソースの型を選択すると、**[次へ]** をクリックします。[ページソースの追加] ダイアログは、選択されたデータソースの使用オプションを表示します (*下のスクリーンショット参照*)。使用するオプションを選択して、**[完了]** をクリックします。



オプションの一部は一部のページソースの型に使用できない可能性があることに注意してください。使用することのできるすべてのオプションは以下にリストされています。

データ型
追加されるデータソース内のデータのフォーマット。*XML*、*HTML* または *JSON* から選択します。ページソースが追加されると、ページソースノード (コンテキストコマンド **データ型**) のコンテキストメニュー内のデータ型を変更することができます。ページソースのルートノードはすべて **`$XML`** です。ですが、が選択されている場合、ルート要素は **`json`** という名前が付けられます。ですが、そのページソースは、JSON と読み込まれ、(XML ではなく) JSON として保存されます。これは**データの保**

存 プロジェクトプロパティを使用して保存アクション保存に適用される点に注意してください。

データの変更
編集可能な XML または*読み取り専用 XML* を選択します。データソースが編集可能に作成されると、ツリー ノード内のデータは変更することができます。読み取り専用のデータソースは変更することができません。両方のタイプのデータはデータの表示に使用することができます。しかし、ユーザーがデータ ノードを書き込むには、データソースを編集可能な XML に作成する必要があります。

データの保持
以下の場合データを指定します: (i) サーバーからクライアントのコピーされる場合、(ii) クライアントに保持される場合、または (iii) サーバー上に保持される場合。この選択を行う場合2つの問題点が挙げられます : (i) クライアントとサーバーでそれぞれ計算が行うことが可能、(ii) プロセスをスピードアップするために、どこにデータを保管するか。

計算に一部は、クライアント側でのみ行われ、(例えば、SMS を送信するために XPath 式を解決する場合)。また、一部の計算はサーバー側でのみ行われます (例えば、チャートの作成、最後のチャートのイメージのみがクライアントに送信されます)。また、計算の一部では、計算はサーバ側とクライアント側両方で行われます (例えば、XML ツリーノードの更新の場合)。すべての計算は最初にクライアントで試みられます。クライアントで計算が可能でない場合、計算はサーバーにパスされます。ですから、処理時間を削減するために、データに最も速くアクセスできる場所にデータを保管することが奨励されます。すべての計算がクライアントで行われる場合、データをクライアント上に保管します。その他の場合は、他の2つのオプションを選択してください。

*クライアント上の永続データ*オプションは、ソリューションが最後に閉じられたときのクライアントのデータを持つソリューションをロードします。クライアントのデータは2つのソリューションが実行される間、永続中と考えられます。

データのロード
ページが初回ロードされる時、ページがロードされる都度、またはページ アクションに指定された場合、データがロードされるか選択します。後で、ソース ツリーのルート ノードの選択オプションはコンテキスト メニューから変更することができます。

データの保存
データソースが (XML、HTML、または DB などの) 外部ファイルである場合、このオプションを有効化することができます。データを以下の場合に保存することを選択します (i) ページ移動時、(ii) ソシューション終了時、(iii) ユーザーが最後の送信オプションをクリックした時、(iv) ページまたはコントロールアクションで定義されている時。ページ ソース アクションを参照してください。

7.4 HTTP/ FTP、REST および SOAP リクエスト

このセクションは、HTTP/FTP、REST、および SOAP リクエストに必要な設定について説明します。これらのリクエストは、外部ソースからデータをロードまたは外部ソースにデータを保存するためのリクエストです。リクエストは、以下のシチュエーションで作成されます:

- ページソースを追加する場合: この場合、リクエストは通常外部ソースからデータをロードするために作成されます。
- ページソースに関連したアクションを定義する場合: ページイベントおよびコントロールイベントのためにアクションを指定することができます。これらのアクション内のリクエストは、外部データソースからロードまたは外部データソースに保存するリクエスト内で使用することができます。

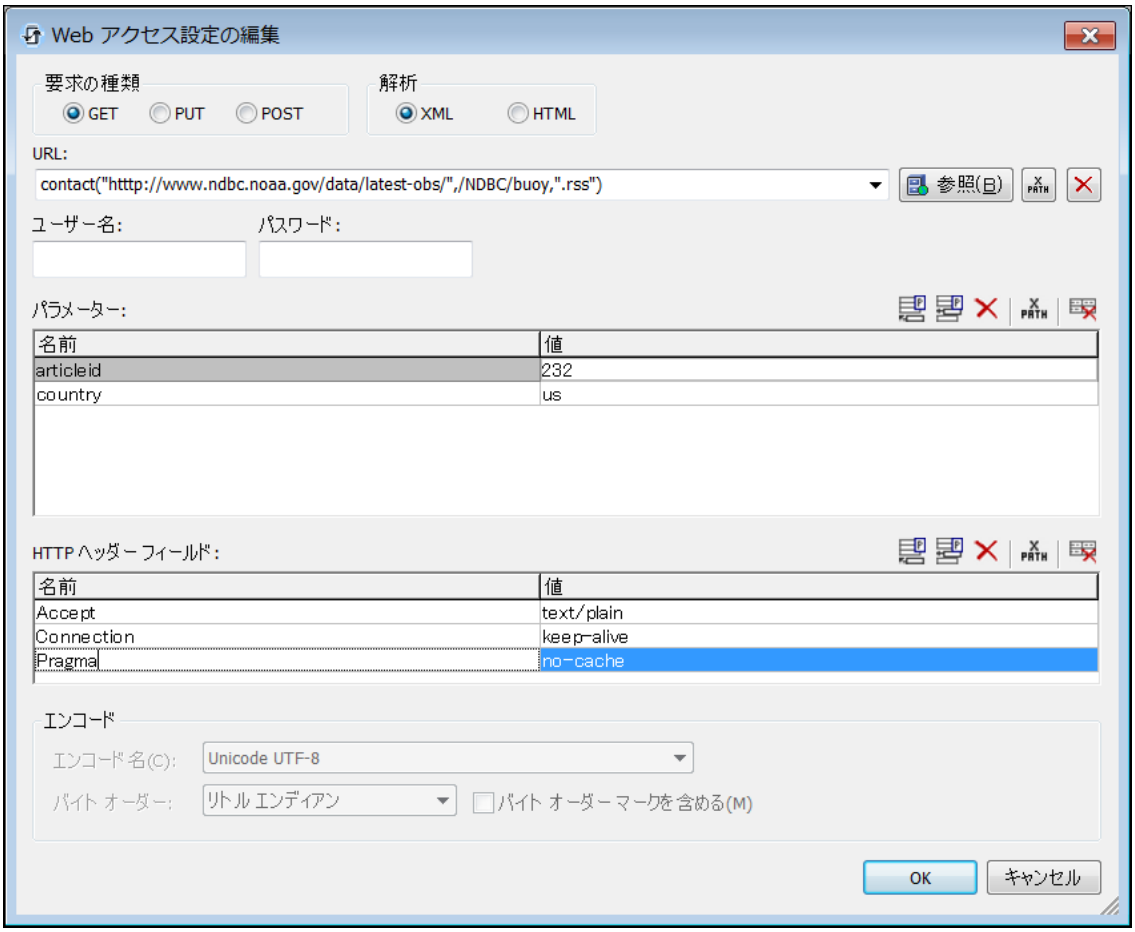
このセクションは以下の各ダイアログについて説明しています:

- HTTP/FTP リクエスト設定
- REST リクエスト設定
- SOAP リクエスト設定

ページソース構造の作成
これらの設定ダイアログで定義されるリクエストは、デザイン内で保存され、ランタイムで実行されます。ページソースは作成されますが、ツリー構造を含みません。構造を作成するためには、XML ファイルから構造をインポートするか、構造を手動で作成する必要があります。例えば、SOAP 応答を XML ファイルとして保存して、ページソースツリー構造を生成するために XML ファイルをインポートすることができます。詳細は、ページソースツリー のセクションを参照してください。

HTTP/ FTP リクエスト設定

HTTP/FTP リクエストのための設定は、[Web アクセス設定の編集] ダイアログ内で定義されます (*下のスクリーンショット*)。リクエストの種類、ターゲットリソースの URL 、ターゲットリソースのデータフォーマット(XML、HTML、または JSON)、ユーザー認証情報、オプションでクエリパラメーターとヘッダーを入力します。下のスクリーンショットでは、例えば、`GET` リクエストは XPath 式を使用して構成されています: `http://www.ndbc.noaa.gov` Web サイトの `.rss` ページをターゲットとしています。RSS ページの名前は、`/NDBC/buoy` ノードから取得されており、ターゲットページは、XML として解析されます。Query パラメーターとヘッダーはリクエストに追加することができます。`charset` ヘッダーは、しかしながら、MobileTogether Designer により自動的に生成され、このダイアログで入力する`charset` ヘッダーにより上書きはされません。



[OK] をクリックすると、リクエストが実行されます。

HTML の取得は修正パーサを使用して行います。この結果、インポートされた HTML 構造が、不足している要素のため、無効なデータオブジェクトモデルを持つ場合、(HTML 5 仕様に従い)、これらの不足している要素は、[ページソース] ペインのデータソースツリーに追加されます。例:

```
<table>
   <tr/>
   <tr/>
</table>
```

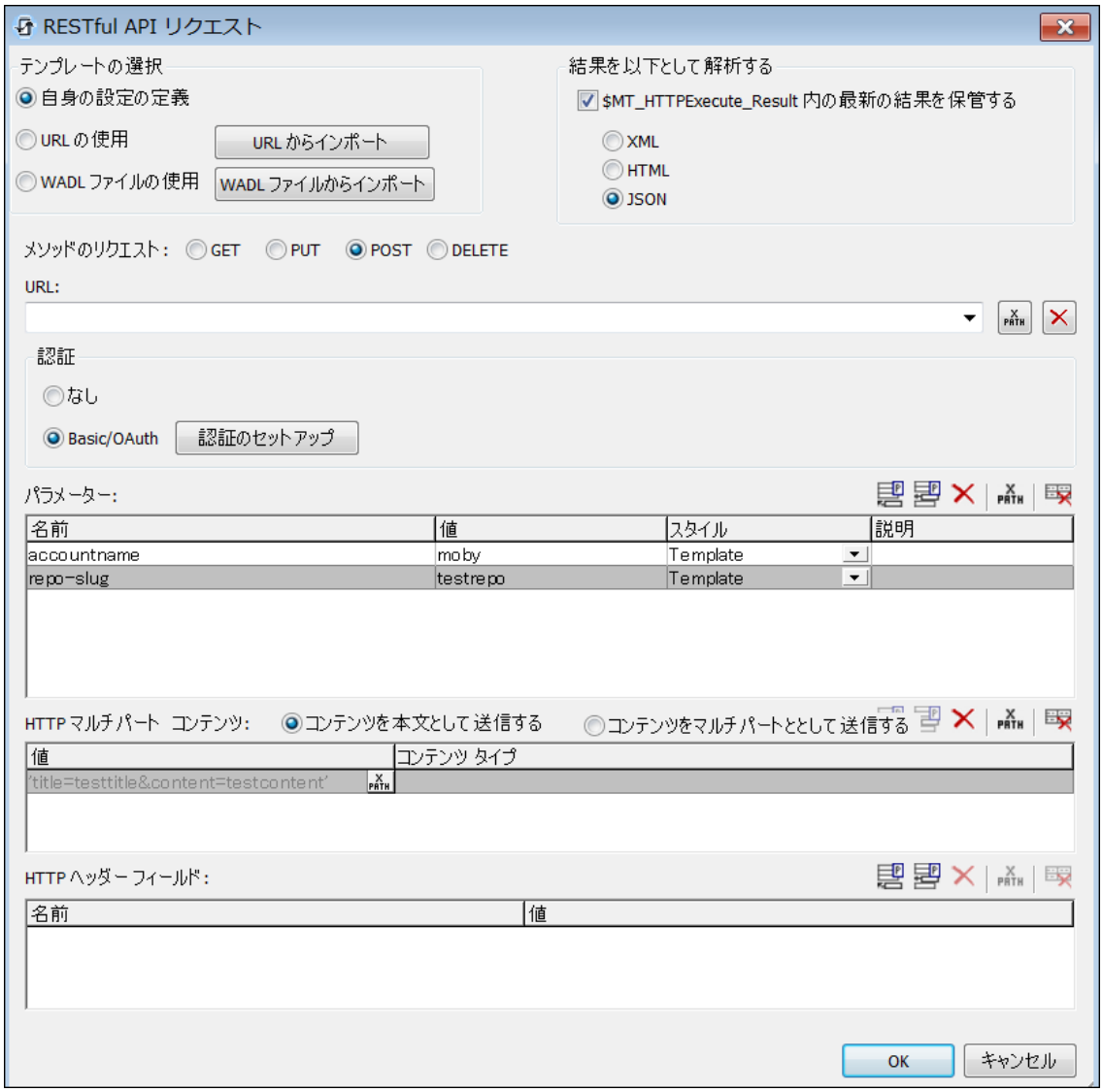
以下に修正されます:

```
<table>
```

```
      <tbody>
         <tr/>
         <tr/>
      </tbody>
   </table>
```

REST リクエスト設定

REST リクエストのための設定は、[RESTful API リクエスト] ダイアログ (*下のスクリーンショット*) 内で定義されます。以下を選択することができます (i) 自身の設定の定義、(ii) URL のインポート、または (iii) WADL ファイルの使用。独自の設定を選択する場合、個別の設定のために独自の定義を指定することができます。URL をインポートまたは WADL ファイルを使用する場合、設定の一部は URL または WADL ファイルにより定義され、変更することはできません。



異なる設定に関しては以下に説明されています:

- `$MT_HTTPExecute_Result` を含むテンプレートと結果の解析
- リクエストメソッド
- URL
- 認証
- パラメーター
- HTTP コンテンツ
- HTTP ヘッダーフィールド

テンプレートと結果の解析

テンプレートを使用して、3つのフレームワーク内(ユーザー自身の設定、URL、またはWADL。*上のスクリーンショット参照*) で設定が定義することができます。適切なラジオボタンを選択して、いつでもテンプレート間を切り替えることができます。URL またはWADL テンプレートオプションが既に選択されている場合、異なるURL またはWADL ファイルを使用する場合は、それぞれ **[URL からインポート]** または **[WADL からインポート]** をクリックします。URL またはWADL オプションを選択することにより、(またはそれぞれの **[インポート]** ボタンをクリックすることにより) 使用するURL またはWADL ファイル を指定するダイアログが開かれます。

自身の設定の定義

ユーザー自身の設定を定義する場合、URL としてREST サーバー にリクエストを入力します。ターゲットリソース(XML、HTML またはJSON) のデータフォーマットを指定し、適切な場所にユーザー認証情報、クエリパラメーターとHTTP コンテンツとヘッダーを入力します。これらの設定に関しては下の詳細を参照してください。

URL の使用

URL が長く複雑な場合、*URL の使用* テンプレート内のURL をインポートすることが*自身の設定の定義* テンプレートを入力するよりも奨励されます。例えば、URL は、下のサンプル(5つのパラメーターを含むGoogle クエリ) で示されるように複数のパラメーターを含む場合があります:

```
https://www.google.at/search?q=REST+WADL&ie=utf-8&oe=utf-
8&gws_rd=cr&ei=89cDVrDHMIP0Up_5vcAB
```

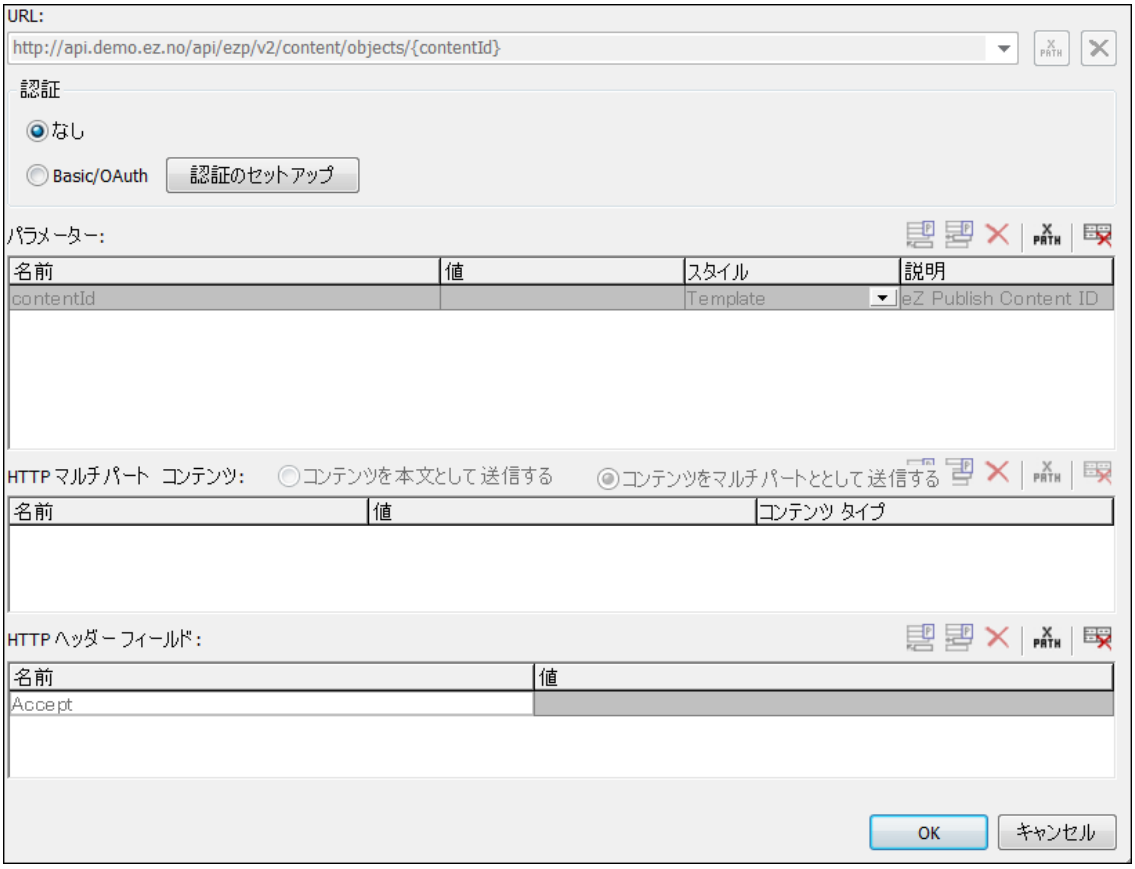
このURL をインポートする場合、テンプレートのURL フィールドに入力しますが、パラメーターは自動的にテンプレートのパラメーターテーブルに入力されます。ターゲットリソース(XML、HTML またはJSON) と*リクエストメソッド*(`GET`、`PUT`、`POST`、または`DELETE`)。のデータフォーマットを選択することができます。パラメーターの値を入力することができますが、パラメータを削除またはパラメーターの型を変更することはできません。URL を変更する場合は、**[URL からインポート]** をクリックします。パラメーターに関する詳細は、下の*パラメーター セクション*を参照してください。

WADL ファイルの使用

WADL ファイルは、Web サービス取りソース間のリレーションシップにより与えられたリソースを定義するXML ドキュメントです。リソースは`resource` 要素により定義されます。各リソースは以下を含みます: (i) リソースの入力を説明する`param` 要素 と (ii) リソースのリクエストと応答を説明する`method` 要素。リクエスト要素は、入力の示し方を指定し、特定のヘッダーに必要な型を指定します。 応答要素はサービスの応答とエラー情報を説明します。

*WADL オプション*を選択すると、使用するWADL ファイルを問うプロンプトが表示されます。 **[OK]** をクリックした後、メソッドの選択ダイアログが表示されます (*下のスクリーンショット*)。このダイアログは、WADL ファイル内で定義されたメソッドを表示します。

使用するメソッドを選択して、**[OK]** をクリックします。リソースのURL は、テンプレートのURL フィールドに、リソースのためにWADL ファイル内で定義されたパラメーターとHTTP コンテンツとヘッダーは、テンプレートのそれぞれのテーブルに入力されます (*下のスクリーンショット参照*)。テンプレート内では、パラメーターとHTTP コンテンツとヘッダーの値とを入力することができますが、パラメーター名は編集することができず、パラメーターを削除することもできません。



以下として結果を解析
結果は、リクエストへの応答内でWeb サービスにより返された内容です。*自身の設定* とURL テンプレートでは、MobileTogether が結果を正確に処理できるように、結果が(XML、HTML またはJSON として) どのように解析されるかを指定する必要があります。WADL テンプレート内では、結果フォーマットに関する情報は、WADL ファイル内の定義から取得され、自動的に選択されます。この結果このテンプレートではラジオボタンは無効化されています。

結果が $MT_HTTPExecute_Result 内に保管されるかを選択することができます。保管される場合は、この変数を使用して結果をデザイン内の他の場所で使用することができます。

リクエストメソッド

自身の設定 と*URL* テンプレート内では、リクエストメソッド(GET, PUT, POST, または DELETE) を指定する必要があります。WADL テンプレートでは、リクエストメソッドは、[リソースのためのメソッドを選択] ダイアログでの選択により決定され (*上のスクリーンショット参照*) 自動的にテンプレート内で選択されます。この結果このテンプレートではラジオボタンは無効化されています。

URL

URL フィールドは、*自身の設定の定義* テンプレート内で編集することができます。このテンプレート内に、URL を直接または XPath 式として入力することができます。[リセット] ボタンを使用して URL フィールド内のエントリを消去します。

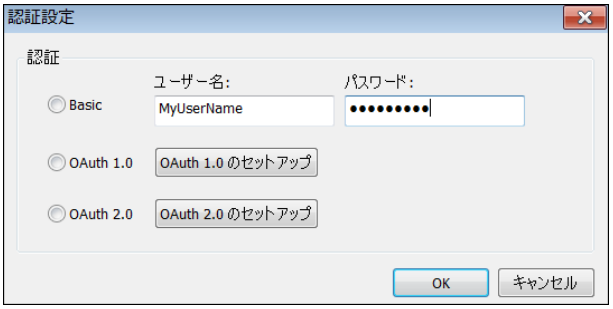
認証

サーバーへのアクセスで認証情報が必要な場合は、必要な情報を提供してください。認証が必要ない場合は、[RESTful API リクエスト] ダイアログで無しを選択してください (*上のスクリーンショット参照*)。

認証 情報は、以下の方法で提供することができます:

- Basic: ユーザー名とパスワード (*下のスクリーンショット参照*)
- OAuth 1.0
- OAuth 2.0

認証 情報をセットアップするには、[RESTful API リクエスト] ダイアログ内の **[認証]** をクリックしてください (*上のスクリーンショット参照*)。これによりサーバーが必要とする認証の型を選択して、必要な認証情報を入力することのできる [認証設定] ダイアログが表示されます (*下のスクリーンショット*)。



OAuth 認証

OAuth は基本的に MobileTogether Designer を認証し、URL により識別された Web サービスのリソースへのアクセスを認証します。これにより、Web サービスは、OAuth をサポートします。OAUTH をサポートする Web サービスの例として、BitBucket documentation about OAuth 1.0 が挙げられます。

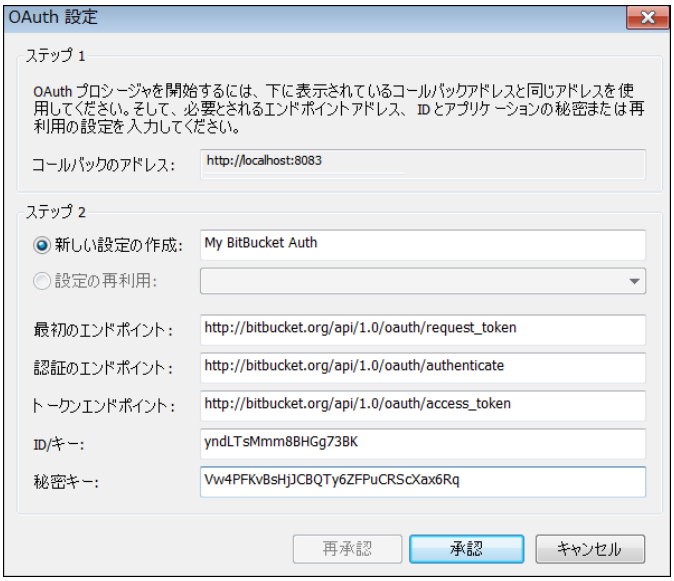
OAuth システムは広い意味で以下のとおり作動します:

1. Web サービスでは、OAuth キー (または ID) と秘密キーを作成します。これらは合わせて OAuth コンシューマーとして知られています。

2. Web サービスのOAuth エンドポイントに注意してください。これらは、OAuth 1.0 のための3つのエンドポイント(最初のエンドポイント、認証 エンドポイント、および トークン エンドポイント) とOAuth 2.0 のための2つのエンドポイント(認証 エンドポイント と トークン エンドポイント)です。エンドポイントは通常すべてのコンシューマーのための定数です。
3. Web サービスにアクセスするためのアプリケーションを、5つの(OAuth 1.0) および4つの(OAuth 2.0) 認証情報を使用してセットアップします。

Web サービスからキー と秘密キーおよび必要なエンドポイントを取得した後、MobileTogether Designer をWeb サービスへアクセスするセットアップの準備が整いました。以下の手順を行ってください:

1. 認証 設定 ダイアログ(*上のスクリーンショット*) 内の**[OAuth 1.0 のセットアップ]** または**[OAuth 2.0 のセットアップ]** をクリックするとOAuth 設定 ダイアログ(*下のスクリーンショット*) が表示されます。



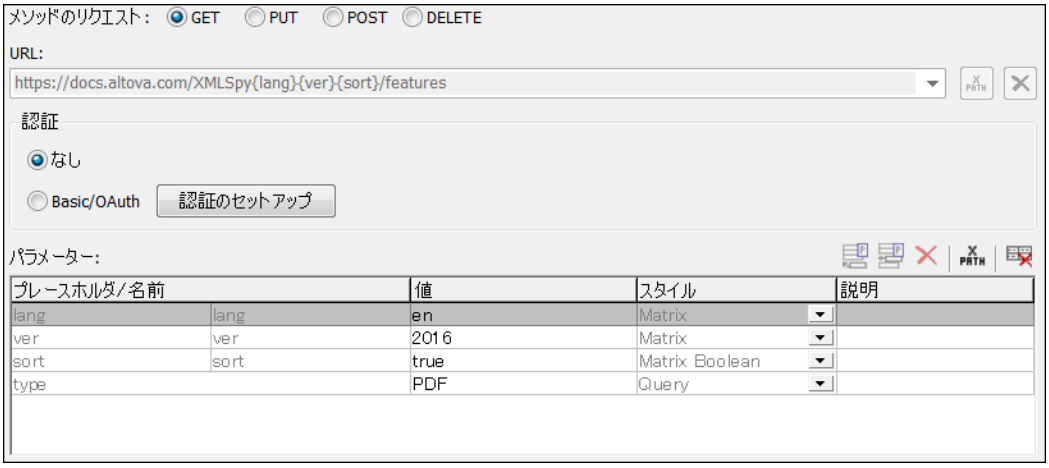
2. *コールバックのアドレス*を`http://localhost:8083` に設定します。このアドレスは固定されており、現在作業中のマシンのアドレスです。
3. *新しい 設定の作成*: 設定に名前を付けてください。これにより(*設定の再利用*コンボボックスオプションを使用して) 同じリソースを使用する他のソリューションで設定を再利用することができます。
4. Web サービスにより宣言されているエンドポイントを入力します。これらは通常 Web サービスのすべてのコンシューマーのためサービス上で同じです。
5. キー(またはID) と秘密キーを入力します。
6. **[認証]** をクリックして完了します。

パラメーター

自身の設定の定義 テンプレート内で、パラメーターを自由に追加、編集、および削除することができます。*URL の使用* と *Use WADL ファイルの使用* テンプレート内では、パラメーターの値のみ編集することができます。パラメーターを追加または削除することはできず、また名前を編集することもできません。次のパラメータの型(またはスタイル)を追加することができます(*下のスクリーンショットの スタイル 列を参照してください*):

- *テンプレート*: テンプレート パラメーターは、ランタイム中にURL 内の値を代替するためにプレースホルダー を使用します。例えば、下のスクリーンショット内では、プレースホルダー`{product}`を持つ テンプレート パラメーターが1つあります。このプレースホルダーは、URL (*下のスクリーンショット参照*)内で使用されています。周りの中かっこはプレースホルダーであることを示します。URL がランタイムで使用されると、プレースホルダはURL 内の対応する箇所で代替されます。URL の対応する箇所は以下に対して解決されます: `https://docs.altova.com/XMLSpy.../features`.

- *マトリックス*: マトリックス パラメーターの場合、URL 内のプレースホルダーは、`name=value` ペアと置き換えられます。下のサンプルのスクリーンショットでは、プレースホルダー `{language}` と`{version}`によりURL 内に与えられた2 つのマトリックス パラメーターがあります。URL 内のこれらのプレースホルダーは青色では以来とされた箇所で表示されています: `https://docs.altova.com/XMLSpy;lang=en;ver=2016.../features`。セミコロンセパレーター `;` は代替の一部としての各パラメーターのプレフィックスです。
- *マトリックス ブール値*: マトリックス ブール パラメーターの値が`true` に設定されている場合、パラメーターのプレースホルダーはパラメーターの名前と置き換えられます。値が`false` に設定されていると、パラメーターのプレースホルダーは空の文字列(つまり空) と置き換えられます。ですから、下のスクリーンショットのサンプルでは、マトリックス ブール値のプレースホルダーはハイライトされている箇所で解決されます: `https://docs.altova.com/XMLSpy;lang=en;ver=2016;sort/features`。セミコロンセパレーター `;` は、代替の一部としての各パラメーターのプレフィックスです。
- *クエリ*: クエリパラメーター はプレースホルダーを使用しません。すべてのクエリパラメーターは、クエリ文字列に集められ、この文字列はURL のパスの一部としてランタイムに追加されます。以下で解決されます。例えば、下のスクリーンショットに表示されるURL は以下に対してランタイムに解決されます: `https://docs.altova.com/XMLSpy;lang=en;ver=2016;sort/features?type=PDF`。疑問符セパレーター **?** は、クエリ文字列のプレフィックスです。追加クエリはアンパサンドセパレーター **&** がプレフィックスです。2 つのクエリを持つクエリ文字列はいかのようになります: `?type=PDF&about=json`.



パラメーターテーブルの列

パラメーター テーブルには4 つの列があります。最初の3列の使用方法は上記のパラメーターの型で説明されています。テンプレート、マトリックス とマトリックス ブール値のパラメーターはプレースホルダーを使用していることに注意してください。テンプレート パラメーター プレースホルダー は、値とおきかえられていますが、マトリックス とマトリックス ブール値のパラメーター プレースホルダーはそれぞれ名前 値ペア と名前と置き換えられています。クエリパラメーターにはプレースホルダはなく、その名前 値ペアがURL パスの一部として追加されています。詳細列はMobileTogether Designer ユーザーのためのパラメータの詳細を含みます。

パラメーターテーブルアイコンとテーブルの編集

パラメーター テーブルの右上に、テーブル内のエントリを[管理] することを有効化するアイコンがあります。

- *パラメーターの追加および挿入*: Use **[追加]** を使用して、新しいパラメーターをテーブル内の最後のパラメーターとして追加します。**[挿入]** を使用して現在選択されているパラメータの上に直接パラメータを挿入します。テーブル内のパラメーターの入力順序は重要ではありません。ですが、URL 内のプレースホルダーの順番は重要です。すべてのクエリパラメーターは、ランタイムでのURL のパスの一部として追加されるクエリ文字列に集められます。
- *パラメーターの削除*: **[削除]** をクリックして、選択されたパラメーターを削除します。
- *パラメーター値のためのXPath 式*: パラメーターが選択されると、**[XPath]** をクリックして[XPath/XQuery 式の編集] ダイアログを開き、文字列に対して解決される式を入力します。この文字列は、パラメーターの値として入植されます。この場合、**[XPath]** アイコンがパラメーターの値列内に表示されます。このアイコンをクリックすることによりXPath 式を編集することができます。

- *パラメーター値のリセット:* **[パラメーターのリセット]** をクリックしてパラメーターの値を削除します。
- *パラメーター名前と値の編集:* 対応するフィールドをクリックして編集します。

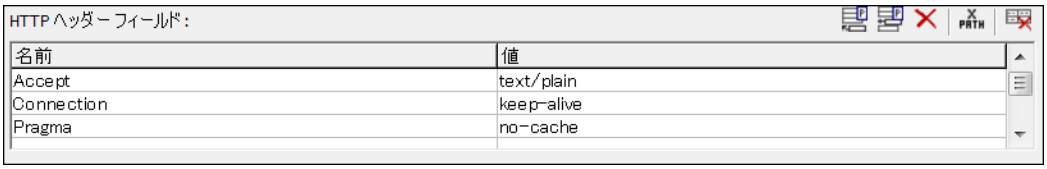
HTTP コンテンツ

HTTP PUT および POST リクエストを使用して、送信するコンテンツを指定することができます。コンテンツをリクエストの本文内の単一アイテムとして送信 (コンテンツを本文として送信) または複数のアイテムとしてマルチパートのリクエスト内で (コンテンツをマルチパートとして送信) 送信することができます。これらの2つのオプションから適切なラジオボタンを選択してください。

コンテンツエントリを右上のアイコンを使用して追加または挿入します。そして、送信されるコンテンツの部分とコンテンツパートのコンテンツ型を入力します。コンテンツの値は実際に送信されるコンテンツです。上のスクリーンショットでは、コンテンツは XPath 式を使用して取得されます。イメージは Base64 フォーマットとして送信されます。

HTTP ヘッダーフィールド

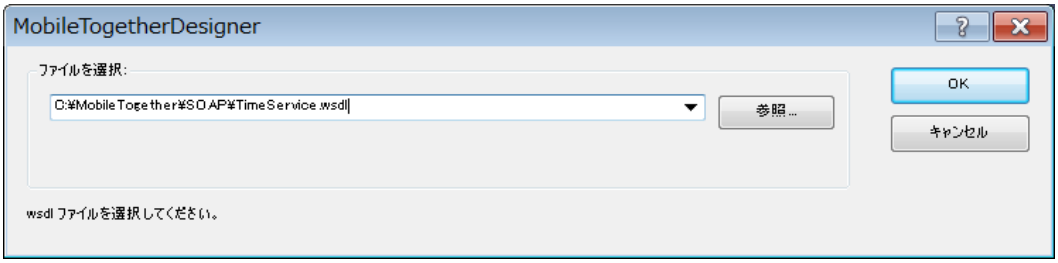
HTTP ヘッダーフィールドは、コロンにより区切られた名前 値ペアです。 例えば、: Accept:text/plain。各ヘッダーにエントリを追加または挿入し、ヘッダー名と値を入力します (*下のスクリーンショット参照*)。



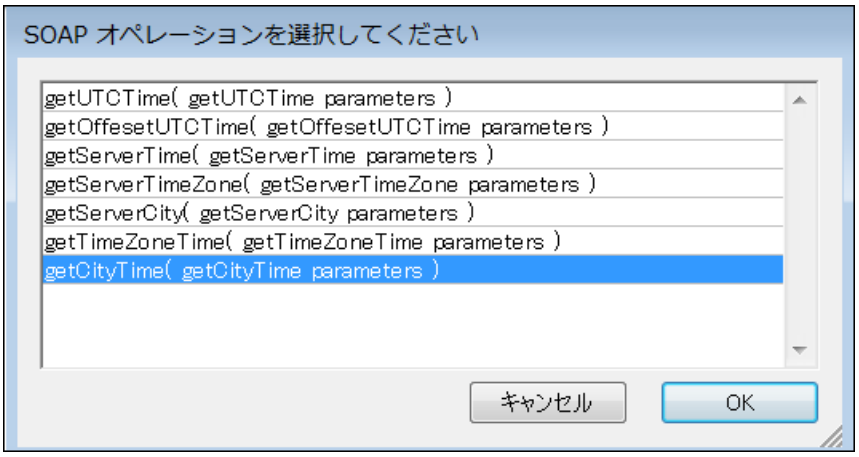
SOAP リクエスト設定

MobileTogether Designer は、WSDL を使用してSOAP リクエストを作成することを有効化します。WSDL ファイルは、どのオペレーションが与えられたWeb サービスで提供されるかを説明します。SOAP プロトコールは、HTTP 越にこれらのオペレーションを呼び出すために使用されます。リクエストを作成するためのMobileTogether Designer での手順は、以下のとおりです:

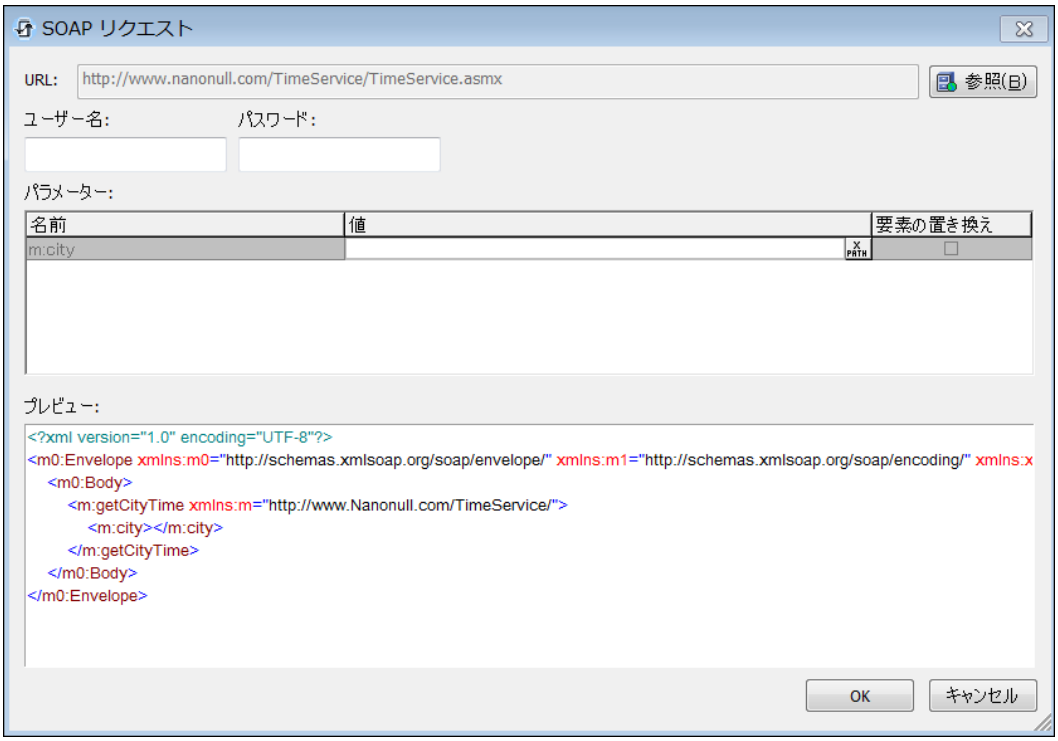
1. ページソースペイン内の**[ページソースの追加]** をクリックして、ページソースを追加します。 パラメーター付きの新規 *HTTP/FTP 要求を選択*して、*SOAP ラジオボタンを選択* します。次のスクリーンでページソース設定を完了して**[完了]**をクリックします。
2. WSDL ファイルを参照または入力するよう促すダイアログが表示されます(*下のスクリーンショット*)。リクエストする Web サービスオペレーションを定義するWSDL ファイルを選択し、**[OK]** をクリックします。



3. **[OK]** をクリックすると、*SOAP オペレーションの選択 (下のスクリーンショット)* が表示されます。このダイアログは WSDL ファイル内で説明されるWeb サービスオペレーションを表示します。リクエストするオペレーションを選択して、**[OK]** をクリックします。



4. SOAP リクエスト ダイアログ (*下のスクリーンショット*) が表示されます。フィールド内のURL は、Web サービスのURL です。[プレビュー] ペインは、SOAP リクエストのテキストを表示します。リクエストがパラメーターを含む場合、これらは[パラメーター] ペインにリストされ、パラメータの値を生成するXPath 式 を入力することができます。下のスクリーンショットでは、例えば、パラメーター`m:city` はXPath 式 `"Boston"` により生成された値を与えられています。 Web サービスにアクセスするための認証情報を入力する必要がある場合、ユーザー名とパスワードをそれぞれのフィールドに入力します。URL フィールドの右横は**[参照]** ボタンです。他のWSDL ファイルを選択するためにクリックして、他のSOAP リクエストを作成します。



5. 完了すると**[OK]** をクリックします。SOAP リクエストは保存され、ランタイム時に送信されます。
6. SOAP 応答を確認するためにシミュレーションを実行します。

7.5 ルートノード

各ページデータソースはツリーとしてコンセプト化されています。各ツリーの各ツリーの**[ルート ノード]** は、各 MobileTogether デザイン(プロジェクト) 内で一意の変数名により識別されます。この用語は以下に示されるページソースツリーの基本的部分に焦点をあて説明するために使用されます。プロジェクトのすべてのページソース変数のリストは、**[プロジェクト] [すべてのページソース変数の使用方法のをリスト]** コマンドをクリックするとユーザーコントロールおよびアクションと共にメッセージウィンドウに表示されることができます。

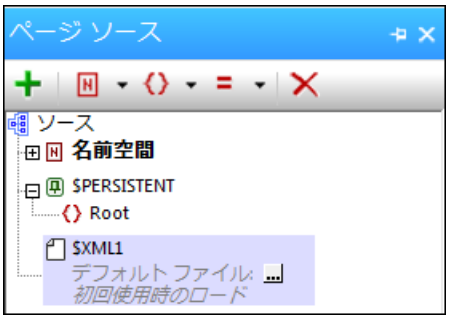
/ = $RootNodeName	• **[ルート ノード]** はツリーの最も外側にあるのノードです。 • RootNodeName が特定のツリーのルートノードを識別する箇所で、ルートノードはフォーム**$RootNodeName** の変数により表示されます。下の*ルートノードの名前* を参照してください。
**<RootElement> <Element-1/> ... <Element-N/> </ RootElement> **	• **[ルート要素]** は XML ドキュメントの最も外側にある要素で、ドキュメントのその他すべての要素を含みます。(XML 言語では、ルート要素は、**[ドキュメント要素]** と呼ばれることもあります。) • ルート要素はルートノードの子と考えられています。

XPath を使用してツリーノードにアクセスする

ツリーのノードはXPath 式を使用してアクセスすることができます。XPath 式を使用する際、コンテキスト ノードが何であるか確認してください。ページのすべてのページ ソースの1 つのノードがページのXPath コンテキストとして設定できます(そのノードを右クリックして、**[ページをXPath コンテキストとして設定]** を選択します)。このノードがページの全てのXPath 式のコンテキスト ノードとなります。例外は、対応するページソースリンクが式のコンテキストノードとしてコントロール内で使用されている、チャートおよび繰り返しテーブル コントロールなどのコントロールです。ページ コンテキスト ノードが設定されているかに関わらず、ツリーのノードは、ツリーのルート ノードのXPath 式を開始することで常にアクセスすることができます。例: **$XML1/root/element1**。

ルートノードの名前

ページデータソースが追加されると、ルート ノードの名前は自動的に生成されます。下のスクリーンショットは、例えば、$PERSISTENT と$XML1 という名前の2 つのルート ノードを表示しています。ルートノードの名前はダブルクリックして編集することができます。



自動的に生成されるルート ノードの名前はページソースの型に左右されます: XML、HTML、HTTP-にアクセスされた DB、XQuery、または FlowForce ジョブ。

$PERSISTENT	$PERSISTENT ツリーはクライアント上に保存されたツリーです。 • Root と呼ばれる空のルート要素付きの新しいデザインに、このツリーは作成されます (*下のスクリーンショット参照*)。手動またはコンテキスト メニューの **[インポート]** コマンドを使用して、XML 構造をインポートしてツリーのために構造を作成する必要があります。 • $PERSISTENT ツリーのノードに静的または動的なデータを割り当てることができます。(XPath 式を使用して、またはコントロールに $PERSISTENT ノードを割り当てることにより) • (ページソースリンクとして) として割り当てられた $PERSISTENT ツリーノードは、クライアントで更新することができます。これは、クライアントでソリューションがロードされると、コントロールに割り当てられているノードが、他のデータソースではなく、$PERSISTENT ツリーからデータを取得するからです。
$XML	手動で作成された、またはインポートされた XML ドキュメント。デフォルトのデータソースは任意です。
$HTML	手動で作成された、またはインポートされた HTML ドキュメント。デフォルトのデータソースは任意です。
$HTTP	HTTP/FTP、REST、または SOAP を使用してアクセスされたドキュメント。リクエストされたリソースはデータを提供します。
$DB	選択されたデータベース テーブルが構造とデータを提供します。
$XQ	XQuery ドキュメント。
$FLOWFORCE	FlowForce ジョブ。
$MT_CONTACTS	連絡先の読み取りアクションが追加されると追加されます。クライアントのアドレスブックからのデータを使用してファイルされます。
$MT_CALENDAR	カレンダーへのアクセス アクションが追加されると追加されます。クライアントのカレンダーからのデータと共に記入されます。
$MT_EMBEDDEDMESSAGE	**埋め込まれたメッセージ** イベントが有効化されると、または、埋め込まれたメッセージをバックする アクションが追加されると追加されます。
$MT_FILEINFO	ファイルの情報の取得 または フォルダーの読み取りアクションが追加されると追加されます。これらのアクション内で指定されたファイルとフォルダーに関する情報を保管します。

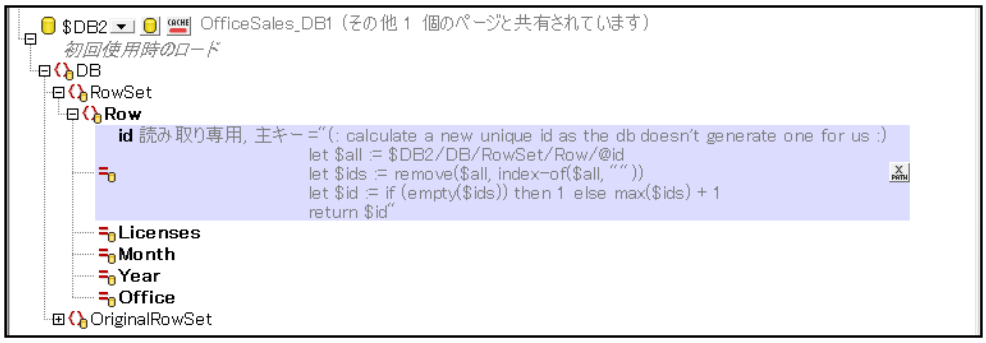
$MT_GEOLOCATION	位置情報のトラッキングの開始/停止 アクション または 位置情報の読み取り アクションがデザインに追加されると追加されます。
$MT_NFC	NFC 開始/停止 アクションが追加されると追加されます。最後に検出されたNFC タグからのデータを使用してフィルされます。
$MT_PUSHNOTIFICATION	**プッシュ通知の受信時** イベントにアクションが追加されるとデザインタイムに作成されます。受信されたプッシュ通知 (PN) のペイロードからのデータがフィルされます。*次を参照してください: プッシュ通知: 受信されるソリューション.*
$MT_SERVICE	サービスデザインが作成されると作成されます。サービストリガーに関するランタイムデータを保持します。

メモ [ページ ソース] ペインの名前をダブルクリックして編集することで、デザイン プロセス中いつでもルートノードの名前を変更することができます。XPath 式の古い名前への参照はデザイン全体で新しい名前に変更されます。

7.6 ページソース ツリー

ツリー構造

ページソースはXML ツリー構造を持ちます。下のスクリーンショットは、DB テーブルであるページソースのXML ツリー構造を表示しています。



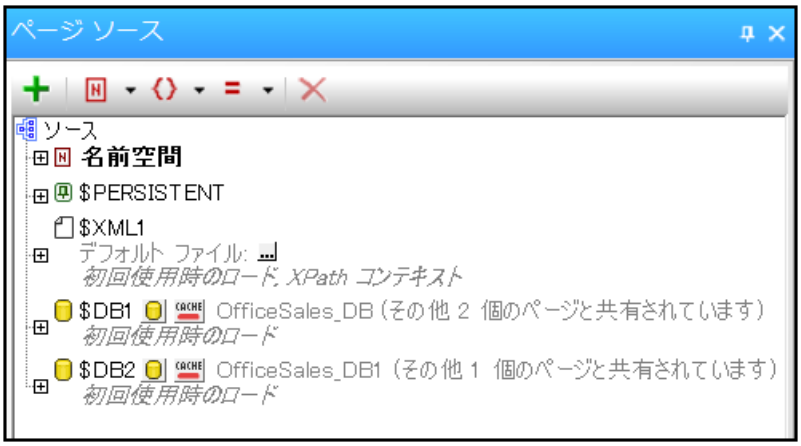
ページ ソースの構造は次の手順で作成されます:

- ページ ソースが追加された時に、XML またはHTML ドキュメント、またはXQuery ステートメントからインポートされます。(*ページ データソースの追加を参照してください*)。
- 要素と属性を新しい空のXML ツリーに追加することにより、(ペインのツールバー アイコンを使用して) [ページ ソース] ペインで手動で作成されます。

ツリー ノードへのアクセス

ページのすべてのページ ソースのノードは、XPath 式によりページ内のどの場所でもアクセスすることができます。

各ページで、(ノードを右クリックして、**[ページをXPath コンテキストとして設定]** を選択することにより) すべてのノードが **ページのXPath コンテキスト ノード** として設定することができます。ページの全てのXPath 式は、ノードのコンテキスト内で評価されます。ページのXPath コンテキスト ノードは、テキスト*XPath コンテキスト*と表示されています。下のスクリーンショットでは、**$XML1** がページのXPath コンテキスト ノードです。ページのすべてのXPath式は**$XML1** に対応して評価されます。



ページ コンテキスト ノードが設定されているにかかわらず、特定のツリーのルート ノード付きロケーター式を開始することにより、すべてのノードをアドレスすることができます。例えば、このページの最初のスクリーンショットでハイライトされている XQuery ステートメント内で、2番目のラインには、ルート ノード**$DB2** から始まるロケーター式付きの**@id** ノードを検索する**let** 式があります。

ソースノード (またはページソースリンク) は、コントロールに関連したツリーノードです。

- ソースノードをツリーからドラッグすることでソースノードはコントロールに関連付けることができます。
- ソースツリーノードがページソースリンクとして定義づけられるとページソースツリー内で太字で表示されます。
- 通常、ページソースリンクはコントロール内のソースノードのコンテンツを表示されるために使用されます。例えば、ラベルコントロールは、関連したページソースリンクのコンテンツを表示します。
- チャートと繰り返しテーブルの場合、コントロールのソースノードは、コントロールのプロパティを定義するために使用されたすべての XPath 式のコンテキストノード(XPath オリジン) として役割を持ちます。

ツリー データ

MobileTogether ソリューションで使用されたデータは、プロジェクトのページ ソース ツリーのノードに保管されています。このデータは以下の方法で取得することができます:

- データソースのデフォルトファイルとしてファイルは指定されます。このファイルは、ページソースの構造に対応する必要があります。ファイルのデータは、その後ページ ソースのデータをして使用することができます。
- ノードのコンテキスト メニューの**[ページのロード時に存在することを確認(固定値)]** コマンドを使用して、ノードに固定した値を与えることができます。
- ノードのコンテキスト メニューの**[ページのロード時に存在することを確認(XPath 値)]** を使用して、ノードに XPath 式を与えることができます。XPath 式はノードのコンテンツを生成します。この値は、デフォルトのファイルからインポートされた値を上書きします。
- イベントがを*ノードの更新アクション*をトリガーするように、または、ノードが機能を編集できるコントロールのソースノード(例えば、コンボ ボックスとフィールドの編集コントロールなど) であると定義されると、ノードは更新されます。

ページソースリンク

ソースノード (またはページソースリンク) は、コントロールに関連したページソースツリーノードです。コンボボックスや編集

フィールドコントロールなど、編集機能を与えるコントロールの場合、エンドユーザーにより編集されたデータは、ツリーノードにパスされます。ソースノードは、[ページソース] ペインからコントロールにドラッグすることでコントロールに割り当てることができます。

コントロールのページソースリンクは、データソースツリー内で**太字で表示**されています。ページソースリンクをポイントすると、デザイン内で関連付けられたコントロールについての情報がポップアップします。一方、ページソースリンクに関連付けられたコントロールは、コントロールの左上にアイコンが表示されます。アイコンをポイントすると関連付けられたページソースリンクについての情報が表示されます。

ツリー構造

ページにデータソースを追加すると、[ページ ソース] ペインにページソースのルート ノードが作成されます(下のスクリーンショット)。追加されたページデータソースの種類 により、ツリー構造は自動的に作成されるされない可能性があります。例えば、外部 XML ファイルからインポートされた構造で新しいXML ツリーが作成されると、ページソースが追加される時に、構造は自動的に作成されます。一方、HTTP リクエスト を介してページソースが追加された場合、たとえば、ツリー構造は作成されません。

ページソースが追加されて、ルートノードが作成された後、ツリー構造は以下の方法で追加することができます:

- 外部 XML ファイルからXML 構造をインポートすることにより追加
- [ページ ソース] ペインのツールバーのコマンドを使用して、手動でツリーを構築することにより追加

これらの方法は下で説明されています。

XML 構造のインポート

[ページ ソース] ペインのルートノードを右クリックして、**[XML から構造をインポート] をクリックして、使用する**XML **ファイルを参照します。** 以下が可能な結果です:

- ルート ノードに子孫ツリー構造がない場合、XML ファイルのルート要素とツリー構造はインポートされます。XML ファイルのルート要素は、ページソースのルート要素として追加されます。
- ルート ノードがルート要素を持つ場合、このルート要素とそのすべての子孫は、XML ファイルのルート要素とツリー構造により置換えられます。

インポートされたツリーの構造は、下に述べられている通り、手動で変更することができ、データはノードに割り当てられることができます。(この点については、次のセクション ツリー データで説明されています)。

手動によるツリー構造の作成

要素と属性は(ルート ノードを含む)ツリー構造のノードに対応して追加することができます。データソースのノードを選択して、適切なツールバーコマンドをクリックします (*下部のツールバーのスクリーンショット参照*)。一時的な要素と属性は、計算のためにデータを保つ、または他の理由のためにファイルに保存されません。一時的ノードのデータは保存されません。

アイコン	コマンド	方法 ...
+	**ソースの追加**	ページソースの追加ダイアログを表示します。ルート ノードが追加されたデータソースのために作成されます。ルートノードには子要素 1 つだけ追加できます。
N	**名前空間の追加**	名前空間宣言を*名前空間* エントリの下に挿入または追加します。デフォルトのプレフィックスを必要に応じて編集して、名前空間を入力します。
{}	**要素の追加**	選択されたノードに関連した子要素を挿入、追加します。
=	**属性の追加**	選択されたノードに関連した子要素を挿入、追加します。
×	**削除**	選択されたノードを削除します。

ノードコンテンツを手動で追加する

選択されたノードのコンテキストメニューの 2 つのコマンドにより個別ノードにコンテンツを手動で追加することができます:

- *ページのロード時に存在することを確認 (固定値):* 固定された文字列の値がノードのコンテンツとして追加されツリーに表示されます。
- *ページのロード時に存在することを確認 (XPath 値):* XPath 式によりノードのコンテンツが提供されます。XPath 式とXPath の編集アイコンがツリーに表示されます。

ノードのコンテンツはページがロードされる前に生成され、ページはこのノード コンテンツと共にクライアントにパスされます。

この方法で手動により追加されたコンテンツは、デフォルトファイルまたは**ページのロード時に存在することを確認**コマンドにより追加されたコンテンツをオーバーライドします。

ツリーデータ

編集可能なおよび読み取り専用データ

ツリーが編集可能なデータソースまたは読み取り専用ページ ソースであるかにより、ツリーのデータは 編集可能 または 編集不可能 (読み取り専用) であることができます。ページ ソースが編集可能または読み取り専用であるかは、ページソースが追加される際に定義されます。編集可能 読み取り専用の定義を変更したい場合、ページソースを削除して、新しい定義で再作成してください。

クライアント アクションにより編集可能ノードのコンテンツを変更することができます。例えば、コンボ ボックスは編集可能なデータソースのノードに関連付けられていて、エンド ユーザーのコンボ ボックスの選択は、関連するノードにパスされ、変更された値となります。この場合、読み取り専用データ ソースは、関連したノードのコンテンツは表示目的のみに使用されます。これらの関連したノードは**ページ ソース リンク**として知られています。ソース ノード リンクは、ツリー ノードをコントロールにドラッグしてコントロールに追加することができます。

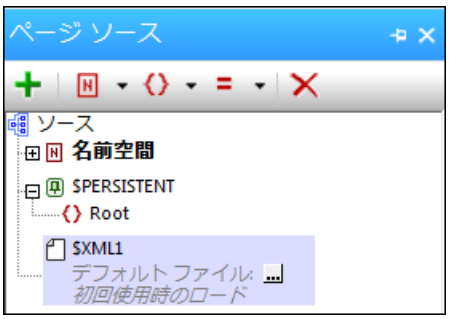
データソースにデータを割り当てる

データは、(編集可能 と読み取り専用 データソースツリー両方に) 次の方法でノードに割り当てられます:

- *デフォルトのファイルの割り当て*: デフォルトのファイルのデータはツリーのノードにパスされ、ノードのコンテンツとなります。デフォルトのファイルの構造はページソースツリーの構造と同じである必要があります。
- *ノードコンテンツを手動で追加する*: 各ノードのコンテキストメニューには、ノードのコンテンツを指定するを許可する (**[存在することを確認]** コマンド) コマンドがあります。ノードにすでに (デフォルト ファイルの使用などの) 他の方法で割り当てられたコンテンツがある場合、手動で追加されたノード コンテンツにより、既に割り当てられたコンテンツは上書きされます。

デフォルトファイルを割り当てる

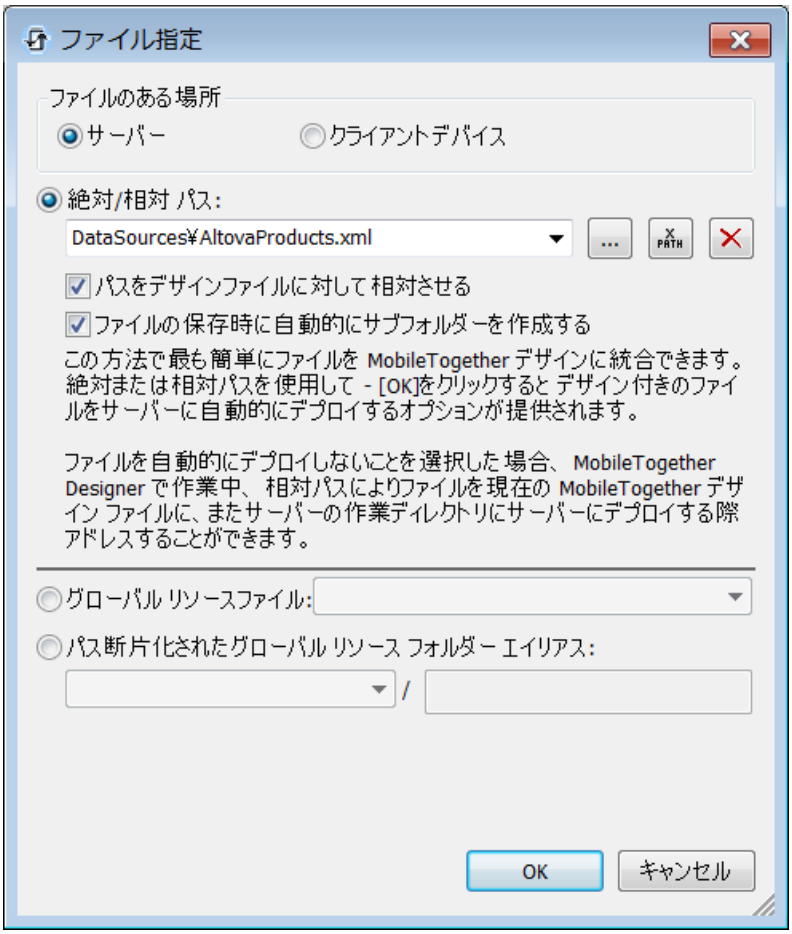
XML データソースは、割り当てられたデフォルトファイルを持つことができます。デフォルトファイルのデータは、データ ツリーとしてページ ソースにパスされます。デフォルトファイルを割り当てるには以下を行います: ページソースのルート ノード名の下が、デフォルト ファイルのエントリです (*下のスクリーンショット参照*)。



[追加ダイアログ] ボタンをクリックして、ファイル指定ダイアログ (*下のスクリーンショット*) を表示し、必要なファイルを選択して、**[OK]** をクリックします。割り当てが行われ、*デフォルト ファイル* エントリにファイル パスが表示されます。デフォルト ファイルが割り当てられると、*デフォルト ファイル* エントリをダブルクリックして、新しいデフォルト ファイルを参照して、割り当てを変更することができます。

デフォルトファイルからのデータはページソースのデータとして使用されます。しかし、データが使用されるためには、デフォルト ファイルは、ページソースと同じ構造を持つ必要があります。デフォルト ファイルがページソースに割り当てられると、その構造は自動的にインポートされます。 XML ファイルの構造をインポートするには、**XML から構造をインポート**のコンテキスト メニュー コマンドを使用します。デフォルト ファイルの構造に一致させるためにデータソースの構造を手動で作成することもできます。

ファイルがサーバーに存在する場合
デフォルトのファイルがサーバーに存在する場合、サーバーのラジオボタンを選択します (*下のスクリーンショット参照*)。このダイアログにより、ファイルを参照することができます (*絶対 /相対パス*) または*グローバルリソース*を使用してファイルを指定することもできます (*ファイルエイリアス*または*フォルダーエイリアス*)。希望するオプションを選択してください。



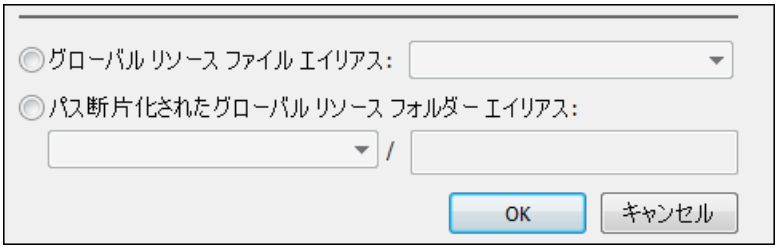
- **絶対 相対 パス:** ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細は*プロジェクト ファイルの場所*を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対 絶対パスがファイルにアクセスするために(サーバーのデータベース) 内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関しては*プロジェクトファイルの場所*のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります 。アクションがファイルの

ロードのみに制限される場合は、このオプションは、存在しません。

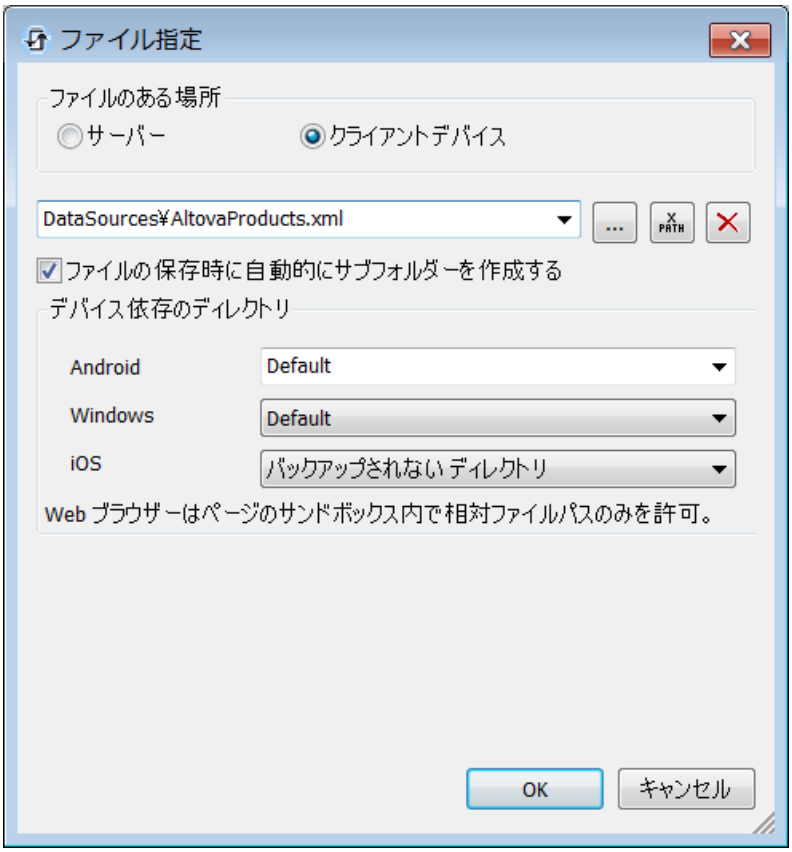
- **グローバル リソース ファイル エイリアス**: コンボ ボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイル リソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova グローバル リソースのセクションを参照してください。

- **パス断片化されたグローバル リソース フォルダ エイリアス**: コンボ ボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがクライアントに存在する場合
デフォルトのファイルがクライアント上に存在する場合、場所を選択することによりパスを指定します。または XPath 式を使用してパスを構築します。**[リセット]** ボタンを使用して現在のエントリを削除します。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (ソースファイル と ターゲットファイル オプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。
- **デバイスに依存するリソース** ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしない

ディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- **シミュレーションのためのファイルのロケーション**: ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ（**ツール | オプション**）内で指定されていなければなりません。

エンドユーザーにより（クライアントデバイス上で）ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子*: ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター*: クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト（例: `txt,html;xml`）、または (ii) 各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式（例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`）

- *オプションのデフォルトのファイル*: 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス*: ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、または XPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する*: もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

  メモ　iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法は iCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

ノードコンテンツを手動で追加する

選択されたノードのコンテキストメニューの 2 つのコマンドにより個別ノードにコンテンツを手動で追加することができます:

- *ページのロード時に存在することを確認（固定値）*: 固定された文字列の値がノードのコンテンツとして追加されツリーに表示されます。
- *ページのロード時に存在することを確認（XPath 値）*: XPath 式によりノードのコンテンツが提供されます。XPath 式と XPath の編集アイコンがツリーに表示されます。

ノードのコンテンツはページがロードされる前に生成され、ページはこのノード コンテンツと共にクライアントにパスされます。

この方法で手動により追加されたコンテンツは、デフォルトファイルまたは**ページのロード時に存在することを確認**コマンドにより追加されたコンテンツをオーバーライドします。

7.7 プロジェクト内の名前空間

名前空間はノードを正確に認識するため、また、式を使用してノードの位置を正確に決めるために重要です。[ページソース] ペイン(*下のスクリーンショット*) の*名前空間* アイテムには、ページデザインビューで現在アクティブなページに関わりなく、プロジェクトで宣言されたすべての名前空間が含まれています。

名前空間を宣言するには次の2 通りの方法があります:

- *データのインポート時に自動的に宣言する* 外部 XML ファイルがページソースとして追加された場合、ソースの名前空間は自動的にデザインにインポートされ、プロジェクト全体のスコープのために宣言されます。[ページソース] ペイン(*上のスクリーンショット参照*) の名前空間アイテムの下に表示されます。デザイン内にあいまいさを引き起こさない場合、名前空間プレフィックスは、自動的にオリジナルのプレフィックスに一致するように設定されています。名前空間の宣言に割り当てられたプレフィックスは、ノード名に使用されており、ページソース内のノードの場所を決定するための XPath 式に使用される必要があります。
- *ユーザーによる定義*: [ページソース] ペイン(*上のスクリーンショット*) のツールバーの**[名前空間]** アイコンをクリックして、名前空間を追加することができます。自身の名前空間をプロジェクトに追加することにより、1 人以上のユーザーにより宣言されるノードを作成することができます。同じローカル名を持つノードのあいまいさをなくすためにこれは有用です。

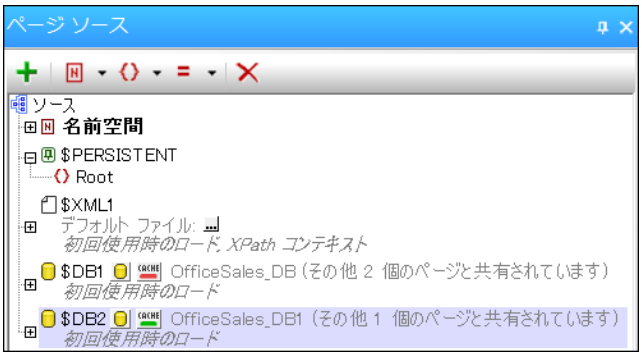
名前空間を削除するには、ペインのツールバーの**[削除]**を選択します。

メモ 名前空間のプレフィックスは、デザインのプロセス中いつでも、[ページソース] ペイン内の名前空間プレフィックスをダブルクリックして編集することにより、名前変更することができます。デザインの XPath 式内の古いプレフィックスへのすべての参照は、新しいプレフィックスに変更されます。

メモ XPath のデフォルトの名前空間 (`xpath-default-ns=''`) は、拡張関数やユーザー定義関数を含むすべての XPath/XQuery 関数に使用されています。

7.8 キャッシュ

ページソースにリンクされている外部リソース(XML ファイルおよびDB) のデータは、サーバーでキャッシュされることができます。データソースがサーバーでキャッシュされたかは、ページソースのキャッシュアイコンの色で表示されます (*下のスクリーンショット参照*)。赤いキャッシュのシンボルはページソースがまだキャッシュされていないことを表示します。緑のシンボルはキャッシュがページソースのために作成されていることを示します。ページソースが(例えば、デフォルトファイルを介して) 外部ソースにリンクされていない場合、キャッシュシンボルは(下のスクリーンショットの `$XML1` に表示されているように) 表示されません。



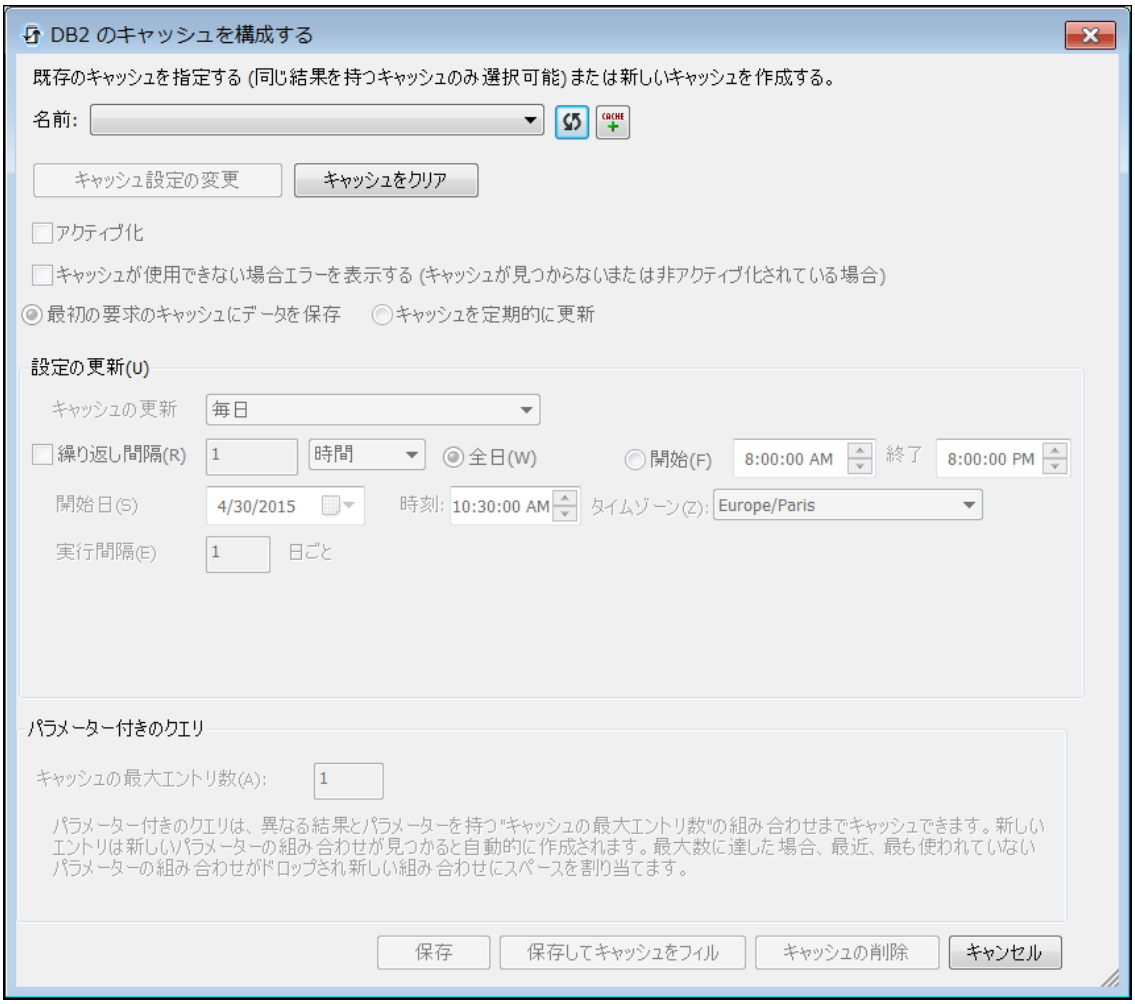
キャッシュする理由

キャッシュの作成には2 つの主な理由があります: (i) ページデータソースがレポートをゆっくり作成する場合 (例えば大きなデータベースの場合) および(ii) ページソースが頻繁に変更されない場合。上記の場合、データがサーバからではなくデータキャッシュより取得された方が、ソリューションの実行は速くなります。

キャッシュを最新状態に保つため、キャッシュが作成される際にキャッシュの更新頻度を設定することができます。MobileTogether Designer でキャッシュが定義されると、異なるデザインのページソースにより使用されることができ、基になるデータ構成に互換性を持たすことができます。ページソースがキャッシュを持つと定義されている場合、キャッシュされたデータがソリューションを実行する際に使用されます。ソリューションがサーバにデプロイされると、キャッシュを使用することができます。

キャッシュの作成

ページソースのキャッシュを作成、または既に作成された、キャッシュの設定を編集するには、ページソースの **[キャッシュ]** アイコンをクリックまたは、ページソースのルートノードのコンテキストメニューから **[キャッシュ設定]** を選択します。これにより、キャッシュの構成ダイアログ(*下のスクリーンショット*) が表示されます。

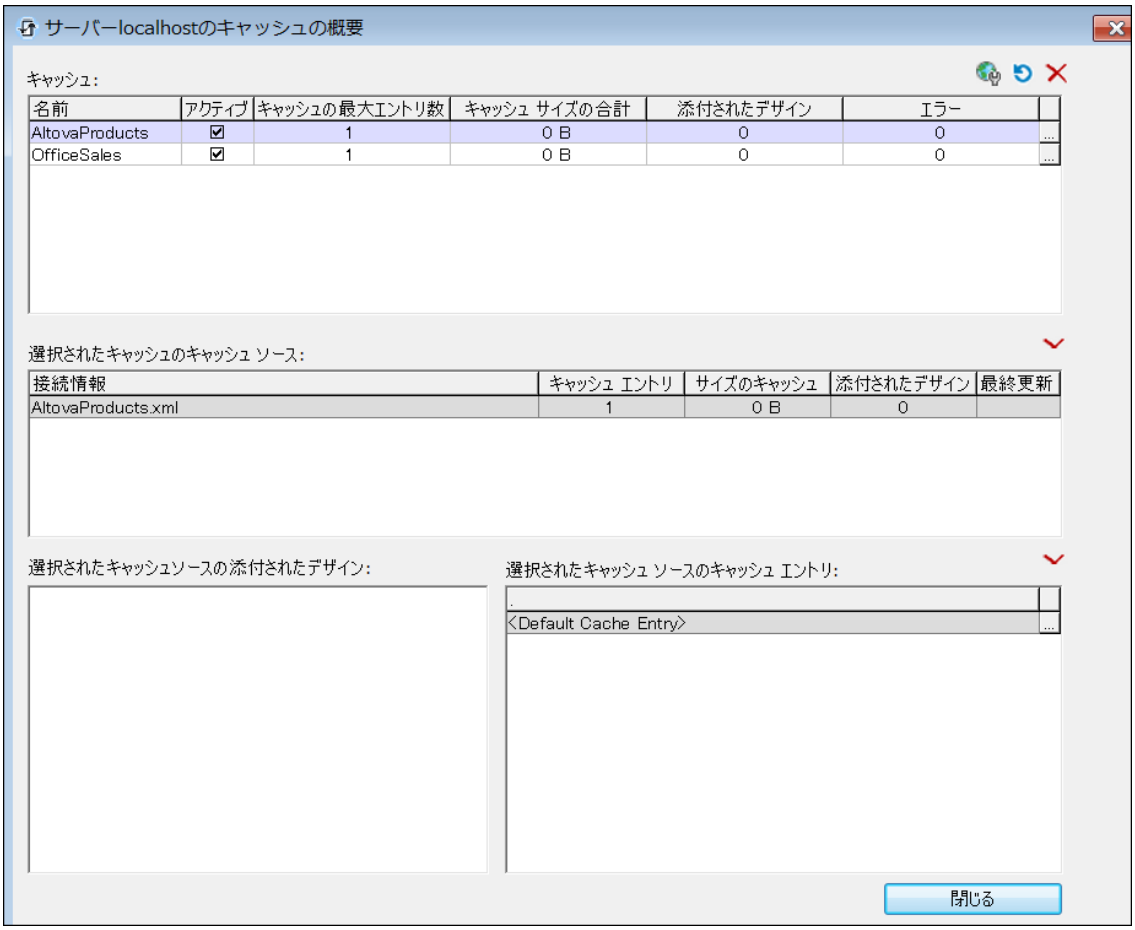


このダイアログでは、以下を行うことができます:

- このページソースにマッチする最新のキャッシュを確認するためにサーバーへの接続を更新します。
- このページソースに新しいキャッシュを追加します。
- キャッシュ設定の変更をクリックして、既存のキャッシュの設定を編集します。
- キャッシュをアクティブ化/非アクティブ化します。
- ソリューションを実行する際、キャッシュがない場合、エラーを表示するか指定します。
- キャッシュの頻度を指定します。

キャッシュの概要

キャッシュの概要ダイアログ (*下のスクリーンショット*) は、**[プロジェクト | キャッシュの概要]** メニュー コマンドからアクセスすることができます。



ダイアログは、サーバー上の全てのキャッシュの概要を提供します。ダイアログにより以下を行うこともできます:

- キャッシュのアクティブ化/非アクティブ化
- キャッシュの削除

7.9 コンテキストメニュー

[ページソース] ペインのアイテムのコンテキストメニューのコマンドは以下で説明されていて、2 つのグループに整理されます:

- ルートノードのコンテキストメニュー
- ツリーノードのコンテキストメニュー。

ルートノードのコンテキストメニュー

以下にリストされたコマンドは、ルートノード ($XML、$DB、$HTML など) のコンテキストメニューで使用することができます。データソース (XML、DB、HTML など) の全ての種類に共通するコマンドに加え、ページソースの一部の種類はその種類 (例えば、DB ページソースのコマンド) に特有のコマンドをもちます。そのようなコマンドの特有性に関しては、関連のある個所で説明されています。

▾ 子の挿入、追加、および追加

選択されたノードに対応した要素と属性の追加を有効化します。**[挿入]** はノードを選択されたノードの前に追加します。**[追加]** はノードをその種類の最後のノードの後に追加します。選択されたノードの直後にノードの追加を希望する場合、直後のノードに移動して **[挿入]** コマンドを使用します。

▾ サーバー上にデータを保存

モバイルデータネットワーク上で送信されるデータの量の減量は、モバイルソリューションのパフォーマンスを向上させます。MobileTogether はどのデータをクライアントデバイスに送信し、どのデータをサーバーに保持するかを指定することができます。例えば、グラフを表示するために特定のデータセットのみが必要な場合、そのデータはサーバー上に保持することができます。グラフのイメージはサーバーにより表示され、基となるデータがモバイルネットワークに送信されることなくクライアントに送信されます。大きなサイズのデータセットではこの設定はパフォーマンスの向上につながります。

この切り替えコマンドは、ツリーのデータがサーバーに保存されるかを指定します。サーバに保存される場合、永続と定義することはできません。下の **[データをクライアント上に保持]** コマンドを参照してください。

▾ 読み取り専用のデータ

ツリーのデータが読み取り専用かを指定する切り替えコマンドです。読み取り専用データツリーは表示と計算のためのデータを提供します。編集が必要なデータを保持することはできません。

▾ クライアント上の永続データ

ツリーを永続ツリーに切り替えるコマンドです。いくつでも永続ツリーとして定義することができます。ツリーが永続と定義されると、ソシューション終了後、ツリー内のデータはクライアントデバイスに保留されます。ソリューションがそのクライアントで次回開かれると、永続データが表示されます。ツリーが永続と定義されていると、サーバーにデータを保存することはできません。上の **[データをサーバー上に保持]** コマンドを参照してください。

▾ データのロード

このコマンドは、次の共通した排他的オプションのサブメニューのロールアウトです。(1 つのみ選択できます):

- *最初の使用時*: 使用されているページに入るとツリーをロードします。一度ロードされると、自動的には再ロードされません。複数のページで同じツリーを共有する場合、ページが初回開かれた際 (トップページまたは、サブページに関わりなく)、ツリーはロードされメモリに残ります。
- *すべてのページ*: トップページまたは、サブページに関わりなく、ページを開くたびにツリーをロードします。このオプションは、ロードに時間がかかるため処理が遅くなる場合があるので注意してください。しかし、このオプションによりデータが各ページより新しく取得されることを保証できます。
- *自動ではない*: ツリーは自動的にロードされません。再ロード、ファイルからロード、またはHTTP からロード アクションを使用してロードする必要があります。代わりに、ノードの追加 とノードの挿入 アクションを使用して、ソースからデータをロードすることなく、最初から作成することもできます。これらの5 つのアクションをデータのロードの設定に関わりなく使用することができます。これらのアクションは、**[データのロード]** が *最初の使用時*または*すべてのページ*に設定されていても、特定の瞬間にツリーを再ロードするために使用することができます。

デフォルトの設定は*最初の使用時*です。

データの保存

[データの保存] コマンドは、サブメニューのロールアウトで、(一度に1 つのみ選択可能な) 次の相互に排他的なオプションを表示します:

- *すべてのページ移動時*: ツリーのデータはツリーを含むページから移動する際保存されます。
- *ソリューションの完了時*: ソリューションの状態、または終了方法に関わらず、ソリューションを完了するとツリーのデータは保存されます。
- *最後の送信時*: ツリーのデータは、ワークフローの処理がデザインされたとおり、最初のページから最後のページまで、最後の**[送信時]** ボタンがタップされた時まで保存されます。もしこのオプションが選択されていて、ソリューションが最後の**[送信時]** ボタンがタップされる前に終了した場合、ツリーのデータは保存されません。
- *自動ではない*: ツリーは自動的に保存されません。保存、ファイルに保存 、またはHTTP/FTP をロード保存 アクションを使用してデータを保存します。

デフォルトの設定は、*自動ではない*。

データをリセットする

このコマンドにより、選択されたルートノード(ページソース)のデータをページを移動する都度リセットできるようになります。2つのオプションから選択することができます: (i) ページソースデータをページを移動する都度リセットする。(ii) ページを移動する際にページソースデータを自動的にリセットしません。(後者の場合、ページの移動以外の条件以外の場合にデータのリセットが必要される場合、リセット アクションを使用してください)。

▾ ページをXPath コンテキストとして設定する

選択されたノードをページのXPath コンテキスト ノードとして設定します。効果の注釈がノード(*下のスクリーンショット参照*) の下に表示されます。ページのXPath コンテキストは、ページのすべてのXPath 式のコンテキストノードです。

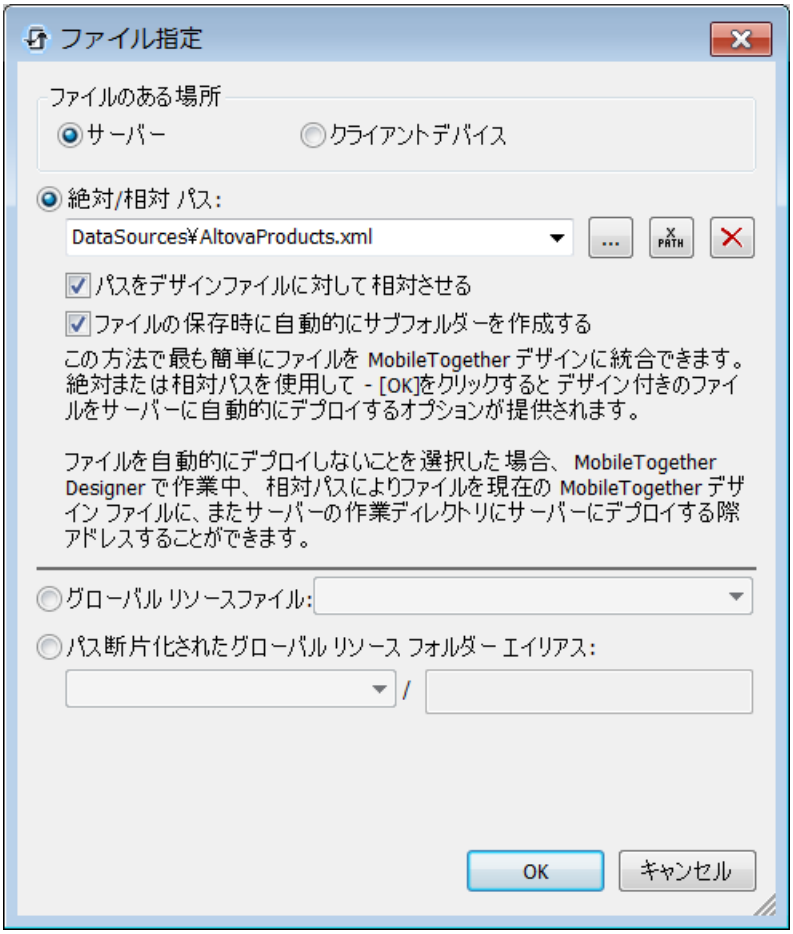
このコマンドは切り替えることができます。ですから、ノードをページのXPath コンテキスト ノード、または、ページのXPath コンテキスト ノードとしてのノードの設定をオフに切り替えることができます。ノードがXPath コンテキスト ノードと設定され、他のノードが既にこの設定をされている場合、設定は前に割り当てられたノードの設定をオフに切り替え、新しく割り当てられたノードの設定をオンに切り替えます。

▾ デフォルトファイルの選択[デフォルトファイルを持つルートノード]

デフォルト ファイルとして使用されるファイルを指定する、ファイル指定ダイアログ(*下のスクリーンショット*) が表示されます。デフォルト ファイルからのデータは、データソースのデータとして使用されます。しかし、データが使用されるには、デフォルト ファイルは、データソースと同じ構造を持つ必要があります。デフォルト ファイルがデータソースに割り当てられると、その構造は自動的にインポートされません。XML ファイルの構造をインポートするには、コンテキスト メニュー コマンド**[XMLから構造をインポート]** を使用します。以下を参照してください。データソースの構造を手動で作成して、デフォルト ファイルの構造にマッチすることができます。

ファイルがサーバーに存在する場合

デフォルトのファイルがサーバーに存在する場合、サーバーのラジオボタンを選択します(*下のスクリーンショット参照*)。このダイアログにより、ファイルを参照することができます(*絶対/相対パス*) またはグローバルリソースを使用してファイルを指定することもできます(ファイルエイリアスまたはフォルダーエイリアス)。希望するオプションを選択してください。



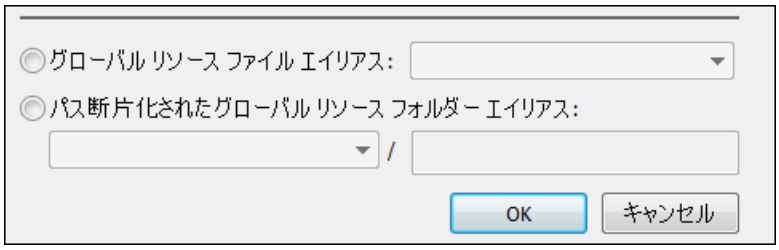
- **絶対 相対 パス**: ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細はプロジェクト ファイルの場所を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対 絶対パスがファイルにアクセスするために(サーバーのデータベース) 内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関してはプロジェクトファイルの場所のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイル パス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

- **グローバル リソース ファイル エイリアス**: コンボ ボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアス

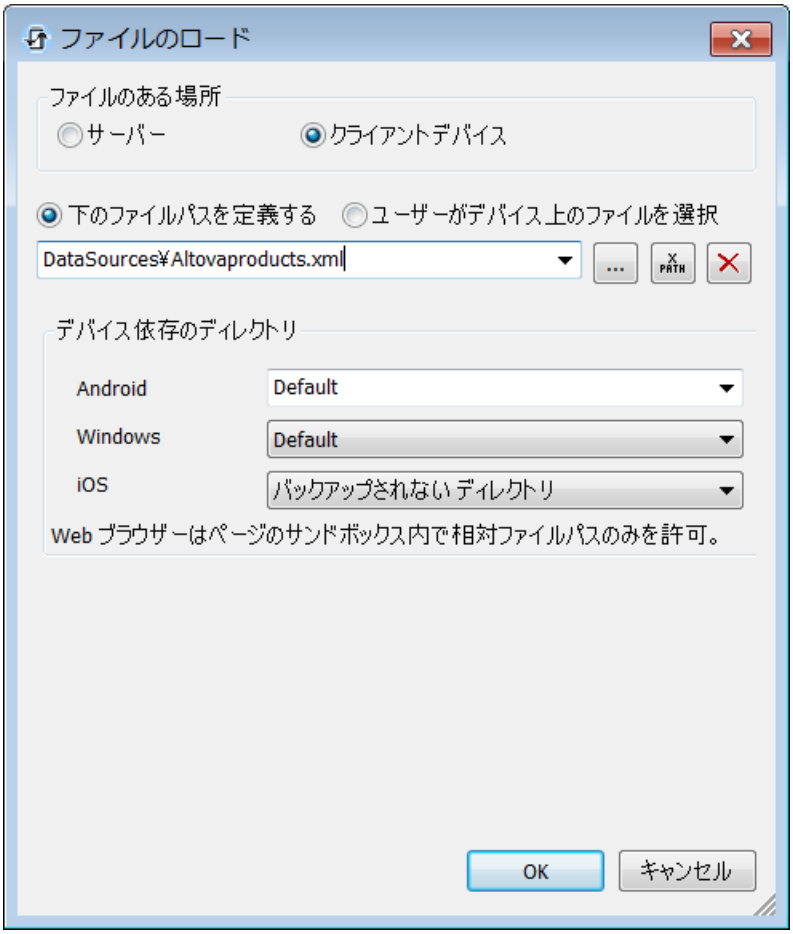
は、異なるファイルリソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova グローバル リソースのセクションを参照してください。

- **パス断片化されたグローバル リソース フォルダ エイリアス**: コンボ ボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがクライアントに存在する場合
デフォルトのファイルがクライアント上に存在する場合、場所を選択することによりパスを指定します。または XPath 式を使用してパスを構築します。**[リセット]** ボタンを使用して現在のエントリを削除します。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイルを指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ　エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷(*ソースファイル*と*ターゲットファイル*オプションがあります)、ファイルのロード/保存、イメージのロード/保存、バイナリのロード/保存。

メモ　クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- **ファイルの保存時に自動的にサブフォルダーを作成する**　ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。
- **デバイスに依存するリソース**: ファイルが存在するディレクトリを選択します。 Windows Phone/RT と

iOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android と Windows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- **シミュレーションのためのファイルのロケーション**: ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ (**ツール | オプション**) 内で指定されていなければなりません。

エンドユーザーにより (クライアントデバイス上で) ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子*: ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター*: クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト (例: `txt,html;xml`)、または (ii) 各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式 (例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`)

- *オプションのデフォルトのファイル*: 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス*: ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、または XPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する*: もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

メモ: iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法は iCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

▾ デザインファイルに XML を埋め込む

デフォルトファイルが割り当てられている XML ページソースのルート ノードが選択されると、このコマンドは有効化されます。選択することにより、デザイン (`.mtd`) ファイルの XML データソースは埋め込まれます。データソースが埋め込まれると、*埋め込まれた*プロパティがルート ノードの注釈に追加されます。(i) 埋め込みの長所と短所、および (ii) 埋め込みの代替手段に関する詳細は、プロジェクト ファイルの場所 とデザイン ファイルの XML の埋め込み を参照してください。

ファイルをデザインと共にデプロイする

この切り替えコマンドはデータソースがデプロイ可能なファイル、通常はデフォルトファイル、に関連付けられると有効化されます。デプロイが可能なファイルは既に[ファイル] ペインにリストされています。

- コマンドを[オン] に切り替えると、[ファイル] ペインのファイルのチェックボックスを選択し、デプロイします。
- コマンドを[オフ] に切り替えると、[ファイル] ペインのファイルのチェックボックスの選択を解除し、ファイルはデプロイされません。

ファイルが最初にデザインに追加されると、ファイルのデプロイを問うプロンプトが表示されることに注意してください。

変数の使用方法のリスト

各ページソースのルートノードは、変数です。例えば、`$XML1` または`$DB1`。 **変数の使用方法** コマンドは、[メッセージ] ペイン内に選択されたルーとノード変数の使用方法をリストします。リスト内のアイテムは変数が使用されるコントロールとアクションです。(変数は通常 XPath 式で使用されます。) リスト内のアイテムをクリックすることにより、コントロールがハイライトされるか変数の使用を含むアクションダイアログが開かれます。

データの型

ロールアウトするサブメニューから**XML HTML** および**JSON** を選択します。選択によりターゲットとするデータソースの型が指定され、MobileTogether Designer が受信および発信するデータを正確に処理できるようにします。この選択はいつでも変更することができます。変更によりデータソースが新しいデータ型のために再解析されます。

構造の再ロード

選択されたページソースの構造を再ロードします。コマンドは構造が、XML ファイルまたはDB などの外部リソースをベースにする場合のみ有効化されます。XML ファイルの場合、デフォルトのファイルがある場合、はコマンドが有効化されます。

XML から構造をインポートする

ページソースツリーからXML 構造をインポートするXML またはHTML ファイルを選択することのできる参照ダイアログを開きます。ツリーに既に構造が含まれる場合、既存の構造の複数のノードを保持するか問われます。既存の構造を保持することを選択すると、新しい構造が既存の構造にマージされることができず、新しい構造は既存の構造の兄弟としてインポートされます。このコマンドは、`json` という名前のルート要素を持ちJSON データソースからデータを期待するツリー構造のために使用することはできません。

メモ　構造がXML ファイルからインポートされると、ファイルはデフォルト ファイルと設定され、ファイルのデータもインポートされます。

- XML に構造をエクスポートする

  ページソースツリーの XML 構造をエクスポートする XML ファイルを選択することのできる参照ダイアログを開きます。既存の XML ファイルを選択または新しいファイルを作成することができます。既存のファイルを選択する場合は、ファイルの既存のデータがエクスポートされた構造により上書きされます。このコマンドはツリー構造のために使用することができません。このコマンドは、`json` という名前のルート要素を持ち JSON データソースからデータを期待するツリー構造のために使用することはできません。

- JSON から構造をインポートする

  ページソースツリーから XML 構造をインポートする JSON ファイルを選択することのできる参照ダイアログを開きます。ツリーに既に構造が含まれる場合、既存の構造の複数のノードを保持するか問われます。既存の構造を保持することを選択すると、新しい構造が既存の構造にマージされることができず、新しい構造は既存の構造の兄弟としてインポートされます。このコマンドは、`json` という名前のルート要素を持ち JSON データソースからデータを期待するツリー構造のためにのみ使用することができます。

メモ 構造が XML ファイルからインポートされると、ファイルはデフォルト ファイルと設定され、ファイルのデータもインポートされます。

- JSON に構造をエクスポートする

  ページソースツリーの XML 構造をエクスポートする XML ファイルを選択することのできる参照ダイアログを開きます。既存の XML ファイルを選択または新しいファイルを作成することができます。既存のファイルを選択する場合は、ファイルの既存のデータがエクスポートされた構造により上書きされます。このコマンドはツリー構造のために使用することができません。このコマンドは、`json` という名前のルート要素を持ち JSON データソースからデータを期待するツリー構造のためにのみ使用することができます。

- DB データソースを選択する [$DB のみ]

  このコマンドはデータベース(`$DB`) 型のルート ノードに対して有効化されます。このコマンドは、DB データソースに接続できる、MobileTogether Designer のデータベース接続ウィザードを開きます。 DB に接続後、データソースとして追加されるテーブルを選択することができます。新しいテーブル データが既存の構造にマージできない場合、新しいテーブル構造が既存の構造の兄弟として作成されます。

  DB が、(データソースとして) デザイン内で共有される場合、次のオプションからの選択が問われます:

  - **共有された構造の修正**: このページに追加されるページソース構造の修正は、DB 構造が使用される他のページで共有されます。

- **構造のコピー**: 構造は新しいページソースにコピーされ、ルート要素は異なる名前が与えられます。元のページソースは削除されます。新しいデータ構造はこれ以上、他のページ上の構造とは共有されません。他のページのデータソースをそのままに残し、このデータソースを修正することができます。
- **キャンセル**: 修正プロセスをキャンセルします。

DB テーブルとビューを選択する [$DB のみ]

このコマンドはデータベース($DB) 型のルート ノードに対して有効化されます。このコマンドは、DB テーブルとビューを選択して、ページソースとしてインポートする、MobileTogether Designer データベース オブジェクト選択ウィンドウを開きます。

OriginalRowSet の作成 [$DB のみ]

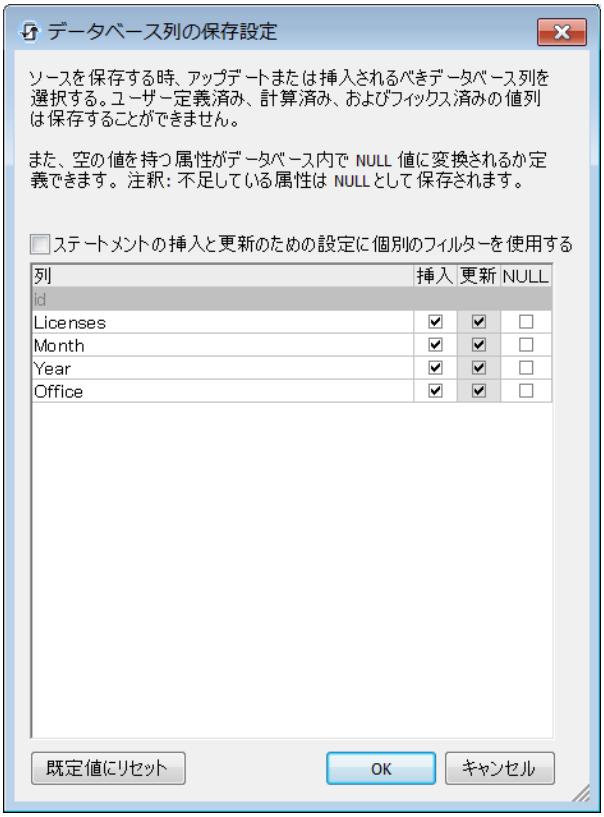
データを編集し、保存するには、ツリーはページ ソースに、RowSet 要素のコピーであるOriginalRowSet 要素を含む必要があります。オリジナルのデータはOriginalRowSet 要素に保存され、編集されたデータはRowSet 要素に保存されます。データ ソースが保存されると、OriginalRowSet とRowSet の 2 つのツリーの差異が計算され、ページソースは差異を基に更新されます。修正に成功すると、修正されたデータは、OriginalRowSet にコピーされ、OriginalRowSetが新しく保存されたDB データを含み、修正プロセスが繰り返すことができます。

ページソースのOriginalRowSet を作成するには、ページソースのルート ノードを右クリックして、コマンド **[OriginalRowSet の作成]** を切り替えます。

[OriginalRowSet の作成] コマンドは、データベース型 ($DB) ルート ノードのために有効化されています。ページソースの元のデータを含むデータ構造を作成/削除する切り替えコマンドです。ユーザーにより修正されたデータはデータ ソースにより作成されたメインの構造に保存されています。修正されたデータがDB に保存されると、OriginalRowSet 構造は新たにDB に保存されたデータを含むために修正されます。これにより元のDB データはツリーで使用可能となります。

列のフィルター [$DB のみ]

このコマンドはデータベース($DB) 型のルート ノードのために有効化されます。このコマンドは、DB データソースに保存される列を指定することのできる、データベース列の保存設定ダイアログを開きます。



ダイアログは、DB データソースのための列を表示します。その列が更新され、または、挿入された値を取ることができるかを指定することができます。(更新は、既存の行要素内の変更されたデータを指します。挿入された値は、新規に追加された行要素を指します)。デフォルトでは、各列の挿入と更新のオプションはペアとして選択されます。ですが、異なるオプションを列の挿入と更新オプションのために指定する場合、挿入慮更新ステートメントのために個別のフィルター設定を使用するチェックボックスをチェックしてください。DB のカラム内の *NULL* チェックボックスをチェックすることで、空の値を持つ属性は、内で NULL 値に変換されることができます。不足する属性は常に NULL として保存できることに注意してください。

(ユーザーにより定義されているため、固定されているため、または計算されているため) 更新することのできない列は、挿入、更新またはオプションチェックボックスを使用することができません。 上のスクリーンショットでは、固定値を持つため、ID 列は、更新されることができません。更新しない列の選択を解除します。全ての列を更新するために、保存設定をリセットする場合は、**[デフォルトにリセット]** をクリックします。

HTTP/ FTP リクエスト設定

コマンドは、HTTP/FTP 型のルートノード (つまり **$HTTP** ルートノード) のために有効化されます。HTTP/FTP、REST、または SOAP を使用して作成された現在のページソースのリクエストに応じて、適切な設定ダイアログが開かれます: Web アクセス設定の編集、RESTful API リクエスト設定、WSDL ファイルの選択。

キャッシュの設定

ルートノードとレーとコールアウトサブメニューにコードと保存の 2 つのオプションが使用できます。

- *ロードのオプション:* ページソースが最初にロードされる場合、ページソースがロードまたはロードされないように自動設定することができます。ページアクションを介して指定することができます。デフォルトは、データソースが初回使用される際にロードされる、*最初の使用時*です。
- *保存のオプション:* ページを移動する場合、ソリューションが完了した場合、最後の送信アクションが実行された場合、および自動的ではない場合、データは自動的に保存されることができます。後者の場合、保存はページアクションを介して指定することができます。デフォルトは*自動ではない*です。

ツリーノードのコンテキストメニュー

下にリストされるコマンドは**ツリー ノード** (ルート ノードを除くすべての要素と属性) のコンテキスト メニューで使用可能です。すべての種類のデータソース(XML、DB、HTML、など) で共通のコマンドに付け加え、一部のデータソースは種類 (DB データソースのコマンド) に特有なコマンドを持ちます。そのようなコマンドの仕様については、関連のある個所で説明されています。

子の挿入、追加、および追加

選択されたノードに対応した要素と属性の追加を有効化します。**[挿入]** はノードを選択されたノードの前に追加します。**[追加]** はノードをその種類の最後のノードの後に追加します。選択されたノードの直後にノードの追加を希望する場合、直後のノードに移動して **[挿入]** コマンドを使用します。

ロード時に存在することを確認 (固定値)

このコンテキスト メニュー アイテムはツリー ノードに対して使用することができます。ページのロード時に選択されたノードに固定値を与えることができます。コマンドをクリックして値を入力します。このコマンドは切り替えコマンドです。ですから、固定値が既に与えられている場合にクリックすると、値が削除されます。

ロード時に存在することを確認 (XPath 値)

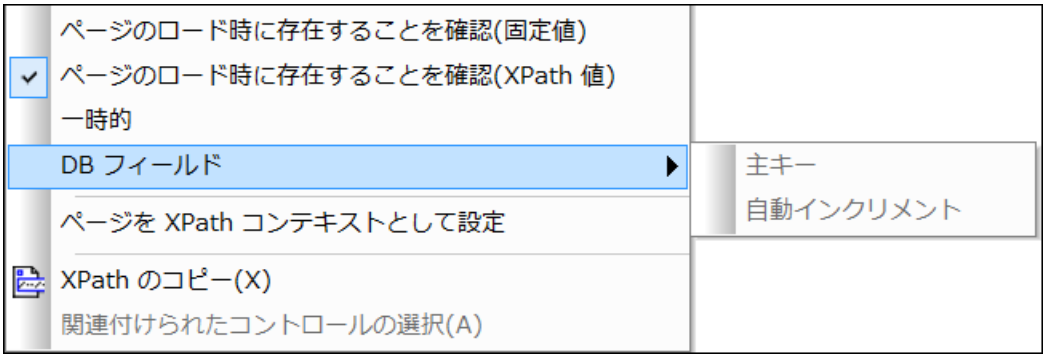
このコンテキスト メニュー アイテムはツリー ノードに対して使用することができます。 ページのロード時に選択されたノードにXPathにより生成された値を与えることができます。コマンドをクリックすると[XPath/XQuery 条件式の編集] ダイアログが表示されます。XPath 式を入力して、ノードの値を生成します。このコマンドは切り替えコマンドです。ですから、XPath 値が既に与えられている場合にクリックすると、値が削除されます。

一時的

選択されたノードを一時的なノードと設定します。ツリーが保存される場合、一時的なノードのデータは保存されません。一時的なノードは、有効なワークフローのデータのフレームワーク外であるため、一時的なノードは、計算または最後のデータの一部として使用されないデータの保管に使用されます。

DB フィールド

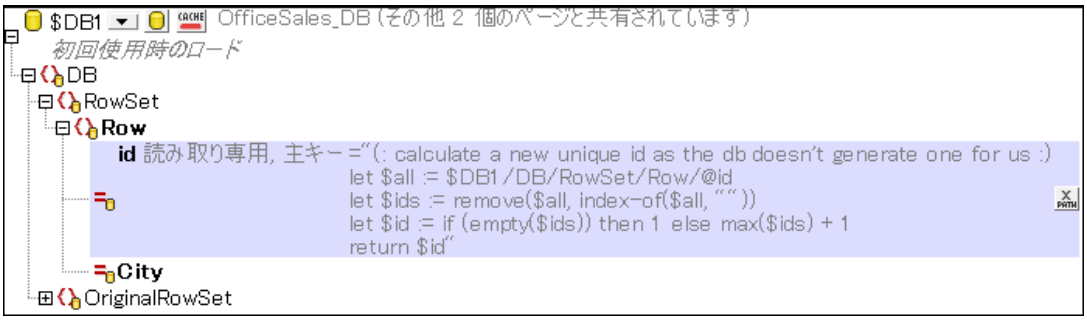
このコンテキスト メニュー アイテムは、DB ノードに対して使用することができ、2 つのコマンドを持つサブメニューを持ちます:



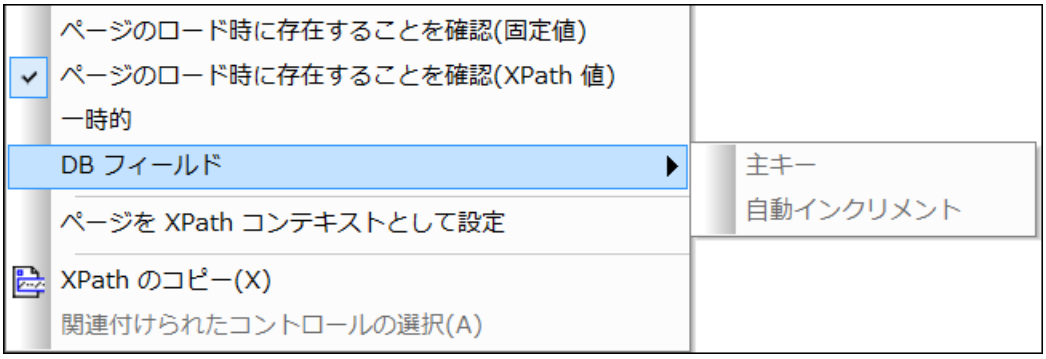
- *主キー*: 主キーがDB からの取得中に自動的に検出されていない場合、選択されたノードを主キー列として設定します。
- *自動インクリメント*: 選択されたノードを自動インクリメントに設定します。ノードはその後読み取り専用となります。

MobileTogether Designer の主キー

DB の主キーは通常、自動インクリメントです。この場合、新しい行がテーブルに追加され、追加された行の主キー列は自動的にインクリメントされます。MobileTogether Designer では、主キーがテーブルから取得されると、自動インクリメント情報が自動的に取得され[ページ ソース] ペインに表示されます(*下のスクリーンショット参照*)。



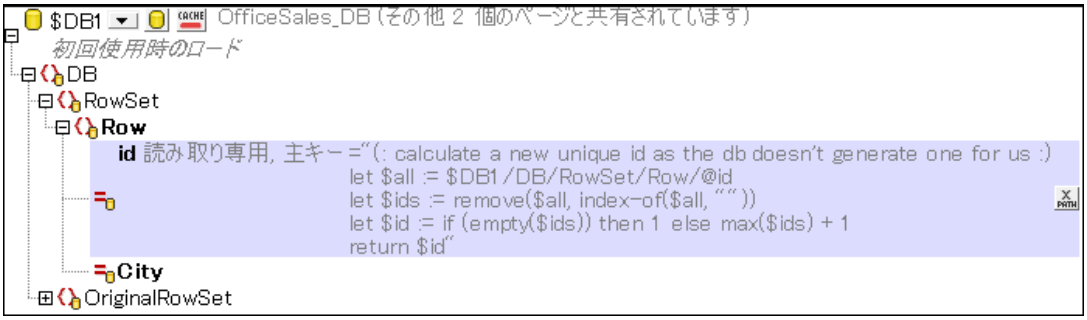
この情報の自動取得が成功しなかった場合、切り替えコマンドを含むツリー ノードのコンテキスト メニューによりノードに正確な注釈をつけることができます (*下のスクリーンショット参照*)。



主キー列が自動インクリメントでない場合、追加された行の新しい主キーの値はXQuery 式を使用して自動的に生成されなければなりません。これは、主キー列を編集することができないからです。XQuery 式は、主キーノードのコンテキストメニューコマンド**[ページのロード時に存在することを確認 (XPath 値)]** を使用して挿入されます。下の例

では、次のXQuery 式を使用して新しい値が主キー`@id` のために生成されます:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```



ページをXPath コンテキストとして設定する

選択されたノードをページのXPath コンテキスト ノードとして設定します。効果の注釈がノード(*下のスクリーンショット参照*) の下に表示されます。ページのXPath コンテキストは、ページのすべてのXPath 式のコンテキストノードです。

このコマンドは切り替えることができます。ですから、ノードをページのXPath コンテキスト ノード、または、ページのXPath コンテキスト ノードとしてのノードの設定をオフに切り替えることができます。ノードがXPath コンテキスト ノードと設定され、他のノードが既にこの設定をされている場合、設定は前に割り当てられたノードの設定をオフに切り替え、新しく割り当てられたノードの設定をオンに切り替えます。

XPath のコピー

ノードのXPath ロケーター式をクリップボードにコピーします。ロケーター式はルート ノードから開始します。例: `$XML1Products/Product` が`Product` ノードのロケーター式です。

- 関連付けられたコントロールの選択

  選択されたノードに関連付けられたデザインダイアグラムのコントロールを選択します。この様な関連は、通常、ノードと [ページ] コントロールの間のページソースリンクです。

チャプター 8

ページ

8 ページ

MobileTogether プロジェクトは、順序付けられたシーケンス内で定義される1つまたは複数のページから構成されています。MobileTogether アプリケーションがクライアントデバイスで開始されると、ワークフローはシーケンスの最初のページから開始され、最後のページまで継続されます。ワークフローシーケンスのトップレベルのページはサブページに分岐され、メインのワークフローに戻ることができます。複数のページを含むプロジェクトにより複雑なワークフローをユーザーに取り更に包括的な部分に構成することができます。プロジェクトのページは、ページペインで管理されています。ページの概要に関しては、サブセクション プロジェクトのページを参照してください。

通常ページに追加される コントロール であるページのデザインコンポーネントは、指定されたデータソースから来ます。これらのデータソースは、ページソースペイン内で定義されています。データソースが定義された後、そのページのコントロールとアクションにより使用されることができます。これらの「ページデータソース」がどのように使用されるかに関しては、ページのデータソースを参照してください。

名前や背景色などのページのアスペクトを定義するページプロパティは多種存在します。これらのプロパティは サブセクション ページプロパティ で リストされ説明されています。

ページの重要な機能は、ページに関連したイベント がトリガーされると、アクション を実行する機能です。例えば、ページがロードされる都度、データソースが更新されるなどです。異なるページイベントとその設定の詳細は、サブセクション ページイベント で説明されています。

このセクション

このセクションは以下サブセクションに整理されています:

- プロジェクトのページ
- ページのデータソース
- ページプロパティ
- ページイベント

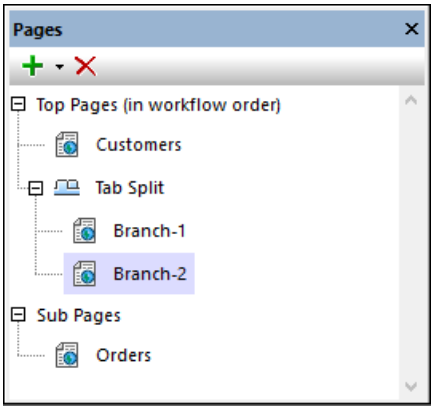
(C) 2018 Altova GmbH

8.1 プロジェクトのページ

MobileTogether プロジェクトは、順序の付けられたシーケンス内で定義されている1つまたは複数のページにより構成されています。MobileTogether アプリケーションがクライアントデバイスで開始されると、ワークフローは、シーケンスの最初のページから開始し、シーケンスの最後のページまで継続されます。ワークフローシーケンスのトップレベルのページは、サブページに分岐し、メインのワークフローに戻ります。複数のページを含むプロジェクトは、ですから、複雑なワークフローをユーザーが理解しやすい方法で部分的に構成することができます。

ページペイン: プロジェクトのページの管理

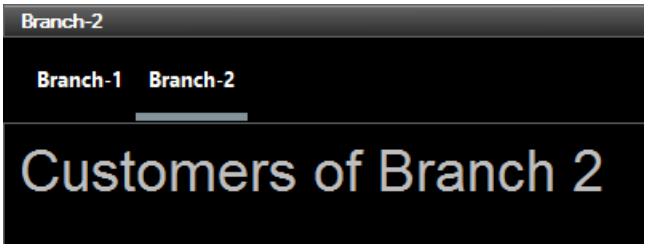
プロジェクトのページは、ページペインで管理します（*下のスクリーンショットを参照*）。プロジェクトに3つの種類のページを追加することができます: (i) トップページ (ii) タブ付けされたページ（タブの分割）、と(iii) サブページ。これはメニューバーの**追加** アイコンを、または、各ページのコンテキストメニューを使用して行います（コンテキストメニューを確認するにはページを右クリックしてください）。ページペイン内でページがリストされる順番は、プロジェクトのシーケンスです。ソリューションがクライアント上で実行されていると、**送信** ボタンをクリックすると、フローは次のページに移動します。プロジェクトのページのシーケンスを変更するには、他のページに相対して、ページを新しい場所にドラッグします。



ページの管理に関しては、ページペインの詳細を参照してください。

タブ付けされたページ（タブの分割）

プロジェクトのページのシーケンスにタブ付けされたページを追加するには、シーケンスにタブの分割を追加します（*上のスクリーンショットを参照*）。タブの分割のコンテキストメニューを使用して、タブの分割に子ページを追加します。例えば、上のスクリーンショットでは、タブの分割には*Branch-1* と*Branch-2* と名前の付けられた2つの子ページが含まれます。下のスクリーンショットで示されるように、デザイン内のタブの分割の子ページはソリューションにのタブ付けされたページになります。下のスクリーンショット内で表示されているように、ソリューション内でタブ付けされたページのタブはソリューション内では、ユーザーはタブを選択して、対応するページを確認することができます。タブ付けされたページの**送信** ボタンをクリックすると、ワークフローはシーケンスの次のページに移動します。ページの管理に関する詳細は、ページペインを参照してください。

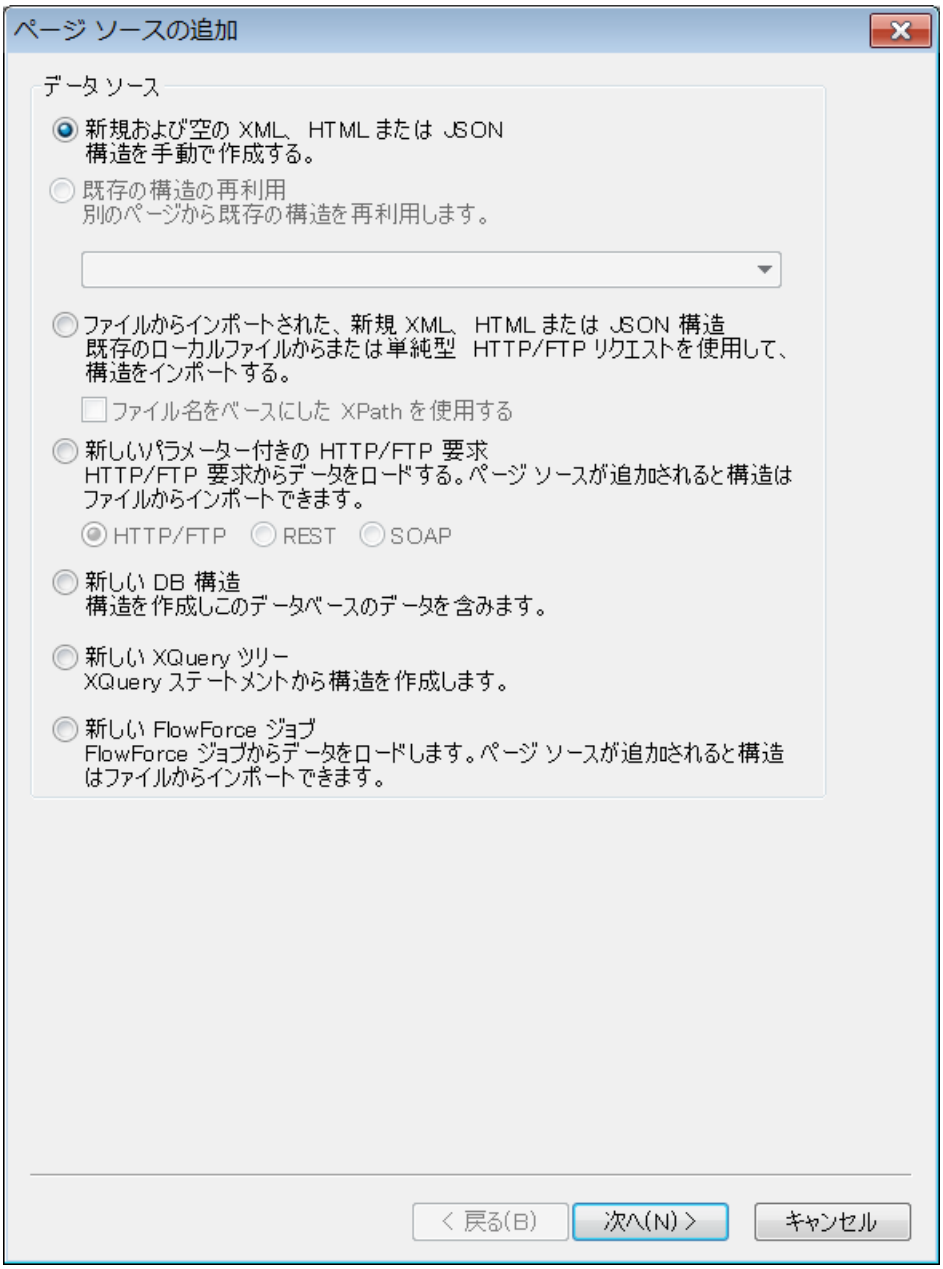


サブページ

ページペインのメニューバー内の**追加** アイコンを使用して、または、各ページのコンテキストメニューを使用してサブページを追加します (コンテキストメニューを確認するためにページを右クリックします)。サブページへの移動 アクション (例えば、ボタンコントロール) を使用してトップページからサブページへフローを移動することもできます。トップページに移動するには、ユーザー MobileTogether Client アプリの**戻る**ボタンをクリックすることができます。または、ページへの移動 アクションによりデザイン内のページに移動することを定義できます。ページの管理に関する詳細は、ページペインを参照してください。

8.2 ページのデータソース

デザインの全てのページには、ページソースペイン内で管理されるデータソースのセットが存在します。データソースを追加するには、ペインの **ソースの追加** ツールバーアイコンをクリックします。これは、ページソースの追加 ダイアログを表示します (*下のスクリーンショットを参照*)。ここで追加するデータソースの型を選択します。既存のページで使用されている同じデータソースを追加するには、既存の構成オプションから再利用を選択して、コンボボックス内に表示されている既存のデータソースから選択します。



ページソースの追加ダイアログの2番目のページにより以下のオプションを設定することができます: (i) データがデータソースからロードされた時、またはデータソースに保存された時、データソースを編集することができるか、そして、データがクライアント、または、サーバーに保管されるか。データソースオプションの詳細に関しては、ページソースオプションを参照してください。

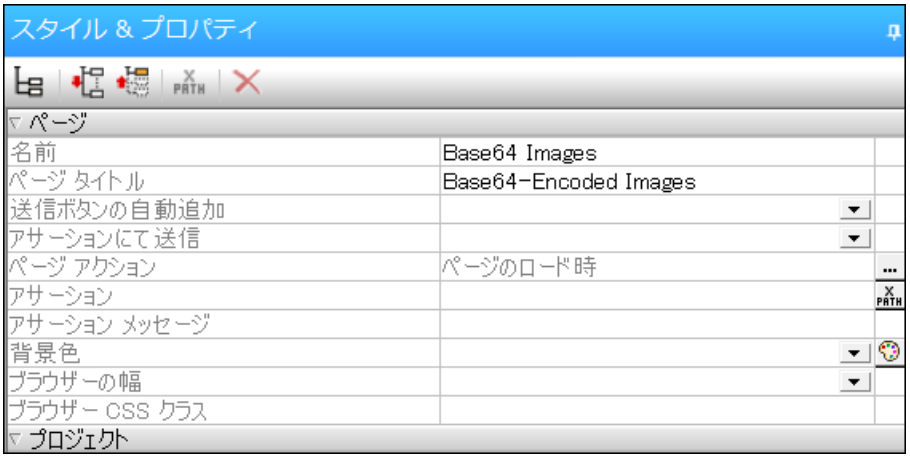
データソースとそのノードのための設定の多くは、個別に設定することができます。異なるノードのコンテキストメニュー コマンド

の完全なリストに関しては、データソースの コンテキストメニュー を参照してください。

データソースに関する詳細は、ページソースペイン とデータソースを参照してください。

8.3 ページ プロパティ

[スタイル & プロパティ] ペインでプロジェクトのプロパティは定義され、以下に説明されています。下のスクリーンショットはデフォルトの値を表示しています。



- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。ページタイトル プロパティ(下を参照) が指定されていない場合、ソリューションのページのタイトル名を指します。値フィールドの内側をダブルクリックして編集します .

- ページタイトルバーの表示

  ページのタイトルバーの設定が表示 (`true`) または非表示 (`false`) に設定されるかを指定します。デフォルトの値は `true` です。このプロパティは、タイトルバー全体の存在に関しての表示です。タイトルバーの内容に関してではなく、タイトルバーの内容は**ページタイトル**プロパティで指定されます。

- ページタイトル

  ソリューションのページのタイトルです。値フィールドの内部をクリックして、名前を入力します。または、ペインのツールバー内の XPath アイコンをクリックして XPath 式を入力します。このプロパティの値が存在しない場合は、名前 プロパティの値がソリューション内のページのタイトルとして使用されます。

- 自動追加送信ボタン

  ブール値の設定で、送信 ボタンが自動的にページに追加されるかを定義します。`true` または `false` をコンボボックスから選択します。デフォルトの値は `true` です。(ソリューション内のページの送信 ボタンは、通常、ページの右上にあり、アクションのためにページのデータを送信します。通常、ワークフローは次のページに移動します。)

- アサーション時に送信

  ページに無効なアサーションが存在するとページの送信を許可または許可しません。以下から値を選択してください:

- 無効: **送信** ボタンは、ページに無効なアサーションが存在する場合、無効化されます。デフォルトの設定です。
- 有効: **送信** ボタンは、ページに無効なアサーションが存在しても、有効化されます。
- 確認: **送信** ボタンは、ページに無効なアサーションが存在しても有効化されます。しかし、無効なアサーションが存在する場合し、送信ボタンがクリックされると、エンドユーザーに対して、送信を実行するか確認するダイアログが表示されます。

デフォルトは無効です。

ページアクション

プロパティの追加アクションダイアログ ボタンをクリックし、ページイベントを選択し、ページイベントがトリガーされた際に実行されるアクションを定義するページアクションダイアログを表示します。詳細はページイベントとアクションのセクションを参照してください。アクションのため定義された、ページイベントはプロパティの値フィールドにリストされています。

オーディオの録音アクション

オーディオの録音イベントは各ページで定義されます。次の2つのイベントを使用することができます: **オーディオの録音時のエラー** と**オーディオの録音の完了時** 。これらのイベントのために定義されるアクションは、**このページのすべてのオーディオの録音に適用されます**。プロパティの**追加ダイアログ**ボタンをクリックすることにより、現在のページのオーディオの録音イベントの定義を含むダイアログが表示されます。それぞれのイベントのために、左側のアクションペインからイベントタブにアクションをドラッグして実行されるアクションを定義します。オーディオの録音イベントダイアログはデザイン内を右クリックして、**オーディオの録音ページ**を選択することによりアクセスすることができます。 *詳細に関しては、 オーディオの録音の機能についての説明を参照してください。*

アサーション

コントロールのアサーションを有効にするための条件を設定します。アサーションが無効な場合、アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールにより表示されます。(複数のアサーション メッセージ コントロールがある場合、すべてのコントロールは`Assertion Message` プロパティのテキストを表示します。)

`Assertion` プロパティのXPath アイコンをクリックしてアサーションを定義するXPath 式を入力します。例えばXPath 式 **`LastName != ""`** がノード`LastName` は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。

アサーションは複数のコントロールに対して定義することができます。ですから、複数のアサーションメッセージが(アサーションが表示されるページのシーケンス内の)最初の無効なアサーションで表示されます。コントロールアサーションはページがアサートする前に評価され、コントロールアサーションはデザイン内に発生する順序で評価されます。

アサーションメッセージ

コントロールのアサーションが有効でない場合、アサーションメッセージが表示されるように設定します。プロパティの値フィールドの内側をダブルクリックしてアサーションメッセージを編集します。または、XPath ツールバーボタンをクリックして、

XPath 式を入力し必要なテキストを生成します。アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールにより表示されます。アサーションの条件はXPath 式により指定されます。例えば XPath 式 `LastName != ""`がノード`LastName` は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。

アサーションは複数のコントロールに対して定義することができます。ですから、複数のアサーションメッセージが(アサーションが表示されるページのシーケンス内の) 最初の無効なアサーションで表示されます。コントロールアサーションはページがアサートする前に評価され、コントロールアサーションはデザイン内に発生する順序で評価されます。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボ ボックスのドロップダウン リストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、XPath 式を入力し、(色のコード) テキストを生成します。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

ブラウザーの幅

ブラウザーの幅を画面の幅または絶対ピクセルの値で指定します。ドロップダウンリストから値を選択します。デフォルトの値はブラウザで自動的に計算されます。

ブラウザー CSS クラス

このページと関連付けるCSS クラスの名前を入力します。このクラスは(プロジェクト プロパティで指定されている) CSS ファイルでこのコントロールに対して、個別にプロパティを割り当てるために使用することができます。

8.4 ページイベント

MobileTogether デザイン内の各ページには、関連する特定の定義済みの**イベント**があります:

- ページのロード時: ページがロードされると実行される一連のアクションを定義することができます。
- ページの更新時: ページが更新されると以下について定義することができます: (i) ページが再度開かれたとき (ii) 間隔タイマーの最中、(iii) **[更新]** ボタンがユーザーにより手動でタップ/クリックされた時、または (iOS では) 更新するためにプルダウンされた時。これらのメソッドのそれぞれに対してアクションを定義することができ、1つまたは全てを使用することができます。
- 戻るボタンのクリック時: ソリューションの**[戻る]** ボタンがタップ/クリックされると、実行されるアクションです。
- 送信ボタンのクリック時: ソリューションの**[送信]** ボタンがタップ/クリックされると、実行されるアクションです。
- サーバーへの接続エラー時: サーバーに接続できないときに実行されるアクションです。エラーは、初期の接続の確立中、または、その後接続が失われたときに起る可能性があります。`MT_ServerConnectionErrorLocation` 変数を使用してこのようなエラーをデバッグします。このイベントの使用方法の概要に関しては、サーバー接続エラーのセクションを参照してください。
- 埋め込まれたメッセージ: Webpage 内に埋め込まれたIFrame からサーバー上のワークフローにポストされたメッセージを生成したアクションです。この機能に関する詳細は、埋め込まれたWeb ページソリューションを参照してください。

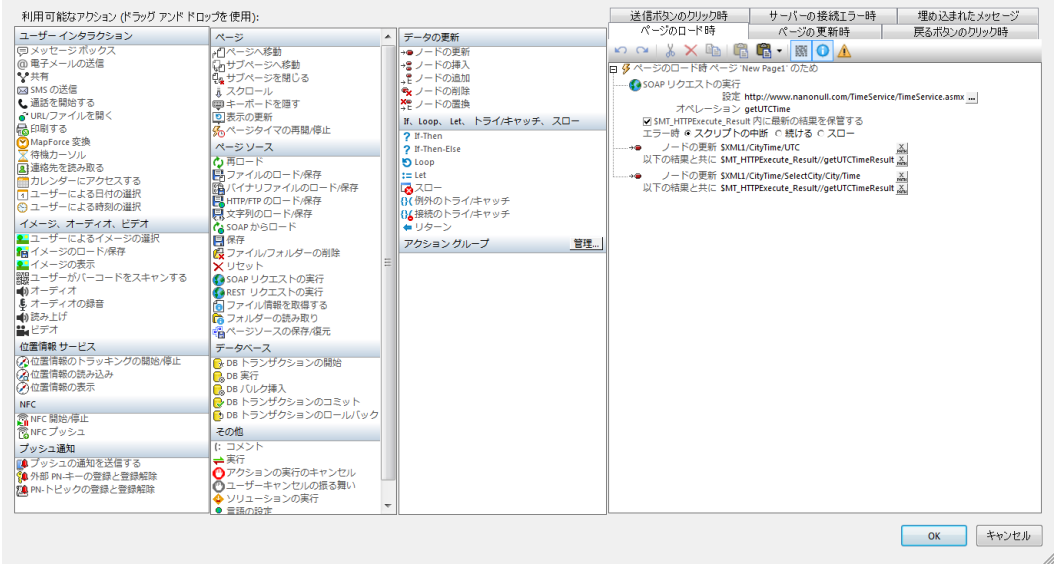
これらのイベントは (コントロール イベントとは対照的に) **[ページイベント]** と呼ばれ、各アクションには1つ以上の**アクション (ページ アクション)** があります。

ページ イベントがトリガーされると実行されるアクションの定義

ページイベントのアクションを定義するには、ページアクションダイアログ (下のスクリーンショット) にアクセスしてください。イベントのタブを選択して、アクションを定義します。

ページアクションダイアログ (下のスクリーンショット) に以下の方法でアクセスすることができます:

- **[ページ] [ページ アクション]** をクリックします
- ページ内を右クリックして、**[ページ アクション]** を選択します
- **[スタイル & プロパティ] [ペイン]** 内のページセクションに移動します。ページ アクション プロパティの**[追加ダイアログ]** ボタンをクリックします。
- **[ページ] [アクションの概要]** をクリックしてアクションの概要ダイアログを表示します。定義したいページイベントの**[追加ダイアログ]** をクリックします。



ページ アクションの定義

ページイベントのアクションを定義するには、左側のペインからアクションをドラッグして、右側のペインのイベントタブにドロップしてください。ページ アクション ダイアログについての詳細および異なる種類のページ アクションに関しては、アクションのセクションを参照してください。

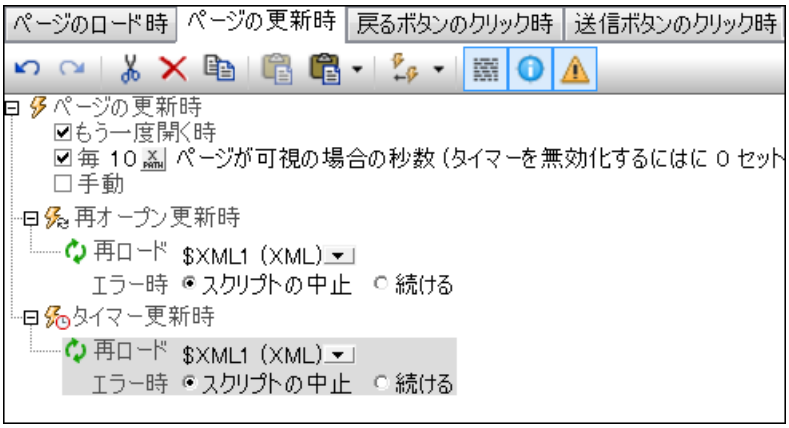
ページのロード時

ページのロード時 イベントは、(トップページとサブページの)全てのページで使用することができます。アクション ページのロード時 イベントのために定義されたアクションは、ページがロードされた時に実行されます。 これらのアクションは、データの値の初期化、また表示内でコンポーネントセットアップする際に役に立ちます。

ページの更新時

ページの更新時 イベントは全てのページで使用することができます。以下の場合には、1回または複数回に渡り発生するように定義することができます (*下のスクリーンショット参照*)。

- ページがもう一度開かれた時 (一時停止した状態から再開した時)。
- ページが最初にロードされた時にタイマーにより決められた間隔に到達した時。間隔は、静的または動的な値であることができ、タイマーは他の (ページまたはコントロール) イベントがトリガーされた時停止また再開することができます。
- ユーザーが手動で、ページの上部分にある **[更新]** ボタンをタップまたはクリックする時、または (iOS で) プルダウンして更新する時。



オプションを選択すると、そのオプションのためのノードがツリー内に表示されます (*タイマーの更新時*、*手動の更新時*)。対応するノードでこれらのオプションのために1つまたは複数のアクションを定義することができます。上のスクリーンショットでは、ページは、2つのシチュエーションでの更新のために設定されています: (i) ページが開かれる都度、**そして** (ii) 10 秒毎。両方の場合、同じ再ロードアクションが定義されています。同じまたは異なるアクションを各ノードのために指定することができます。

もう一度開く時の更新
このオプションのために定義されているアクションはページがもう一度開かれると実行されます。また、一時停止され (背景で実行されていた) ソリューションがもう一度開かれると実行されます。プロジェクトプロパティ*他のソリューションへの切り替え時*と*ソリューションの実行* アクションを参照してください。

タイマーの更新時

- タイマー間隔は、XPath 式 を使用して選択されます。値は数値である必要があります。更新の間隔は秒単位です。 (許可される制度はミリ秒です。ですから、`1.002` の値は許可されており、更新間隔を1 秒と2 ミリ秒に設定します)。デフォルトは、`10` (秒) の静的の値です。動的な値も設定することができます (例えば、データソースノードからの数値、または計算から生成された値など)。
- タイマーは、ページがロードされると開始されます。*タイマーの更新時* アクションは、開始時からの間隔で実行されます。更新間隔を変更すると、タイマーは再起動される必要があります。これは、更新間隔を変更するイベントへ*ページタイマーの再開* アクションを追加することで行うことができます。*SOAP リクエストチュートリアル* をサンプルとして参照してください。
- 更新アクションは、指定された間隔で、タイマーが実行されている限り継続されます。タイマーを停止するには、適切なイベントに*ページタイマーの停止* アクションを追加してください。

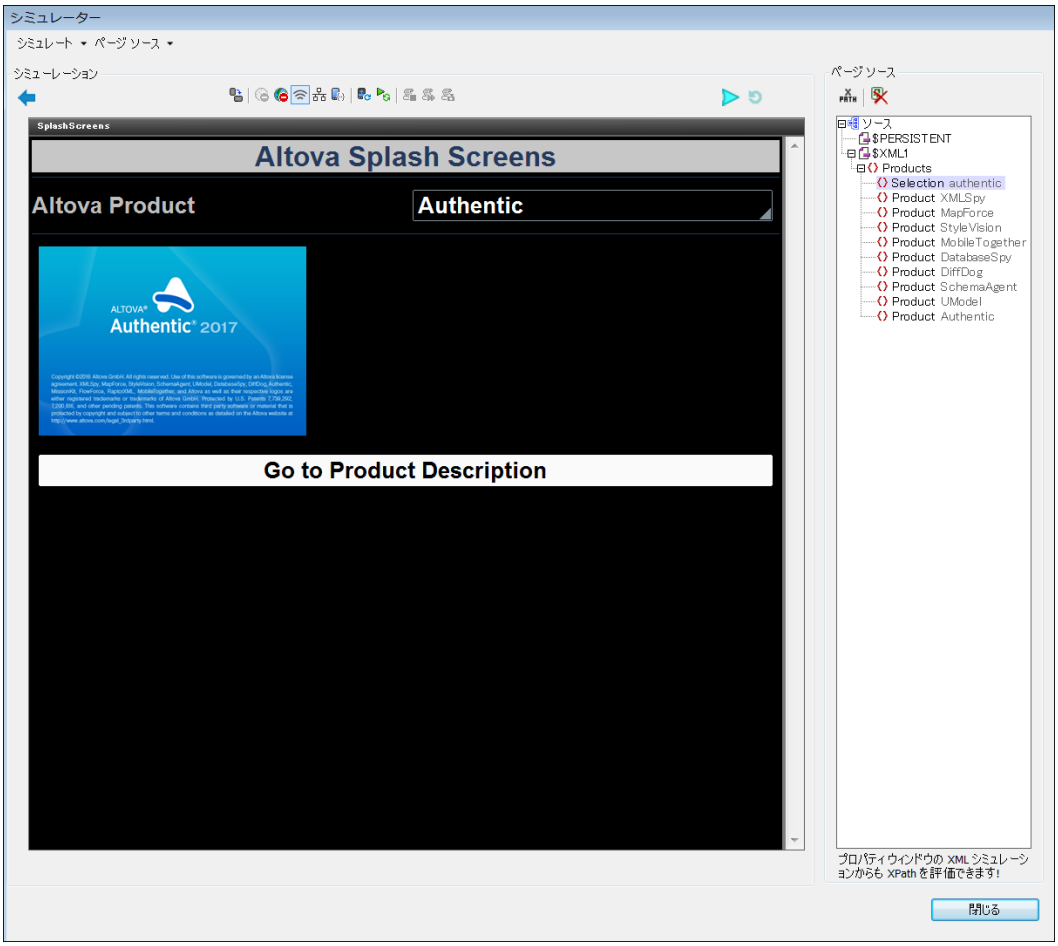
手動の更新
このオプションが選択されている場合、ページは **[更新]** *ボタンを表示します。ユーザーがこのボタンをタップ/クリックすると、または (iOS で) プルダウンして更新する時、手動での更新アクションが実行されます。 SOAP リクエストチュートリアル をサンプルの参照としてください。*

ページ更新のサンプルに関しては、SOAP リクエストチュートリアル を参照してください。

ページの更新のシミュレーション

シミュレーター内では、(*下のスクリーンショット*) ページの更新を次の方法で影響することができます:

- ページの更新ページがもう一度開かれるときに定義されている場合、**[シミュレーション再開 停止]** ボタンが有効化されます。ページがもう一度開かれるとシミュレーションを再開するためにクリックします。
- 時間をベースにした更新が定義されていると、**[タイマーの終了]** ボタンが有効化されます。シミュレーションが開始すると、ページが自動的に x 秒毎に更新されます。 x は更新の間隔です。**[タイマーの終了]** ボタンをクリックして、タイマーを停止することができます。これは、頻繁にページを更新することなくシミュレーションの結果を確認する場合に役に立ちます。タイマーが停止されると、ボタンはタイマーを再開することのできる **[タイマーの開始]** に切り替えられます。
- 手動の更新が定義されると、**[更新]** ボタンを使用することができます。手動で更新オプションのために指定されたアクションを実行するためにこのボタンをクリックします。



戻るボタンのクリック時

戻るボタンのクリック時 イベントは、全てのページ(トップページとサブページ)で使用することができます。ソリューションの表示とシミュレーションでは、**[戻る]** ボタンはスクリーンの左上にあります。

[戻る] ボタンのデフォルトの振る舞いは、以下のページの種類により異なります:

- *トップページ:* ソリューションの終了を問うプロンプトが表示されます。**[はい]** をクリックして終了、**[いいえ]** をクリックしてキャンセルします。
- *サブページ:* 表示は、通常サブページをロードするトップレベルのページである、前のページに切り替えられます。

上記のデフォルトの振る舞いをオーバーライドする、このイベントのためのアクションのセットを定義します。ユーザーが**[戻る]** ボタンをタップ/クリックすると、定義されたイベントアクションが実行されます。

送信ボタンのクリック時

トップページの自動追加送信ボタンプロパティがページプロパティ設定内でデフォルトの値である`true`である場合、**[送信]** ボタンは、すべてのトップページの右上に表示されます。

[送信] ボタンのデフォルトの振る舞いは、ページシーケンス内のページの位置により異なります。

- ページが、ページシーケンスの最後のページの場合、[送信] ボタンをタップ/クリックするとワークフローを終了します。
- ページが、ページシーケンスの最後のページではない場合、ワークフローは次のページに移動します。

ページの **[送信]** ボタンがタップ/クリックされた場合他のアクションを指定するには、ページの送信ボタンのクリック時 イベントに適切なアクションを追加します。

サーバーへの接続エラー時

サーバーへの接続エラーは、初期の接続を確立する途中に起こる場合、また、接続が後に失われた場合に起こります。 このようなエラーの発生時に取られるイベントで以下を定義することができます:

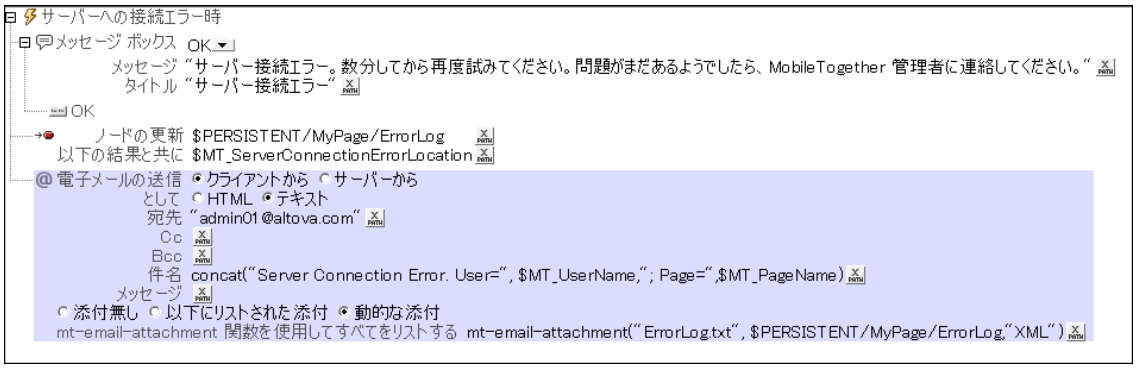
- エンドユーザーへの適切なエラーメッセージ(クライアントデバイス)、および
- その後に取られるアクション

接続エラーをシミュレーター内でシミュレートすることができます。

メモ サーバー接続エラーが発生すると、次のアクションの最初のアクションがトリガーされます: (i) サーバー接続エラーのトライ&キャッチ アクション、(ii) **サーバー接続エラー** イベント(このイベント)のためのアクション、(iii) エラーについての MobileTogether メッセージがトリガーされ、ワークフローが再開します。

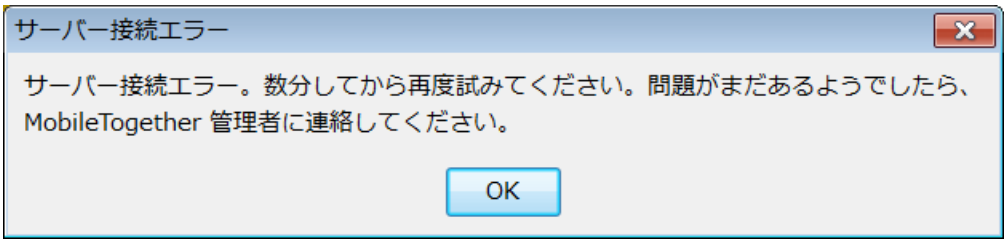
接続エラーが発生した場合にとられる対応を定義する

サーバー接続エラーが発生した時に実行されるアクションを定義することができます。これらのアクションは、それぞれのページのためにページイベント サーバーへの接続エラー時のタブ内で定義することができます。定義されるアクションは通常、エンドユーザーに送信されるメッセージおよび、ワークフローのための次の手順が含まれます。下のスクリーンショットは、実行されるアクションのシーケンスが表示されています。



上のスクリーンショットは、実行される3つのアクションのシーケンスを定義しています:

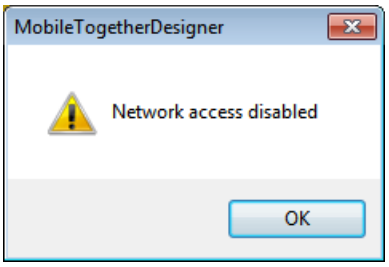
1. クライアントにエラーメッセージを送信します(下のスクリーンショット)。



2. `MT_ServerConnectionErrorLocation` 変数を使用して、サーバーへの接続エラー時 ページイベントをトリガーするアクションスタックを保存します。(変数はデバッグのために使用されます。詳細に関しては、`MT_ServerConnectionErrorLocation` を参照してください)。または、`MT_ServerConnectionErrorLocation` 変数の代わりに、 ノードの更新 アクションを使用して、エラーコードをこの目的のために作成されたノードに書き込むことができます。
3. 管理者に、(クライアントから)エラーメッセージを添付した電子メールを送信します。

メモ サーバーへの接続エラー時 ページイベント のタブ内でアクションが定義されていない場合、モバイルデバイスベジェネリッ

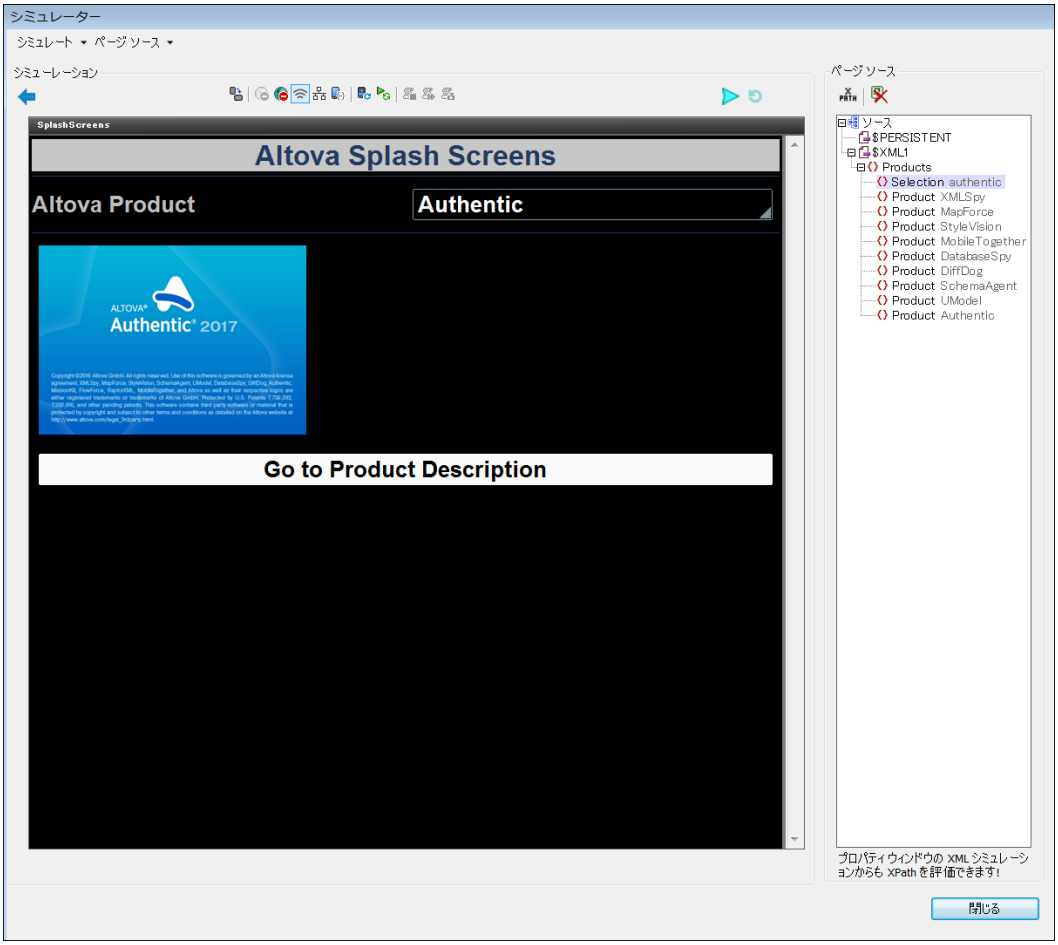
クなネットワークアクセスの無効化のメッセージが送信されます (*下のスクリーンショット*):



(テストのための) サーバー接続エラーのシミュレーション

ソリューションの振る舞いをテストするために MobileTogether Designer 内のシミュレーション と サーバー上でのシミュレーション を使用することができます。これを以下の通りおこなってください!

1. (例えば、**[F5]** を使用して) シミュレーションを開始します 。シミュレーターが開始されます (*下のスクリーンショット*)。



2. **[サーバーアクセスを無効化する]** をクリックします。サーバーへのアクセスが無効化され、ボタンは **[サーバーへのアクセスの有効化]** に切り替えられます。
3. サーバー接続を呼び出すアクションを実行します。アクセスが無効化されたため サーバーへの接続エラー時 ページイベ

ントで定義されたアクションがトリガーされます。

4. 再度サーバーへのアクセスを有効化するには、シミュレーター内の **[サーバーアクセスを有効化する]** をクリックします。

埋め込まれたメッセージ

埋め込まれたメッセージ は、Web ページに埋め込まれたIFrame からワークフローにポストされるメッセージです。ワークフローはシリアル化されたJSON 文字列を受け取ることを期待します。

埋め込まれたメッセージイベントは以下のとおりです:

- *デザイン時*: 埋め込まれたメッセージ イベントのためにアクションが定義されると、`$MT_EMBEDDEDMESSAGE` と呼ばれるJSON ページソースが`json` という名前のルート要素と共に作成されます。追加ノードを手動でページソースに追加することもできます。XPath 式を使用してこのページソースのノードにデザインコンポーネントがアクセスできるようになります。**メモ** (アクションを追加することによる) 埋め込まれたメッセージ イベントの有効化は、デザイン時に`$MT_EMBEDDEDMESSAGE` ページソースを作成する2つの方法の1つです。
- *ランタイム*: 埋め込まれたメッセージイベントは`json` という名前のルート要素を持つ`$MT_EMBEDDEDMESSAGE` という名前のJSON ページソースを作成します。ページソースの構造とコンテンツは、埋め込まれてメッセージから来ます。このページソースはデザイン時で作成されたページソースと同じ構造を持たず、デザインコンポーネントの式によりページソースにランタイムでアクセスすることができません。

メモ 埋め込まれたメッセージイベントはページイベントであるため、デザイン内の各ページは埋め込まれてメッセージイベントのために定義されたアクションを持つことができます。Web ページからメッセージが送信されると、(特定のページを指定することなく) ワークフロー全体を送信します。現在アクティブなページの埋め込まれたメッセージイベントがトリガーされます。

デザインタイム

埋め込まれたメッセージのイベントの処理のために少なくとも1つのアクションが定義されている場合、`$MT_EMBEDDEDMESSAGE` JSON ページソースは自動的に作成されます。ページソースの作成以外に他のアクションを実行しない場合、ワークフローの動作に影響しないアクションを追加してください。例えば、コメント アクションなど。アクションを追加しない場合、`$MT_EMBEDDEDMESSAGE` ページソースは作成されません。定義されたアクションをの後に削除する場合、`$MT_EMBEDDEDMESSAGE` ページソースも削除されます。

`$MT_EMBEDDEDMESSAGE` ページソースが作成されると、`json` と呼ばれるルート要素のみを有します。ページソースペインのツールバー内のコマンドを使用してページソースにJSON 構造を追加することができます。デザインコンポーネントがXPath 式を使用してこれらのノードにアクセスできるようにするために構造は必要とされます。

`$MT_EMBEDDEDMESSAGE` ページソースのデザインタイムとランタイムの構造は一致する必要があります。それ以外の場合、デザインコンポーネント内のXPath 式はランタイムページソースノードを検索できなくなります。

ランタイム

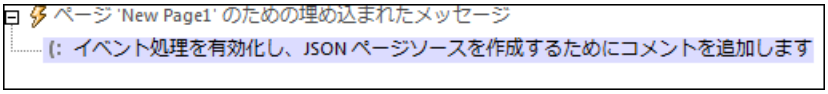
ランタイムでは、JSON 文字列の書式で埋め込まれたメッセージが受信された場合、埋め込まれたメッセージ イベントは`$MT_EMBEDDEDMESSAGE` ページソースを作成します。(それ以外の場合、このページソースは作成されません) `json` と呼ばれるルート要素と共にページソースが作成され、メッセージ内に含まれている構造とデータを持つように作成されます。追加アクションがイベントのために定義されている場合、これらは実行されます。

ソリューション内で埋め込まれたメッセージを使用する

埋め込まれたメッセージイベントは、受信するメッセージがJSON 文字列であることを期待し、この文字列をパースしJSON ページソースである`$MT_EMBEDDEDMESSAGE` ページソースを生成します。

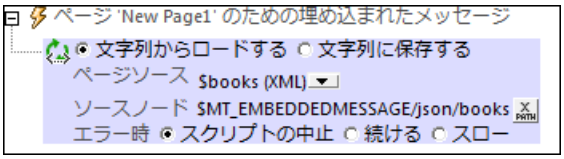
Web ページ内のソースデータのフォーマットにより異なるシナリオが発生します:

- *JSON フォーマット:* JSON ページソース **$MT_EMBEDDEDMESSAGE** はソリューション内で直接使用することができ、このページソースからのデータはWeb ページにパスされます。この場合、(コメント アクションなど) 割り込まないアクションを埋め込まれたメッセージのために定義することができます (*下のスクリーンショットを参照*)。これはイベントの処理は有効化し、**$MT_EMBEDDEDMESSAGE** ページソースの作成条件を満たします。



JSON データとの作業方法に関しての詳細は、JSON データの送信 受信 を参照してください。

- *XML フォーマット:* 埋め込まれたメッセージにより自動的に生成されるJSON ページソース **$MT_EMBEDDEDMESSAGE** に加え、XML ページソースも作成されます。これにより Web ページソースの XML フォーマットに変換し易いフォームでソリューションがデータを送信することができます。**$MT_EMBEDDEDMESSAGE** 内のデータからXML ページソースを作成するには、文字列からロード アクションを使用します (*下のスクリーンショットを参照*)。これが上手く作動するように、**$MT_EMBEDDEDMESSAGE** 内で選択されるノードは、XML として解析することのできる文字列を含んでいる必要があります。



XML データとの作業方法に関しての詳細は、XML データの送信 受信 を参照してください。

チャプター 9

コントロールとコントロールイベント

9 コントロールとコントロールイベント

機能と**外観**などのページのデザインを決定するページ内に表示されるです。

例えば、(i) ページタイトルと(ii) Web サイトにリンクされるボタンから構成される簡単なページを作成する場合、2つのコントロールをそのページに追加します): (i) ページタイトルを表示するラベルコントロール、と(ii) Web サイトにリンクするボタンコントロール(*下のスクリーンショットを参照*)。Web ページはエンドユーザーがボタンをクリックすると開かれます。クリックが Web サイトへリンクされるボタンアクションをトリガーします。このように、ボタンコントロールは**機能**を提供します。同時に2つのコントロール(ラベルとボタン)はコントロールのための広範囲なプロパティを指定することによりスタイル付けすることが可能です。これらのプロパティは、色、次元、および、エンドユーザーとのインタラクションを有効化、または、無効化する設定などを含みます。この意味でのスタイル付けは、コントロールの**外観**だけではなく、ページ全体の外観のスタイル付けも行うことができます。下のスクリーンショットは、同じコントロールが2つの異なるプロパティによりスタイルされている例を示しています。

コントロールの使用方法

MobileTogether 内で使用することのできるコントロールすべては、コントロールペイン内で表示されています。コントロールをデザインに追加するには、コントロールペインからドラッグして、ページデザイン内の希望する箇所にドロップします。以下を行うことができます:

- (ページソースペイン内で使用することのできる) **データソースにコントロールをリンクする**。データソース内のノードからのデータがデザイン内で表示、および、処理することができます。通常、コントロールとノード間の関連付けは(ページソースペイン内の) ソースツリーからデータノードをドラッグしコントロールにドロップすることで定義することができます。このような関連付けはコントロールの**ページソースリンク**と呼ばれます。
- *コントロールアクションの設定*: 1つまたは複数のコントロールアクションを追加して、(例えば、ボタンをクリックするなど) コントロールに関連したイベントがトリガーされると実行することができます。コントロールを右クリックして、コントロールアクションを定義し、**コントロールアクション**を選択します。使用することのできるアクションに関してはアクションのセクションで説明されています。
- *コントロールのプロパティ*: コントロールのプロパティは (ページ上のポジションを含む) コントロールの外観を定義します。コントロールのプロパティを設定するには、コントロールを選択し、[スタイル & プロパティ]ペイン内でプロパティを設定します。コントロールのプロパティはコントロールのセクションで説明されています。ページ上のコントロールのプロパティがページの外観を決定することに注意してください。 .

このセクション

このセクションは以下のように整理されています:

- コントロールは、各コントロールを個別に説明しています: 関数、プロパティ、および、イベント (例えば、ボタンコントロールには1つのイベント、ボタンのクリック時イベント、が存在します)
- コントロールイベントは、1つのテーブル内に異なるコントロールのイベントをリストしています。

9.1 コントロール

ページデザイン(*下のスクリーンショット*) は、コンボボックス、テーブル、イメージなどのページコントロールにより構成されていて、エンドユーザーが閲覧する通りにフォーマットされ列挙されています。コントロールは、[コントロール] ペインからデザインにドラッグされます。デザイン内にコントロールが配置された後、以下を行うことができます:

- ソースをデザイン内で表示し処理するためにコントロールを(ページソースペイン内で使用することのできる)データソースにリンクすることができます。通常、コントロールとデータソースノードは、(ページソースペインで)ソースツリーからデータノードをソースツリーからコントロールにドラッグして定義されます。この型のデータの関連付けは、コントロールの**ページソースリンク**と呼ばれます。
- *コントロールアクションの設定*: コントロールアクションは、(例えば、ボタンがクリックされた時など)コントロールに関連したイベントがトリガーされた際に実行される機能を決定します。コントロールを右クリックし、**コントロールアクション**を選択してコントロールアクションを定義します。コントロールアクションは、アクションのセクションで説明されています。
- *コントロールのプロパティ*: コントロールのプロパティは、(位置も含む)コントロールの外観を定義します。この結果、コントロールのプロパティは、ページの外観も定義します。コントロールのプロパティを設定するには、コントロールを選択して、[スタイル & プロパティ]ペイン内でプロパティを選択します。各コントロールのプロパティはこのセクションのサブセクションで説明されています。

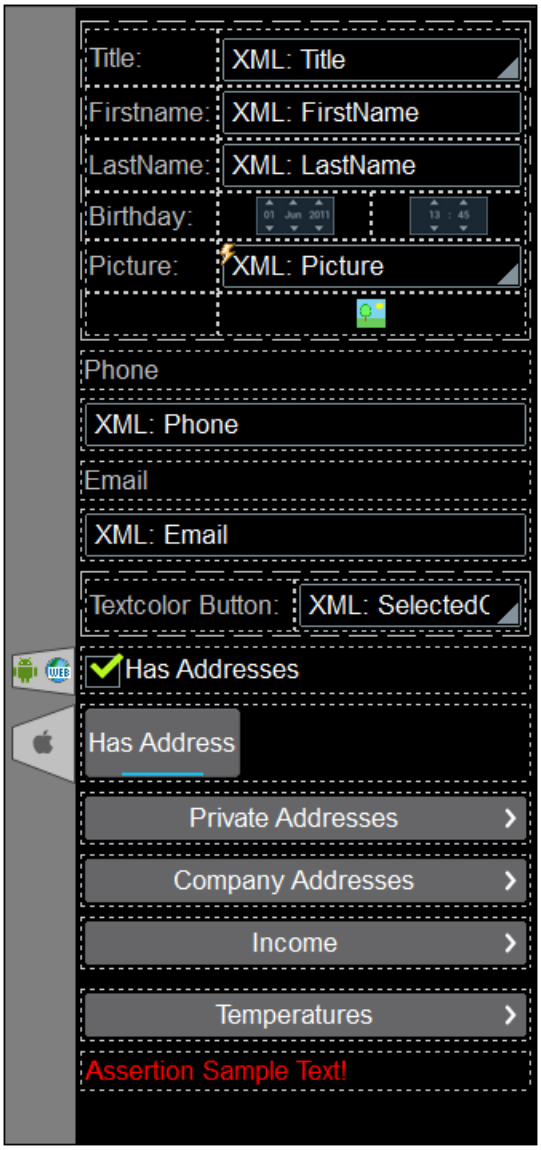
メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページデザインビューの)コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページデザインビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティの XPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザー CSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールの CSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

コントロールのリスト

以下は使用可能なコントロールのリストです。

- アサーション メッセージ
- ボタン
- チャート
- チェックボックス
- コンボボックス
- 日付
- 日時 (iOS)
- 編集フィールド
- 水平線
- 水平スライダー
- イメージ
- ラベル
- ラジオボタン
- リッチテキスト
- 署名フィールド
- スペース
- 切り替え
- テーブル
- 時刻
- 垂直線
- ビデオ



コンテキスト メニュー

ページ デザインの各コントロールには、コンテキスト メニューがあります。以下のコントロールに関連したコマンドは、ほぼすべてのコンテキスト メニューに共通しています。

- ページ ソース リンクの削除
  - *説明*

    このコマンドは、データソース ノードに関連付けることのできる、および関連付けを持つコントロールに対して有効化されます。**[ページソースリンクの削除]** はコントロールとデータソースの関連づけを削除します。コントロールのノードとの関連を削除する方法は他にはないことに注意してください。

- コントロール アクション
  - *説明*

    多種のコントロール イベントを設定することのできる[ページ アクション] ダイアログを表示します。コントロール イベントの使用可能なアクションとコントロール アクションの設定に関しての説明は、[ページデザイン] アクション] を参照してください。

- テキストの入力
  - *説明*

    テキストが入力できるコントロールに対して有効化できます。次のオプションがあるサブメニューをロールアウトします:

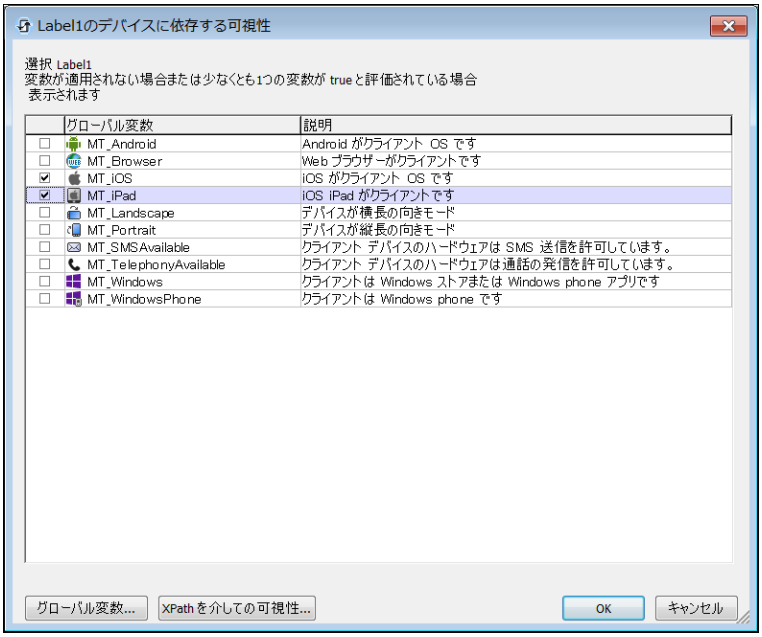
    - *ディレクトリ:* コントロールのテキストとして静的 テキストを直接入力します
    - *XPath:* コントロールのテキストを選択するXPath 式を入力する[XPath/XQuery 式の編集] ダイアログ を表示します。
    - *XML ノード:* データソース ノードを[ページ ソース] ペインからコントロールへドラッグするヒントを表示します。ドラッグアンド ドロップ されたノードはコントロールに関連付けられていて、ノードのコンテンツはコントロールのテキストとして入力されます。

- ローカライズ
  - *説明*

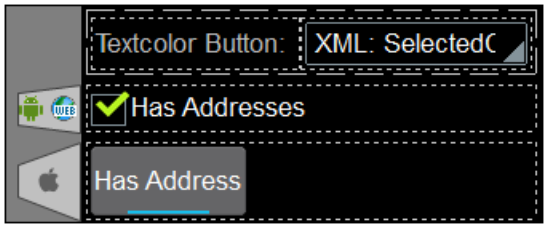
    プロジェクトの多種のコントロールに表示される文字列のローカライズ (翻訳) を定義することのできる ローカライズ? ダイアログ を表示します。このコマンドには **[プロジェクト] [ローカライズ]** コマンドと同じ効果があります。ローカライズに関する詳細は **[プロジェクト] [ローカライズ]** コマンドに移動してください。

- デバイスに依存する可視性
  - *説明*

デバイスに依存する可視性ダイアログ (*下のスクリーンショット*) を表示します。このダイアログはクライアントデバイスの種類のリストを含みます。コントロールを可視化したいクライアントデバイスの種類を選択して、**[OK]** をクリックします。



クライアントデバイスの種類にアイコンがある場合、デザインのコントロールの左側にアイコンが表示されます(*下のスクリーンショット参照*)。



ページアクション

説明

多種のページイベントのアクションを設定する [ページアクション] ダイアログを表示します。このコマンドには **[ページ|ページアクション]** コマンドと同じ効果があります。ページイベントに使用可能なアクションとページアクションの設定方法に関する詳細は **[ページ|ページアクション]** コマンドに移動してください。

アクションの概要

説明

現在アクティブなページのアクション概要ダイアログを表示します。このコマンドには **[ページ| アクションの概要]** コマンドと同じ効果があります。ページイベントに使用可能なアクションとページアクションの設定方法に関する詳細は **[ページ| アクションの概要]** コマンドに移動してください。

アサーション メッセージ

アサーション メッセージ コントロールは、ページの最初の無効なアサーションのアサーション メッセージを表示します。 アサーションはページと一部のコントロールのプロパティです。アサーションは、(例えば、ノードが空であってはならない等の) 特定の条件を指定します。もし、アサーション プロパティの条件が満たされない場合、アサーションは無効であり、アサーション プロパティと関連付けられたアサーション メッセージ **[プロパティ]** が **[アサーション メッセージ コントロール]** に表示されます。

アサーション メッセージ コントロールはデザインのどこにでも置くことができます。常に、ページの最初の無効なアサーションに関連付けられた、アサーション メッセージのプロパティテキストを表示します。複数の無効なアサーションがページにある場合、(ページでアサーション メッセージが表示される順に) 最初の無効なアサーションのアサーション メッセージが表示されます。アサーション メッセージコントロールは、ページに一度のみ挿入されるべきです。複数のアサーション メッセージ コントロールがデザインにおかれると、(最初の無効なアサーションの) 同じアサーション メッセージを表示することになります。

アサーションメッセージは以下のように作動します:

- コントロールまたはページのサーション メッセージのプロパティは、アサーションが有効であるように条件が満たされるように設定します。アサーションの条件はXPath 式を使用して指定されます。
- アサーションが無効な場合、コントロールのアサーション メッセージ プロパティのテキストは、アサーション メッセージ コントロールに表示されます。

例: コントロールまたはページの アサーション プロパティのXPath 式 `LastName != ""` が、ノード`LastName` が空である必要があるとアサートしたとします。このノードが空の場合、コントロールのアサーション メッセージはアサーション メッセージ コントロールが挿入されたページの箇所に表示されます。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

アサーションメッセージイベント

[アサーションメッセージ] コントロールに割り当てられているイベントはありません。

アサーション メッセージ プロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

複数行

複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい | 小さい | 中 | 大きい | 一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの **`TextSize`** 値として使用します: **`mt-font-height('largest', 'px') * 1.2`** 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル) を生成します。この値は、**'一番大きい'** の120% に対応する値を得るために **`1.2`** 倍されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に **`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は **`concat($MT_CanvasX * 0.5, 'px')`** になります。

テキストサイズの自動調整

テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (`true`) | オフ(`false`) | グループ `X`. デフォルトは、オフ(`false`)です。

9つのグループの1 つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。 すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

`Multiline` プロパティが **`true`** に設定されている場合、自動的にフィットする機能を使用することはできません。

メモ　テキストサイズを自動的に調整するはクライアントでは使用することができません。

太字テキスト

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

イタリックテキスト

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

- コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。

$MT_CanvasX と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**$MT_CanvasX * 0.5** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**concat($MT_CanvasX * 0.5, 'px')** になります。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **LabelClassOne LabelClassTwo**。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

ボタン

ボタンはボタンが押された時にアクションを実行する時に使用することができます。ボタンの名前は、(Text プロパティの値として入力された、以下参照) 静的テキストまたは、(ノードをボタンにドラッグして) データソースノードから得られた動的な値であることができます。 または、ボタンルック プロパティコンボボックスで使用することのできるオプションからアイコンを選択することもできます。アイコンが選択されると、テキスト プロパティとすべてのテキストフォーマットプロパティを使用することはできなくなります。これらを使用する場合は、ボタンルック プロパティのアイコン無しのオプションを選択してください。ボタンのクリック時 イベントは、ボタンコントロールに関連付けられています。このイベントのアクションを定義するためには、コントロール アクション プロパティの**[追加ダイアログ]** ボタンをクリックします。これにより、必要とされるアクションを指定することのできる[コントロールアクション] ダイアログが表示されます。

メモ

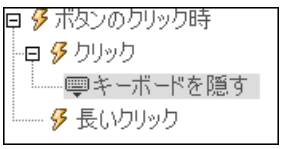
- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

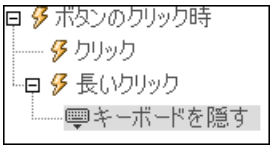
ボタンイベント

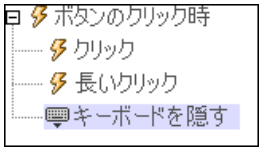
ボタンのクリック時 イベントを使用することができます。ボタンの ボタンのクリック時 イベントためのアクションを定義するには、ボタンを右クリックして、表示されるコンテキストメニューからボタンのクリック時のためのコントロール アクションを選択します。これによりボタンイベントのためのアクションダイアログを表示します。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

ボタンのクリック時 (クリック、長いクリック)

エンドユーザーはコントロールを次の方法でクリックすることができます: 短いタップまたは長押し。異なるアクションのシーケンスは、クリックの各種類により指定することができます。アクションの追加シーケンスは、エンドユーザークリックの後に実行することができます。この追加シーケンスは、長いクリックイベントの後に定義することができます。



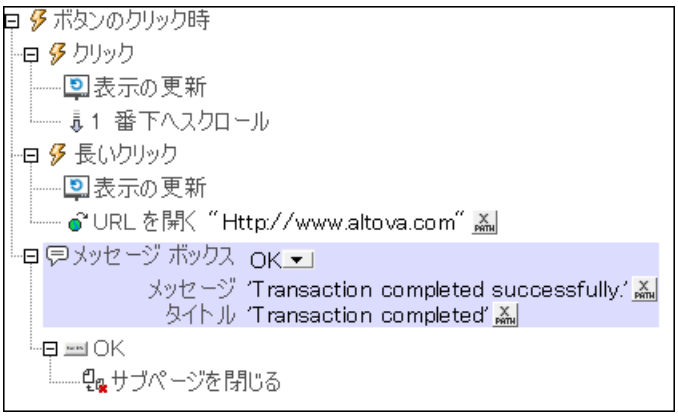




- ***クリック***: コントロールをタップすると実行されるアクション(*左上のスクリーンショットを参照*)。
- ***長いクリック***: タップよりも長くコントロールが押された時に実行されるアクション(*中央のスクリーンショットを参照*)。

- **追加アクション**: クリックまたは長いクリックアクションが実行された後に実行されるアクション(右上のスクリーンショットを参照)。クリックまたは長いクリックにアクションが設定されていない場合、追加アクションがクリックまたは長いクリックに対して直接実行されます。

異なるクリックイベントに対して アクション をまとめることができます。例えば下のサンプルに表示されているようにアクションをまとめることができます。



上のスクリーンショットはクリックと長いクリックイベントに定義されたアクションのシーケンスがあることを表示しています。追加メッセージボックスイベントは長いクリックイベントの後に定義されています。このメッセージボックスイベントは、クリックおよび長いクリックのアクションのシーケンスが終了した後に実行されます。

ボタンプロパティ

[スタイル & プロパティ] ペイン で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- テキスト

  テキストプロパティは以下の値から取得されます:

  - コントロールで表示される固定値テキスト文字列
  - データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

  編集する値フィールドをダブルクリックします、または、**XPath** ツールバー ボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキスト メニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

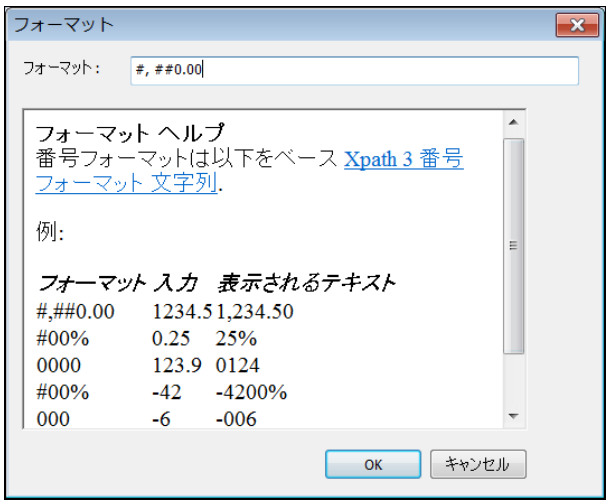
メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 複数行

複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

- 番号フォーマット文字列

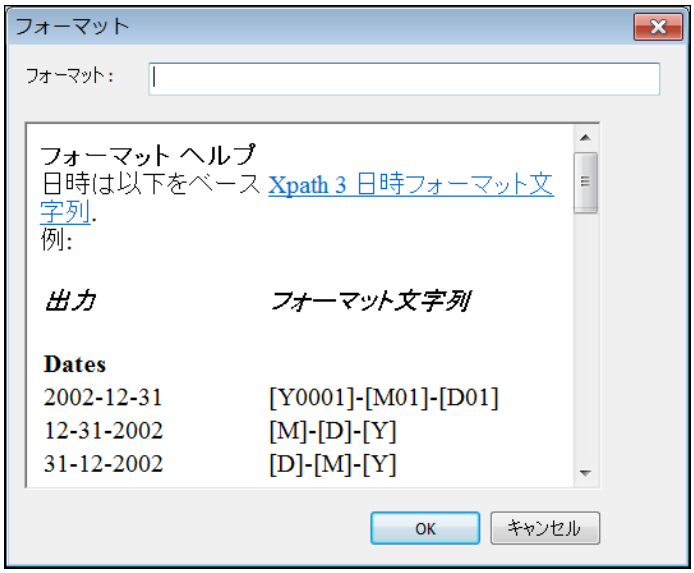
  [追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに番号フォーマットを入力します (*下のスクリーンショット*)。



コンテンツが数字の場合、フォーマットはコントロールのコンテンツに適用され、デザインではなくソリューションで表示されます。

- 日付 時刻フォーマット文字列

  [追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します (*下のスクリーンショット*)。



フォーマットは、`xs:date` (日付コントロール)、 `xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`

- *xs:dateTime:* `2014-12-31T23:59:59`

日付/時刻フォーマット言語

コンボボックスのドロップダウンリストでサポートされる言語 (英語、ドイツ語、スペイン語、日本語) から言語を選択します。選択された言語は、日付/時刻 フォーマット文字列 プロパティで設定された日付/時刻 フォーマットに使用されます。(*上の説明を参照*)。月と週の曜日の名前がフォーマット文字列で使用されている場合、これらはこのプロパティで選択された言語で表示されます。デフォルトの言語は英語です。

ボタンルック

定義済みのアイコンにボタンとして表示するアイコンを(テキストの代わりに) 追加します。プロパティのコンボボックスのドロップダウンリストからアイコンを追加することができます。以下のアイコンを使用することができます: プラス、マイナス、共有、電子メール、削除、編集、再生、再開、停止、マイクロフォン、早送り、巻戻し、検索。アイコンの大部分にはボタンの背景色として透明と不透明を使用することができます。ボタンルックプロパティのデフォルトの値はアイコンではありません。アイコンがボタンを表示するように追加されると、以前に入力されたボタンのテキストは削除されます。更に、すべてのコントロールのテキストフォーマットプロパティも削除されます。テキストを(アイコンの代わりに)追加する場合、アイコン無しのオプションを選択し、テキストプロパティの値としてボタンテキストを入力、またはボタンテキストを直接入力します。

コントロールアクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクションダイアログを表示します。コントロールイベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクションダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または `false()` を評価するXPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。** 使用された場合、検証エラーの結果が発生します。

有効化された編集可能な

コントロールは、プロパティの値が `true` (有効化) または `false` (無効化) であるにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の `true` または `false` を評価するXPath 式であることもできます。デフォルトの値は `true` です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が `true` と評価した場合、コントロールが有効化されます。

テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。

- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

テキストの色 (無効化済み)

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの 有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの **`TextSize`** 値として使用します: **`mt-font-height('largest', 'px') * 1.2`** 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル)を生成します。この値は、 **'一番大きい'**の120% に対応する値を得るために**`1.2`** 倍されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

テキストサイズの自動調整

テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (`true`) | オフ(`false`) | グループ X. デフォルトは、オフ(`false`)です。

9つのグループの1つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有

するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

Multiline プロパティが **true** に設定されている場合、自動的にフィットする機能を使用することはできません。

メモ　テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストから `true` または `false` を選択します。デフォルトは `false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから `true` または `false` を選択します。デフォルトは `false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから `true` または `false` を選択します。デフォルトは false です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値は XPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

高さのコントロール

高さのコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 高さのコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 高さのコントロールを、コンテンツが必要なだけの幅にします。

効果としては、`fill_parent` は最大の高さを作成し、`wrap_content` は最小の高さを作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは、`fill_parent` です。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのド

ロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。 **「ページ| タブの順序を表示 定義する|** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

入力 エスケープ時

以下の3つの値の1つを取ります:

- 入力時: **入力** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- エスケープ時: **エスケープ** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- 無し キーが押された場合でも、アクションは発生しません。これはデフォルトの値です。

XPath 式が値を生成するために使用されている場合、式は "**入力時** " または "**エスケープ時** " に対して評価される必要があります。ページ上の1 つ以上のコントロールに同じ値 (入力時 または エスケープ時) が与えられると、キーがタップされると、値を持つ最初に表示され、有効化されているコントロールが選択されます (`Visible` と**有効 /編集可能** プロパティを参照してください)。

この設定は コントロールの **クリック時** アクションを設定するために、ダイアログを使用しておこなうことができます (次を参照してください: コントロール イベント)。

メモ　この機能はWeb クライアントとWindows クライアント上のみで使用することができます。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照: ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

チャート

チャート コントロールによりソースデータファイルからのデータをチャートのフォームとして表示することが可能になります。使用可能なチャートの種類 は次のとおりです: 円グラス 棒グラス 折れ線グラス 面グラス ローソク足チャート、ゲージチャート。X軸、Y軸、および他のチャートのコンポーネントはXPath 式により選択されます。XPath 式のコンテキスト ノードは、ソースデータツリーからドラッグして、デザインのチャート コントロールにドロップして設定します。

ページ内のチャートの表示設定は[スタイル & プロパティ] ペインで定義されます。チャートの種類、データの選択、および表示の設定は、チャート構成ダイアログで定義されます。このダイアログは チャート設定プロパティの [追加ダイアログ] ボタンをクリックして、またはデザインのチャートをダブルクリックしてアクセスすることができます。

チャートの構成についての詳細はチャートのセクションを参照してください。

メモ

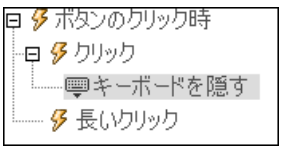
- コントロールがデータソースノード(ページソースリンク) と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

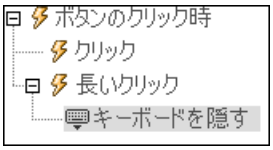
チャートイベント

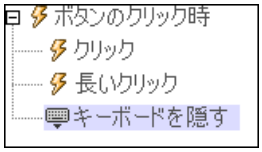
チャートのクリック時 イベント を使用することができます。チャートの チャートのクリック時 イベントためのアクションを定義するには、 チャートを右クリックして、表示されるコンテキストメニューからチャートのクリック時のためのコントロールアクションを選択します。これによりチャートイベントのためのアクションダイアログを表示します。 このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

チャートの クリック時 (クリック時、長いクリック時)

エンドユーザーはコントロールを次の方法でクリックすることができます: 短いタップまたは長押し。異なるアクションのシーケンスは、クリックの各種類により指定することができます。アクションの追加シーケンスは、エンドユーザークリックの後に実行することができます。この追加シーケンスは、長いクリックイベントの後に定義することができます。



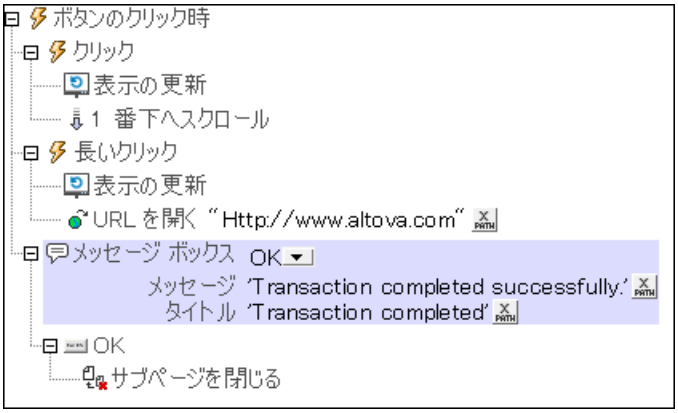




- ***クリック***: コントロールをタップすると実行されるアクション(*左上のスクリーンショットを参照*)。

- **長いクリック**: タップよりも長くコントロールが押された時に実行されるアクション(*中央のスクリーンショットを参照*)。
- **追加アクション**: クリックまたは長いクリックアクションが実行された後に実行されるアクション(*右上のスクリーンショットを参照*)。クリックまたは長いクリックにアクションが設定されていない場合、追加アクションがクリックまたは長いクリックに対して直接実行されます。

異なるクリックイベントに対してアクションをまとめることができます。例えば下のサンプルに表示されているようにアクションをまとめることができます。



上のスクリーンショットはクリックと長いクリックイベントに定義されたアクションのシーケンスがあることを表示しています。追加メッセージボックスイベントは長いクリックイベントの後に定義されています。このメッセージボックスイベントは、クリックおよび長いクリックのアクションのシーケンスが終了した後に実行されます。

チャートプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- チャート設定

  [追加ダイアログ] ボタンをクリックしてチャート構成ダイアログを表示します。このダイアログでする設定は、現在デザインで選択されているチャートに適用されます。チャートの構成方法についての説明はチャートのセクションを参照してください。

- ID

  このプロパティはチャートを繰り返しテーブルまたは動的テーブルの繰り返し行内に配置する時に入力される必要があります。ID プロパティはどの文字列を使用することができますが、インスタンスを作成するチャートのために異なる ID に対して評価されなければなりません。これは、動的 XPath 式をプロパティの値として割り当てることで達成できます。

- ロード前に作成する

  コンボボックスで、希望する値を以下から選択します: `true` または `false`。`true` の場合、チャートまたは Base64 イメージはページのロード前に作成されます。の `false` 場合は、ページソースアクションはチャートまたはイメージの作成に使用されなければなりません。デフォルトは `true` です。

- チャート作成の幅

生成されるチャートの幅をピクセルで設定します。**[XPath 編集]** アイコンをクリックして、表示されるダイアログに数字の値を返す式を入力します。この値が生成されるチャートのピクセルで示される幅となります。

- チャート生成の高さ

  生成されるチャートの高さをピクセルで設定します。**[XPath 編集]** アイコンをクリックして、表示されるダイアログに数字の値を返す式を入力します。この値が生成されるチャートのピクセルで示される高さとなります。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  `true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウン リストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウン リストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは `fill_parent` です。`Image` および `Chart` コントロールのデフォルトは `wrap_content` です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの 1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

- コントロールの高さをキャンバスの高さに制限する

  許可されている値 (`true` または`false`) をコンボボックスから選択します。コントロールの高さがデバイスの高さを超える場合、`true` の値はデバイスの表示の高さを制約します。デフォルトは`true` です。

- 余白

  コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

  コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係

することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

入力/エスケープ時

以下の3つの値の1つを取ります:

- 入力時: **入力** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- エスケープ時: **エスケープ** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- 無し: キーが押された場合でも、アクションは発生しません。これはデフォルトの値です。

XPath 式が値を生成するために使用されている場合、式は "**入力時**" または "**エスケープ時**" に対して評価される必要があります。ページ上の1つ以上のコントロールに同じ値 (入力時 または エスケープ時) が与えられると、キーがタップされると、値を持つ最初に表示され、有効化されているコントロールが選択されます (`Visible` と**有効/編集可能** プロパティを参照してください)。

この設定は コントロールの **クリック時** アクションを設定するために、ダイアログを使用しておこなうことができます (次を参照してください: コントロールイベント)。

メモ　この機能はWeb クライアントとWindows クライアント上のみで使用することができます。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (次を参照 *ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

チェック ボックス

チェックボックスによりノードのコンテンツに1 つまたは2 つの値を入力することが可能になります。この方法で、ユーザーは1 つまたは2 つの特定の値から選択するように制約されます。チェックボックスを挿入する時、チェックボックスがチェックボックスのテキストの右側または左側になるかを指定することができます。チェックボックスには次の2 つの重要なプロパティがあります:

- チェックボックスに伴うテキスト。テキストは(テキスト プロパティの値として入力された、*下を参照*) 静的テキストまたはXPath 式を介して得られた動的な値であることができます。
- チェックされたまたはチェックされていないチェックボックスにそれぞれ割り当てられた値。これらの値はチェック ボックス プロパティ(*下を参照*)で割り当てられます。ノードが値を受け取るように指定する場合、(チェックボックスにノードをドラッグして)データソースノードへのチェックボックスからのページソースリンクを作成します。

チェックボックスには、エンドユーザーがチェックボックスの選択を行うとトリガーされる編集の終了時 イベントがあります。このイベントのアクションを定義するには、コントロール アクションのプロパティの**[追加ダイアログ]** ボタンをクリックします。これにより必要なアクションを指定することのできる[コントロール アクション] ダイアログが表示されます。

メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザイン ビューの)コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

チェックボックス イベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

チェック ボックスのプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- テキスト

  テキストプロパティは以下の値から取得されます:

  - コントロールで表示される固定値テキスト文字列
  - データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

  編集する値フィールドをダブルクリックします、または、**XPath** ツールバーボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキスト メニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

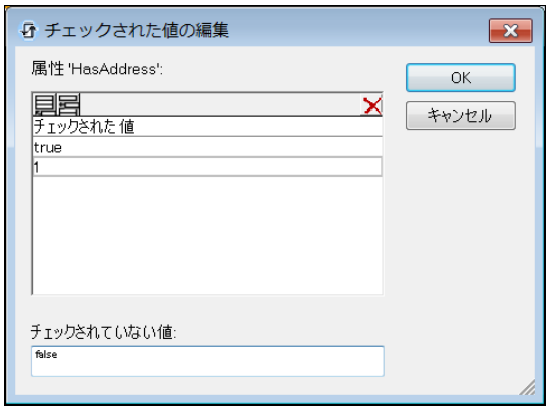
メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

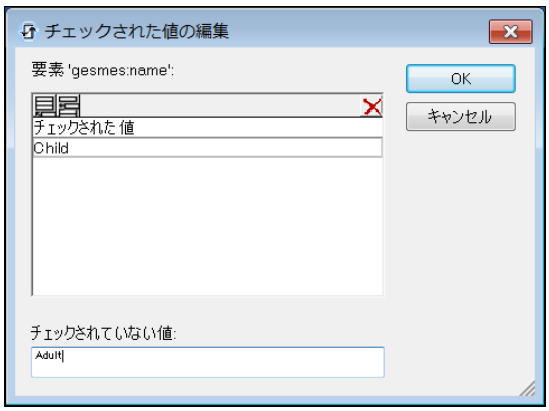
- 複数行

  複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

- チェックされた値 (true/false)

  選択された/選択されていない状態のコントロールにXML データの値を与えます。**[追加ダイアログ]** ボタンをクリックして、チェックされた値の編集ダイアログを表示します(*下のスクリーンショット*)。チェックされた(選択された) またはチェックされていない(選択されていない)状態の値を入力します。エンドユーザーにより選択されたコントロールの状態(選択された/選択されていない)に対応した値がコントロールに関連付けられたXML ノードのデータとして入力されます。





- XPath から値を取得する

  XPath 式により返された値はコントロールに表示されます。これにより、特定のコントロールに表示する値を入力する別の方法が有効化されます。例えば、コンボボックスはページソースノードまたは **XPath** から値を取得 プロパティの返り値から表示する値を得ることができます。

  メモ **$MTControlValue** 変数を、XPath から値を取得 プロパティの値を生成するために使用することは**できません。** 使用されると検証エラーの結果が発生します。

- 値の自動修正

  コントロールには 2 つの状態があります: チェックされたおよびチェックされていない状態です。各状態は少なくとも 1 つの XML 値に関連付けられています。これらの XML 値はチェックされた値 プロパティで定義されています。

  値の自動修正 プロパティには 2 つの可能な値があります: true または false です。プロパティが true に設定されている場合、XML 値は、チェックされた値 プロパティのチェックされた、およびチェックされていない状態で定義された値に自動的に修正されます。例えば、チェックされた XML 値が [子供] の値を持ち、そして、チェックされていない XML 値が [大人] の値を持ち、コントロールがチェックされている場合、何かが入力されると、XML 値は [子供] に修正されます。コントロールがチェックされていない場合、XML 値は [大人] になります。1 つ以上の XML 値がチェックされた状態にある場合、最初の XML 値が修正に使用されます。例えば、コントロールがコントロールの状態に無効な XML 値のコンテンツを持つノードと関連付けられている場合、修正が必要です。

- チェックマークの位置

  コントロールのテキストに相対するチェックボックスの位置をテキストの左または右側に設定します: デフォルトはオペレーティングシステムのデフォルトです。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  true() または false() を評価する XPath 式です。式が、false() のみを評価する場合、false() と評価した場合、コントロールは可視ではありません。 式が true() を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは true() です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ **$MTControlValue** 変数は、可視 プロパティの評価のために使用することは、**できません。** 使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が true (有効化) または false (無効化) であるにより、有効化または無効化されています。(コンボ ボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで) 値を直接入

力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

- テキストサイズ

  コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

  `mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの`TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2` 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル) を生成します。この値は、**'一番大きい'**の120% に対応する値を得るために`1.2` 倍されます。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- テキストサイズの自動調整

  テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです：個別 (true) | オフ(false) | グループ X. デフォルトは、オフ(false)です。

  9つのグループの1つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

  Multiline プロパティが **true** に設定されている場合、自動的にフィットする機能を使用することはできません。

  メモ テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストから true または false を選択します。デフォルトは false です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから true または false を選択します。デフォルトは false です。

- テキストの下線

  コンボボックスのドロップダウンリストから true または false を選択します。デフォルトは false です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、#FF0000)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、#FF0000)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

- 余白

  コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。**「ページ」 タブの順序を表示 定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ　タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **LabelClassOne LabelClassTwo**。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

コンボ ボックス

コンボ ボックス コントロールは2 つのパートから成るメカニズムを提供します:

- *入力テキストとXML 値*: コンボ ボックスの入力テキスト(エンドユーザが閲覧するドロップダウン リスト) およびそれに対応する(データソースに入る) XML値は、コンボボックス入力値 プロパティ(詳細は下を参照) で指定されます。
- *関連付けられたXML ノード*: [ページ ソース] ペイン からのノードはコンボ ボックスにドロップされます。これは関連付けられた、エンドユーザーが選択したコンボ ボックスの値を受けるXML ノード(またはページ ソース リンク) です。エンドユーザーが選択した値をページの他の箇所で使用する場合、ノードのコンテンツは、ノードをコントロールにドラッグして、またはXPath 式でノードを選択することにより参照されます。

メモ

- コントロールがデータソースノード(ページソースリンク) と関連付けられると、マウスを([ページ デザイン ビュー]の) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、([ページ デザイン ビュー]の) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択して[ペインのツールバー]の**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、[ペインのツールバー]の[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

コンボボックス イベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

コンボボックスのプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

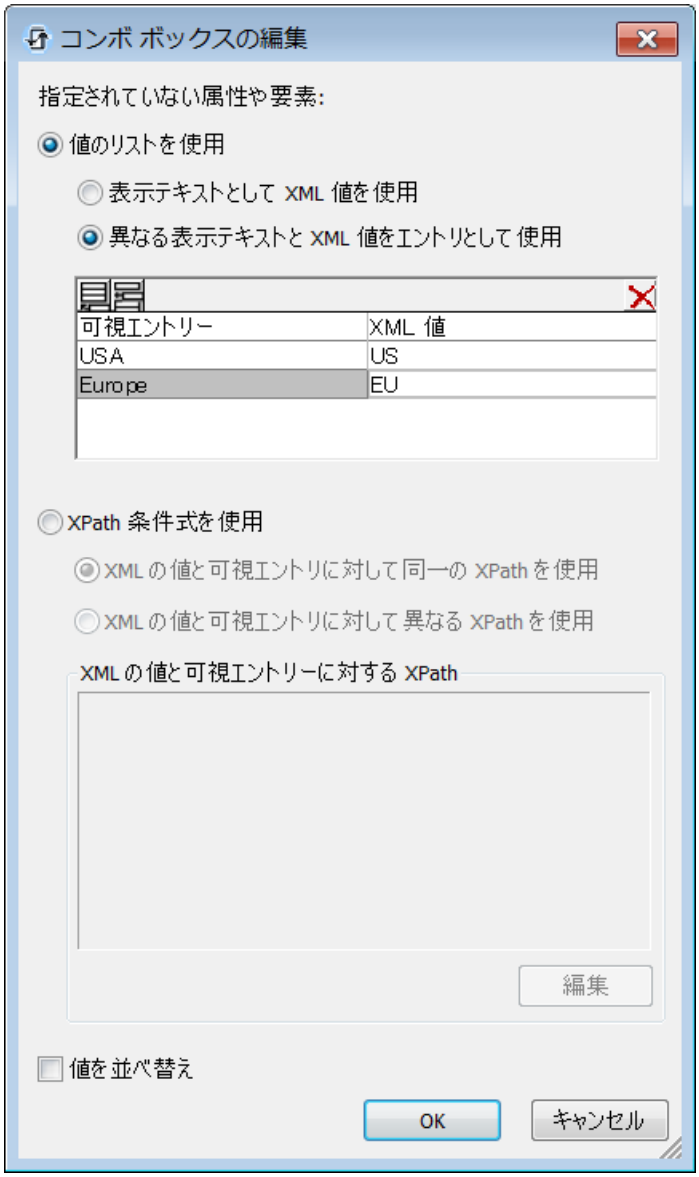
複数行

複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

コンボボックス入力値

コンボ ボックスのドロップダウン リストのアイテムのXML データを提供します。**[追加ダイアログ]** ボタンをクリックしてコンボ ボックスの編集ダイアログ(*下のスクリーンショット参照*) を表示します。別の方法は、コンボボックスをダブルクリックして

コンボ ボックスの編集ダイアログを表示します。



コンボ ボックスのエントリと値を定義するには以下を行います:

1. 適切なラジオ ボタンをクリックして値を選択して、エントリと値を定義する方法を選択します: (i) 値のリスト、または (ii) XPath 式。
2. *値のリストを使用* を選択した場合、コンボ ボックスのドロップダウン リストに表示されるテキスト、およびこれらのエントリにマップする値を指定することができます。以下の 2 つの選択肢を使用します (i) XML 値としてのエントリテキスト、および (ii) テキストから値 マッピング。XPath 式を使用して、値の可視エントリと XML の値を作成することもできます。XPath 式により返されたシーケンスのアイテムは、可視エントリおよび/または XML 値に使用されます。以下を指定することができます: (i) 同じ XPath 式が可視エントリと XML 値に使用される、または (ii) 異なる XPath 式が可視エントリと XML 値に使用される。後者の場合、1 つに 1 つずつの 2 つのシーケンス間のアイテムのインデックスのマッピングは、XML 値への可視エントリの対応を決定します。2 つのシーケンスのアイテムの数が同じでない場合、エラーが報告されます。
3. コンボ ボックスのドロップダウン リストに表示されるアイテムを保管したい場合、*値を並べ替え* チェック ボックスをチェックします。

4. **[OK]** をクリックして終了します。

メモ

- XPath 式を使用してコンボ ボックスのアイテムのリストを選択することにより、データ ソースからの動的なエントリを持つコンボ ボックスを作成することができます。
- コンボ ボックスのドロップダウン リストのアイテムはデータ ソースから取得され、デフォルトではドキュメントの順番に表示されます。

XPath から値を取得する

XPath 式により返された値はコントロールに表示されます。これにより、特定のコントロールに表示する値を入力する別の方法が有効化されます。例えば、コンボボックスはページソースノードまたは **XPath** から値を取得 プロパティの返り値から表示する値を得ることができます。

メモ **$MTControlValue** 変数を、XPath から値を取得 プロパティの値を生成するために使用することは**できません。**使用されると検証エラーの結果が発生します。

値の自動修正

2 つの可能な値があります: true または false です。true に設定されている場合、ドロップダウン リストの表示アイテムを自動的に修正して、リストで定義されたアイテムが表示されます。定義されていないアイテムは削除されます。例えば、コンボ ボックスのページ ソース リンク ノードを介して定義されていないアイテムをリストに入れることができます。票視される値が自動修正された場合、ドロップダウン リストのアイテムは定義された順番に表示されます。プロパティのデフォルトの値は false です。

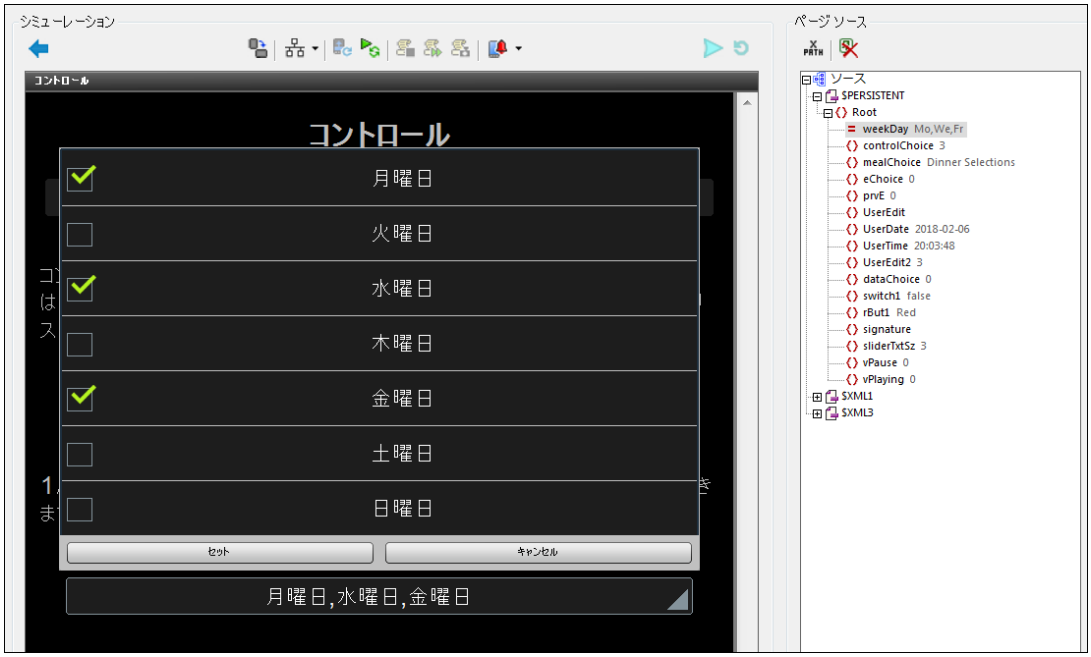
複数の選択

コンボボックスに複数の値の選択を許可するか(true/false).設定します。デフォルトは、false です。

コンボボックスは、通常1つのみを選択することのできる複数のオプションを提供します。MobileTogether ソリューション内で、選択されたオプションに対応する値はページソースノードに書き込まれます。

複数の選択 プロパティが true に設定されている場合、ユーザーは1つ以上のオプションを選択することができます。選択されたオプションのそれぞれに対応する値は、単一の文字列に結合され、関連したページソースノードに入力されます。文字列が構成されると、選択された値が**複数の選択セパレーター** プロパティ内で指定されているセパレーターにより釘枯れます。

下のスクリーンショット内で、選択されたオプションに対応する値は、(カンマのセパレーターを持つ) 単一の文字列に結合され、**weekDay** 属性ノードに挿入されます。セパレーターは**複数の選択 セパレーター** プロパティを使用して選択されます (*下を参照*)。



メモ 複数の選択を含むノード内の値が個別に読み取られる必要がある場合、文字列は文字列を構築する際に使用されたセパレーターと同じセパレーターを使用してトークン化される必要があります。

複数の選択セパレーター

複数の選択 プロパティが`true` に設定されている場合、このプロパティは有効化にされます。返される文字列内でセパレーターが使用されるように指定します。詳細に関しては、複数の選択 プロパティを参照してください。

使用することのできる選択肢から選択、または、セパレーターとして使用する文字を入力します。

コントロール アクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場

合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるかにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

- テキストサイズ

  コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

  `mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの**`TextSize`** 値として使用します: **`mt-font-height('largest', 'px') * 1.2`** 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル)を生成します。この値は、**'一番大きい'**の120% に対応する値を得るために**`1.2`** 倍されます。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビュ

ーポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- テキストサイズの自動調整

  テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (true) | オフ(false) | グループ X. デフォルトは、オフ(false)です。

  9つのグループの1つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

  Multiline プロパティが**true** に設定されている場合、自動的にフィットする機能を使用することはできません。

  メモ テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボ ボックスのドロップダウン リストからtrue またはfalse を選択します。デフォルトはfalse です。

- イタリック テキスト

  コンボ ボックスのドロップダウン リストからtrue またはfalse を選択します。デフォルトはfalse です。

- テキストの下線

  コンボ ボックスのドロップダウン リストからtrue またはfalse を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、#FF0000)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、#FF0000)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

- 余白

  コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。

$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。
$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。
$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。 **「ページ| タブの順序を表示 定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの

順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

日付

日付コントロールはデータソースのノードから得られた日付をフォーマットするために使用されます。異なる地域のために異なる日付をフォーマットする際に役に立ちます。例えば `12-31-2014` (米国フォーマット) と`31-12-2014` (欧州フォーマット)。日付/時刻 フォーマット文字列プロパティでフォーマットは定義されています (*詳細は下を参照*)。ソース ノード コンテンツは、XSD 仕様で定義されたとおりの正確な構文フォーマットである必要があります: `YYYY-MM-DD`。

メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティの XPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザー CSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールの CSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

日付イベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

日付のプロパティ

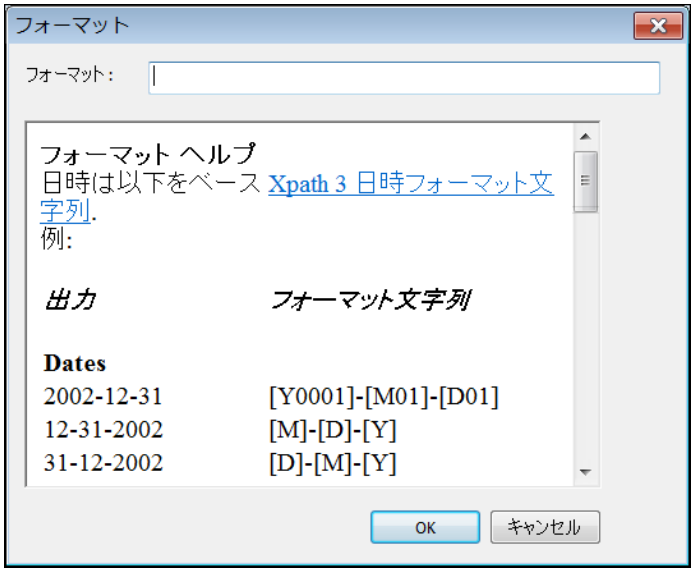
[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

日付 /時刻フォーマット文字列

[追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します (*下のスクリーンショット*)。



フォーマットは、`xs:date` (日付コントロール)、`xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`
- *xs:dateTime:* `2014-12-31T23:59:59`

日付/時刻フォーマット言語

コンボボックスのドロップダウンリストでサポートされる言語 (英語、ドイツ語、スペイン語、日本語) から言語を選択します。選択された言語は、日付/時刻 フォーマット文字列 プロパティで設定された日付/時刻 フォーマットに使用されます。(*上の説明を参照*)。月と週の曜日の名前がフォーマット文字列で使用されている場合、これらはこのプロパティで選択された言語で表示されます。デフォルトの言語は英語です。

時間処理

値の時間の部分をどのように処理するか指定します。3 つの使用可能なオプションがあります:

- *時間のカット*は時間の部分を削除します。
- *時間をゼロにセット*は時間の部分をキープしますが、`00:00:00` に変更します。
- *時間をキープ*は時間の部分をキープします。

コントロール アクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価し

た場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ　スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ　`$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化 /編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

- アサーション

コントロールのアサーションを有効にするための条件を設定します。アサーションが無効な場合アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールに表示されます。(複数のアサーション メッセージ コントロールが存在する場合、これらのコントロールは アサーションメッセージ プロパティのテキストを表示します。) プロパティの**XPath** アイコンをクリックして、アサーションを定義するXPath 式を入力します。例えば XPath 式 **`LastName != ""`** はノード`LastName` は空であってはならないとアサートします。もしこのノードが空の場合、コントロールのアサーションメッセージがページのアサーション メッセージ コントロールに表示されます。

他のコントロールとページもアサーションを持つことができます。複数の無効なアサーションがページに存在する場合、最初の無効のアサーションが表示されます。 コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- アサーション メッセージ

コントロールのアサーションが有効でない場合、アサーションメッセージが表示されるように設定します。プロパティの値フィールドの内側をダブルクリックしてアサーションメッセージを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力し必要なテキストを生成します。 アサーションメッセージはアサーション メッセージコントロール内に表示されます。例えば コントロールのアサーションのXPath 式が**`LastName != ""`** の場合、がノード`LastName` は空であってはならないとアサートするとします。 は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。

複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウン リストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

テキストの色 (無効化済み)

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい | 小さい | 中 | 大きい | 一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次の XPath 式をの **`TextSize`** 値として使用します: **`mt-font-height('largest', 'px') * 1.2`** 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル) を生成します。この値は、**'一番大きい'**の120% に対応する値を得るために**`1.2`** 倍されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

テキストサイズの自動調整

テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (`true`) | オフ(`false`) | グループ X. デフォルトは、オフ(`false`)です。

9つのグループの1 つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

`Multiline` プロパティが**`true`** に設定されている場合、自動的にフィットする機能を使用することはできません。

メモ　テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- ヒント

  エンドユーザーにテキスト化したヒントを与えます。例えば、エンドユーザーがコントロールを使用して実行するアクションについてのヒントを提供します。プロパティの値フィールドの内側をダブルクリックして、テキスト化したヒントを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力して必要なテキストを生成します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポ**

イントをデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。 **「ページ | タブの順序を表示 定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます:

LabelClassOne LabelClassTwo。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

日時 (iOS)

日時コントロールはiOS クライアントデバイスのみで使用が可能で、データソースのノードから得られた日時をフォーマットするために使用されます。異なる地域のために異なる日付をフォーマットする際に役に立ちます。例えば 12-31-2014 (米国フォーマット) と31-12-2014 (欧州フォーマット)。日付/時刻 フォーマット文字列プロパティでフォーマットは定義されています (*詳細は下を参照*)。ソース ノード コンテンツは、XSD 仕様で定義されたとおりの正確な構文フォーマットである必要があります: YYYY-MM-DDTHH:MM:SS。任意のタイムゾーンとミリ秒の部分は許可されています。

メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

日時イベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

日時プロパティ

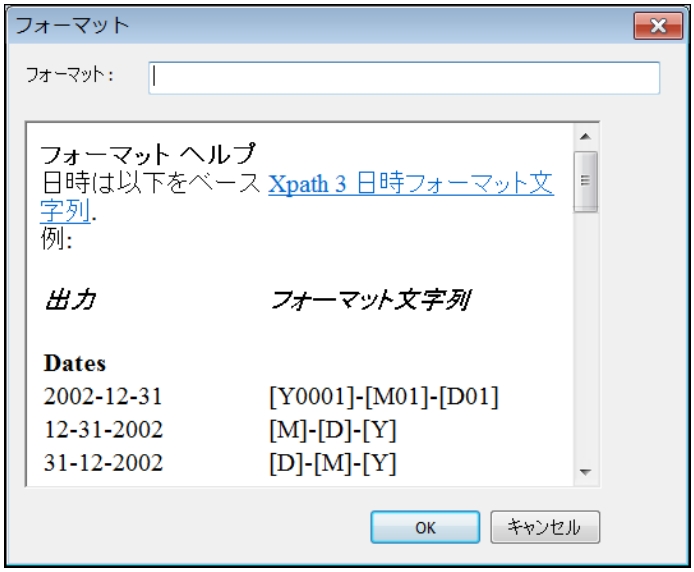
[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

日付/時刻フォーマット文字列

[追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します (*下のスクリーンショット*)。



フォーマットは、`xs:date` (日付コントロール)、`xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`
- *xs:dateTime:* `2014-12-31T23:59:59`

日付/時刻フォーマット言語

コンボボックスのドロップダウンリストでサポートされる言語 (英語、ドイツ語、スペイン語、日本語) から言語を選択します。選択された言語は、日付/時刻 フォーマット文字列 プロパティで設定された日付/時刻 フォーマットに使用されます。(*上の説明を参照*)。月と週の曜日の名前がフォーマット文字列で使用されている場合、これらはこのプロパティで選択された言語で表示されます。デフォルトの言語は英語です。

コントロール アクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ　スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

有効化された編集可能な

コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるかにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

アサーション

コントロールのアサーションを有効にするための条件を設定します。アサーションが無効な場合アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールに表示されます。(複数のアサーション メッセージ コントロールが存在する場合、これらのコントロールは アサーションメッセージ プロパティのテキストを表示します。) プロパティの**XPath** アイコンをクリックして、アサーションを定義するXPath 式を入力します。例えば XPath 式 `LastName != ""` はノード`LastName` は空であってはならないとアサートします。もしこのノードが空の場合、コントロールのアサーションメッセージがページのアサーション メッセージ コントロールに表示されます。

他のコントロールとページもアサーションを持つことができます。複数の無効なアサーションがページに存在する場合、最初の無効のアサーションが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

アサーション メッセージ

コントロールのアサーションが有効でない場合、アサーションメッセージが表示されるように設定します。プロパティの値フィールドの内側をダブルクリックしてアサーションメッセージを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力し必要なテキストを生成します。アサーションメッセージはアサーション メッセージコントロール内に表示されます。例えば コントロールのアサーションのXPath 式が`LastName != ""` の場合、がノード`LastName` は空であってはならないとアサートするとします。は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。

複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウン リストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの'**一番大きい**' に対応

する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの`TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2` 。関数は`'一番大きい'` サイズに対応する数値 (ピクセル) を生成します。この値は、`'一番大きい'`の120% に対応する値を得るために`1.2` 倍されます。

- *iOS デバイス上でのポイントとピクセル*

  長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

  MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 太字テキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- ヒント

  エンドユーザーにテキスト化したヒントを与えます。例えば、エンドユーザーがコントロールを使用して実行するアクションについてのヒントを提供します。プロパティの値フィールドの内側をダブルクリックして、テキスト化したヒントを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力して必要なテキストを生成します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

幅のコントロール

幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が

計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

編集フィールド

編集フィールドはエンドユーザからの入力を許可するために使用されます。エンドユーザーの入力をXML ノードに保管する場合、(データソースからコントロールにノードをドロップすることにより) デザインに このノードとの関連付けが必要になります。この場合、有効 /編集可能 プロパティを true に設定して、編集フィールドが編集可能であることを確認してください。関連付けられたノードの値は編集フィールドに表示されます。エンドユーザーが編集フィールドを修正すると、関連付けられたノードは自動的に更新されます。

ノードを編集フィールドに関連付ける代わりに、編集フィールドのコンテンツを、テキストプロパティのXPath 式により定義することができます(下を参照)。XPath 式がXML ノードをロケートする場合、効果はコントロールにノードをドロップすることと同じです。XPath 式は出力の計算、および、編集フィールドの表示にも使用することができます。この場合、有効 /編集可能プロパティはに自動的に false 設定され、編集することができません。機能は[ラベル] コントロールと同じになります。

メモ

- コントロールがデータソースノード(ページソースリンク) と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

編集フィールドイベント

入力中 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

編集フィールドのプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

キーボード

データ入力をアクティブ化するためにエンドユーザーのキーボードの種類を設定します。コンボボックスのドロップダウンリストの次のオプションから1 つ選択してください:

- テキスト
- 番号
- パスワード
- 電子メール
- URI
- 電話番号

テキスト

テキストプロパティは以下の値から取得されます:

- コントロールで表示される固定値テキスト文字列
- データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

編集する値フィールドをダブルクリックします、または、**XPath** ツールバーボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキスト メニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

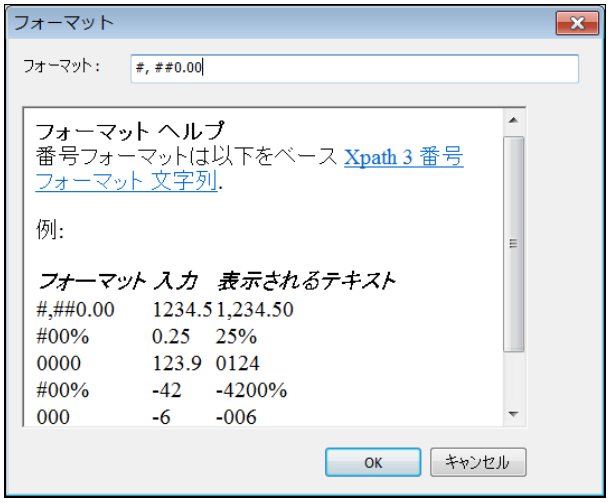
メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

複数行

複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

番号フォーマット文字列

[追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに番号フォーマットを入力します (*下のスクリーンショット*)。

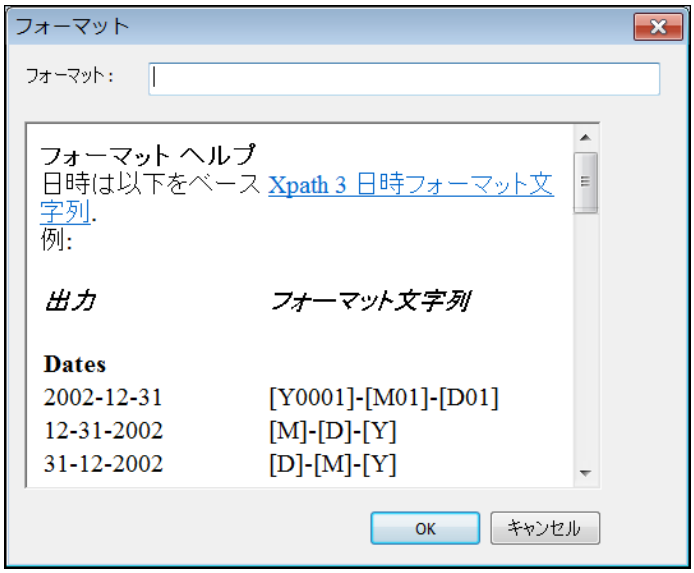


コンテンツが数字の場合、フォーマットはコントロールのコンテンツに適用され、デザインではなくソリューションで表示されます。

日付 時刻フォーマット文字列

[追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します (*下の*

スクリーンショット)。



フォーマットは、`xs:date` (日付コントロール)、`xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`
- *xs:dateTime:* `2014-12-31T23:59:59`

日付/時刻フォーマット言語

コンボボックスのドロップダウンリストでサポートされる言語 (英語、ドイツ語、スペイン語、日本語) から言語を選択します。選択された言語は、日付/時刻 フォーマット文字列 プロパティで設定された日付/時刻 フォーマットに使用されます。(*上の説明を参照*)。月と週の曜日の名前がフォーマット文字列で使用されている場合、これらはこのプロパティで選択された言語で表示されます。デフォルトの言語は英語です。

タイプ中のコントロールアクションのトリガー

タイプが開始するとコントロールアクションがすぐにトリガーされます。例えば、タイプを開始するとページを下までスクロールすることができます、またはタイプ中にノードを更新することもできます。プロパティの値は`true` または`false` です。デフォルトの設定は`true` です。HTML に関しては、設定は常に`false` です。

コントロールアクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクションダイアログを表示します。コントロールイベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクションダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの**[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または`false()`を評価するXPath 式です。式が`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。式が`true()` を評価し、または他の値を返す場合、コントロールは可

視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるかにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化 /編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

- アサーション

コントロールのアサーションを有効にするための条件を設定します。アサーションが無効な場合アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールに表示されます。(複数のアサーション メッセージ コントロールが存在する場合、これらのコントロールは アサーションメッセージ プロパティのテキストを表示します。) プロパティの**XPath** アイコンをクリックして、アサーションを定義するXPath 式を入力します。例えば XPath 式 **`LastName != ""`** はノード`LastName` は空であってはならないとアサートします。もしこのノードが空の場合、コントロールのアサーションメッセージがページのアサーション メッセージ コントロールに表示されます。

他のコントロールとページもアサーションを持つことができます。複数の無効なアサーションがページに存在する場合、最初の無効のアサーションが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- アサーション メッセージ

コントロールのアサーションが有効でない場合、アサーションメッセージが表示されるように設定します。プロパティの値フィールドの内側をダブルクリックしてアサーションメッセージを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力し必要なテキストを生成します。アサーションメッセージはアサーション メッセージコントロール内に表示されます。例えば コントロールのアサーションのXPath 式が**`LastName != ""`** の場合、がノード`LastName` は空であってはならないとアサートするとします。は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。

複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウン リストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの 有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの`TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2` 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル)を生成します。この値は、 **'一番大きい'**の120% に対応する値を得るために`1.2` 倍されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

太字テキスト

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

イタリックテキスト

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

テキストの下線

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

背景色 (無効化済み)

オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

ヒント

エンドユーザーにテキスト化したヒントを与えます。例えば、エンドユーザーがコントロールを使用して実行するアクションについてのヒントを提供します。プロパティの値フィールドの内側をダブルクリックして、テキスト化したヒントを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力して必要なテキストを生成します。

テキストヒントの色

コントロールのヒントのテキストの色を設定します。このテキストの色はヒント プロパティで定義されたテキストの色になります(上を参照)。色選択するために次を行います:

- 色パレットをクリックして背景色を選択します
- コンボボックスのドロップダウンリストから色を選択します
- 値フィールドをダブルクリックして色コードを入力します (例えば、`#FF0000`)、または**XPath** ツールバーボタンをクリックして XPath 式を入力して必要なテキストを生成します。

水平の配置

水平の配置を左、中央、右に設定します。デフォルトは中央です。 値は XPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

幅のコントロール

幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**

に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。 **「ページ| タブの順序を表示 定義する|** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ　タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

水平線

水平線コントロールにより色と幅をスタイルできる線を追加することができます。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

水平線 イベント

水平線コントロールに関連したイベントはありません。

水平線のプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

線の幅

選択された水平線の幅を (太さ) ピクセルで設定します。コンボボックスのドロップダウンリストから幅の値 (ピクセル) を選択します、または値フィールドをダブルクリックして、数字の値を入力します。数字の値はピクセルの値として理解されます。このために、単位は指定する必要はありません。

線のスタイル

選択された水平線のスタイルを指定します。コンボボックスのドロップダウンリストからオプションを選択します。デフォルトの値は実線です。

線の色

選択された水平線の色を選択します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

可視

`true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを*

適用する)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **`LabelClassOne LabelClassTwo`**。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

水平線スライダー

水平線スライダーコントロールによりユーザーはスライダーのスケールで値を選択することができます。選択された値は、ページソースノードの値としてページソースリンクを使用して、入力することができます（下の *メモ* 参照してください）。[スタイル & プロパティ] ペイン内では、スライダーの最小と最大の値を設定することができ、（下の *水平線スライダーのプロパティを参照してください*）、スライダーを移動中、またはスライダーの移動が完了した後、実行する コントロールアクション を指定することができます。

コントロールの実行方法には（スライディングアクションをベースにした）以下にリストされる3つの方法があります:

アクションの実行	スライディング中のトリガーコントロールアクション	スライダーのトリガーの間隔（ミリ秒）
スライディングが完了後トリガーされるアクション	false（デフォルト）	—
継続的にトリガーされるアクション	true	0
間隔にトリガーされるアクション	true	1 – 10000

メモ

- コントロールがデータソースノード（ページソースリンク）と関連付けられると、マウスを（ページデザインビューの）コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、（ページデザインビューの）コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- （[スタイル & プロパティ] ペインの）スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- （[スタイル & プロパティ] ペインの）スタイルまたはプロパティの XPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの **[XPath の編集]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために（ブラウザー CSS クラスプロパティを介して）1 つ以上のクラスを定義し、（[ブラウザー設定] ダイアログで指定する）外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールの CSS プロパティは [スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

水平線スライダーイベント

スライドイベント を使用することができます。水平線スライダーイベントのために定義することのできるアクションについての説明はアクションセクションを参照してください。

水平線スライダープロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリッ

クして編集します。

- スライダー最小値

  スライダーの最小値を設定します。許可されている値は -10000 から10000 です。デフォルトの値は0 です。0 から100 の間の優先する値を選択し、XPath 式を入力します。

- スライダー最大値

  スライダーの最大値を設定します。許可されている値は -10000 から10000 です。 デフォルトの値は100 です。0 から100 の間の優先する値を選択し、XPath 式を入力します。

- スライディング中のコントロールアクション

  true に設定されている場合、コントロールアクションがスライディング**中**にトリガーされます。false に設定されている場合、コントロールアクションは、**スライディングの完了後**にトリガーされます。デフォルトは false です。

- スライディングインターバル上のコントロールアクション(ms)

  コントロールアクションがトリガーされる間隔をミリ秒で指定します。値の範囲は0 から10000 です。デフォルトは 1000 ms です。値をドロップダウンリストから選択するか、値を入力してください。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  true() または false() を評価するXPath 式です。式が、false() のみを評価する場合、false() と評価した場合、コントロールは可視ではありません。 式が true() を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは true() です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ **$MTControlValue** 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 上余白

  コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポ**

イントをデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

イメージ

イメージ コントロールはデザインにイメージを挿入します。挿入に選択されたイメージはURL により参照されるイメージ ファイル、またはBase64 エンコード イメージ データの文字列となることができます。 イメージ ソースの種類のプロパティはイメージの2 つの種類を指定します: URL により配置されるファイル、またはBase64 文字列。指定するために、データソースノードからイメージ(URL またはBase64 文字列) を得て、ノードをイメージ コントロールにドロップします。イメージコントロールのプロパティは下にリストされています。

メモ イメージ ソースが(URL またはBase64 文字列) がシミュレーション中またはシミュレーションの実行中に変更された場合、イメージは明示的に再ロード アクションで再ロードされる必要があります。例えば、コンボ ボックスがイメージの選択を変更すると、イメージをターゲットする再ロード アクションがコンボ ボックスで定義されなければなりません。

メモ

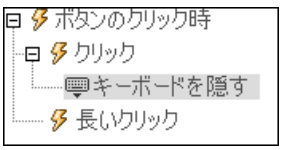
- コントロールがデータソースノード(ページソースリンク) と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

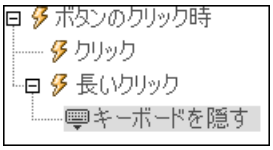
イメージイベント

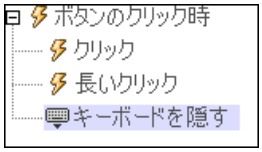
イメージのクリック時 イベントを使用することができます。イメージの たイメージのクリック時 イベントのアクションを定義するには、 イメージを右クリックして、表示されるコンテキストメニューからイメージのクリック時のためのコントロールアクションを選択します。これによりイメージイベントのためのアクションダイアログを表示します。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

イメージの クリック時 (クリック時、長いクリック時)

エンドユーザーはコントロールを次の方法でクリックすることができます: 短いタップまたは長押し。異なるアクションのシーケンスは、クリックの各種類により指定することができます。アクションの追加シーケンスは、エンドユーザークリックの後に実行することができます。この追加シーケンスは、長いクリックイベントの後に定義することができます。



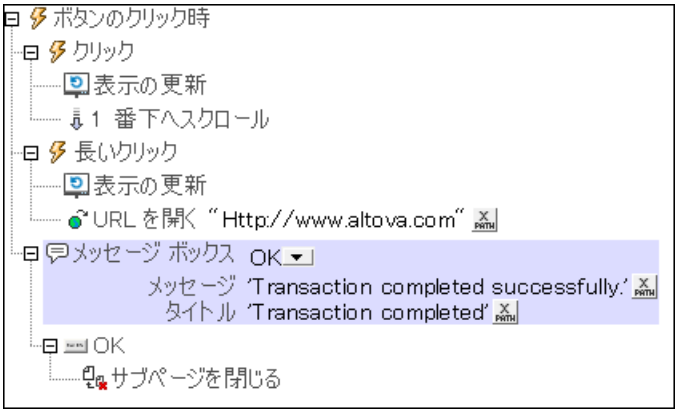




- ***クリック***: コントロールをタップすると実行されるアクション(*左上のスクリーンショットを参照*)。

- **長いクリック**: タップよりも長くコントロールが押された時に実行されるアクション(*中央のスクリーンショットを参照*)。
- **追加アクション**: クリックまたは長いクリックアクションが実行された後に実行されるアクション(*右上のスクリーンショットを参照*)。クリックまたは長いクリックにアクションが設定されていない場合、追加アクションがクリックまたは長いクリックに対して直接実行されます。

異なるクリックイベントに対して アクション をまとめることができます。例えば下のサンプルに表示されているようにアクションをまとめることができます。



上のスクリーンショットはクリックと長いクリックイベントに定義されたアクションのシーケンスがあることを表示しています。追加メッセージボックスイベントは長いクリックイベントの後に定義されています。このメッセージボックスイベントは、クリックおよび長いクリックのアクションのシーケンスが終了した後に実行されます。

イメージプロパティ

[スタイル & プロパティ] ペイン で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

イメージソース

イメージソースのプロパティの値は次の方法でイメージを参照します:

- (PNG、BMP などの) バイナリイメージファイルの URL です。プロパティの値は URL である必要があります。ファイルの指定ダイアログで選択されます (下の説明参照)。
- Base64 エンコード文字列としてイメージファイルが表示されます。プロパティの値は Base64 エンコード文字列である必要があります。直接入力または XML ノードから取得される文字列を XPath 式が提供します。

イメージソースの種類は イメージ ソースの種類 (下の次のプロパティを参照) プロパティで設定されます。デフォルトでイメージ ソースの種類は URL に設定されています。イメージ ソース プロパティは自動的に対応するダイアログを開きます: `url` (*下を参照*) のためのファイルの指定ダイアログ、および `base64` (*Base64 エンコード イメージ参照*) のための XPath/XQuery の編集ダイアログ。

メモ イメージソースが (URL または Base64 文字列) がシミュレーション中またはシミュレーションの実行中に変更された場合、イメージは明示的に再ロード アクションで再ロードされる必要があります。例えば、コンボ ボックスがイメージの選択を変更すると、イメージをターゲットする再ロード アクションがコンボ ボックスで定義されなければ

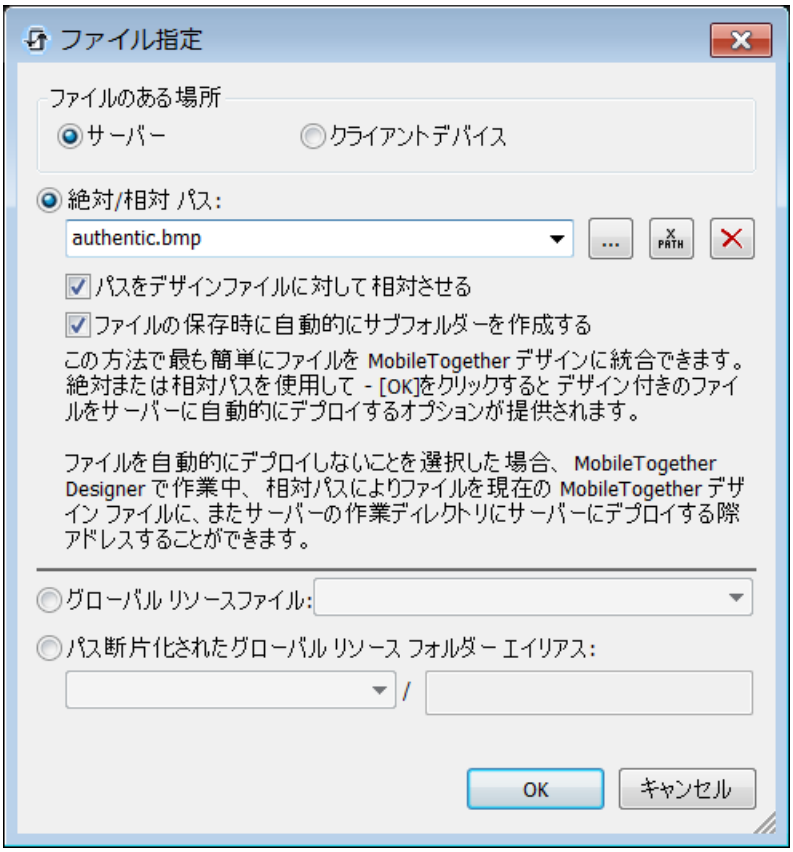
なりません。

- *ファイルの指定ダイアログ*

ファイルをサーバーまたはクライアントから選択する事ができます。相対するラジオボタンのオプションを選択してください。

ファイルがサーバーに存在する場合

イメージファイルがサーバーに存在する場合、(絶対/*相対パス*を)参照または(ファイルエイリアスまたはフォルダーエイリアス)グローバルリソースを使用して指定する事ができます。使用するオプションを選択してください(下のスクリーンショット参照)。



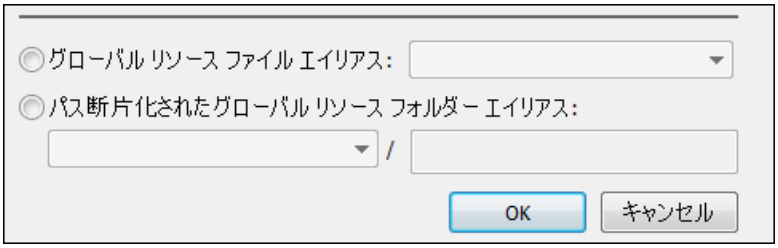
- **絶対 相対 パス** ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルトファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細はプロジェクト ファイルの場所を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対 絶対パスがファイルにアクセスするために(サーバーのデータベース) 内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合 : (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関してはプロジェクトファイルの場所のセクションを参

照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

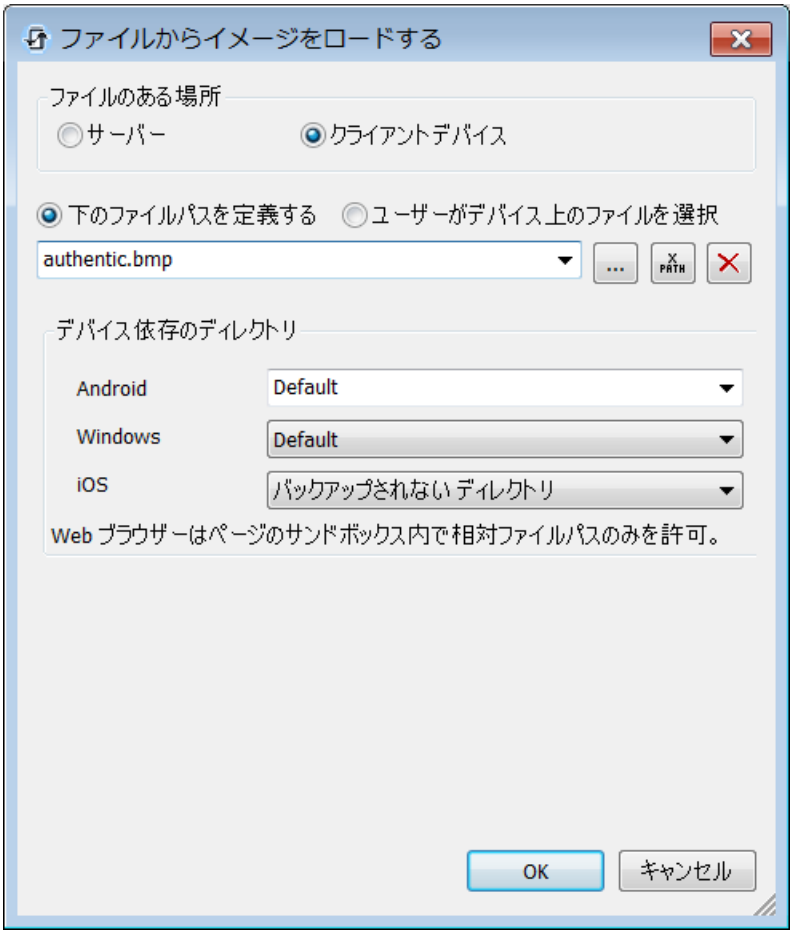
- **グローバル リソース ファイル エイリアス:** コンボボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイルリソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova グローバル リソースのセクションを参照してください。

- **パス断片化されたグローバル リソース フォルダ エイリアス:** コンボボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがクライアントに存在する場合
イメージファイルがクライアントに存在する場合、パスを入力またはファイルの場所を選択して指定してください。または、式を使用して XPath パスを構築することもできます。**[リセット]** ボタンを使用して現在のエントリを削除してください。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (*ソースファイル* と *ターゲットファイル* オプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。

- **デバイスに依存するリソース**: ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- **シミュレーションのためのファイルのロケーション**: ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ (**ツール | オプション**)内で指定されていなければなりません。

エンドユーザーにより(クライアントデバイス上で)ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子*: ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター*: クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト(例: `txt,html;xml`)、または(ii)各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式 (例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`)

- *オプションのデフォルトのファイル*: 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス*: ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、またはXPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する*: もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

  メモ　iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法はiCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

イメージソースの種類

上のイメージソースプロパティより選択されたイメージの種類の設定をします。方法は2 通りあります:

- URL: `PNG` または`BMP` イメージ ファイルなどのバイナリイメージ ファイル。
- `base64`: base64-エンコード文字列

デフォルトの設定はURL です。

ユーザー名

コンテンツにアクセスするためのユーザー名を設定します。プロパティの値フィールドをダブルクリックして編集します。

- パスワード

  ユーザーがリソースにアクセスするためにパスワードを設定します。プロパティの値フィールドをダブルクリックして編集します。

- ロード前に作成する

  コンボボックスで、希望する値を以下から選択します: `true` または `false`。`true` の場合、チャートまたは Base64 イメージはページのロード前に作成されます。の `false` 場合は、ページソースアクションはチャートまたはイメージの作成に使用されなければなりません。デフォルトは `true` です。

- イメージの埋め込み

  イメージ ソースの種類が URL の場合はプロパティが有効化されています。設定は `true` または `false` です。`true` の場合、イメージはデザイン ファイルに埋め込まれます。イメージ ファイル (`PNG`、`BMP`、など) のデータはテキスト ベースの Base64 エンコードに変換されます。このテキストがデザイン ファイルに埋め込まれます。プロパティのデフォルトの値は `false` です (ファイルは変換されず、埋め込まれません)。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  `true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。** 使用された場合、検証エラーの結果が発生します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値は XPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウン リストから値を選択するか、直接値を入力します。

- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは `fill_parent` です。`Image` および `Chart` コントロールのデフォルトは `wrap_content` です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に **`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は **`concat($MT_CanvasX * 0.5, 'px')`** になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に **`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は **`concat($MT_CanvasX * 0.5, 'px')`** になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が

計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 入力/エスケープ時

以下の3つの値の1つを取ります:

- 入力時: **入力** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- エスケープ時: **エスケープ** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
- 無し: キーが押された場合でも、アクションは発生しません。これはデフォルトの値です。

XPath 式が値を生成するために使用されている場合、式は "**入力時**" または "**エスケープ時**" に対して評価される必要があります。ページ上の1つ以上のコントロールに同じ値 (入力時 または エスケープ時) が与えられると、キーがタップされると、値を持つ最初に表示され、有効化されているコントロールが選択されます (`Visible` と**有効 /編集可能** プロパティを参照してください)。

この設定は コントロールの **クリック時** アクションを設定するために、ダイアログを使用しておこなうことができます (次を参照してください: コントロールイベント)。

メモ　この機能はWeb クライアントとWindows クライアント上のみで使用することができます。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **LabelClassOne LabelClassTwo**。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

ラベル

ラベルは情報を表示するために使用されます。デザイナーでラベルのテキスト プロパティとして入力され、表示された情報は静的である可能性があります。(ページソースリンクとして知られる) ページデータソースの一部からのランタイムから得られた動的な情報であることもあります。ラベル(ページソースリンク)の動的なコンテンツを指定するには、[ページ ソース] ペインから必要なデータツリーノードをドラッグするか、またはラベルのテキスト プロパティとしてノードからデータを得るまたは出力を計算する XPath 式を入力します。

メモ

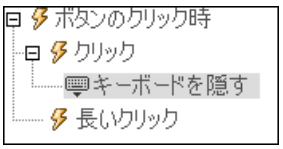
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

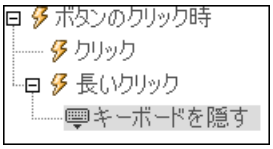
ラベルイベント

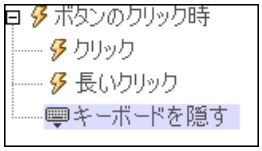
ラベルのクリック時 イベントを使用することができます。ラベルの ラベルのクリック時 イベントためのアクションを定義するには、 ラベルを右クリックして、表示されるコンテキストメニューから ラベルクリックの時のためのコントロールアクション を選択します。これによりラベルイベントのためのアクションダイアログを表示します。 このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

ラベルの クリック時 (クリック時、長いクリック時)

エンドユーザーはコントロールを次の方法でクリックすることができます: 短いタップまたは長押し。異なるアクション のシーケンスは、クリックの各種類により指定することができます。アクションの追加シーケンスは、エンドユーザークリックの後に実行することができます。この追加シーケンスは、長いクリックイベントの後に定義することができます。

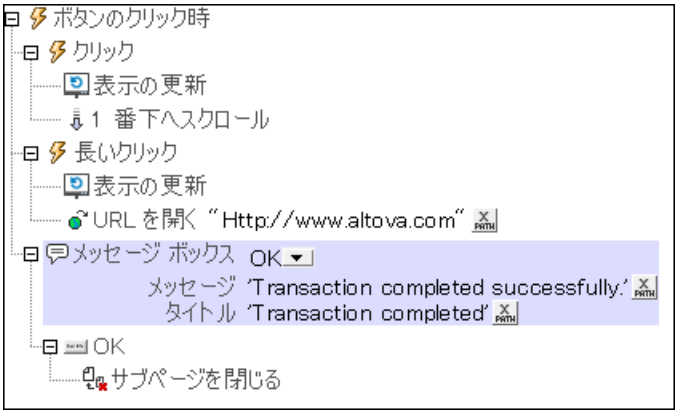






- **クリック**: コントロールをタップすると実行されるアクション(*左上のスクリーンショットを参照*)。
- **長いクリック**: タップよりも長くコントロールが押された時に実行されるアクション(*中央のスクリーンショットを参照*)。
- **追加アクション**: クリックまたは長いクリックアクションが実行された後に実行されるアクション(*右上のスクリーンショットを参照*)。クリックまたは長いクリックにアクションが設定されていない場合、追加アクションがクリックまたは長いクリックに対して直接実行されます。

異なるクリックイベントに対して アクション をまとめることができます。例えば下のサンプルに表示されているようにアクションをまとめることができます。



上のスクリーンショットはクリックと長いクリックイベントに定義されたアクションのシーケンスがあることを表示しています。追加メッセージボックスイベントは長いクリックイベントの後に定義されています。このメッセージボックスイベントは、クリックおよび長いクリックのアクションのシーケンスが終了した後に実行されます。

ラベルプロパティ

[スタイル & プロパティ] ペイン で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- テキスト

  テキストプロパティは以下の値から取得されます:

  - コントロールで表示される固定値テキスト文字列
  - データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

  編集する値フィールドをダブルクリックします、または、**XPath** ツールバー ボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキスト メニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

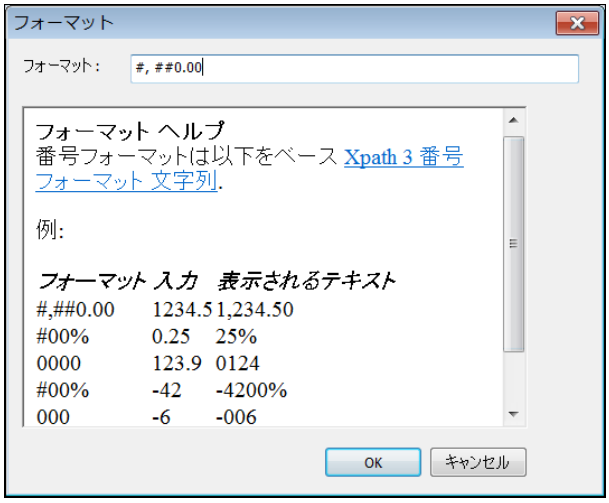
メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 複数行

  複数行の入力 表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

- 番号フォーマット文字列

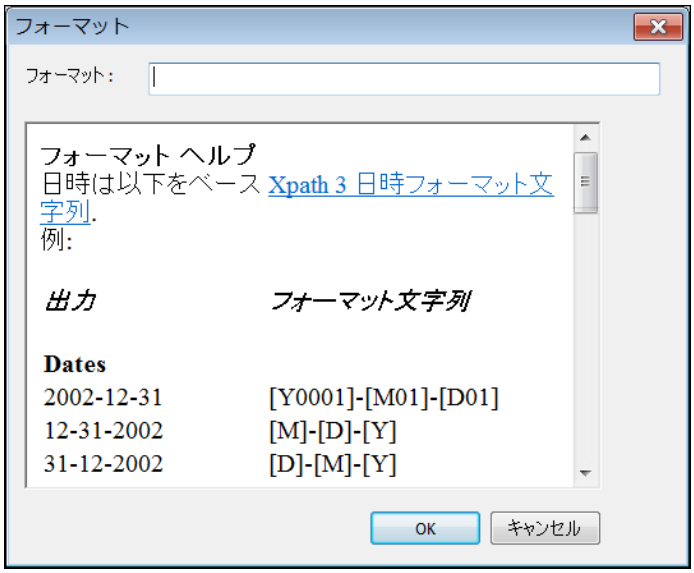
  [追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに番号フォーマットを入力します (*下のスクリーンショット*)。



コンテンツが数字の場合、フォーマットはコントロールのコンテンツに適用され、デザインではなくリレーションで表示されます。

- 日付/時刻フォーマット文字列

  [追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します (*下のスクリーンショット*)。



フォーマットは、`xs:date` (日付コントロール)、`xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`
- *xs:dateTime:* `2014-12-31T23:59:59`

- 日付/時刻フォーマット言語

  コンボボックスのドロップダウンリストでサポートされる言語 (英語、ドイツ語、スペイン語、日本語) から言語を選択します。選択された言語は、日付/時刻 フォーマット文字列 プロパティで設定された日付/時刻 フォーマットに使用されま

す。(*上の説明を参照*)。月と週の曜日の名前がフォーマット文字列で使用されている場合、これらはこのプロパティで選択された言語で表示されます。デフォルトの言語は英語です。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  `true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が `true` (有効化) または `false` (無効化) であるかにより、有効化または無効化されています。(コンボ ボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の `true` または `false` を評価する XPath 式であることもできます。デフォルトの値は `true` です。典型的な 有効化 /編集可能なプロパティの XPath 式は、ノードの存在または値をチェックします。式が `true` と評価した場合、コントロールが有効化されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように

設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

- テキストサイズ

  コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい | 小さい | 中 | 大きい | 一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

  `mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの'**一番大きい**'に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの`TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2` 。関数は'**一番大きい**'サイズに対応する数値 (ピクセル)を生成します。この値は、'**一番大きい**'の120% に対応する値を得るために`1.2` 倍されます。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- テキストサイズの自動調整

  テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (`true`) | オフ(`false`) | グループ X. デフォルトは、オフ(`false`)です。

  9つのグループの1つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

  `Multiline` プロパティが`true` に設定されている場合、自動的にフィットする機能を使用することはできません。

  メモ　テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

背景色 (無効化済み)

オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

水平の配置

水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

幅のコントロール

幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポ**

イントをデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビュ

ーポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

  長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

  MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 入力 /エスケープ時

  以下の3 つの値の1 つを取ります:

  - 入力時: **入力** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
  - エスケープ時: **エスケープ** キーがタップされると、このコントロールにフォーカスが与えられるように指定し、クリックされます。
  - 無し キーが押された場合でも、アクションは発生しません。これはデフォルトの値です。

  XPath 式が値を生成するために使用されている場合、式は **"入力時"** または **"エスケープ時"** に対して評価される必要があります。ページ上の1 つ以上のコントロールに同じ値 (入力時 または エスケープ時) が与えられると、キーがタップされると、値を持つ最初に表示され、有効化されているコントロールが選択されます (`Visible` と**有効 /編集可能** プロパティを参照してください)。

  この設定は コントロールの **クリック時** アクションを設定するために、ダイアログを使用しておこなうことができます (次を参照してください: コントロール イベント)。

  メモ　この機能はWeb クライアントとWindows クライアント上のみで使用することができます。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照: ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

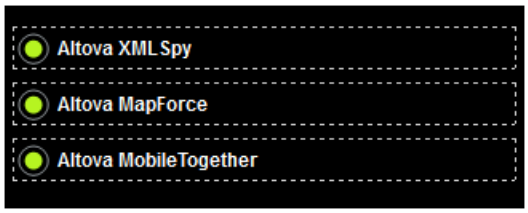
- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

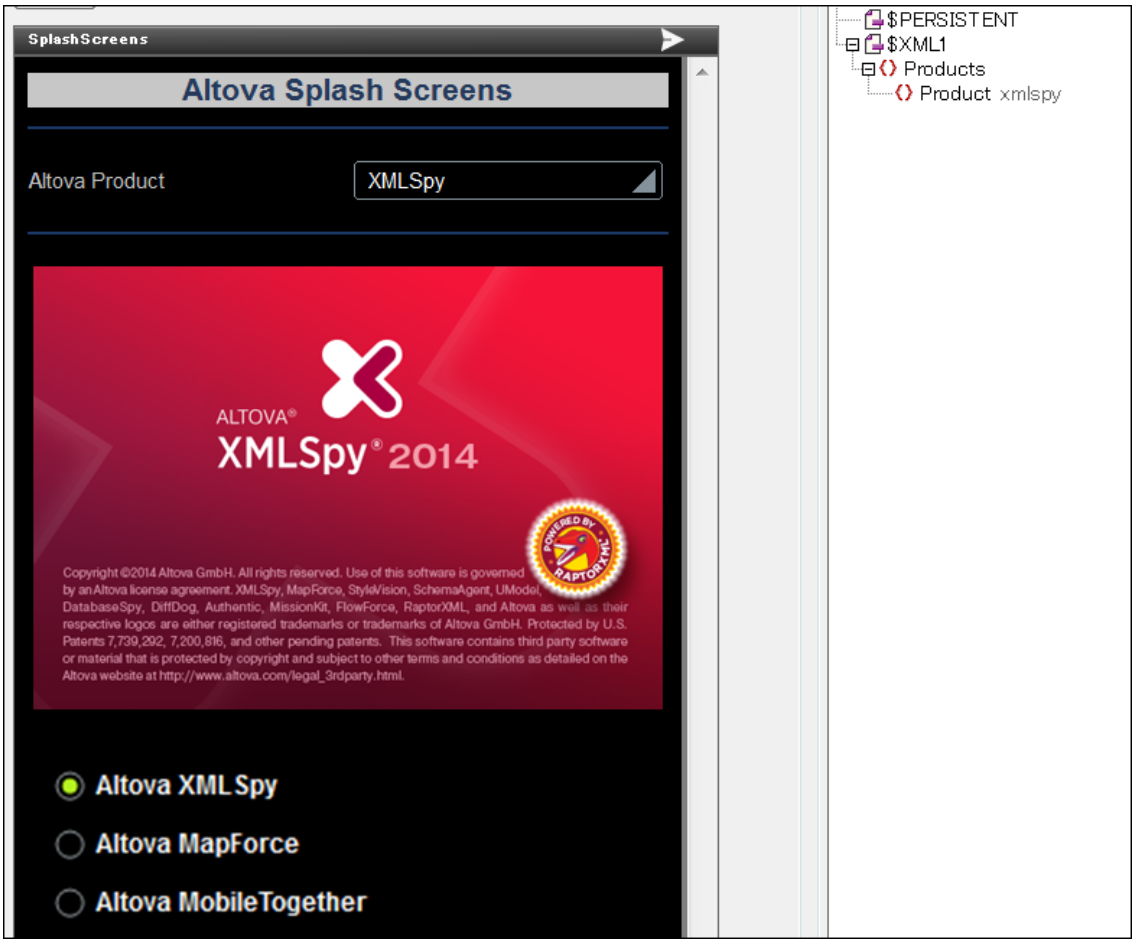
ラジオボタン

ラジオボタンは特定の値のセットの1 つをユーザーが選択することを制約するために使用されます。各ラジオボタンは1 つの特定の値に関連付けられています。 ラジオボタンのセットは、単一のデータソースノードに関連付けることで相互に排他的なセットにグループ化されています。 「相互に排他的」とは、ユーザーがセットから1 つのラジオボタンのみを選択するように制約されていることを意味します。選択されたラジオボタンの値は関連付けられたデータソースノードに入力されます。ラジオボタンコントロールを挿入する時、ラジオボタンがテキストの左、右、またはデフォルトのシステムの位置に挿入するかを指定できます。

例えば、下のスクリーンショットでは、3 つすべてのラジオボタンがXML データソースノードに関連付けられています: 要素は `Product` と呼ばれます。これにより相互に排他的なラジオボタンが`Product` ノードのために作成されます。各ラジオボタンは チェックされた値 プロパティから値を与えられます。ボタンテキストはボタンの テキスト プロパティの値です。



ソリューションが実行されると、最初、3 つすべてのラジオボタンは空白で表示されます (*のスクリーンショット参照*)。ユーザーがラジオボタンを選択すると、ボタンのチェックされた値(サンプル内の製品の名前) は関連付けられたノードにパスされます (`Product` 要素; *下のスクリーンショット内のXML データツリーを参照してください*)。



ラジオボタンの2 つの主要なプロパティは以下の通りです:

- ラジオボタンに付帯するテキスト。このテキストは(Text プロパティの値として入力された) 静的テキストまたは、XPath 式から取得された動的テキストです。
- ラジオボタンのチェックされた値は、チェックされた値 プロパティと共に割り当てられます (*下を参照*)。

エンドユーザーがラジオボタンを選択することでトリガーされる編集の終了時 イベントを持ちます。このイベントのためにアクションを定義するには、コントロールアクション プロパティの **[追加ダイアログ]** ボタンをクリックします。これにより、必要なアクションを指定することのできる[コントロールアクション] ダイアログが表示されます。サンプルでは、イベントはスプラッシュスクリーンのイメージの再ロードをトリガーします。

メモ

- コントロールがデータソースノード(ページソースリンク) と関連付けられると、マウスを(ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの [**XPath の編集**] をクリックします。
- デザイン内で他の場所に [**コントロールをコピーする**] 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために (ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

ラジオボタンイベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

ラジオボタンのプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- テキスト

  テキストプロパティは以下の値から取得されます:

  - コントロールで表示される固定値テキスト文字列
  - データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

  編集する値フィールドをダブルクリックします、または、**XPath** ツールバーボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキスト メニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

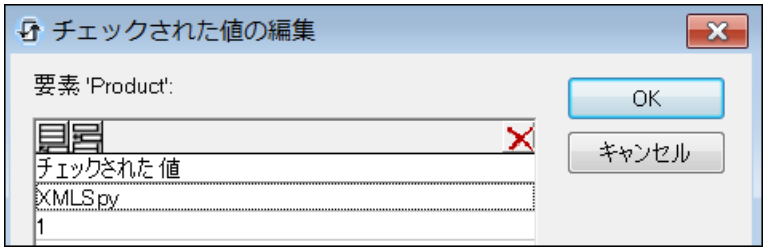
  メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。** 使用された場合、検証エラーの結果が発生します。

- 複数行

  複数行の入力/表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

- チェックされた値

  選択された状態のコントロールのデータのXML 値を提供します。**[追加] ダイアログ** ボタンをクリックして、[チェックされた値の編集] ダイアログを表示します (下のスクリーンショット)。チェックされた(選択された) 状態に対しての値を入力します。ラジオボタンがエンドユーザーにより選択された場合、リストの最初の値がノード内でコントロールと関連付けられたデータとして入力されます。



- XPath から値を取得する

  XPath 式により返された値はコントロールに表示されます。これにより、特定のコントロールに表示する値を入力する別の方法が有効化されます。例えば、コンボボックスはページソースノードまたは **XPath** から値を取得 プロパティの返り値から表示する値を得ることができます。

  メモ **$MTControlValue** 変数を、XPath から値を取得プロパティの値を生成するために使用することは**できません。**使用されると検証エラーの結果が発生します。

- 値の自動修正

  コントロールには 2 つの状態があります: チェックされたおよびチェックされていない状態です。各状態は少なくとも 1 つの XML 値に関連付けられています。これらの XML 値はチェックされた値 プロパティで定義されています。

  値の自動修正 プロパティには 2 つの可能な値があります: true または false です。プロパティが true に設定されている場合、XML 値は、チェックされた値 プロパティのチェックされた、およびチェックされていない状態で定義された値に自動的に修正されます。例えば、チェックされた XML 値が[子供]の値を持ち、そして、チェックされていない XML 値が[大人]の値を持ち、コントロールがチェックされている場合、何かが入力されると、XML 値は[子供]に修正されます。コントロールがチェックされていない場合、XML 値は[大人]になります。1 つ以上の XML 値がチェックされた状態にある場合、最初の XML 値が修正に使用されます。例えば、コントロールがコントロールの状態に無効な XML 値のコンテンツを持つノードと関連付けられている場合、修正が必要です。

- チェックマークの位置

  コントロールのテキストに相対するチェックボックスの位置をテキストの左または右側に設定します: デフォルトはオペレーティングシステムのデフォルトです。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  true() または false() を評価する XPath 式です。式が、false() のみを評価する場合、false() と評価した場合、コントロールは可視ではありません。 式が true() を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは true() です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

有効化された編集可能な

コントロールは、プロパティの値が `true` (有効化) または `false` (無効化) であるかにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の `true` または `false` を評価するXPath 式であることもできます。デフォルトの値は `true` です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が `true` と評価した場合、コントロールが有効化されます。

テキストの色

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

テキストの色 (無効化済み)

コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

テキストサイズ

コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

`mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの `TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2`。関数は **'一番大きい'** サイズに対応する数値 (ピクセル)を生成します。この値は、**'一番大きい'**の120% に対応する値を得るために `1.2` 倍されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係

することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- テキストサイズの自動調整

  テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (true) | オフ(false) | グループ X. デフォルトは、オフ(false)です。

  9つのグループの1つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。 すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

  Multiline プロパティが**true** に設定されている場合、自動的にフィットする機能を使用することはできません。

  メモ　テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストからtrue またはfalse を選択します。デフォルトはfalse です。

- イタリックテキスト

  コンボボックスのドロップダウンリストからtrue またはfalse を選択します。デフォルトはfalse です。

- テキストの下線

  コンボボックスのドロップダウンリストからtrue またはfalse を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、#FF0000)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。

- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 垂直の配置

  垂直の配置を上、中央、下、または下に設定します。デフォルトは中央です。値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。

- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

高さのコントロール

高さのコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 高さのコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 高さのコントロールを、コンテンツが必要なだけの幅にします。

効果としては、`fill_parent` は最大の高さを作成し、`wrap_content` は最小の高さを作成します。イメージおよびチャートコントロール以外の全てのコントロールのデフォルトは、`fill_parent` です。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に `$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は `concat($MT_CanvasX * 0.5, 'px')` になります。

- タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。**「ページ | タブの順序を表示/定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ　タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

リッチテキスト

リッチテキストコントロールにより、フォーマット(スタイル)と共にマークアップされているテキスト内のページソースから書式設定されたテキストを表示することが可能になります。更に、Web クライアントとWindows PC クライアント上で、リッチテキストコントロールを編集、書式設定し、ページソースに戻し保存することができます。リッチテキストの機能の広義の説明に関しては、「オブジェクト機能のデザイン」「リッチテキスト」を参照してください。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

リッチテキストイベント

編集の終了時 イベント を使用することができます。このイベントのために定義することのできるアクションの詳細に関しては、アクションセクションを参照してください。

リッチテキストプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

テキスト

テキストプロパティは以下の値から取得されます:

- コントロールで表示される固定値テキスト文字列
- データソースのノードからデータを取得し、コントロールでこの値を表示するXPath 式

編集する値フィールドをダブルクリックします、または、**XPath** ツールバーボタンをクリックして、必要なテキストを生成するXPath 式を入力します。コントロールの一部にはプロパティ値を入力するための2 通りの方法があります: プロパティを右クリックして、コンテキストメニューから方法を選択します (固定値またはXPath)。他のコントロールに関しては、プロパティの入力方法は1 つしかありません。

メモ `$MTControlValue` 変数を、テキスト プロパティの値を生成するために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして

XPath 式を入力または編集します。可視プロパティは、XPath 式が評価されるかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるかにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true` または`false` を評価するXPath 式であることもできます。デフォルトの値は`true` です。典型的な有効化 /編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true` と評価した場合、コントロールが有効化されます。

- ヒント

  エンドユーザーにテキスト化したヒントを与えます。例えば、エンドユーザーがコントロールを使用して実行するアクションについてのヒントを提供します。プロパティの値フィールドの内側をダブルクリックして、テキスト化したヒントを編集します。または、**XPath** ツールバー ボタンをクリックして、XPath 式を入力して必要なテキストを生成します。

- テキストヒントの色

  コントロールのヒントのテキストの色を設定します。このテキストの色はヒント プロパティで定義されたテキストの色になります(上を参照)。色選択するために次を行います:

  - 色パレットをクリックして背景色を選択します
  - コンボボックスのドロップダウン リストから色を選択します
  - 値フィールドをダブルクリックして色コードを入力します (例えば、`#FF0000`)、または**XPath** ツールバー ボタンをクリックしてXPath 式を入力して必要なテキストを生成します。

- リッチテキストスタイルシート

  デザインのリッチテキストスタイルシートがリッチテキストのコントロール内のテキストのレイアウトのために使用されるかを指定します。プロパティのコンボボックスのドロップダウンリストは、(リッチテキストスタイルシート ダイアログにより作成された)プロジェクトのリッチテキストスタイルシートのすべてを表示します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロ

ールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

リッチテキストの高さ

リッチテキストコントロールの高さを指定します。デフォルトの振る舞いにより、コンテンツが増えるとコントロールの高さがスクリーンの高さまで増加します。高さの最高値まで達すると、コントロールのコンテンツの部分にスクロールバーが表示されます。また、コントロールのために高さを(ピクセルで) 設定することができます。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポ**

イントをデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

  コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- タブの順序

  タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

  タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。**「ページ| タブの順序を表示 定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

  メモ タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

署名フィールド

署名フィールド コントロールは、エンドユーザーの署名をグラフィックファイルとして保管することを有効化します。これはとても役に立つ機能です。例えば、配達ビジネスソリューションでは、署名が配達の受け取りの確認として使用されています。エンドユーザーが署名フィールドに署名を記入すると、Base64 イメージエンコードは、データソースノードに書き込まれます。ノード内で指定されるデータソースが保存されると、Base64 イメージデータが保存されます。

署名イメージには、以下のデフォルトのプロパティがあります。背景色は、ページの背景色の反転です。署名自身は、ページ背景色と同じ色です。イメージの幅はデバイスのビューポートのディメンションより小さく、イメージの高さは幅の半分です。XPath 式を使用して、これらの値が計算されるかは、これらの設定をカスタム化するために使用することのできるコントロールプロパティと共に下のテーブルに表示されています。

署名 プロパティ	デフォルトの値	コントロール プロパティを使用したカスタム値 ...
署名 色	`$MT_PageBackgroundColor`	テキストの色
署名 背景色	`mt-invert-color` (`$MT_PageBackgroundColor`)	背景色
署名 イメージの幅	`min (`$MT_CanvasX`, `$MT_CanvasY`)`	署名作成の幅
署名 イメージの高さ	`min (`$MT_CanvasX`, `$MT_CanvasY`) div 2`	署名作成の高さ

署名フィールドの主な設定は、以下のとおりです:

- ページソースリンクは、署名イメージデータが保管されるデータソースノードです。データソースノードをコントロールにドラッグして、コントロールのページソースリンクを作成または変更します。ページソースリンクを削除すると、関連付けがクリアされます (下のメモ参照)。

1. 署名作成の幅 と署名作成の高さ プロパティ。作成されるイメージのディメンションを指定します。
2. テキストの色 と背景色 プロパティ。署名テキストと背景色を指定します。
3. 保存アクションの一部は、署名イメージデータをデータソースに保存します。このようなアクションが実行されるまで、データは一時的な XML ツリーにのみ保存されています。

メモ

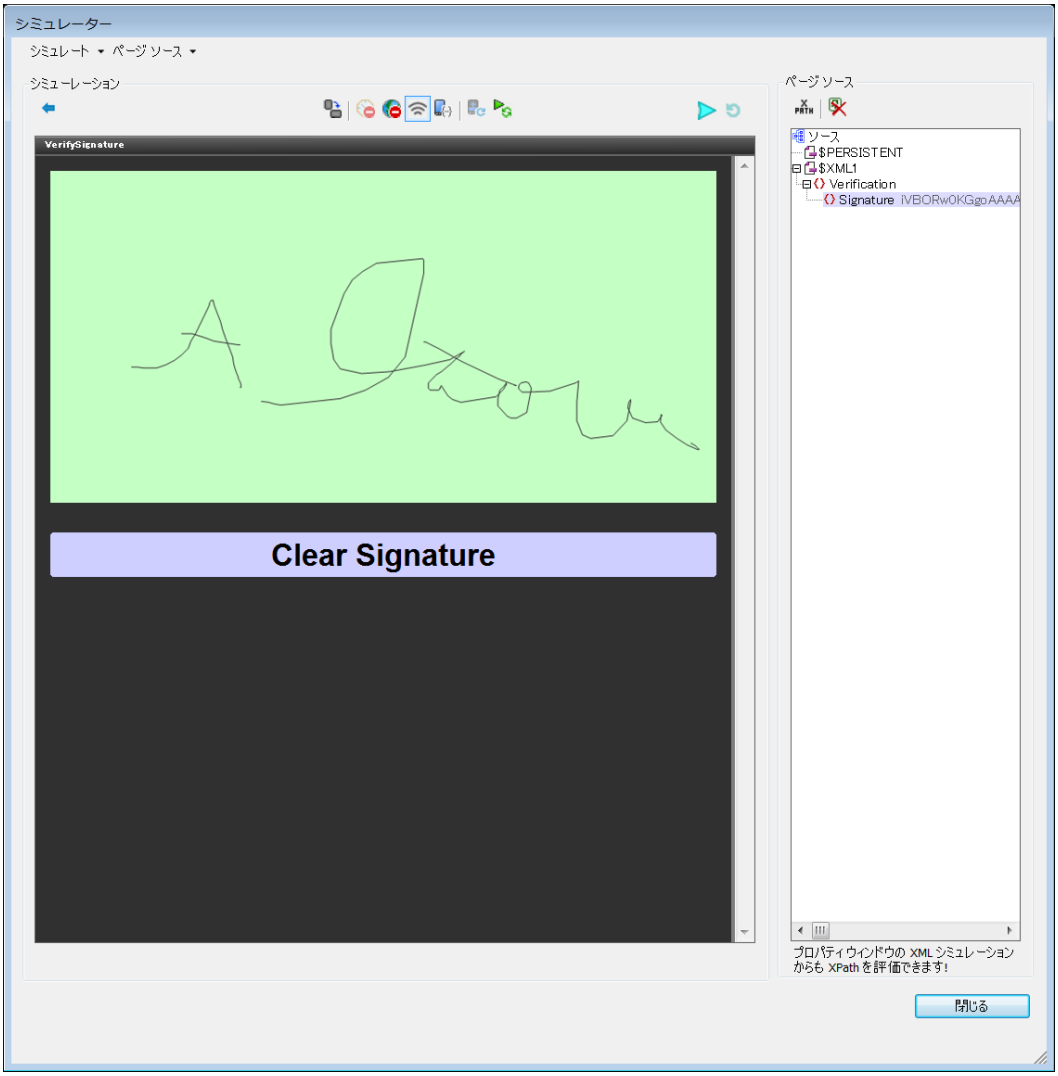
- コントロールがデータソースノード (ページソースリンク) と関連付けられると、マウスを (ページ デザイン ビューの) コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティの XPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの [**XPath の編集**] をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために (ブラウザー CSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。

- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

ユーザーによる署名の編集を許可する

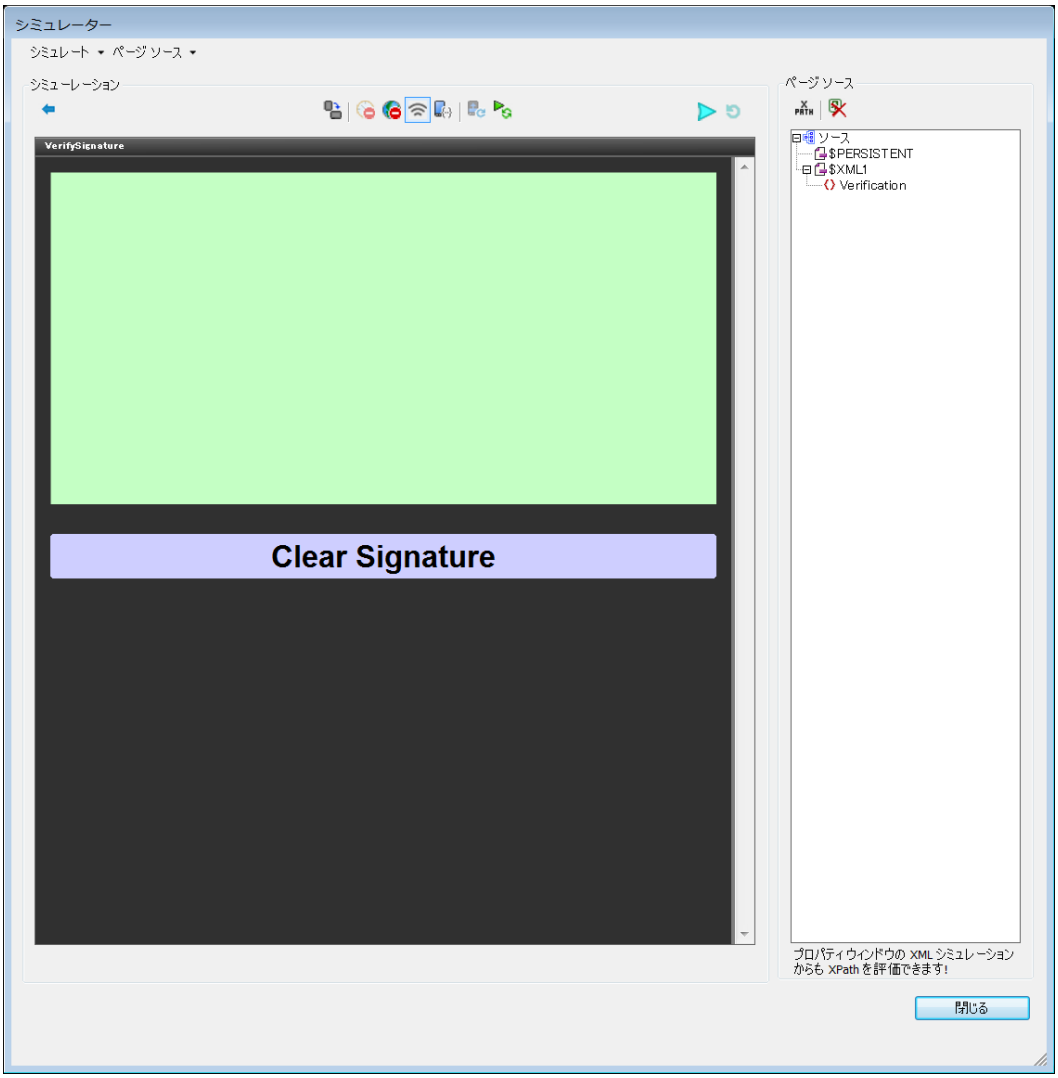
エンドユーザーの署名は、ページソースノード内でイメージとして作成されます。この場合、エンドユーザーは、開始された署名の描画にのみデータを追加することができます。イメージを編集する場合、例えば、エンドユーザーが署名に満足していない場合、イメージデータはノードから削除されなくてはなりません(または、ノード自身が削除される必要があります)、そして署名を再度入力します。この機能を実行する最も簡単な方法は、ノードを削除するボタンコントロールを作成することです。以下の手順をおこなってください:

1. 署名フィールドコントロールの近くにボタンコントロールを作成します(*下のスクリーンショット参照*)。
2. ボタンの クリック時 イベントをしてノードの削除アクションを追加し、署名のページソースノードを削除するノードを設定します。



3. ボタンをシミュレートして確認します。上のスクリーンショットでは、署名が入力されイメージデータが**署名**ノードに保存されています。下のスクリーンショットは、ボタンがクリックされた後に撮られました。ノードが削除され、署名フィールドが

クリアされていることを確認してください。



4. 署名フィールド内に署名が入力されると、新しい署名のイメージデータと共に署名ノードが再び作成されます。

メモ　または、空の文字列を使用して署名のページソースノードを更新するボタンアクションを設定することもできます (*ノードのアップデート アクション参照*)。これによりノードからイメージデータを削除し、このため署名フィールドがクリアされます。ですがノード自身は削除されません。

署名フィールドイベント

署名 フィールド コントロールに関連したイベントは存在しません。

署名フィールドプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

- 名前

  ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

- 可視

  `true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

  メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が `true` (有効化) または `false` (無効化) であるにより、有効化または無効化されています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の `true` または `false` を評価する XPath 式であることもできます。デフォルトの値は `true` です。典型的な 有効化 /編集可能な プロパティの XPath 式は、ノードの存在または値をチェックします。式が `true` と評価した場合、コントロールが有効化されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの 有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- イメージのアンチエイリアシング

  コントロールに関連したイメージに対してアンチエイリアシング(イメージ内のジャギーを削除するテクニック)が使用されるか定義します。 プロパティは、`true` または `false` の値をとります。*ヒント:* イメージの色が後に変更される場合、プロパティの値を `false` に設定することが奨励されます。これは、元のイメージのアンチエイリアシングの色情報がターゲットの色のスキーム内で適切なアンチエイリアシングの色に変換されないためです。

- 署名作成の幅

  署名イメージの幅を指定するピクセルの数。署名が描かれるキャンバスの幅です。例えば、 `400` の値は、署名イメージに400 ピクセルの幅を割り当てます。これは作成されたイメージの幅であり、モバイルデバイスに表示されるイメージの幅では必ずしもないことに注意してください。表示されるイメージは、**幅のコントロール** プロパティ内で定義された幅に相対してスケールされます。例えば、署名イメージが高さ300 ピクセル、幅 400 ピクセル、そして **幅のコントロール** プロパティが `80%` である場合、1000 ピクセルの幅のデバイスでは、イメージは (デバイスの80% である) 幅 800 ピクセルにスケールされます。

- 署名作成の高さ

  署名イメージの高さを指定するピクセルの数。署名が描かれるキャンバスの高さです。例えば、 `200` の値は、署名イメージに200 ピクセルの高さを割り当てます。これは作成されたイメージの高さであり、モバイルデバイスに表示されるイメージの高さでは必ずしもないことに注意してください。表示されるイメージは、**幅のコントロール** プロパティ内で定義された幅に相対してスケールされます。例えば、署名イメージが高さ300 ピクセル、幅 400 ピクセル、そして **幅のコントロール** プロパティが `80%` である場合、1000 ピクセルの幅のデバイスでは、イメージは (デバイスの80% である) 幅 800 ピクセル、(イメージのアスペクトの比率を保つ) 高さ600 ピクセルにスケールされます。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

- コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

高さのコントロール

高さのコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- `fill_parent`: 高さのコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
- `wrap_content`: 高さのコントロールを、コンテンツが必要なだけの幅にします。

効果としては、`fill_parent` は最大の高さを作成し、`wrap_content` は最小の高さを作成します。イメージおよびチャートコントロール以外の全てのコントロールのデフォルトは、`fill_parent` です。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。

$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。
$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。
$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、

ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

スペース

スペース コントロールによりデザインに垂直空間を追加することができます。これはデザイン コンポーネント間のスペースを正確に管理するために役に立ちます。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

スペース イベント

スペースコントロールに関連したイベントはありません。

スペースのプロパティ

[スタイル & プロパティ] ペイン で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

スペースの高さ

選択されたスペース オブジェクトの高さをピクセルで設定します。高さの値 (ピクセル) をコンボ ボックスのドロップダウン リストから選択、または値フィールドをダブルクリックして数値を入力します。数値はピクセルの値で理解されます。このため単位は指定される必要がありません。

可視

`true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。** 使用された場合、検証エラーの結果が発生します。

スタイルシート

スタイルシート プロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウン リストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (次を参照 *ユーザの作成したスタイルシートを*

適用する)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **`LabelClassOne LabelClassTwo`**。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

切り替え

切り替えはデータソース ノードと関連づけられています。エンド ユーザーは次の2 つの状態から選択することができます: *選択されたまたは選択されていない状態*。選択されている状態により、対応するXML 値が関連したノードのコンテンツとして入力されます。(*選択されたまたは選択されていない*)各状態に対応する値がチェックされた値(`true/false`) プロパティで定義されます (下のプロパティの詳細を参照)。典型的なケース: 最初にエンドユーザーの切り替えの選択によりノードのコンテンツを決定します。次に、このノードのコンテンツにより、コントロールの有効化/非有効化などのページアクションを決定します。

メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザインビューの)コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザインビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

切り替え イベント

編集の終了時 イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

切り替えのプロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

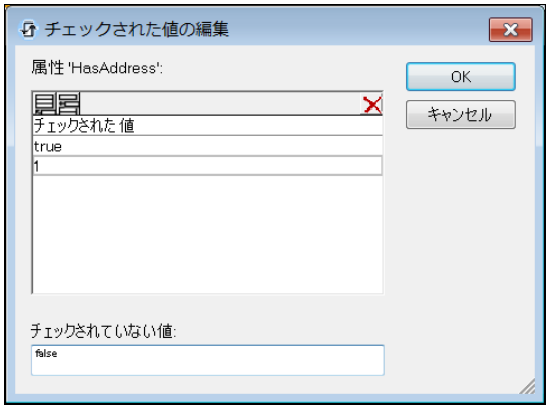
ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

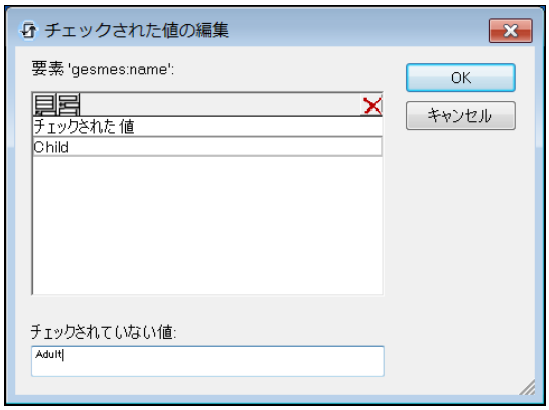
複数行

複数行の入力/表示のオンまたはオフ(`true/false`) を設定します。デフォルトは`false` です。

チェックされた値 (true/false)

選択された/選択されていない状態のコントロールにXML データの値を与えます。**[追加ダイアログ]** ボタンをクリックして、チェックされた値の編集ダイアログを表示します(*下のスクリーンショット*)。チェックされた(選択された) または チェックされていない(選択されていない)状態の値を入力します。エンドユーザーにより選択されたコントロールの状態(選択された/選択されていない)に対応した値がコントロールに関連付けられたXML ノードのデータとして入力されます。





XPath から値を取得する

XPath 式により返された値はコントロールに表示されます。これにより、特定のコントロールに表示する値を入力する別の方法が有効化されます。例えば、コンボボックスはページソースノードまたは **XPath** から値を取得 プロパティの返り値から表示する値を得ることができます。

メモ **$MTControlValue** 変数を、XPath から値を取得プロパティの値を生成するために使用することは**できません。**使用されると検証エラーの結果が発生します。

値の自動修正

コントロールには 2 つの状態があります: チェックされたおよびチェックされていない状態です。各状態は少なくとも 1 つの XML 値に関連付けられています。これらの XML 値はチェックされた値 プロパティで定義されています。

値の自動修正 プロパティには 2 つの可能な値があります: true または false です。プロパティが true に設定されている場合、XML 値は、チェックされた値 プロパティのチェックされた、およびチェックされていない状態で定義された値に自動的に修正されます。例えば、チェックされた XML 値が [子供] の値を持ち、そして、チェックされていない XML 値が [大人] の値を持ち、コントロールがチェックされている場合、何かが入力されると、XML 値は [子供] に修正されます。コントロールがチェックされていない場合、XML 値は [大人] になります。1 つ以上の XML 値がチェックされた状態にある場合、最初の XML 値が修正に使用されます。例えば、コントロールがコントロールの状態に無効な XML 値のコンテンツを持つノードと関連付けられている場合、修正が必要です。

テキストの切り替えオン

コントロールの切り替えがオンになっているとき、表示されるテキストを設定します。値フィールドの内側をダブルクリックして編集します。

- テキストの切り替えオフ

  コントロールの切り替えがオフになっているとき、表示されるテキストを設定します。値フィールドの内側をダブルクリックして編集します。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  `true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ　スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ　`$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 有効化された編集可能な

  コントロールは、プロパティの値が `true` (有効化) または `false` (無効化) であるかにより、有効化または無効化されています。(コンボ ボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の `true` または `false` を評価する XPath 式であることもできます。デフォルトの値は `true` です。典型的な 有効化 /編集可能な プロパティの XPath 式は、ノードの存在または値をチェックします。式が `true` と評価した場合、コントロールが有効化されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバー ボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボ ボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバー ボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合

は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの 有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

- テキストサイズ

  コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

  `mt-font-height` 関数を使用して、他の値を生成することができます。例えば、デバイスの'**一番大きい**'に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの`TextSize` 値として使用します: `mt-font-height('largest', 'px') * 1.2` 。関数は'**一番大きい**'サイズに対応する数値 (ピクセル)を生成します。この値は、'**一番大きい**'の120% に対応する値を得るために`1.2` 倍されます。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- テキストサイズの自動調整

  テキストサイズをコントロールの幅にフィットするようにフォントのサイズの50%まで縮小するかを設定します。プロパティの値は、以下のとおりです: 個別 (`true`) | オフ(`false`) | グループ X. デフォルトは、オフ(`false`)です。

  9つのグループの1 つに対して設定されている場合、コントロールのグループに属するコントロールがグループの番号により識別されます。グループにコントロールを追加する場合、コントロールを選択し、グループに割り当てます。すべてのコントロールのテキストサイズは、自動的にフィットするために計算が行われた後、グループ内の最も小さいフォントサイズと同じ大きさに自動的に調整されます。このプロパティにより、コントロールの選択されたセットが、統一された適切なサイズを有するように設定することができ、試行錯誤することなく、コントロールのグループの最適なサイズを見つけることができます。

  `Multiline` プロパティが`true` に設定されている場合、自動的にフィットする機能を使用することはできません。

  メモ テキストサイズを自動的に調整するはクライアントでは使用することができません。

- 太字テキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントと

は、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

- 余白

  コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

  コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのド

ロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

タブの順序

タブの順序のプロパティは、値として整数を取り、または、整数を評価するXPath 式を取ります。この整数の数はタブの順序のシーケンス内のコントロールの位置です。

タブの順序は、**Tab** キーをクライアントデバイスのユーザーがクリックすると、フォーカスを受けるコントロール内のシーケンスを指します。**「ページ | タブの順序を表示 定義する」** メニュー コマンドを使用してページのすべてのコントロールの視覚的なコンテキスト内のタブの順序のシーケンスの全体を素早く設定することができます。個別のコントロールのタブの順序 プロパティは、そのコントロールのみのシーケンスのポジションを設定します。

メモ タブの順序機能は、Web とWindows クライアント上でのみ使用することができます。

スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、スタイルシートを参照してください。

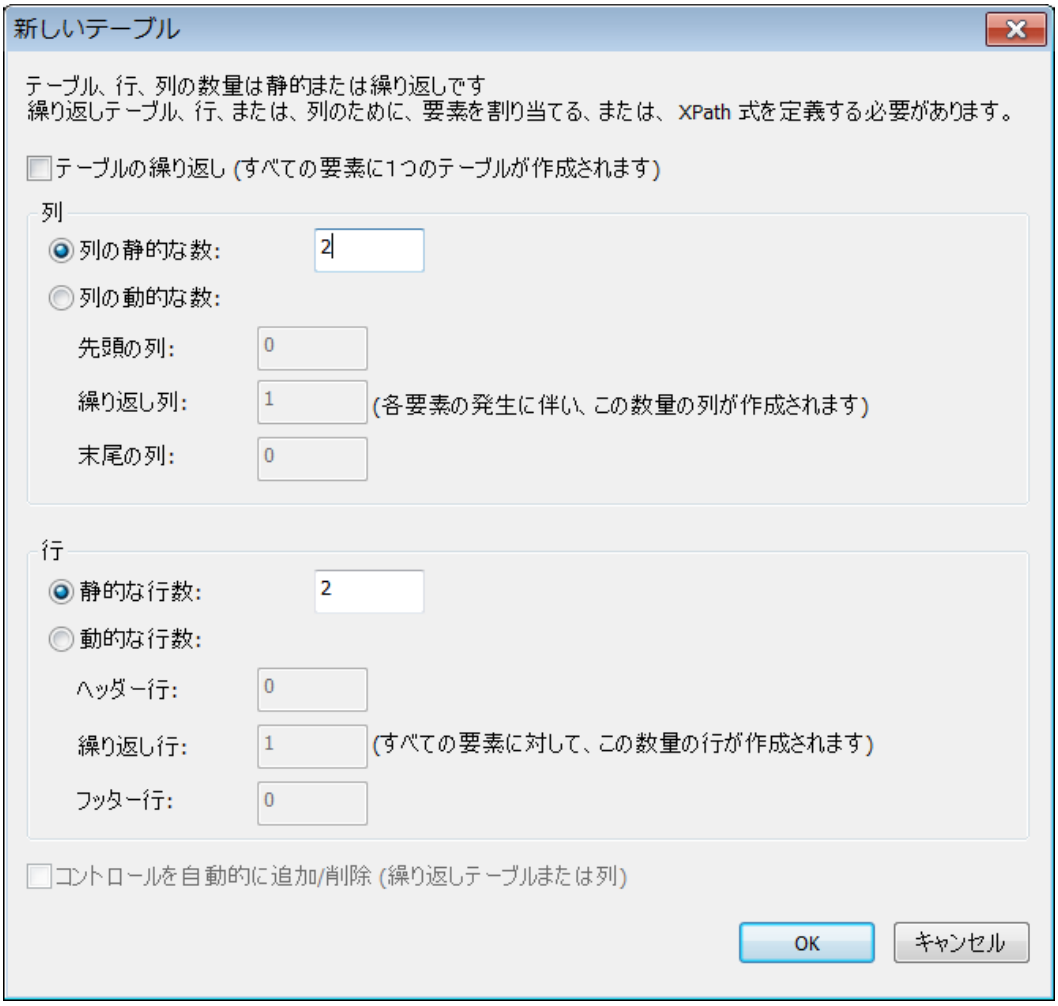
ブラウザーCSS クラス

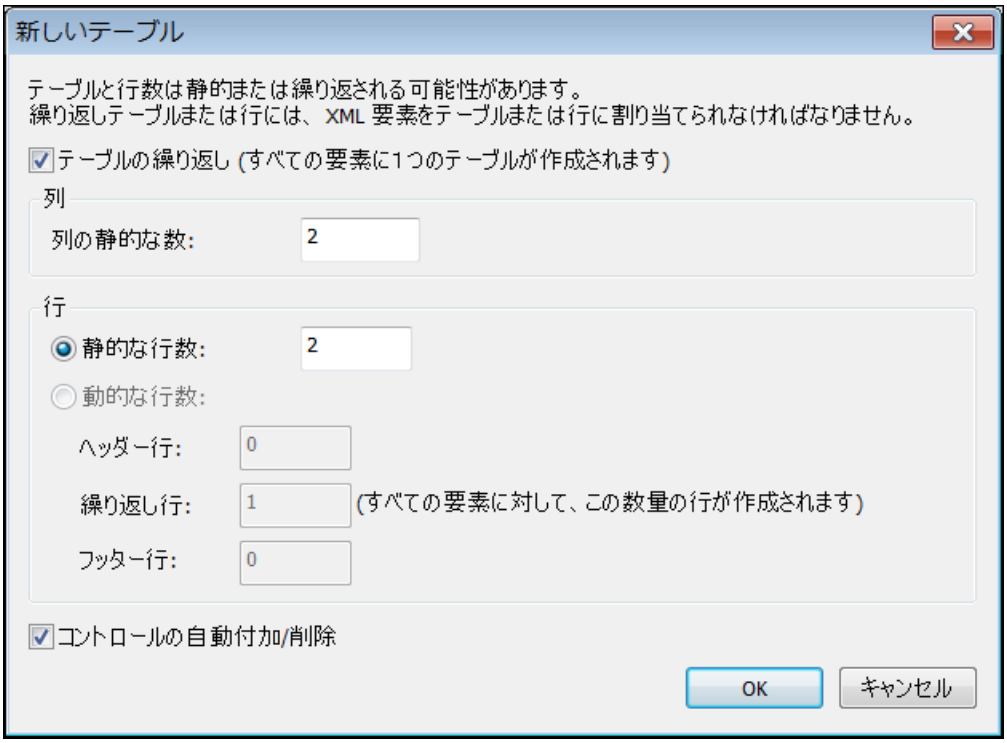
このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

テーブル

[テーブル] コントロールは、静的テーブル、繰り返しテーブル、または 動的 テーブルを挿入します。デザインにコントロールをドロップすると新しいテーブルダイアログが表示されます(*下のスクリーンショット*)。

- 列と行の静的な数を指定すると(*下のスクリーンショット左側*)、繰り返しテーブルが作成されます。
- 繰り返しテーブルは、テーブルに関連付けられた要素が発生する都度作成される新しいテーブル作成されるテーブルです。この種類のテーブルを作成するには、*テーブルの繰り返しチェックボックス*を選択します。要素の繰り返される各*インスタンス*が新しいテーブルとして作成されます。列と行の数は静的な数として指定されます(*下のスクリーンショット右側*)。
- 動的 テーブルでは、テーブル全体ではなく、テーブルの行のグループが繰り返されます。動的 テーブルを作成するには、*テーブルの繰り返しチェックボックス*の選択が解除されていて、行の動的な行数のラジオボタンが選択されていることを確認してください。繰り返される要素の各*インスタンス*が、ユーザーにより定義される行の数として作成されます。列数は、(静的に) 固定、または(動的に)に繰り返すことができます。
- 動的な列が存在するテーブル内では、選択された要素ノードに対応する各行、列が繰り返されます。このように新しい列を動的に対応する要素の発生に対して作成することができます。動的な列を作成するには、列の動的な列のラジオボタンを使用します。 動的な列 繰り返しテーブル と動的な行が存在するテーブルのために生成することができます。





繰り返しテーブルと動的な行が存在するテーブルには、コントロールの自動追加/削除を追加することができます。追加された場合、繰り返される要素の各インスタンスには削除コントロールが横に表示されます。これにより、エンドユーザは要素のこのインスタンスを削除することができます。繰り返される要素が(繰り返しテーブル) テーブルまたは(動的な行を持つテーブル) 行として作成されるかにより、新しいテーブルまたは行 (対応する要素のインスタンス) を追加する追加コントロールが追加されます。

テーブルのセルは(静的および動的) 以下を含むことができます:

- 静的なテキスト
- データソースからのノード
- ページコントロール
- ネストされたテーブル

テーブルのフォーマットのプロパティは[スタイル & プロパティ] ペインおよび、テーブルのコンテキストメニューで使用することができます。

詳細に関しては、テーブルのセクションを参照してください。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

テーブルのイベント

テーブルコントロールに関連したイベントはありません。

テーブルプロパティ

[スタイル & プロパティ] ペイン で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

テーブルセル

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

列グループのスパン

このプロパティは、行グループの最初の列で使用することができます。このプロパティは **true** または **false** の値を取ることができます。デフォルトは、**false** です。プロパティが **true** に設定されている場合、デザイン内の列グループが1つまたは複数の列により構成されているかに関わらず、全ての列グループの全ての列が単一の列にスパンされます。

変換のプロセスは2つのステップにより発生すると考えることができます: (i) **ジョイン** コマンドが使用されたかのように、デザイン内では、列グループ内の(静的な型を含む) すべての列が1つの列にジョインされます。(ii) 出力では、繰り返し要素の全てのインスタンスが単一の列として作成されます。XPath 式: (i) はデザイン内の列に存在します。(ii) は、エラーを返す列を出力するために個別の要素インスタンスを検索します。

下のスクリーンショットは、列グループ内で作成された動的な列の単純なサンプルを表示しています。デザイン内の列グループは、**day** 要素に関連付けられた単一の列グループを含んでいます。この列グループは、(データソース内では **day** 要素の親である) **week** 要素に関連付けられた(繰り返し) テーブル内に存在します。**week** 要素は繰り返すため、新しいテーブルが各 **week** 要素のために作成されます。データソース内で、要 **week** 素の複数の **day** 子要素が存在する場合、そして、デザイン内で、列グループの動的な列がスパンされていない場合、このデザインから生成される (各 **week** のための) テーブルには、**day** 子要素と同じ数の列が存在します。しかしながら、列グループのスパン プロパティを `true` に設定すると、生成されたテーブル内の列はスパンされ、テーブルには1つの列のみが存在するようになります。

列グループに関する詳細は、動的な列 を参照してください。列のスパンに関しては、行 列のジョインとスパンを参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

テーブル列

- 幅

  オブジェクトの幅を設定します。`fill_parent`、`wrap_content` またはパーセンテージの値をコンボ ボックスのドロップダウン リストから選択します。パーセンテージの値は親オブジェクトのパーセンテージです。

  - `fill_parent`: 幅は親と同じ幅になります。例えば、テーブルの行などです。
  - `wrap_content`: 幅はコンテンツに必要とされる幅になります。
  - *パーセントの値*: 親オブジェクトのパーセンテージ。ドロップダウン リストから値を選択します。または直接値を入力します。
  - *ピクセルの値*: ドロップダウン リストから値を選択します。または直接値を入力します。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 幅の最大値

  テーブル幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウン リストから値を選択します、または、直接値を入力します。
  - *ピクセルの値*: ドロップダウン リストからピクセルの値を選択します、または、直接値を入力します。

- 可視

  `true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価し

た場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ　スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ　`$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

背景色に関する注意点

異なる背景色がテーブルの行とテーブルの列に与えられている場合、共通部分にあるセルは、列の色が割り当てられます。この設定は、セルの背景色の設定によりオーバーライドすることができます。

テーブル行

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ　スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ　`$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

▾ ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: **`LabelClassOne LabelClassTwo`**。これらのクラスは、(ブラウザー設定ダイアログで指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

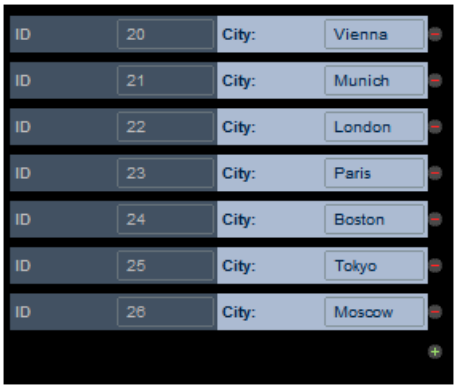
⊟ *背景色に関する注意点*

異なる背景色がテーブルの行とテーブルの列に与えられている場合、共通部分にあるセルは、列の色が割り当てられます。この設定は、セルの背景色の設定によりオーバーライドすることができます。

テーブル行グループ

▾ コントロールの自動追加 削除

構造を表示するためにコントロールを自動追加 削除します。ソリューションの **[追加]** ボタンにより他の繰り返し構造を追加します。ソリューションのボタンは繰り返し構造の各発生に使用することが可能で、その構造の発生を削除します。したのスクリーンショットは、ソシューションの繰り返しテーブル構造の中のこれらのボタンを表示しています。



このプロパティにより **[追加]** および **[削除]** ボタンが自動的にデザインに追加されるか定義します。プロパティの値は `true` または`false` です。値はいつでも変更することができます。

▾ 各アイテムのための作成

プロパティのXPath 式により返されるアイテムの数に対して繰り返されるテーブルの数を設定します。

XPath 式はデータツリーの繰り返しノードを選択することができます。これは、特定のノードの発生を希望する回数生成し、繰り返しの回数を動的に設定する際に役に立ちます。このように、選択されたノードの発生回数は変更し、繰り返されるテーブルの数も自動的に変更されます。例えば、XPath 式のプロパティが`$XML1/Office/Department` とします。そして、テーブルが生成される回数はの`$XML1/Office/Department`要素の数に等しくなります。

XPath 式はデータソースに対する関連付けから解除されることもできます。この場合、作成されるテーブルはから外されることもできます。この場合、返されるシーケンスにテーブルが各アイテムのために作成されます。例えば

- `1 to 9` の式の場合は9 回
- `"John", "Mary"` の場合は2 回
- `45, true()`の場合は2 回

▾ 可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

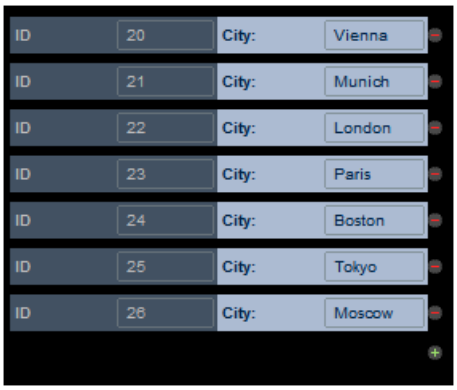
テーブル

▾ 繰り返し

テーブルが繰り返しまたは静的かを定義します。値は`true` または`false` です。このプロパティの値は繰り返し (`=true`) または静的 (`=false`) テーブルが作成された際に自動的に作成されます。特定の種類(繰り返しまたは静的)のテーブルが作成されると、種類はプロパティの**繰り返し**値を変更することで変更できます。動的テーブルには**繰り返し**プロパティを使用することはできません。

▾ コントロールの自動追加 削除

構造を表示するためにコントロールを自動追加 削除します。ソリューションの**[追加]** ボタンにより他の繰り返し構造を追加します。ソリューションのボタンは繰り返し構造の各発生に使用することが可能で、その構造の発生を削除します。したのスクリーンショットは、ソシューションの繰り返しテーブル構造の中のこれらのボタンを表示しています。



このプロパティにより**[追加]** および**[削除]** ボタンが自動的にデザインに追加されるか定義します。プロパティの値は`true` または`false` です。値はいつでも変更することができます。

▾ 各アイテムのための作成

プロパティのXPath 式により返されるアイテムの数に対して繰り返されるテーブルの数を設定します。

XPath 式はデータソリューの繰り返しノードを選択することができます。これは、特定のノードの発生を希望する回数生成し、繰り返しの回数を動的に設定する際に役に立ちます。このように、選択されたノードの発生回数は変更し、繰り返されるテーブルの数も自動的に変更されます。例えば、XPath 式のプロパティが `$XML1/Office/Department` とします。そして、テーブルが生成される回数はの `$XML1/Office/Department` 要素の数に等しくなります。

XPath 式はデータソリューに対する関連付けから解除されることもできます。この場合、作成されるテーブルはから外されることもできます。この場合、返されるシーケンスにテーブルが各アイテムのために作成されます。例えば

- `1 to 9` の式の場合は9 回
- `"John", "Mary"` の場合は2 回
- `45, true()` の場合は2 回

テーブル幅の最大値

テーブル幅の最大値 プロパティは、テーブルの幅: (i) ピクセル値で、(ii) デバイスのスクリーン幅に相対するか、または (iii) 列のために最適化された値 (`wrap_content`) を指定します。デフォルトは、`wrap_content` です。プロパティのコンボボックスのドロップダウンリストから値を選択します。テーブルの幅がスクリーンの幅よりも大きい場合は、テーブルは左右のスクロールバーで表示されます。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

詳細に関しては、テーブルプロパティを参照してください。

テーブルの高さの最大値

テーブルの高さの最大値 プロパティは、デバイス上でのテーブルの高さを、ピクセルまたはデバイスのスクリーンの高さに相対して指定します。プロパティのコンボボックスから値を選択します。例えば、`50%` を選択すると、デバイスのスクリーンの高さの半分の高さのテーブルが表示されます。テーブルが上下に長く、割り当てられたスクリーンのスペースに入りきらない場合、テーブルには上下のスクロールバーが表示され、エンドユーザーはテーブルの残りを、割り当てられたスクリーンのスペースの内外でスクロールする事ができます。(このサンプルではスクリーンの高さの50%)。テーブルの上でデザインコンポーネントが起こる場合、これら全てのコンポーネントはテーブルの上に表示されます。テーブル自身は、このプロパティにより指定された絶対または相対的な高さを持ちます。

ピクセルの値とスクリーンの高さのパーセンテージを除いて、**テーブルの高さの最大値** プロパティは2 つの他の値を取ることができます:

- *スクリーンの高さの残り(最大値):* テーブルの高さは、ページの残りが表示されるようにできるだけ最小化されます。
- *スクリーンの高さの残り(常に):* このオプションによりスクリーンの高さ全体を使用して、ページを表示することが

できます。テーブルにページを埋めるだけの上下の余白がない場合、追加スペースがテーブルの下に追加され、ページの最後のコンポーネントがスクリーンの下のすぐ上に表示されます。

- *iOS デバイス上でのポイントとピクセル*

  長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

  MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

詳細に関しては、テーブルプロパティを参照してください。

- 上下にスクロール

上下にスクロール プロパティは、**テーブルの高さの最大値** プロパティのために値が設定されると使用することができます。1つまたは2つの値をとることができます:

- *テーブル全体*: テーブル全体は、**テーブルの高さの最大値** プロパティを使用してテーブルに割り当てられたテーブルの高さをの内外をスクロールします。*テーブル全体* はデフォルトの値です。
- *ヘッダーとフッターを除く行*: テーブルのヘッダーとフッタはビュー内では固定されています。テーブルのボディ行は残りのテーブルの高さ内でスクロールします。

詳細に関しては、テーブルプロパティを参照してください。

- **行グループチャンクサイズ**

行グループチャンクサイズ プロパティは、スクロールすることのできるテーブルの**テーブルの高さの最高値** プロパティのために値が設定されると使用することができます。ロードされた行グループの数を指定することができます。ユーザーが下にスクロールして、最後にロードされたチャンクの最後の行グループにたどり着くと、次のチャンクがロードされます。このプロパティのためのデフォルトの値は存在しません。

詳細に関しては、以下を参照してください: テーブルプロパティ。

- 可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式がを評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

背景色

オブジェクトの背景色を設定します。以下を行い色を選択します:

- 色パレットをクリックして背景色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

水平の配置

水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

垂直の配置

垂直の配置を上、中央、下、または下に設定します。デフォルトは中央です。値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。

$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、$MT_CanvasX と$MT_CanvasY 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。$MT_CanvasX と$MT_CanvasY 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に$MT_CanvasX * 0.5 等しいピクセルの幅を与えます。イメージの幅の XPath 式はconcat($MT_CanvasX * 0.5, 'px') になります。

- テーブルパディング

Tableテーブル パディング プロパティは、iOS デバイス上でテーブルのためにパディングが使用されるかを設定します。デフォルトの値は、true です。テーブルパディングが使用されると、テーブルの右側に9px のパディングが、左側に5px のパディング与えられます。

- スタイルシート

スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照: ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: LabelClassOne LabelClassTwo。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

- *ネストされたテーブルのプロパティに関しての注意点*

ネストされたテーブルの**全体** が選択されている場合: (i) ネストされているテーブルのプロパティは見出しのテーブルの下にリストされます、(ii) しかしながら、親テーブルのプロパティに含まれるセル、列、行はそれぞれ、以下の見出しの下にリストされます: *親テーブルのセル、親テーブルの行、および親テーブルの列。ネストされているテーブルは上下または左右に配置することができます。*

時刻

時刻コントロールはデータソースノードから得られた時刻をフォーマットするために使用されます。これは特別な方法で時刻をフォーマットする際に役にたちます。日付/時刻　フォーマット文字列プロパティでフォーマットは定義されています（*詳細は下を参照*）。ソースノードコンテンツは、XSD 仕様で定義されたとおりの正確な構文フォーマットである必要があります: `HH:MM:SS`。最適のタイムゾーンとミリ秒は許可されています。

メモ

- コントロールがデータソースノード（ページソースリンク）と関連付けられると、マウスを（ページデザインビューの）コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、（ページデザインビューの）コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- （[スタイル & プロパティ] ペインの）スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- （[スタイル & プロパティ] ペインの）スタイルまたはプロパティの XPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの [**XPath の編集]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために（ブラウザー CSS クラスプロパティを介して）1 つ以上のクラスを定義し、（[ブラウザー設定] ダイアログで指定する）外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールの CSS プロパティは [スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

時刻（コントロール）イベント

編集の終了時　イベント を使用することができます。このイベントのために定義されることのできるアクションの詳細は、アクションのセクションを参照してください。

時刻のプロパティ

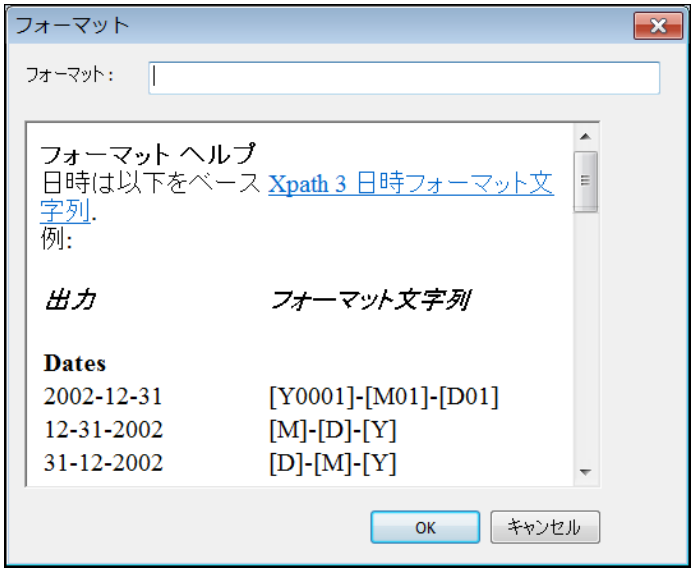
[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

日付/時刻フォーマット文字列

[追加ダイアログ] ボタンをクリックして、表示されるフォーマットダイアログに日付、時刻または日時を入力します（*下のスクリーンショット*）。



フォーマットは、`xs:date` (日付コントロール)、`xs:time` (時刻コントロール)、または`xs:dateTime` (日付、時刻、日時コントロール)構文フォームが正しい場合、コントロールのコンテンツに適用されます。基本的な例は

- *xs:date:* `2014-12-31`
- *xs:time:* `23:59:59`
- *xs:dateTime:* `2014-12-31T23:59:59`

コントロール アクション

[追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

可視

`true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

有効化された編集可能な

コントロールは、プロパティの値が`true` (有効化) または`false` (無効化) であるにより、有効化または無効化さ

れています。(コンボボックスから選択または、値フィールドをダブルクリックして希望する値を入力することで)値を直接入力することができます。値は、ブール値の`true`または`false`を評価するXPath 式であることもできます。デフォルトの値は`true`です。典型的な有効化/編集可能なプロパティのXPath 式は、ノードの存在または値をチェックします。式が`true`と評価した場合、コントロールが有効化されます。

- アサーション

  コントロールのアサーションを有効にするための条件を設定します。アサーションが無効な場合アサーション メッセージのプロパティ(*下を参照*) のテキストはアサーション メッセージ コントロールに表示されます。(複数のアサーション メッセージ コントロールが存在する場合、これらのコントロールは アサーションメッセージ プロパティのテキストを表示します。) プロパティの**XPath** アイコンをクリックして、アサーションを定義するXPath 式を入力します。例えば XPath 式 **`LastName != ""`** はノード`LastName`は空であってはならないとアサートします。もしこのノードが空の場合、コントロールのアサーションメッセージがページのアサーション メッセージ コントロールに表示されます。

  他のコントロールとページもアサーションを持つことができます。複数の無効なアサーションがページに存在する場合、最初の無効のアサーションが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- アサーション メッセージ

  コントロールのアサーションが有効でない場合、アサーションメッセージが表示されるように設定します。プロパティの値フィールドの内側をダブルクリックしてアサーションメッセージを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力し必要なテキストを生成します。アサーションメッセージはアサーション メッセージコントロール内に表示されます。例えば コントロールのアサーションのXPath 式が**`LastName != ""`**の場合、がノード`LastName`は空であってはならないとアサートするとします。は空であってはならないとアサートするとします。もしこのノードが空の場合、コントロールのアサーション メッセージがページのアサーション メッセージ コントロールが挿入された箇所に表示されます。

  複数の無効なアサーションがページに存在する場合、(ページにアサーションが表示される順番の) 最初の無効なアサーションのアサーション メッセージが表示されます。コントロールアサーションはページアサーションの前に評価され、アサーションはページで発生する順番に評価されます。

- テキストの色

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

- テキストの色 (無効化済み)

  コントロールのテキストの色を設定します。以下を行うことにより、色を選択することができます:

  - 色パレットからテキストの色を選択します。
  - コンボボックスのドロップダウン リストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**テキスト**プロパティを使用します。

- テキストサイズ

  コンボボックスのドロップダウンリストからサイズを選択します。許可されている値は、以下のとおりです: 一番小さい|小さい|中|大きい|一番大きい。各プラットフォームまたはデバイスは文字のサイズのための自身のピクセルの高さを有します。ですから、デフォルトのピクセルでのテキストのサイズは、クライアント特有のピクセルの高さであり、中 の値に対応します。

  mt-font-height 関数を使用して、他の値を生成することができます。例えば、デバイスの **'一番大きい'** に対応する数値よりも120% 大きいサイズを表示する場合、次のXPath 式をの **TextSize** 値として使用します: **mt-font-height('largest', 'px') * 1.2** 。関数は **'一番大きい'** サイズに対応する数値 (ピクセル)を生成します。この値は、 **'一番大きい'**の120% に対応する値を得るために**1.2** 倍されます。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**$MT_CanvasX * 0.5** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**concat($MT_CanvasX * 0.5, 'px')** になります。

- 太字テキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- イタリックテキスト

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトは`false` です。

- テキストの下線

  コンボボックスのドロップダウンリストから`true` または`false` を選択します。デフォルトはfalse です。

- 背景色

  オブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- 背景色 (無効化済み)

  オブジェクトが無効化されているときのオブジェクトの背景色を設定します。以下を行い色を選択します:

  - 色パレットをクリックして背景色を選択します。
  - コンボボックスのドロップダウンリストから色を選択します。
  - 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

  メモ　コンテキストに従いコントロールを有効化/無効化することができます。例えば、個人情報に関する書式の場合は、ユーザーは生命保険の有無について入力し、生命保険の詳細に関するフィールドが有効化されるように設定することができます。それ以外の場合は、これらのフィールドは無効化することができます。フィールドを有効化または無効化するには、対応するコントロールの有効化/無効化 プロパティを使用します。オブジェクトが無効化されている場合の背景色を設定するには、**背景色 (無効化済み)** プロパティを使用します。

- ヒント

  エンドユーザーにテキスト化したヒントを与えます。例えば、エンドユーザーがコントロールを使用して実行するアクションについてのヒントを提供します。プロパティの値フィールドの内側をダブルクリックして、テキスト化したヒントを編集します。または、**XPath** ツールバーボタンをクリックして、XPath 式を入力して必要なテキストを生成します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウンリストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

  - *iOS デバイス上でのポイントとピクセル*

    長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

    MobileTogether Designer では、**`$MT_CanvasX`** と**`$MT_CanvasY`** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**`$MT_CanvasX`** と**`$MT_CanvasY`** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**`$MT_CanvasX * 0.5`** 等しいピクセルの幅を与えます。イメージの幅の XPath 式は**`concat($MT_CanvasX * 0.5, 'px')`** になります。

- コントロール幅の最大値

  コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができま

す:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができ

ます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件付に設定できることです (*次を参照* *ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

垂直線

垂直線コントロールによりテーブル内で垂直線を追加することができます。色、幅などのプロパティのために垂直線のスタイルを変更することができます。使用することのできるプロパティは以下にリストされています。**Ctrl**+ ドラッグアンドドロップを使用して、垂直線をコピーして他の場所にコピーする場合は、テーブル内の場所にのみコピーして張り付けることができる点に注意してください。

メモ

- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの **[リセット]** をクリックします。
- デザイン内で他の場所に **[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。

垂直線イベント

垂直線コントロールに関連したイベントは存在しません。

垂直線プロパティ

[スタイル & プロパティ] ペインで使用可能なコントロールのプロパティは以下に表示される順にリストされています。

名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

線の幅

選択された水平線の幅を (太さ) ピクセルで設定します。コンボボックスのドロップダウンリストから幅の値 (ピクセル) を選択します、または値フィールドをダブルクリックして、数字の値を入力します。数字の値はピクセルの値として理解されます。このために、単位は指定する必要はありません。

線のスタイル

選択された水平線のスタイルを指定します。コンボボックスのドロップダウンリストからオプションを選択します。デフォルトの値は実線です。

線の色

選択された水平線の色を選択します。以下を行うことにより、色を選択することができます:

- 色パレットからテキストの色を選択します。
- コンボボックスのドロップダウンリストから色を選択します。
- 値フィールドをダブルクリックして色のコードを入力します (例えば、`#FF0000`)、または、**XPath** ツールバーボタンをクリックして、必要とするテキストを生成します。

可視

`true()` または `false()` を評価する XPath 式です。式が、`false()` のみを評価する場合、`false()` と評価した場合、コントロールは可視ではありません。 式が `true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは `true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして

XPath 式を入力または編集します。可視プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

メモ スパンされた列または行の可視性に関してはテーブルプロパティを参照してください。

メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

水平の配置

水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができ

ます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5`等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')`になります。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシートを表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照* *ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

ビデオ

ビデオコントロールは、のコントロール ビデオソース プロパティ内で指定されたビデオを表示します（*下を参照*）。複数のコントロールをページに設置することができます。各コントロールは、名前で識別され、ビデオソース プロパティ内で指定されたビデオコントロールを再生します。[スタイル & プロパティ]ペインでコントロールのプロパティを設定することができます。(冗長であるため)ローカルのビデオファイルを再生する機能は有効化されていません。ビデオのファイルフォーマットに関する詳細はオーディオ/ビデオ フォーマットを参照してください。

以下の点に注意してください：

- ビデオリソースはURL を使用して検索できます。URL はページソースリンク、XPath 式を使用して、または直接静的アドレスを入力することにより指定することができます。クライアントデバイスで、ビデオファイルを指定するには、デバイス依存ディレクトリに対してURL が相対的である必要があります（*下の ビデオソース プロパティに関する詳細を参照してください*）。ビデオファイルフォーマットに関しては、オーディオ/ビデオ フォーマットを参照してください。
- ビデオをダウンロード中にコントロールのサイズを指定することができます。ダウンロードが完了すると、実際のサイズに合わせてコントロールは拡張または縮小されます。初期の幅 および 初期の高さ プロパティを参照してください。
- ビデオはページがロードされると(ロードすると再生 プロパティ)再生されます。ビデオが後で再生される場合はビデオの開始アクション(例えば、ボタン)を使用します。
- ビデオの再生アクションを管理するボタンがコントロール内で表示されるかを(コントロールの表示 プロパティを使用して)指定することができます。ボタンを表示しない場合は、ビデオアクションはカスタムボタンを作成するために使用することができます。
- ビデオアクションには、ビデオの再生の開始、一時停止、再開、停止、特定の箇所にスキップして時間が定義されているセグメントを再生するなどの機能が搭載されています。
- ビデオイベントがトリガーされると実行されるアクションのセットを指定することもできます（*下を参照*）。
- ビデオファイルおよび再生の属性を与えるMobileTogether XPath 拡張関数の一部は条件付処理を作成するために使用することができます。例えば、`mt-video-is-playing` 関数が`true()` または`false()`を返す場合、代替のアクションのセットを定義することもできます。

メモ

- コントロールがデータソースノード(ページソースリンク)と関連付けられると、マウスを(ページ デザイン ビューの)コントロールにかざすと、関連したノードがポップアップとして表示されます。
- データソースツリーの全てのページソースリンクは太字で表示されています。
- デザインツリー内のページソースリンクにマウスをポイントすると、関連したコントロールの情報が表示されます。
- データソースノードの関連を削除するには、(ページ デザイン ビューの) コントロールを右クリックし、**[ページソースリンクの削除]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティをリセットする場合、プロパティを選択してペインのツールバーの**[リセット]** をクリックします。
- ([スタイル & プロパティ] ペインの) スタイルまたはプロパティのXPath 式を編集するには、スタイルまたはプロパティを選択して、ペインのツールバーの[**XPath の編集**] をクリックします。
- デザイン内で他の場所に**[コントロールをコピーする]** 場合は、**[Ctrl]** を押して、コントロールを希望する場所にドラッグアンドドロップします。
- コントロールのための特定のプロパティを割り当てるには、コントロールのために(ブラウザーCSS クラスプロパティを介して) 1 つ以上のクラスを定義し、([ブラウザー設定] ダイアログで指定する) 外部 CSS ファイル内でクラスのためのルールを割り当てます。
- コントロールのCSS プロパティは[スタイル & プロパティ] ペインおよび/または、外部 CSS ファイルで定義することができます。[スタイル & プロパティ] ペインで定義されたプロパティが優先されます。

ビデオイベント

ビデオイベントは、コントロールのコンテキストメニュー(右クリック) または コントロールの コントロールアクション プロパティ(*下のプロパティ参照*)からアクセスすることができます。ビデオイベントのためのアクションを定義するには、左側のアクションペインからア

クションをビデオイベントタブにドラッグアンドドロップします。

- **ビデオの開始時**: イベントが発生する前に(すなわち、ビデオの再生が開始する前に)、ビデオファイルの詳細を使用することはできない場合は、ビデオの高さ 幅、時間、および現在の位置を取得する関数を呼び出してはなりません(下を参照)。この時点では、`mt-video-is-playing` 関数が有効な情報を返します。このイベントは、例えば、(*ノードの更新アクション*などを仕使用して)ビデオの再生の再生をXML ツリーノードにログするために使用できます。
- **ビデオのエラー時**: 可能性のあるエラー: ファイルが見つからないエラー、ファイルフォーマットのエラー、またはダウンロード/再生の中断エラーなど。MobileTogether XPath 実行関数 `mt-external-error` を使用してエラーに関する情報を取得することができます。アクションがイベントのために定義されている場合、これらのアクションは実行されます。それ以外の場合は、エラーはメッセージボックスに表示されます。
- **ビデオの完了時**: ビデオの再生は(停止アクションが実行されることなく)ファイルまたは指定されてセグメントの再生が最後に到達すると完了したものと考えられます。これらのアクションはビデオが中断(プロジェクトプロパティ*他のソリューションに切り替える*)または一時停止されると**実行されません**。

ビデオプロパティ

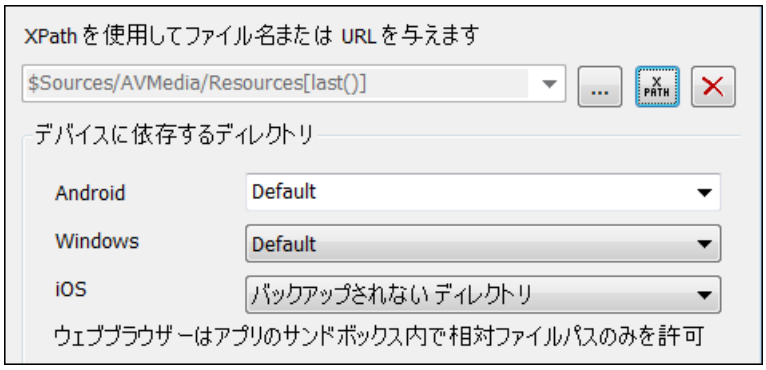
[*スタイル & プロパティ*] *ペイン*で使用可能なコントロールのプロパティは以下に表示される順にリストされています。

▾ 名前

ページまたはプロジェクトの他の場所からのコントロールを参照するために使用されます。値フィールドの内側をダブルクリックして編集します。

▾ ビデオソース

ビデオコントロール内で再生されるビデオを選択します。リモートファイルを検索するURL を入力、または、URL を選択または生成するXPath を入力します。下のスクリーンショットでは、例えば、再生するビデオファイルを検索するために使用される最後の`$Sources/AVMedia/Resource` 要素に含まれているURL などです。



クライアントデバイス上のビデオファイルを指定するには、URL は、それぞれのデバイス依存ディレクトリに対して相対的なURL に対して相対的である必要があります(*上のスクリーンショット参照*)。

- *Android:* Android デバイスディレクトリをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。しかしながら、Android デバイスがルートされていない限り(MobileTogether 以外の)他のアプリは、MobileTogether サンドボックスディレクトリへアクセスすることができません。ですから、他のアプリを使用してMobileTogether サンドボックス内でファイルを開こうと試みると失敗する可能性があります。
- *Windows RT:*Windows Phone または、Windows RT デバイスフォルダーをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレク

トが選択されます。

- *iOS:* MobileTogether は、ユーザーのデバイスに2つのディレクトリを作成します: (i) オペレーティングシステムにより場悪アップされるファイルのためのバックアップディレクトリで、後に使用することができます。例えば、復元時に使用することができます。 (ii) バックアップの必要ないファイルのためのバックアップされないディレクトリであり、再生のため速いバッファーが必要な場合選択されます。バックアップディレクトリまたはバックアップされないディレクトリを必要に応じて選択してください。
- *Web ブラウザー:* 選択できるオプションはありません。相対的パスはブラウザー内のサンドボックスのコンテキストで解決されます。

ビデオファイルフォーマットの詳細に関しては、オーディオ/ビデオ フォーマットを参照してください。

メモ オーディオ/ビデオファイルは、MobileTogether Designer プロジェクトのサーバーメカニズムにデプロイを使用してMobileTogether Server にデプロイすることはで**きません**。しかしながら、URL を使用してストリームすることはできませんが、サーバーに手動でオーディオ/ビデオファイルをコピーすることができます。MobileTogether Server に存在する、オーディオ/ビデオファイルをストリームするには、以下を行います : (i) バイナリのロード アクションを使用して、バイナリのオーディオ/ビデオデータをデータソースノードにロードします。(ii) バイナリの保存 アクションを使用して、このノード内のデータをクライアント上のファイルに保存します。(iii) オーディオ/ビデオの再生アクション を使用して、クライアントデバイス上に保存されたファイルを再生します。または、オーディオ/ビデオファイルをMobileTogether Server に保存する代わりにWeb サーバーに保存し、URL を使用して、Web サーバーからオーディオ/ビデオファイルをストリームします。

キャッシュされたビデオソース

プロパティ**ロードされると再生** (下を参照) が`false` に設定されている場合、ビデオが再生されるファイルからのキャッシュを指定することができます。キャッシュファイルが指定されると、(**ビデオソース** プロパティを使用して)再生のために選択されたビデオファイルがダウンロードされるとクライアントデバイス上でキャッシュされます。ローカルのキャッシュファイルが与えられたビデオコントロールのために既存の場合は、キャッシュファイルが今トールで再生され、新しいダウンロードは行われません。

ローカルキャッシュファイルを指定するには、このプロパティの値とする相対的な URL を入力します。この URL は、デバイス依存ディレクトリペイン内の各 OS のために選択されたフォルダーに対して解決されます。クライアントデバイス上に存在しない場合、相対的 URL のサブフォルダーが作成されるかを指定することができます。

- *Android:* Android デバイスディレクトリをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。しかしながら、Android デバイスがルートされていない限り (MobileTogether 以外の)他のアプリは、MobileTogether サウンドボックスディレクトリへアクセスすることができません。ですから、他のアプリを使用してMobileTogether サンドボックス内でファイルを開こうと試みると失敗する可能性があります。
- *Windows RT:*Windows Phone または、Windows RT デバイスフォルダーをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。
- *iOS:* MobileTogether は、ユーザーのデバイスに2つのディレクトリを作成します: (i) オペレーティングシステムにより場悪アップされるファイルのためのバックアップディレクトリで、後に使用することができます。例えば、復元時に使用することができます。 (ii) バックアップの必要ないファイルのためのバックアップされないディレクトリであり、再生のため速いバッファーが必要な場合選択されます。バックアップディレクトリまたはバックアップされないディレクトリを必要に応じて選択してください。
- *Web ブラウザー:* 選択できるオプションはありません。相対的パスはブラウザー内のサンドボックスのコンテキストで解決されます。

ユーザー名

コンテンツにアクセスするためのユーザー名を設定します。プロパティの値フィールドをダブルクリックして編集します。

パスワード

ユーザーがリソースにアクセスするためにパスワードを設定します。プロパティの値フィールドをダブルクリックして編集します。

- コントロールの表示

  ビデオコントロールのボタンが表示されるかを指定します。プロパティの値は、`true` または、`false` です。デフォルトの値は、`true` です。値が`false` に設定されている場合、再生はビデオアクションにより開始されます。

- ロード時に再生

  ページがロードされた直後にビデオが再生されるかを指定します。プロパティの値は、`true` または、`false` です。デフォルトの値は、`true` です。値が`false` に設定されている場合、再生はビデオの開始アクションにより開始されます。

- コントロール アクション

  [追加ダイアログ] ボタンをクリックして、コントロールのアクション ダイアログを表示します。コントロール イベントがトリガーされるとアクションが実行されるように設定することができます。コントロールのイベントは定義済みで、アクション ダイアログの右側のペインのタブに表示されます。アクションのライブラリは左側のペインに表示されます。イベントのタブに左側のペインからアクションをドラッグして、アクションのプロパティを定義します。各イベントに関しては、複数のアクションを設定することができます。上から下へ発生する順番に実行されます。

  コントロールのアクションを定義した後、プロパティの **[追加ダイアログ]** ボタンをクリックしていつでも編集することができます。

- 可視

  `true()` または`false()`を評価するXPath 式です。式が、`false()`のみを評価する場合、`false()`と評価した場合、コントロールは可視ではありません。 式が`true()` を評価し、または他の値を返す場合、コントロールは可視です。デフォルトは`true()` です。値フィールドをダブルクリックして、または、**[XPath]** ボタンをクリックして XPath 式を入力または編集します。可視 プロパティは、XPath 式が評価するにかかわらず、オブジェクトを可視に表示するために使用できます。この結果、オブジェクトの表示は、データの構造とデータに基づき動的に行うことができます。

  メモ スパンされた列または行の可視性に関してはテーブル プロパティを参照してください。

  メモ `$MTControlValue` 変数は、可視 プロパティの評価のために使用することは、**できません。**使用された場合、検証エラーの結果が発生します。

- 水平の配置

  水平の配置を左、中央、右に設定します。デフォルトは中央です。 値はXPath 式を使用しても定義することができます (これにより値は動的に生成されることができます)。

- 幅のコントロール

  幅のコントロールを設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

  - `fill_parent`: 幅のコントロールを、たとえば、テーブルのセルまたはテーブルなどの、親と同様にします。
  - `wrap_content`: 幅のコントロールを、コンテンツが必要なだけの幅にします。
  - *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウン リストから値を選択するか、直接値を入力します。
  - *ピクセルの値*: ドロップダウン リストからピクセルの値を選択するか、または、直接値を入力します。

  効果としては、`fill_parent` は最大幅を作成し、`wrap_content` は最小幅を作成します。イメージ および チャート コントロール以外の全てのコントロールのデフォルトは`fill_parent` です。`Image` および`Chart` コントロールのデフォルトは`wrap_content`です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

コントロール幅の最大値

コントロール幅の最大値を設定します。プロパティのコンボボックスから値を選択します。以下の値を使用することができます:

- *パーセンテージの値*: ページ幅のパーセンテージ。ドロップダウンリストから値を選択します、または、直接値を入力します。
- *ピクセルの値*: ドロップダウンリストからピクセルの値を選択します、または、直接値を入力します。

初期の幅

ビデオコントロールの初期の幅を指定するピクセルの値。これはビデオのダウンロード中のビデオコントロールの幅です。ビデオが開始すると、コントロールの幅は実際のビデオの幅に拡張または縮小されます。デフォルト初期の高さは`$MT_CanvasX` の値です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と`$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と`$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

初期の高さ

ビデオコントロールの初期の高さを指定するピクセルの値。これはビデオのダウンロード中のビデオコントロールの高さです。ビデオが開始すると、コントロールの幅は実際のビデオの高さに拡張または縮小されます。デフォルト初期の高さは`($MT_CanvasX * 9) div 16` です。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、

ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 余白

コントロールの余白を周りのオブジェクトまたはオブジェクトを含む囲いに相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- 左余白

コントロールの左余白をオブジェクトの左またはオブジェクトを含む囲いの左に相対して設定します。コンボ ボックスのドロップダウン リストからピクセル値を選択、または値フィールドにピクセル値を入力します。

- *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセ

ルの幅を与えます。イメージの幅の XPath 式は `concat($MT_CanvasX * 0.5, 'px')` になります。

右余白

コントロールの右余白をオブジェクトの右またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と `$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と `$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に `$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は `concat($MT_CanvasX * 0.5, 'px')` になります。

上余白

コントロールの左余白をオブジェクトの上またはオブジェクトを含む囲いの上に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、`$MT_CanvasX` と `$MT_CanvasY` 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2つの変数によりデザイン内で使用されるピクセルとして返されます。`$MT_CanvasX` と `$MT_CanvasY` 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に `$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は `concat($MT_CanvasX * 0.5, 'px')` になります。

下余白

コントロールの下余白をオブジェクトの下またはオブジェクトを含む囲いの左に相対して設定します。コンボボックスのドロップダウンリストからピクセル値を選択、または値フィールドにピクセル値を入力します。

iOS デバイス上でのポイントとピクセル

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントと

は、一インチの1/72 に等しい単位ではありません。 iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に`$MT_CanvasX * 0.5` 等しいピクセルの幅を与えます。イメージの幅の XPath 式は`concat($MT_CanvasX * 0.5, 'px')` になります。

- スタイルシート

  スタイルシートプロパティは、コントロールのために使用されるスタイルシートを設定します。スタイルシート プロパティのコンボボックスのドロップダウンリストがユーザーが作成したすべてのスタイルシート を表示します。コントロールのために使用するスタイルシートを選択してください。または、XPath 式を使用して使用するスタイルシートを選択します。後者の方法の利点はスタイルシートの一部を条件月に設定できることです (*次を参照 ユーザの作成したスタイルシートを適用する*)。詳細に関しては、*スタイルシート*を参照してください。

- ブラウザーCSS クラス

  このコントロールと関連付けられる1 つまたは複数のCSS クラス。スペースを使用して複数のクラスを割り当てます: `LabelClassOne LabelClassTwo`。これらのクラスは、(*ブラウザー設定ダイアログ*で指定された) CSS ファイル内で、このコントロールのためにプロパティを割り当てるために使用することができます。

9.2 コントロール イベント

ページ デザインで使用されるコントロールには、関連付けられた定義済みの**イベント**があります(*下のテーブル参照*)。例えば、ボタン コントロールには ボタンのクリック時 に関連付けられたイベントがあります。これらのイベントは**コントロール イベント**と呼ばれ、各イベントには定義づけられた**アクション (コントロール アクション)** があります。MobileTogether ソリューションの実行中にコントロールイベントがトリガーされると、定義されたコントロール アクションが実行されます。

下のテーブルはコントロールとそのイベントのリストです。

コントロール	イベント
アサーションメッセージ	? なし ?
ボタン	ボタンのクリック時
チャート	チャートのクリック時
チェックボックス	編集の終了時
コンボボックス	編集の終了時
日付	編集の終了時
日付 (iOS)	編集の終了時
編集フィールド	入力時
水平線	? なし ?
水平線スライダー	スライド時
イメージ	イメージのクリック時
ラベル	ラベルのクリック時
ラジオボタン	編集の終了時
編集フィールド	? なし ?
スペース	? なし ?
切り替え	編集の終了時
テーブル	? なし ?
時刻	編集の終了時
垂直線	? なし ?
ビデオ	ビデオの開始時 ビデオのエラー時 ビデオの完了時

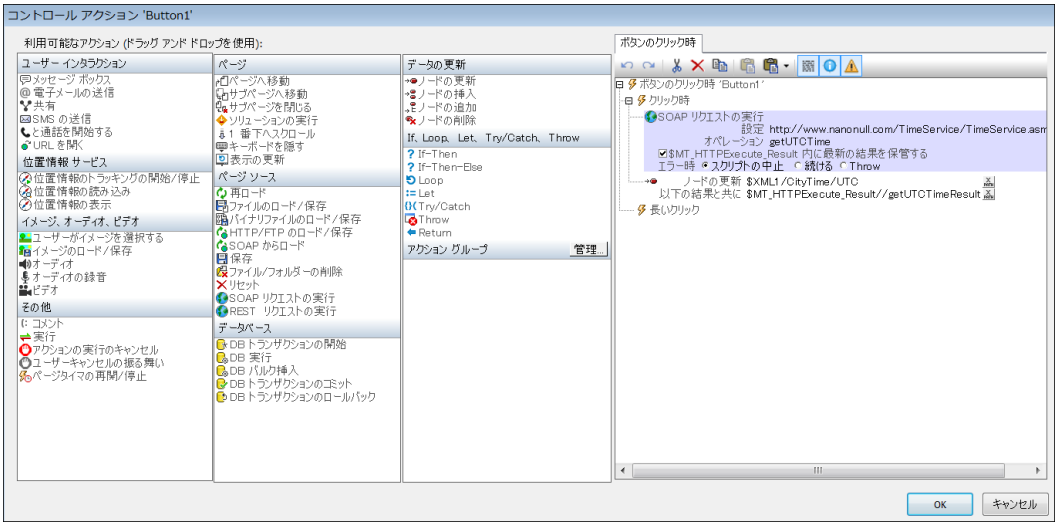
コントロールイベントのアクションの定義

コントロールイベントのアクションは、アクションダイアログ内で定義されています(*下のスクリーンショット参照*)。希望するアクションを左側のペインから右側のペインのイベントタブにドラッグします。

します。コントロールのアクションダイアログ には以下の方法でアクセスすることができます:

- コントロールを右クリックして、[**...に対するコントロール**] を選択します。
- [スタイル & プロパティ] ペインのコントロール アクション プロパティの [**追加ダイアログ**] ボタンをクリックします。

- **[ページ] [アクションの概要]** をクリックして、アクションの概要ダイアログを表示します。定義するコントロール イベントの**[追加ダイアログ]** ボタンをクリックします。



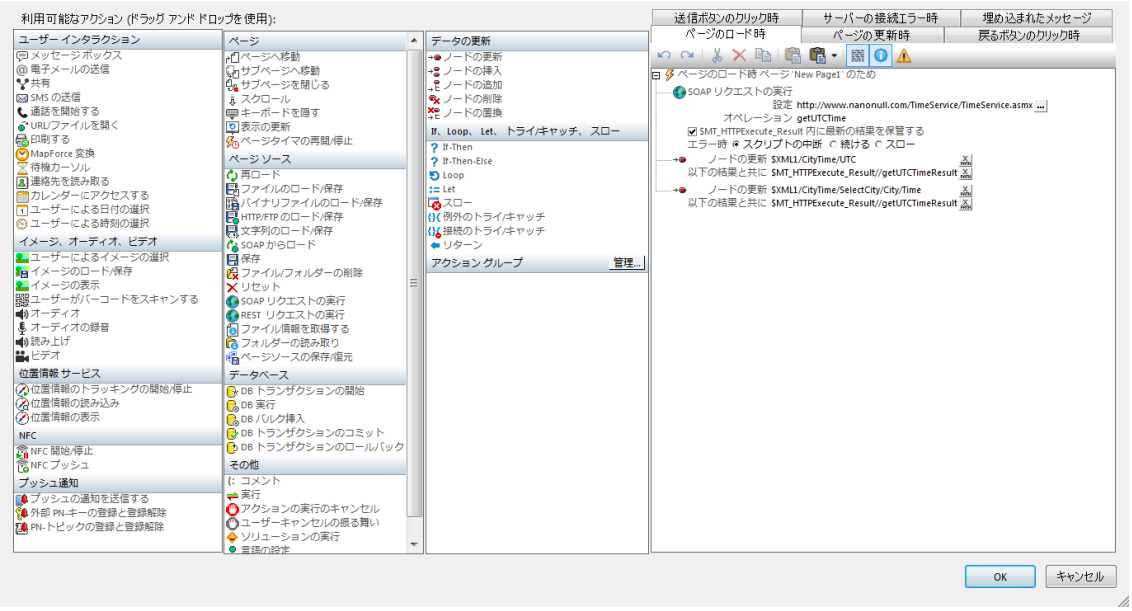
使用可能な異なる種類のコントロール アクションについてはアクションのセクションを参照してください。

チャプター 10

アクション

10 アクション

コントロール イベント とページイベントでアクションを定義することができます。両方の種類のアクションはアクション ダイアログ (*下のスクリーンショット参照*) で定義されます。左側のペインには機能性によりグループ化されたアクションが含まれています。右側のペインにはページまたはコントロール イベントが含まれています。右側のペインで各イベントは個別のタブで表示されており、各イベントのアクションがイベントのタブで定義されています。



選択されたイベントのために特定のアクションを定義するには、左側のペインからイベント タブにアクションをドラッグします。アイコンを使用してアクションを追加することができます。アクションが、コンボ ボックスからのエントリの選択、または XPath 式の入力など、アクションが更なる定義を必要とする場合、(*上のスクリーンショット参照*) ? これらのステップを完了させます。

アクションの追加 アイコン

複数のアクションのためのイベントも定義することができ、アクションに**子アクション**を追加することもできます。上のスクリーンショットでは、例えば、ページのロード時 イベントに 3 つのアクションが定義されており、キャンセル オプションがソリューションの終了アクションをトリガーします。コントロールまたはページ アクションの定義が終わったら**[OK]** をクリックします。同じレベルで複数のアクションが存在する場合、定義された順番に実行されます。

以下のショートカットを使用することができます:

ショートカット	イベントペイン内の選択	結果
[Ctrl]+ダブルクリック アクション	メインイベント (例: ラベルのクリック時)	イベントの最後のアクションとして追加されたアクション
	サブイベント (例: クリック時、長いクリック時)	なし
	アクション	次のアクションとして追加されたダブルクリックされたアクション
Alt+ダブルクリック アクション	メインイベント (例: ラベルのクリック時)	最初のサブイベントの最後のアクションとして追加されたアクション

	サブイベント (例: クリック時、長いクリック時)	サブイベントの最後のアクションとして追加されたアクション
	アクション	なし

アクションまたはアクショングループの使用方法を全て表示する

使用することのできるアクションペイン内のアクションまたはアクショングループを右クリックします。表示されるコンテキストメニュー内の **「メッセージログ内のアクションの使用法を表示する」** をクリックします。デザイン内のアクションの全ての使用法が メッセージペイン 内に表示されます。使用法はページごと、そして、コントロールごとにリストされています。

アクションの処理

(チャートの生成、埋め込まれていないファイルのロード、など) アクションの一部はサーバーで実行され、(メッセージボックス、SMS の送信など) アクションの一部はクライアント上で実行されます。このため、パフォーマンスの向上のためには、アクションの順序は処理のためのサーバーとクライアント間の切り替えを最小化して定義することをおすすめします。

アクションの無効化

イベントのために定義された個々のアクションを一時的に無効化にすることができます。無効化されたアクションは無視され、アクションが定義されていないこととして処理は継続されます。アクションを無効化するには、イベントのアクションの定義を右クリックして、**[アクションの無効]** を選択します。これは切り替えコマンドで、もう一度選択するとアクションを有効化することができます。

再帰的にログを抑制する

ログはメッセージペイン内に表示されるメッセージです。個別のアクションに関連するメッセージとそのサブアクション(子アクション)を抑制することができます。これを行うには、アクションを右クリックし、 **再帰的にログを抑制する」** を選択します。

再帰的にログが抑制されるアクションの子アクションのログを有効化するには、子アクションを右クリックし、 **有効化されたログを再帰的に強制する」** を選択します。

ログを抑制、または、強制的に有効化するアクションには現在の状態を示すアイコンが存在します。下のスクリーンショットは ページへ移動 アクションのアイコンを表示しています。

	オーバーレイ無し
	ログの抑制
	ログの強制的な有効化

イベント ペイン ツールバー

イベント ペイン ツールバー(*下のスクリーンショット*) には左から順に次のコマンドが含まれます:

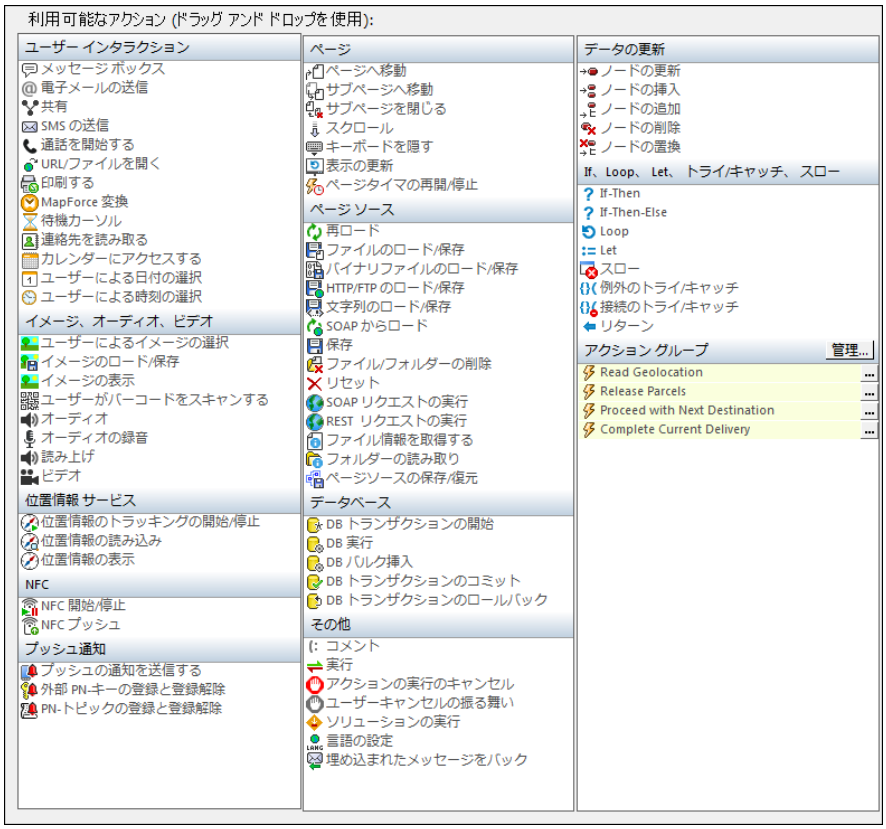
- *元に戻す、やり直す:* イベントペインの編集を元に戻し、やり直します。
- *切り取り、削除:* 選択されたイベントペインのアイテムを削除します。*切り取り*は選択されたアイテムをクリップボードに貼り付けます。
- *コピー:* 選択されたアイテムをクリップボードにコピーします。
- *貼り付け、クリップボードの内容を追加:* クリップボードの内容を選択された場所に貼り付けます。
- *ステートメントを複数行で表示する:* XPath 式などのステートメントの複数行の表示のオン/オフの切り替えをします。
- *ほとんど使用されないオプションを隠す:* ほとんど使用されないオプションの表示のオン/オフの切り替えをします。
- *アクションのための警告を隠す:* 警告の表示のオン/オフの切り替えをします。

(C) 2018 Altova GmbH

10.1 ユーザー インタラクション

アクションダイアログのユーザー インタラクション グループで次のアクションを使用することができます(下のスクリーンショット):

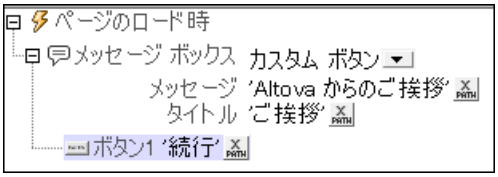
- メッセージボックス
- 電子メールを送信する
- 共有
- SMS を送信する
- 通話を発信する
- URL を開く
- 印刷する
- MapForce 転送
- 待機カーソル
- 連絡先の読み取り
- カレンダーへのアクセス
- ユーザーによる日付の選択
- ユーザーによる時刻の選択

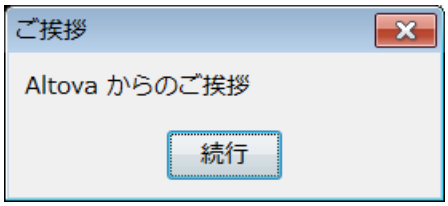


このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

メッセージ ボックス

イベントがトリガーされた時、表示されるメッセージ ボックスです。コンボ ボックスによりメッセージ ボックスに表示されるボタンを選択することができます。定義済みのボタンは**[OK]**、はい、いいえ、キャンセルです。 1 から3 までのカスタム ボタンを定義することもできます。カスタム ボタンのテキストはXPath 式により指定されます (例えば、下の左のスクリーンショットは : **"Proceed"**です)。

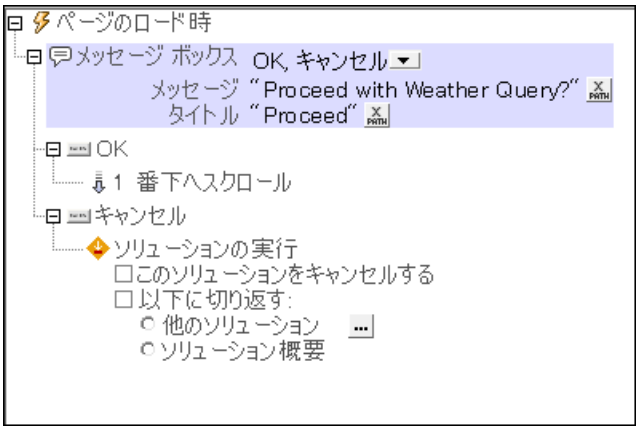




上のスクリーンショットはメッセージ ボックス(右) とその定義(左) を表示しています。定義はカスタム ボタン付きのメッセージ ボックスのためにあります。タイトル、メッセージ、およびボタンのテキストが定義されていることを確認してください。

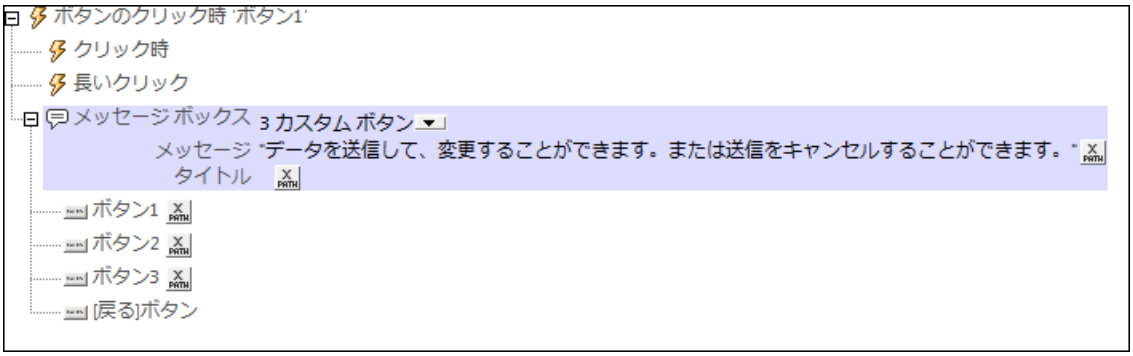
複数のユーザーオプション付きのメッセージボックス

メッセージ ボックスによりユーザーに複数のオプションが与えられます。これは、自身のアクションにリンクされた複数のボタンのフォームにより行われます。例えば、下のスクリーンショットでは、**[OK]** と**[キャンセル]** の2 つのボタンがコンボ ボックスで選択されることにより定義されています。ボタンは、メッセージ ボックスの子オブジェクトとして階層形式のツリーで自動的に表示されます。各ボタンにアクションを割り当てることができます (*スクリーンショット参照*)。これらのアクションはユーザーによりボタンが押されると実行されます。例えば、下のスクリーンショットで定義されている **[キャンセル]** ボタンがクリックされると、ソリューションの終了アクションが実行されます。



カスタムボタン

カスタムボタンにより、シチュエーションに応じてメッセージを構成することのできる柔軟性が与えられます。メッセージを1 つから3 つのボタンで作成することができます。下のメッセージの例は、3 つのボタンを使用しており、それぞれが特定のアクションに関連しています (*スクリーンショットを参照してください*) 。更に、モバイルデバイスの**戻る**ボタンへアクションを割り当てることもできます。**戻る**ボタンの下に対応するアクションをドラッグして行ってください。アクションが割り当てられていない場合、**戻る**ボタンがタップされても何も起こりません。


ボタンのクリック時 'ボタン1'
クリック時
長いクリック
メッセージ ボックス
3 カスタム ボタン
メッセージ "データを送信して、変更することができます。または送信をキャンセルすることができます。"
タイトル
ボタン1
ボタン2
ボタン3
[戻る]ボタン

電子メールの送信

モバイルデバイスの電子メールアプリケーションから電子メールを送信、またはサーバーからダイアログを表示せずに複数の宛先に電子メールを送信します。電子メールはHTML またはテキストとして送信することができます。XPath 式を使用して、宛先、件名、本文のメッセージを指定することができます。更に、添付のテキストとイメージも生成することができます。電子メールの送信アクションの設定は下のスクリーンショットで表示されており、詳細は以下に説明されています。

メモ `mobiletogether:// スキーム`を使用するリンクはGmail 内および他の電子メールアプリケーション内では、作動しませんが、AquaMail、K9、およびMailWise などのよく使用されるクライアントでは作動します。AquaMail およびK9 内でテスト済みでこれらのアプリケーションでは正常に作動します。`mobiletogether:// スキーム`は、リンクを介してMobileTogether ソリューションを開くまたはリンクを介してクライアントデバイス上のサーバー設定を更新するなどのMobileTogether 特有のタスクのために使用されます。

クライアントまたはサーバーから電子メールを送信する

エンドユーザーのクライアントデバイスの電子メールアプリケーション、またはサーバーから電子メールが送信されるかを選択します。下のスクリーンショット(クライアント) と上のスクリーンショット(サーバーから送信される電子メール) を比較してください。サーバーオプションにはサーバーオプションはクライアントオプションよりも以下の3 つの追加設定があります: としてフィールド、差出人フィールド、エラー時のアクション。

- **クライアント**: アクションが実行されると、電子メールを電子メールアプリケーションで開きます。電子メールはアクション設定で指定された詳細 (アドレスフィールド、件名、メッセージ、添付) で埋められます。エンドユーザーはメールを編集し送信、またはお送信せずに閉じることができます。メールがHTML またはテキストで送信されるかは、クライアントにより異なり、デザインで指定することはできません。
- **サーバー**: 電子メールが(MobileTogether) サーバーを使用して送信される場合、MobileTogether Server はISP のSMTP サーバーにアクセスできるように構成されている必要があります。(構成方法の詳細に関しては、MobileTogether Server ドキュメント を参照してください。基本的には、SMTP サーバ

ーアドレスとポート、および差出人の電子メールのユーザー名とパスワードはMobileTogether Server の設定で構成されている必要があります。) エンドユーザーが、アクションをトリガーするイベントを実行する場合、エンドユーザーとの更なるインタラクション無しで、静的に電子メールが送信されます。サーバーを使用して電子メールを送信する選択には、次の3つの追加オプションがあります: としてフィールド、差出人フィールド、エラー時に実行されるアクション。としてフィールドで電子メールがHTML またはテキストとして送信されるかを指定することができます。差出人フィールドは、以下に説明されています。サーバーからのメール送信時にエラーが発生した場合、送信アクションを続行または中止するかを指定することができます。

宛先、Cc、Bcc

これらのフィールドに入力する電子メールアドレスは、XPath 式を介して入力することができます。入力方法には以下の方法があります (i)(上のスクリーンショットに表示されてあるとおり) XPath 式に文字列として直接入力する、または (ii) データソース内のノードから生成する(下の*XPath* 式を参照してください)。複数の受信者これらのフィールドで指定される場合は、シーケンスを返す XPath 式を使用することが奨励されます。 2 つの電子メール間を(セミコロンまたはカンマなどの)セパレーターをハードコードすることは、異なる電子メールクライアントが異なるセパレーターを使用するため、奨励されません。以下は、電子メールアドレスラインを生成するためにXML データソース内のノードを使用する XPath の例です:

```
if ( $MT_iOS=true() ) then iosGroup/Person/Email else otherGroup/Person/
Email
```

式は`Person/Email` ノードのシーケンスを反復し、各反復は、1つの電子メールアドレスを含むことが予想されます。この場合、iOS クライアントと非 iOS クライアントは、複数の宛先を文字列のシーケンスとして与えられます: `("contact1@altova.com", "contact2@altova.com")`。

差出人と返信

MobileTogether Server を使用して電子メールの送信を選択する場合、差出人の設定を使用することができます。この設定内で差出人のメールアドレスを指定することができます。差出人のメールアドレスが必須のSMTP サーバーを使用して電子メールが送信される場合、差出人の設定は入力されている必要があります。SMTP サーバーがこの項目を必要としない場合、差出人の設定を空にすることができます。

受信者が閲覧することのできる「擬似の」電子メールアドレスから自動応答は送信されます。「擬似の」アドレスを使用する場合、このアドレスを差出人の設定に入力します。この場合、受信者からの連絡を受けることができ、実際使用することのできる電子メールアドレスを返信先の設定に入力することができます。受信者が電子メールクライアント内で**返信**コマンドをクリックすると、返信先設定に入力された電子メールアドレスに宛てられた新規の電子メールが開かれます。

電子メールの件名と本文

(電子メールのメッセージと件名のフィールド) の設定 のXPath 式は、文字列またはXML データソースからの各テキストから生成されることができます。電子メールの本文フィールドのためのXPath 式は、有効な HTML サブ構造を持つ`<html>` 要素である(下を参照)構成されたHTML を生成する必要があります。

上のスクリーンショットでは、件名のフィールドのためのXPath 式は、直接入力された文字列です。電子メールのメッセージのためのXPath 式は、`"2015-04-15"` に等しい`date` 属性を持つ、`Message` 要素のテキストコンテキストを返します。

電子メールの本文は構造されているHTML である必要があり、子ノードを含む`<html>` 要素として構造される必要があります。HTML サポートのレベルは電子メールが送信されるソースにより異なります:

- *サーバーまたはiOS クライアントから* 標準 HTML がサポートされます。
- *Android:* `<b>` などの、少ないHTML 構造がサポートされます。
- *Windows Phone とWindows App (タブレットとタッチスクリーンPC)*: HTML サポート無し。

フルフォーマットされたHTML を送信する場合は、サーバーオプションから(送信) を選択します。

添付

ファイルとイメージはメッセージに添付することができます。以下の3 つのオプションから1 つ選ぶことができます:

- 添付無し(デフォルトで選択されています)
- 下にリストされる添付
- 動的な添付

下にリストされる添付

このオプションにより、添付を個別に作成することができます。新しい添付を追加するには、 をクリックします。下のスクリーンショットは2 つの添付を持つメッセージを表示しています。添付を削除するには、**[削除]** アイコンをクリックします。

それぞれの添付は次のプロパティを持ちます:

- *ファイル名 (XPath):* ファイル名は拡張子を持つことができます。ファイル名は、添付の(メッセージ内) 表示としてのみ使用することができ、実際のパスとしては使用することができません。

- *コンテンツ(XPath):* XML ツリー フラグメントを選択することができ、単一のXML ノード、1つ以上のノードのテキストコンテンツ、または添付されたファイルのコンテンツとなる文字列を直接入力することができます。(次の) コンテンツ型プロパティ内の選択に従いコンテンツは解析されます。
- *コンテンツ型 (コンボボックス: XML/テキスト):* コンテンツの型が`XML` の場合、コンテンツプロパティ(前のプロパティ)により選択されたコンテンツは、XML データとして解析されます。XML ノードセットが期待され、ノードセットが電子メールに添付されます。コンテンツの型が`Base64` の場合は、Base64 によりエンコードされたコンテンツが期待されます。また、このコンテンツはデコードされます。ですから、コンテンツがイメージのBase64 エンコードの場合、イメージが生成され電子メールに添付されます。コンテンツの型がテキスト の場合、テキストがコンテンツとして期待され、このテキストが電子メールに添付されます。コンテンツプロパティの値は、コンテンツ型プロパティのための選択に従う必要があります。

動的な添付
XPath 式は、**`mt-email-attachment`** XPath 拡張関数を使用して、添付を作成します。

▾ mt-email-attachment

`mt-email-attachment(Filename` *`xs:string`* を*`xs:string`* として、**`Content`** を *`item()`* として **`, ContentType`** を*`xs:string`* として **`) array(*)`** として
`Content` 引数により与えられたXML、Base64 または、テキストをコンテンツを電子メールの添付として準備します。コンテンツが XML、Base64 または、テキスト として解析されるかは、値として、`XML`、`Base64`、または テキストを受け入れる**`ContentType`** 引数により決定されます。添付に関連したファイル名が**`Filename`** 引数により与えられます。

メモ `mt-email-attachment` は、電子メールの送信アクションの動的な添付のオプションを使用する際の必要条件です。

メモ HTML として送信される電子メールに関しては、電子メールの内容が正確な HTML 、つまり、**`html`** 要素から始まる、HTML である必要があります。有効な内容は次のXPath/XQuery 構成により開始することができます: **`element html { element body { "Test" } }`**

メモ 添付はクAndroid およびiOS ライアントのみで使用することができます。

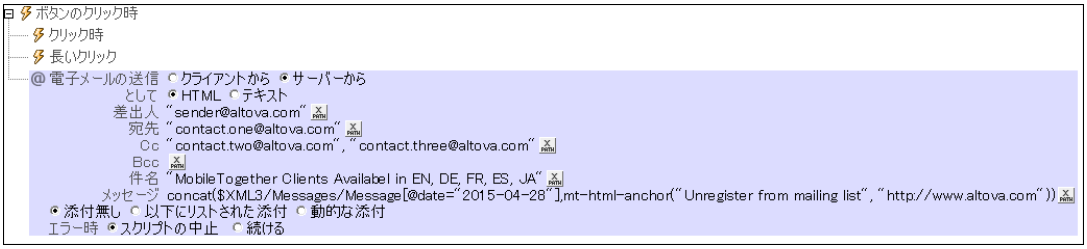
サンプル

- **`mt-email-attachment`**`('MTNewFeatures.txt', $XML2/Releases/Release[@date='2015-04-15']/Features, 'XML')` は 機能ノード を返します。
- **`mt-email-attachment`**`('MTLogo.jpg', $XML4/Images/Image[@name='MTLogo'], 'Base64')` は イメージファイル を返します。

メモ 添付はAndroid とiOS クライアントのみで使用することができます。

▾ 電子メールのメッセージにリンクを追加する

HTML フォーマットで送信される電子メールのテキストにハイパーリンクを追加することができます。この機能は、テキストとして送信される電子メールでは使用することができません。インターネットページまたはMobileTogether ソリューションをターゲットにすることができます。電子メールの本文にリンクを追加するには、メッセージオプションのXPath 式の内の**`mt-html-anchor`** 関数を使用します (*下のスクリーンショット参照*)。



mt-html-anchor 関数は以下の2 つの引数を必要とします: **LinkText** および **targetURL**。HTML ハイパーリンク要素を作成するには2 つの引数を使用します: `<a href="targetURL">LinkText</a>`

サンプル:

```
mt-html-anchor('Unregister from mailing list', 'http://www.altova.com')
```

は、HTML コードフラグメントを生成します:

```
<a href="http://www.altova.com'">Unregister from mailing list</a>
```

上のサンプルは、インターネットにリンクされます。MobileTogether ソリューションにリンクするには、ソリューションへのハイパーリンクを参照してください。

共有

(*下のスクリーンショットでハイライトされている*) 共有アクションを使用することによりユーザーはテキストとイメージを共有することができます。テキストはHTML または テキストとして送信することができ 共有アクションのテキストフィールド内のXPathにより選択されます (*下のスクリーンショット参照*)。更に、テキストとイメージの添付も生成されることができます。共有アクションがモバイルデバイスでトリガーされると、デバイスの共有オプションが表示されます。(これらは、モバイルデバイスにインストールされている、メッセージまたはソーシャルネットワーキングアプリです)。エンドユーザーはこれらのアプリを選択することができ 共有することができます。

- メッセージのタイトルとテキスト

  これら2つの設定(メッセージのタイトルと本文テキスト)のXPath 式は、文字列またはXML データソースからの対応するテキストを生成することができます。

  上のスクリーンショットでは、*タイトルフィールドのための*XPath 式は、直接入力された文字列です、しかし、本文テキストの式は`Location/@Geolocation` ノードのコンテンツを返します。このノードは位置情報トラッキングの開始 / 停止 と位置情報データの読み込み アクションを使用して取得されたモバイルデバイスの位置情報の座標を提供します (*上のスクリーンショット参照*)。

  メモ　HTML/テキストの選択は、iOS デバイスに影響をもたらしません。アプリの一部は既存の`html` タグを自動的に解釈する可能性があります。

- 添付

ファイルとイメージはメッセージに添付することができます。以下の3つのオプションから1つ選ぶことができます:

- 添付無し(デフォルトで選択されています)
- 下にリストされる添付
- 動的な添付

下にリストされる添付

このオプションにより、添付を個別に作成することができます。新しい添付を追加するには、 をクリックします。下のスクリーンショットは2つの添付を持つメッセージを表示しています。添付を削除するには、**[削除]** アイコンをクリックします。

それぞれの添付は次のプロパティを持ちます:

- *ファイル名 (XPath):* ファイル名は拡張子を持つことができます。ファイル名は、添付の(メッセージ内) 表示としてのみ使用することができ、実際のパスとしては使用することができません。
- *コンテンツ(XPath):* XML ツリー フラグメントを選択することができ、単一のXML ノード、1つ以上のノードのテキストコンテンツ、または添付されたファイルのコンテンツとなる文字列を直接入力することができます。(次の) コンテンツ型プロパティ内の選択に従いコンテンツは解析されます。
- *コンテンツ型 (コンボボックス XML/テキスト):* コンテンツの型が`XML` の場合、コンテンツプロパティ(前のプロパティ)により選択されたコンテンツは、XML データとして解析されます。XML ノードセットが期待され、ノードセットが電子メールに添付されます。コンテンツの型が`Base64` の場合は、Base64 によりエンコードされたコンテンツが期待されます。また、このコンテンツはデコードされます。ですから、コンテンツがイメージのBase64 エンコードの場合、イメージが生成され電子メールに添付されます。コンテンツの型がテキストの場合、テキストがコンテンツとして期待され、このテキストが電子メールに添付されます。コンテンツプロパティの値は、コンテンツ型プロパティのための選択に従う必要があります。

動的な添付

XPath 式は、**`mt-email-attachment`** XPath 拡張関数を使用して、添付を作成します。

▾ mt-email-attachment

`mt-email-attachment(Filename` *`xs:string`* を*`xs:string`* として、**`Content`** を *`item()`* として **`, ContentType`** を*`xs:string`* として **`) array(*)`** として **`Content`** 引数により与えられたXML、Base64 または、テキストをコンテンツを電子メールの添付として準備します。コンテンツが XML、Base64 または、テキスト として解析されるかは、値として、`XML`、`Base64`、または テキストを受け入れる**`ContentType`** 引数により決定されます。添付に関連したファイル名が**`Filename`** 引数により与えられます。

メモ `mt-email-attachment` は、電子メールの送信アクションの動的な添付のオプションを使用する際の必要条件です。

メモ HTML として送信される電子メールに関しては、電子メールの内容が正確な HTML 、つまり、**`html`** 要素から始まる、HTML である必要があります。有効な内容は次のXPath/XQuery 構成により開始することができます: **`element html { element body { "Test" } }`**

メモ 添付はクAndroid およびiOS ライアントのみで使用することができます。

サンプル

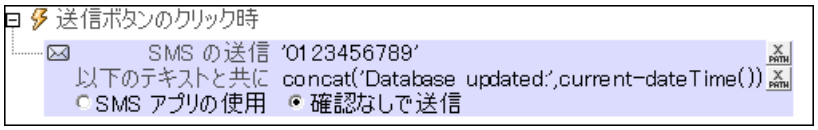
- `mt-email-attachment('MTNewFeatures.txt', $XML2/Releases/Release[@date='2015-04-15']/Features, 'XML')`は 機能ノード を返します。
- `mt-email-attachment('MTLogo.jpg', $XML4/Images/Image[@name='MTLogo'], 'Base64')`は イメージファイル を返します。

メモ 添付はAndroid とiOS クライアントのみで使用することができます。

チュートリアル 位置情報の共有 は共有 アクションの使用方法について説明しています。

SMS を送信する

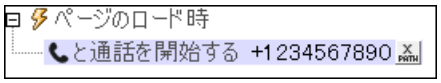
指定された番号に指定されたテキストのSMS を送信します。受信者の番号とSMS のテキストは、XPath 式を使用して指定されています(*下のスクリーンショット参照*)。番号にはかっこを使用しないでください。複数の受信者の番号はセミコロンで区切られなければなりません。**$MT_SMSAvailable** という名の静的なグローバル変数を使用して、クライアントデバイスで通話サービスが使用可能かどうか確認することができます。変数は **true()** または **false()** です。



モバイルデバイスのSMS アプリケーションを使用して、またはダイアログを表示せずにメッセージの送信を選択することができます。

通話を発信する

定義のXPath 式で指定された電話番号に通話を発信します (*下のスクリーンショット参照*)。番号にはかっこを使用しないでください。**$MT_TelephonyAvailable** という名の静的なグローバル変数を使用して、クライアント デバイスで通話サービスが使用可能かどうか確認することができます。変数は **true()** または **false()** です。



URL/ ファイル を開く

クライアントデバイス上でアクション内で指定されているURL 、または、ファイルを開きます。URL は、クライアントのデフォルトのインターネットのブラウザーにより開かれます。アクションの構成オプションから、URL を開く または、ファイルを開くラジオボタンを選択し、詳細を入力してください。

URL を開く

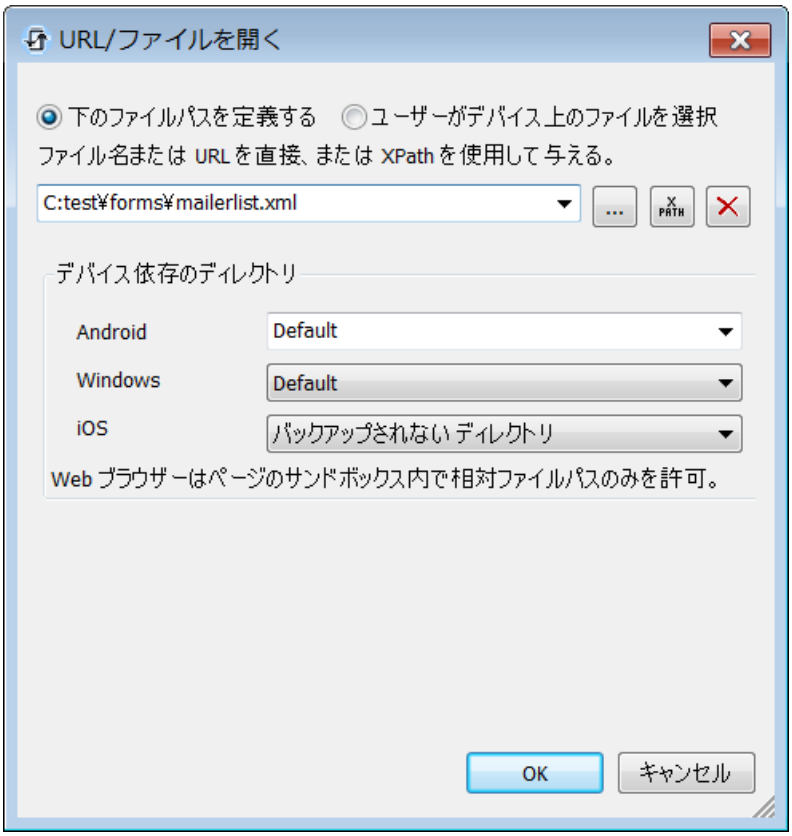
URL を開くアクションのXPath 式 ダイアログ内に、開かれるURL の文字列を評価する式を入力します (*下のスクリーンショットを参照*)。

ファイルを開く

このアクションは、クライアントデバイス上にあるファイルを開きます。デザイン内でファイルを直接指定する、または、エンドユーザーがファイルを選択することができます。ファイルを開くアクションの **編集** ボタンをクリックすると、ファイルを開くためのダイアログが表示されます (*下のスクリーンショット*)。適用するオプションを選択します。

ファイルの場所がデザイン内に保管されている場合

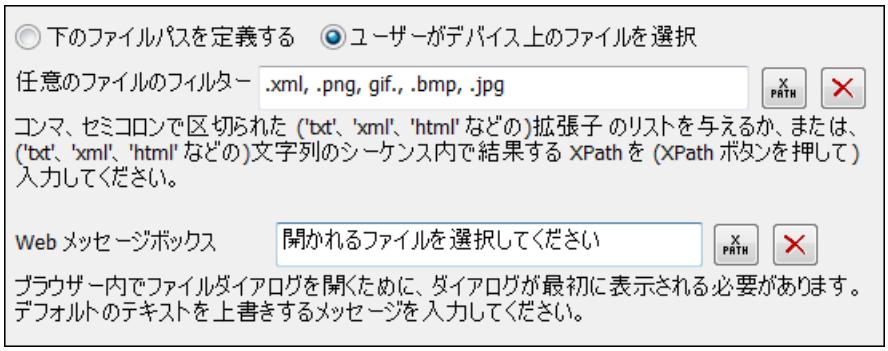
(デザイン内で) 開くファイルを直接指定するためには、「下のファイルパスを定義する」を選択します (*次のスクリーンショット参照*)。絶対的、または、相対的ファイルパス、またはそのようなファイルパスを評価するXPath 式を入力します。相対的パスを入力した場合、このパスは、そのデバイスの型のために指定したベースディレクトリに相対して解決されます (*リストと下のスクリーンショットを参照してください*)。



- *Android:* Android デバイスディレクトリをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。しかしながら、Android デバイスがルートされていない限り (MobileTogether 以外の) 他のアプリは、MobileTogether サウンドボックスディレクトリへアクセスすることができません。ですから、他のアプリを使用して MobileTogether サンドボックス内でファイルを開こうと試みると失敗する可能性があります。
- *Windows RT:* Windows Phone または、Windows RT デバイスフォルダーをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。
- *iOS:* MobileTogether は、ユーザーのデバイスに2つのディレクトリを作成します: (i) オペレーティングシステムにより場悪アップされるファイルのためのバックアップディレクトリで、後に使用することができます。例えば、復元時に使用することができます。 (ii) バックアップの必要ないファイルのためのバックアップされないディレクトリであり、再生のため速いバッファーが必要な場合選択されます。バックアップディレクトリまたはバックアップされないディレクトリを必要に応じて選択してください。
- *Web ブラウザー:* 選択できるオプションはありません。相対的パスはブラウザー内のサンドボックスのコンテキストで解決されます。

ユーザーによりデバイス上のファイルが選択される場合

モバイルデバイス上でユーザーがファイルを選択するには、*ユーザーがデバイス上のファイルを選択する*を選択します (*下のスクリーンショットを参照してください*)。アクションがランタイムで処理されると、エンドユーザーは、ファイルを参照する、または、開くファイルの名前を入力することができます。



このオプションは以下のオプションをします:

- *任意のファイルのフィルター:* クライアントデバイス上で開かれたブラウザーダイアログが開かれるファイルの型を、ここで定義されているファイルの拡張子のみが許可されるようにフィルターします。入力方法: (i) カンマ、または、セミコロンで区切られた拡張子のリスト(例: `txt,html;xml`)、または(ii) それぞれの文字列アイテムがファイル型拡張子である箇所の文字列アイテムのシーケンスを返す XPath 式: `'txt','html,'xml'`)。
- *Web メッセージボックス:* クライアントデバイス上で「ファイルを開く」ダイアログが開かれる前に、メッセージボックスが表示されます。このメッセージボックスのデフォルトのテキストを上書きするためにメッセージを入力することができます。メッセージのテキストを直接 XPath 式を使用して入力します。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ キャッチ アクションの変数内に保管された例外をスローします。トライ キャッチ アクション の キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ キャッチ アクションのセクションを参照してください。

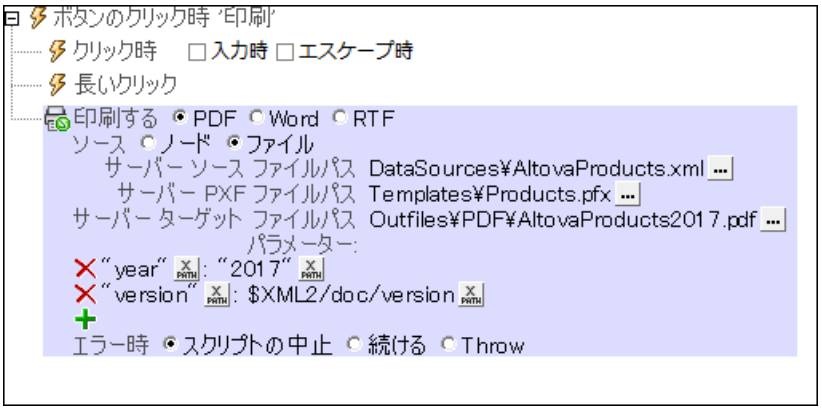
ファイルが存在しない場合のみ、エラーが報告されます。ファイルが存在する場合は、デバイスアプリがファイルを開くことができない場合でも、成功として報告されます。

印刷する

印刷 アクション(下のスクリーンショット) では、Altova StyleVision Server (バージョン2017sp1 または以降) を使用して、PDF、Word。または、RTF 出力ドキュメントを生成することができます。仕組みは次のとおりです: デザインのデータソース ノード または、外部 XML ファイル内のXML データがXSLT スタイルシートであるPXF ファイルを使用して処理されます。Altova StyleVision Server は、XSLT 変換するために使用されます。印刷 アクションを実行するには、Altova Web サイト でダウンロードすることのできるStyleVision Server アプリケーションが(ローカルのシミュレーションのため) とMobileTogether Server (サーバーでのシミュレーションとデプロイで使用されるために) MobileTogether Designer がインストールされているマシンにインストールされている必要があります。

PXF ファイル とStyleVision Server に関するメモ

- ポータブルなXML フォーム(PXF) ファイルは、Altova により開発されたXSLT スタイルシートで、複数の出力を1つのファイルにパッケージするためのファイルフォーマットです。多種のXSLT スタイルシートが単一のデザインから Altova StyleVision を使用して作成することができます。
- デザイン内で、XSLT スタイルシート内に含まれているPXF ファイル内で定義されているパラメーターを定義することができます。ランタイムでは、値がスタイルシート内のパラメーターにパスされることができます。
- StyleVision Server は、多種のフォーマットの出力を生成するXSLT 変換を実行するために使用される軽量のコマンドラインアプリケーションです。
- PXF フォーマットに関する詳しい情報は、Altova StyleVision ドキュメントを参照してください。StyleVision Server に関する詳しい情報は、Altova StyleVision Server ドキュメントを参照してください。



印刷 アクションで以下のオプションをおこなうことができます:

PDF/Word/RTF
印刷される出力ドキュメントの型を選択します: `.pdf`、`.docx`、または `.rtf`.

ソース
以下を選択することができます: (i) デザインの データソースのXML ノードの1つ、または、(ii) クライアントデバイス、または、サーバー上の外部 XML ファイル

ソース ノード
ノードをソースとして選択する場合、出力ドキュメントのためのソースデータとして使用するノードのためのデザインのデータソースツリーを参照することのできる、ダイアログが表示されます。選択するノードはこのオプションの値として入力されます。

ソース ファイル
ファイルをソースとして選択すると、ソース ファイル オプションの **「編集」** ボタンをクリックして、ソースファイルを選択します。

サーバーファイル、または、クライアントファイルを選択することのできるダイアログが表示されます。使用することのできるオプションに関しては、下の*ファイルの場所* セクションを参照してください。ファイルを選択するには、ファイルがサーバーにデプロイされるかを問われます。ファイルがデプロイされない場合、ファイルはソリューションの作業ディレクトリ(または子孫ディレクトリ)に保存される必要があります。この場合、この(*ソース ファイル*) オプションのために入力されるパスは、ランタイムでソースファイルに正確にアクセスするために、正確に設定されている必要があります 。

PXF ファイル
PXF ファイルは、StyleVision Server が出力ドキュメントを生成するために使用するスタイルシートコンテナーファイルです。下の *ファイルの場所* セクション内のサーバーロケーションのために説明されているメソッドの1つを使用するPXF ファイルを選択してください。PXF ファイルをデプロイしない場合、ソリューションの作業 ディレクトリ(または 子孫ディレクトリ) に保存してください。この場合、この(*ソース ファイル*) オプションのために入力されるパスは、ランタイムでPXF ファイルに正確にアクセスするために、正確に設定されている必要があります 。

ターゲット ファイル
出力ファイル名 と(サーバー または クライアント上の) その場所を指定します。下の セクション ファイルの場所 で説明されている方法を使用してファイルの場所を指定します。

パラメーター
ランタイムでは、パラメーター値をPXF ファイルのスタイルシートにパスすることができます。この オプションは、複数のパラメーターボタンを指定することを有効化します。 **「パラメーターの追加」** アイコンをクリックして、パラメーターのエントリを追加します。パラメーターの名前と値をXPath 式として入力します。上のスクリーンショットでは、例えば、最初のパラメーターには **"year":"2017"** の**name:value**ペアがあります。2番目のパラメーターは、ノード `$XML2/doc/version` からの値を取ります。追加することのできる パラメーター値 に制限はありません。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション の キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

ファイルの場所

ソースファイル、*PXF ファイル*、および *ターゲット ファイル* オプションの **「追加ダイアログ」** ボタンをクリックすると、ファイルの指定ダイアログが、表示され、ファイルをロードするか、保存するかを指定することができます。

- *ソースファイル*: サーバーまたはクライアント上にあるXML データ ファイルを選択します。ローカルまたはネットワークマシン上のファイルを選択する場合、以下を行います: (i) デザインのファイルをサーバーにデプロイします、または、(ii) ファイルを ソリューションのサーバー上の作業ディレクトリ(または、子孫ディレクトリ)に保存します。
- *PXF ファイル*: サーバー上にあるPXF ファイルを選択します。ローカル または ネットワーク マシン上にあるファイル

を選択すると、以下をおこないます: (i) サーバーにデザインファイルをデプロイします。または (ii) ソリューションのサーバー上の作業ディレクトリ (または子孫ディレクトリ) にファイルを保管します。 クライアント ベースの PXF ファイルを指定することはできません。

- *ターゲット ファイル:* サーバー または クライアントの場所に出力を生成します。

ファイルの指定ダイアログで使用することのできるオプションは、ファイルがロードされている (*ソースファイル* または *PXF ファイル*) か、または、保存されている (*ターゲット ファイル*) かにより異なります。

サーバー上のファイルの場所

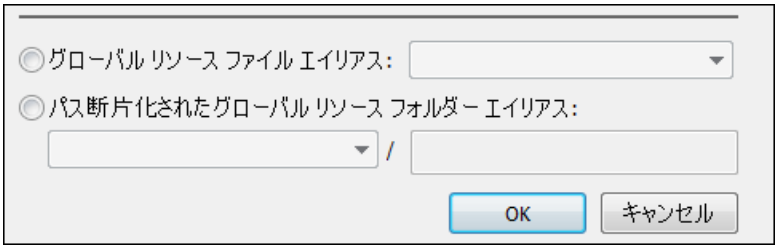
サーバーの場所という用語は、ランタイムのファイルの場所を指します。ロードするファイルを選択する際、ネットワーク上の場所を選択することができます。しかしながら、ランタイムでファイルにアクセスするには、以下を行います: (i) ファイルをサーバーにデプロイする、または、(ii) ソリューションのサーバー上の作業ディレクトリにファイルを保存する。ロードするファイルを指定し、(絶対的な/*相対的な* パス) ファイルの場所を参照することができます、または、グローバル リソースを利用してファイルを指定します (*ファイル エイリアス* または *フォルダー エイリアス*)。ダイアログ内で、使用するオプションを選択します。

- **絶対 相対 パス:** ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細はプロジェクト ファイルの場所を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対/絶対パスがファイルにアクセスするために (サーバーのデータベース) 内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で (MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関してはプロジェクトファイルの場所のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイル パス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

- **グローバル リソース ファイル エイリアス:** コンボ ボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイル リソースを (**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova グローバル リソースのセクションを参照してください。

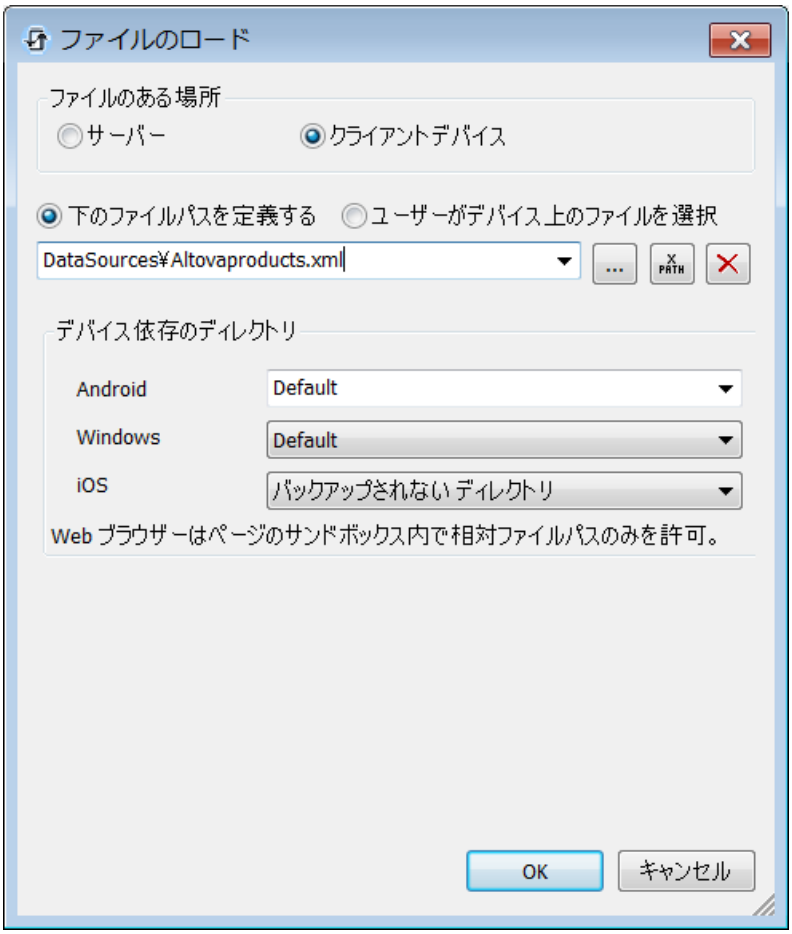
- **パス断片化されたグローバル リソース フォルダ エイリアス:** コンボ ボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アク

ティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがクライアント上にある場合
ファイルがクライアント上にある場合、ファイルの場所のパスを指定するか、XPath 式を用いてパスを構築してください。**「リセット」**ボタンを使用して、現在のエントリを削除します。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (*ソースファイル* と *ターゲットファイル* オプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上の SD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する**　ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。

- **デバイスに依存するリソース**: ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- **シミュレーションのためのファイルのロケーション**: ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ（**ツール | オプション**）内で指定されていなければなりません。

エンドユーザーにより（クライアントデバイス上で）ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子*: ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター*: クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト（例: `txt,html;xml`）、または（ii）各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式（例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`）

- *オプションのデフォルトのファイル*: 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス*: ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、またはXPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する*: もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

  メモ　iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法はiCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

MapForce 転送

MapForce 転送アクションにより、データ構造のセットを他のデータ構造に変換 (マップ) することができます。出力セットの各データ構造はファイル、または、ページデータソースのノードに書き込むことができます。

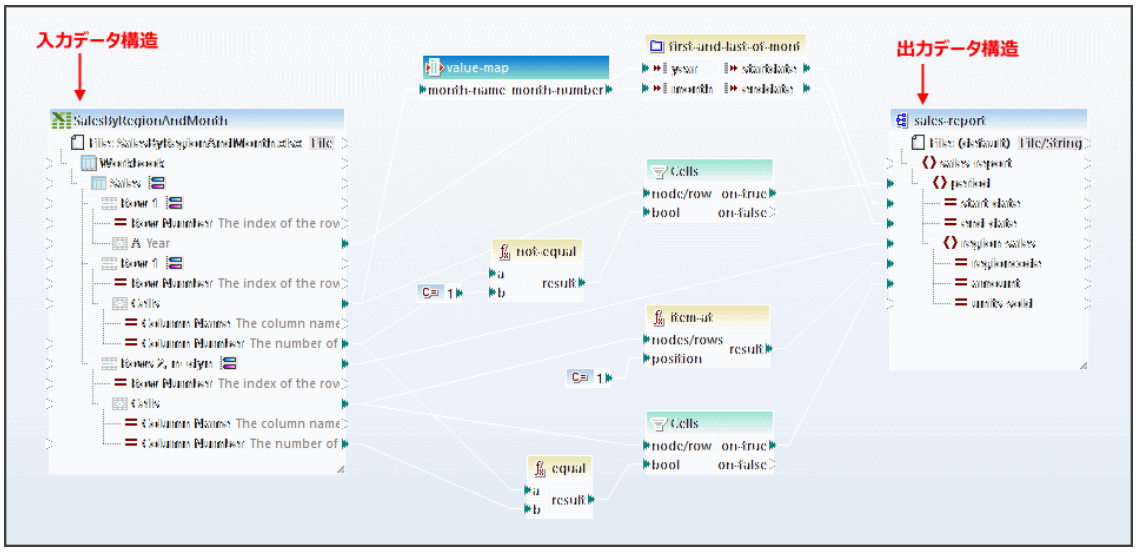
MapForce 転送を実行するには、次の追加 コンポーネントを使用する必要があります:

- 入力データ構造のセットのマッピングを出力データ構造のセットにデザインする*Altova MapForce* 。マッピングがデザインされると、MapForce Server 実行可能ファイル (MFX、または、`.mfx`) として生成されます。
- *Altova MapForce Server* は MobileTogether Server と同じマシンにインストールされている必要があります。MobileTogether により呼び出され、MFX ファイルを処理し、出力データ構造を生成します。

メモ MapForce 転送アクションをローカルシミュレーションまたはクライアント上でトライアルランでテストする場合、MapForce Server 内のアクションは MobileTogether Designer と同じマシンにインストールされている必要があります。

MapForce デザイン

MapForce デザインは Altova MapForce 内で作成されます。入力データ構造を1つ、または、複数の出力データ構造にマップします。下に表示されるサンプルデザインには、Excel ファイルである1つの入力データ構造と、XML ファイルである1つの出力データ構造が存在します。各データ構造 (または、MapForce 用語でコンポーネント) には名前が存在します。デザインは複数の入力コンポーネントと複数の出力コンポーネントを持つことができます。



マッピングデザインが完了すると、MapForce の **「ファイル | MapForce Server 実行可能ファイルにコンパイルする」** コマンドを使用して MFX を生成します。ランタイムでは、MapForce Server により MFX ファイルが使用され出力データ構造を生成します。

MobileTogether の MapForce 転送アクションを使用するためには、MapForce デザインの次の詳細を確認する必要があります:

- 使用する入力と出力コンポーネントの名前。(上に表示されるデザイン内には次が存在します **`SalesByRegionAndMonth`** と **`sales-report`**)。複数の入力と出力コンポーネントを指定することができます。デザイン内で、複数の入力コンポーネントが1つの出力データ構造を生成するために必要とされる場合、(入力と出力) パラメーターとして MapForce 転送アクションに提供される必要があります。
- MapForce デザインの一部では、値パラメータはマップされた出力データ構造を生成するプロセスの一部として使用

されます。このようなパラメーターはそれぞれ値を取り、出力データ構造を生成します。正確に出力データ構造を生成するために、このようなパラメーターの名前を確認する必要があります。

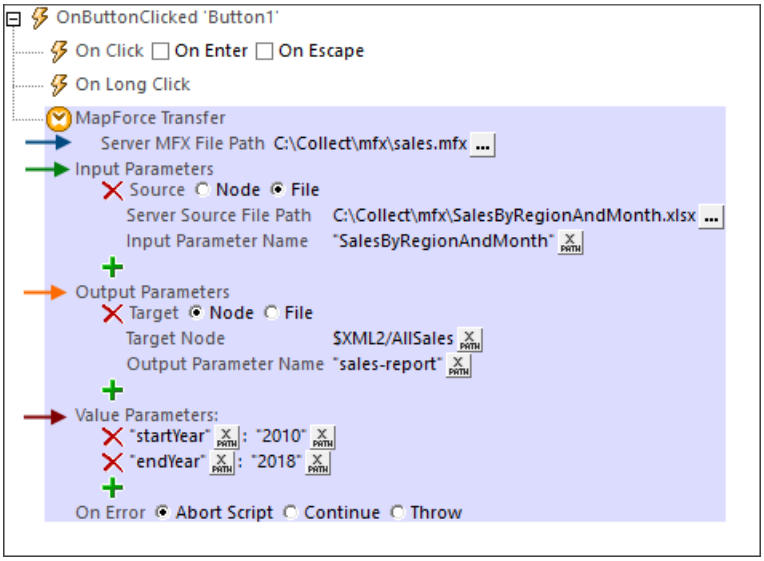
MapForce デザインとマッピングの作動方法に関しての詳細は、次を参照してください：MapForce ユーザーマニュアル。次を参照してください：MapForce Server ユーザーマニュアル。

MobileTogether の MapForce 転送アクション

MapForce 転送アクションは MapForce Server を呼び出し処理するための MFX ファイルと入力データを提供します。下のスクリーンショットは上の示されるマッピングデザインをベースにした MapForce 転送アクションを表示します。

アクションの設定は以下のとおりです:

- 使用する MFX ファイルを検索するサーバー *MFX ファイルパス*。
- *入力パラメーター*: 各入力パラメーターをこの入力コンポーネントのために使用されるデータと共にマッピングの入力コンポーネントを指定します。複数の入力パラメーターを指定することができます。必要とする出力データ構造を正確に生成するために必要とされる全ての入力コンポーネントを指定してください。*入力パラメーター名* 設定は入力コンポーネントの名前 (MapForce デザインから) を指定します。ページソースノード、または、ファイルから入力コンポーネントのためにデータを提供することもできます。ノード、または、ファイルを選択します。 下のスクリーンショット内で指定されている入力コンポーネントは **`SalesByRegionAndMonth`** です。このコンポーネント Excel データ構造であるため、入力コンポーネントと同じ同じ構造を持つ Excel ファイルが提供されます。これはマッピングが成功するために必要なステップです。



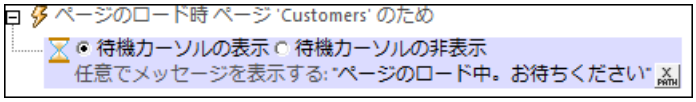
- *出力パラメータ*: 各出力パラメーターはマッピングの出力コンポーネントを出力データ構造が書き込まれる場所と共に指定されます。複数の出力パラメータを指定することができます。*出力パラメーター名* 設定は (MapForce デザイン内で) 生成する出力コンポーネントの名前を指定することができます。ページソースノード、または、ファイルに出力データ構造が書き込まれるかを選択します。ノード、または、ファイルを選択します。上のスクリーンショット内で指定されている出力コンポーネントは **`sales-report`** で、**`$XML2`** ページソースのルート要素 (**`AllSales`**) に保存されます。出力データ構造全体が指定されたノードにコピーされることに注意してください。XML フラグメントが生成されると、XML フラグメント全体が指定されたノードにコピーされます。ターゲットノードの構造に生成されたデータ構造が合うことを確認してください。
- *値パラメータ*: これらはマッピング内で使用するための入力として値を与える *名前 :値* ペアです。ペアの名前の部分は、MapForce 内の入力コンポーネントの名前です。*値* の部分はランタイムにこのコンポーネントにパスされる値です。作成する値パラメータの数に制限はありません。

メモ ランタイムでは、このアクションのために入力されるパラメーター設定はMapForce Server への呼び出しのパラメーターとして送信されます。このため、(入力、出力、値パラメータのセット全体の)各パラメーター名 は一意である必要があります。

メモ MapForce Server はMobileTogether Server と同じマシンにインストールされている必要があります。

待機カーソル

待機カーソル アクション (*下のスクリーンショット参照*) がトリガーされると、プラットフォームに固有の待機カーソルがクライアントに表示されます。任意で、追加メッセージが同時に追加することができます。待機カーソルの表示は、待機カーソルの非表示アクションがトリガーされるまで継続されます。MobileTogether タスクに時間がかかる場合は、待機カーソルの表示はユーザーにタスクが進行中であることを通知するために役に立ちます。



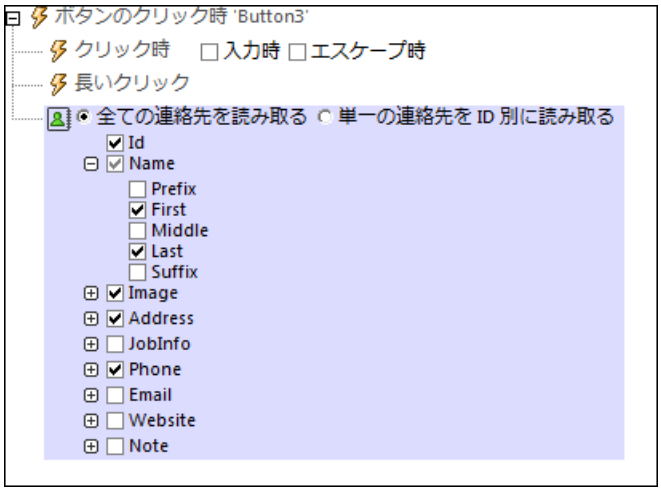
待機カーソル アクションを以下のように使用します:

1. (待機カーソルアクションで待機カーソルの表示を選択して) 待機カーソルの使用を希望するアクションの前に待機カーソル アクションを追加します。
2. 待機カーソルの使用を希望するアクションを追加します。待機カーソルの表示の兄弟 (または、子) アクションとして追加します。
3. (待機カーソルアクションで待機カーソルの非表示を選択して) 待機カーソルの非表示アクションを追加します。(表示するためのアクション (ステップ 2) が完了すると、待機カーソルは自動的に表示になります。

アクションのシーケンスを含むイベントがトリガーされると、以下が発生します: (1) 待機カーソルが表示され (2) 待機カーソルの使用を希望するアクションがトリガーされます。(3) このアクションが完了すると、待機カーソルの非表示アクションがトリガーされカーソルの表示が停止されます。待機カーソルアクションを使用してカーソルの表示を停止することができますが、必須ではありません。

連絡先の読み取り

連絡先の読み取りアクションがデザインに追加されると、**$MT_CONTACTS** データソースツリーは自動的にデザインに追加されます(*下のスクリーンショット*)。アクションがトリガーされると、デバイスの アドレス帳から連絡先が読み取られ、**$MT_CONTACTS** ツリーに保存されます。



アクションを定義する際に、以下を指定することができます:

- すべての連絡先を読み取る、または、ID に基づき一件の連絡先のみを読み取る。
- 各連絡先のデータから読み取り保管するフィールドを選択する。これは読み取るフィールドをクリックすることにより行うことができます(*スクリーンショットを参照してください*)。

メモ ID はプラットフォームに依存します(また、単一のプラットフォームのバージョン間により異なる場合もあります)。このため、連絡先のID を検索するには、全ての連絡先を(ID と共に)読み取り、他のフィールドをベースに必要な ID を検索することが必要となります。

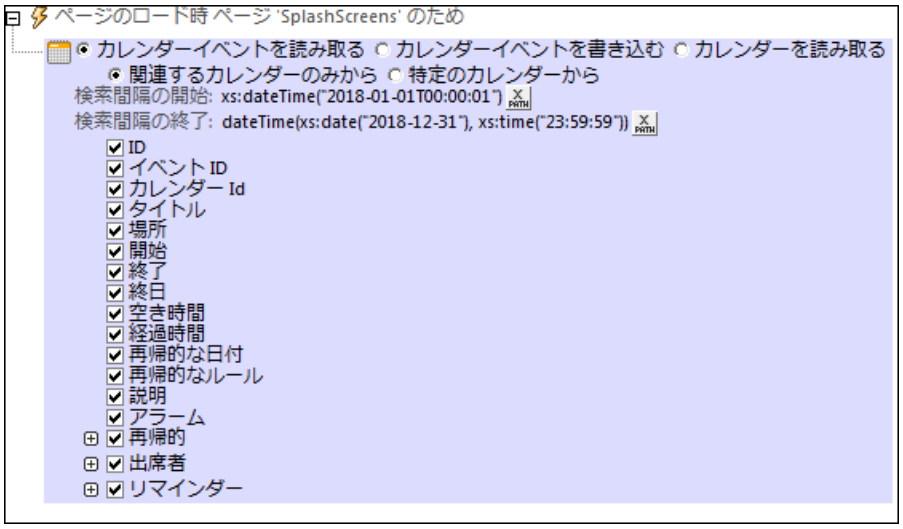
メモ すべての連絡先のすべてのフィールドを読み取る作業は時間とメモリを必要とします。このため、連絡先とフィールドの読み取りの幅を狭める必要があります。例えば、2つのステップにより行うことができます: (i) 名前とID を読み取る。(ii) 最後の読み取りのために、必要とするフィールドで必要とするID のみを読み取る。

メモ デバイスのアドレス帳をシミュレートするには、サンプルの連絡先ファイルを作成し使用することができます。詳細に関しては、次を参照してください: 連絡先のサンプルファイル。または、オプションダイアログ.のシミュレーションタブ内の対応するオプションを選択し、アプリケーションからの連絡先を使用することができます。

カレンダーへのアクセス

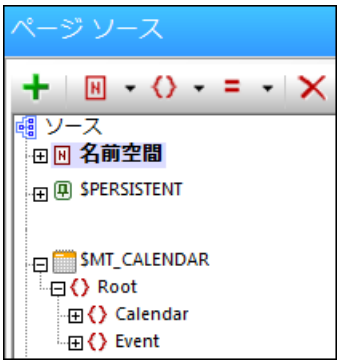
カレンダーへのアクセス アクション (下のスクリーンショット) がデザインに追加されると、**$MT_CALENDAR** データソース ツリーがデザインに自動的に追加されます。 ランタイムでは、選択されたカレンダーアクションにより、(i) デバイスのカレンダーからの情報が読み取られ、**$MT_CALENDAR** ツリー内に保管、または、(ii) カレンダー イベントのためのエントリがデバイスのカレンダーアプリ内で開かれます。ユーザーはこのエントリを編集し、保存することができます。

メモ　Windows 8 クライアントデバイスはカレンダーイベントをサポートしないことに注意してください。



カレンダーのアクションには3つの種類が存在します:

- *カレンダーの読み取りイベント:* デバイス上のカレンダー内のイベント情報を読み取り、この情報を **$MT_CALENDAR** ツリー内に保管します。各イベントは個別の**Event** 要素に保管されます (*下の* **$MT_CALENDAR** *ツリー構造を参照してください*)。ツリー内のデータは後にソリューション内で使用されます。

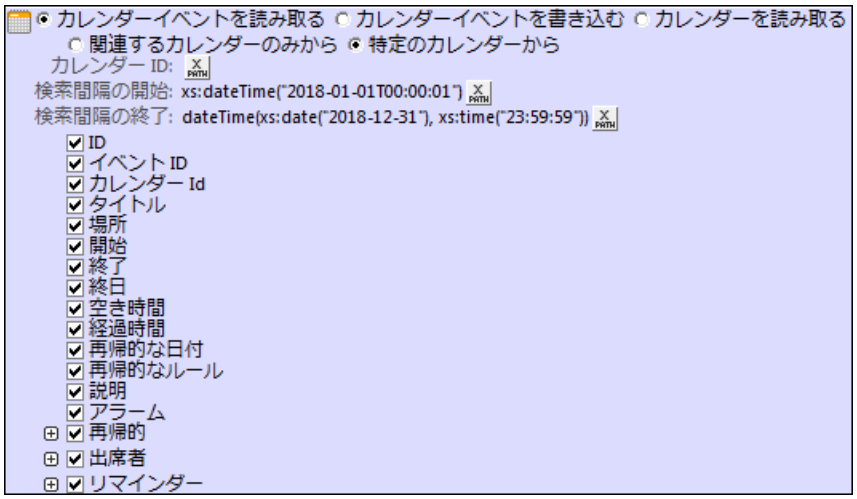


- *カレンダーへの書き込みイベント:* ランタイムでは、エンドユーザーデバイスのカレンダーアプリ内でイベントエントリを開きます。このイベントエントリは、アクションの設定内に入力された情報と共に記入されます。エンドユーザーはエントリを編集し、デバイスのカレンダーにエントリを保管することができます。
- *カレンダーの読み取り:* デバイス上のカレンダーに関する情報を読み取り、この情報を**$MT_CALENDAR** ツリーに保存します。各カレンダーは個別の**Calendar** 要素として保存されます (*下の* **$MT_CALENDAR** *ツリー構造のスクリーンショットを参照してください*)。ツリー内のデータはソリューション内で後に使用することができます。

これら3つの種類のカレンダーアクションは下で更に詳しく説明されています。

カレンダーの読み取りイベント

このアクションはデバイス上のカレンダー内のイベントに関する情報を読み取ります (*下のスクリーンショット参照*)。デバイス上の関連するカレンダーのみ選択することもできます (*上の カレンダーの読み取りを参照してください*) 、または、ID 別にカレンダーを読み取るように指定することもできます(文字列のシーケンスとして複数の ID が与えられます)。`$MT_CALENDAR` ツリー内の `Event` 要素として各イベントは保管されます。



読み取るイベントのデータフィールドを選択することができます:

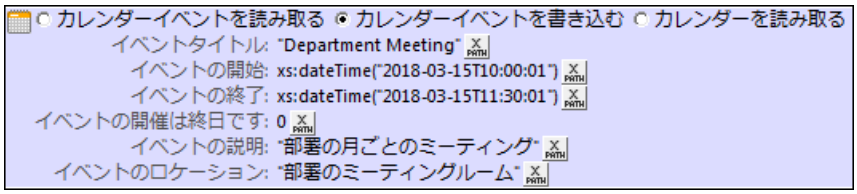
- *ID*、*カレンダー ID*: カレンダーの識別子である文字列。
- *イベント ID*: イベントの識別子である文字列。
- *タイトル*: イベントの名前。
- *場所*: イベントの場所。
- *開始*、*終了*: イベントの開始と終了時刻。
- *終日*: イベントが終日かどうか。*終日* は開始と終了時間が指定されていると設定されます。このプロパティは `true` の値を持ち、それ以外の場合は `false` の値を持ちます。
- *空き時間*: カレンダーのユーザーの空き時間です。
- *経過時間*: イベントの期間を分数で表します。
- *再帰的な日付*、*再帰的なルール*: イベントの再帰的な日付と(例えば 毎週、または、毎木曜日など)再帰的なルールです。
- *説明*: イベントの説明。
- *アラーム*: アラームが設定されているかどうか: はいの場合は、`true` 、いいえの場合は `false` です。
- *再帰的*: イベントが再帰的に発生する開始と終了時間です。
- *参加者*: 各参加者の詳細は個別の *参加者* アイテム内に保管されています。
- *リマインダー*: リマインダーの頻度と、リマインダーのメソッドなどのリマインダーの詳細です。

メモ カレンダーに特定のフィールドの情報が含まれていない場合、そのフィールドのために返される値はありません。

カレンダーへの書き込み イベント

When このアクションが実行されると、デバイスのカレンダーアプリが開かれ、アクション内に入力されたデータを含むイベントのエントリが作成されます (*下のスクリーンショット参照*)。例えば、下のスクリーンショット内で表示されているイベントエントリは、

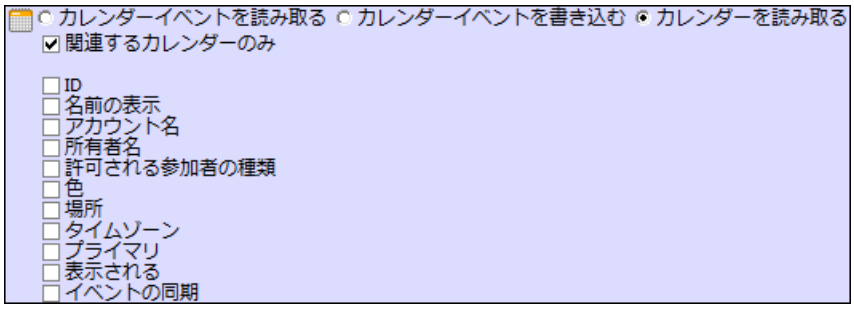
イベントのタイトル、開始と終了時刻、説明とイベントの場所が含まれています。下のスクリーンショット内では、*イベントは終日*のための値は(false の場合) **0** 、または、(true の場合) **1** であることができにことに注目してください。



クライアントデバイス上でアクションがトリガーされると、イベントは直接カレンダーに保存されません。代わりに、イベントエントリは、カレンダーアプリで開かれ、ユーザーはすぐにイベントを編集し、希望する場合はカレンダーに保存することができます。

カレンダーの読み取り

このアクションは、デバイス上のカレンダーに関する情報を読み取ります。デバイスには国際的な祝日、または、特定の業界の貿易フェアなど追加カレンダーが存在する場合があります。これらのカレンダーは通常新規のイベントを追加するために使用され、必須とは考慮されません。カレンダーの読み取り時、これらの「必須ではない」カレンダーを(*関連するカレンダーのみを選択し*)フィルターすることができます(*下のスクリーンショット参照*)。カレンダーの選択がフィルターされていない場合、デバイス上のすべてのカレンダーが読み取られます。**$MT_CALENDAR** ツリー内で**カレンダー**要素として各カレンダーは保管されます。



読み取るカレンダーの情報のデータフィールドを選択することができます:

- *ID:* カレンダーの識別子である文字列。
- *名前:* これらの名前は、カレンダー間を区別するために使用されます。カレンダーの表示名、アカウント名、管理者の名前から1つ、または(デバイスに複数のアカウントが存在する場合があるため)複数を選択します。
- *許可される参加者の種類:* 任意、または、必須などの値。
- *色、場所、タイムゾーン:* カレンダーの色、場所、タイムゾーン(`+/-HH:MM` の書式で通常表示されています)。
- *プライマリ:* 通常各デバイスにはプライマリカレンダーと複数のセカンダリカレンダーが存在します。この値は、カレンダーがプライマリ(**true**) 、または、セカンダリ(**false**) かを示しています。
- *表示される:* カレンダーが表示 (**true**) 、または、非表示 (**false**) かどうか。
- *イベントの同期:* カレンダーがイベントを同期するか(**true**) 、または、しないか(**false**) どうか。

メモ　カレンダーに特定のフィールドの情報が含まれていない場合、そのフィールドのために返される値はありません。

デバイスのカレンダーをシミュレーションする方法

デバイスのカレンダーアプリをシミュレーションするには2つのオプションがあります:

- Microsoft Outlook のカレンダー
- **$MT_CALENDAR** ツリーの構造を持つXML ファイル

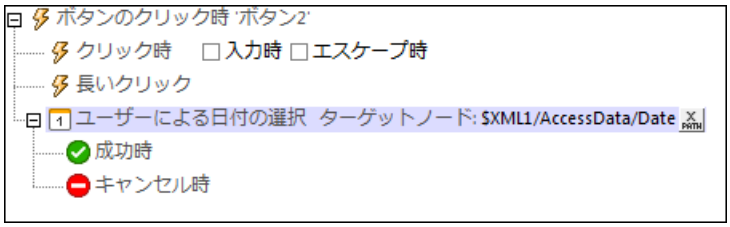
オプションダイアログ（**「ツール | オプション」**）のシミュレーションタブ内のオプションを選択します。

シミュレーションに使用されるサンプルカレンダーファイルの構造

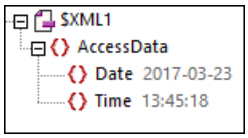
```
<Root>
  <Calendar Id="1" Name="Business">
    <Event Id="1" Title="Quarterly Meeting" Start="2018-04-04" End="2018-04-04" AllDay="1"
Location="Meeting Room 2">
      <Attendee Name="Bob" Status="Accepted" Type="Required" Relationship="Speaker"/>
    </Event>
    <Event Id="1" Title="New Customer Lunch" Start="2018-05-14T12:30:00" End="2018-05-
14T14:00:00" Location="Sushi Restaurant">
      <Attendee Name="Alice" Status="Accepted" Type="Optional" Relationship="Attendee"/>
    </Event>
  </Calendar>
  <Calendar Id="2" Name="Private">
    <Event Id="1" Title="Family Dinner" Start="2018-05-18T19:00:00" End="2018-05-18T23:00:00"
Location="Home"/>
    <Event Id="2" Title="Summer Vacation" Start="2018-07-09" End="2018-07-22" AllDay="1"
Location="Home"/>
  </Calendar>
</Root>
```

ユーザーによる日付の選択

*ユーザーによる日付の選択 アクション*によりクライアントデバイス上に日付ピッカーが表示されます (*下のスクリーンショット参照*) 。この時時刻ピッカーでエンドユーザーが選択する日付はこの*アクション*内で指定されているターゲットページソース内に保存されます (*スクリーンショット参照*)。日付は以下のフォーマットで保管されます **YYYY-MM-DD**。

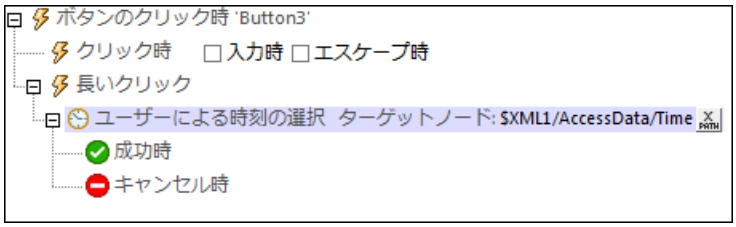


下のスクリーンショットは、クライアントデバイス上で日付が選択された後の[シミュレーション](#)内の対応するページソースノードを表示しています。

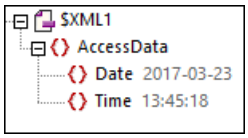


ユーザーによる時刻の選択

ユーザーによる時刻の選択 アクションによりクライアントデバイス上に時計が表示されます (*下のスクリーンショット参照*)。 この時計内でエンドユーザーが選択する時刻はこのアクション内で指定されているターゲットページソース内に保存されます (*スクリーンショット参照*)。時刻は24時間のフォーマットで保管されます: **HH:MM:SS**。



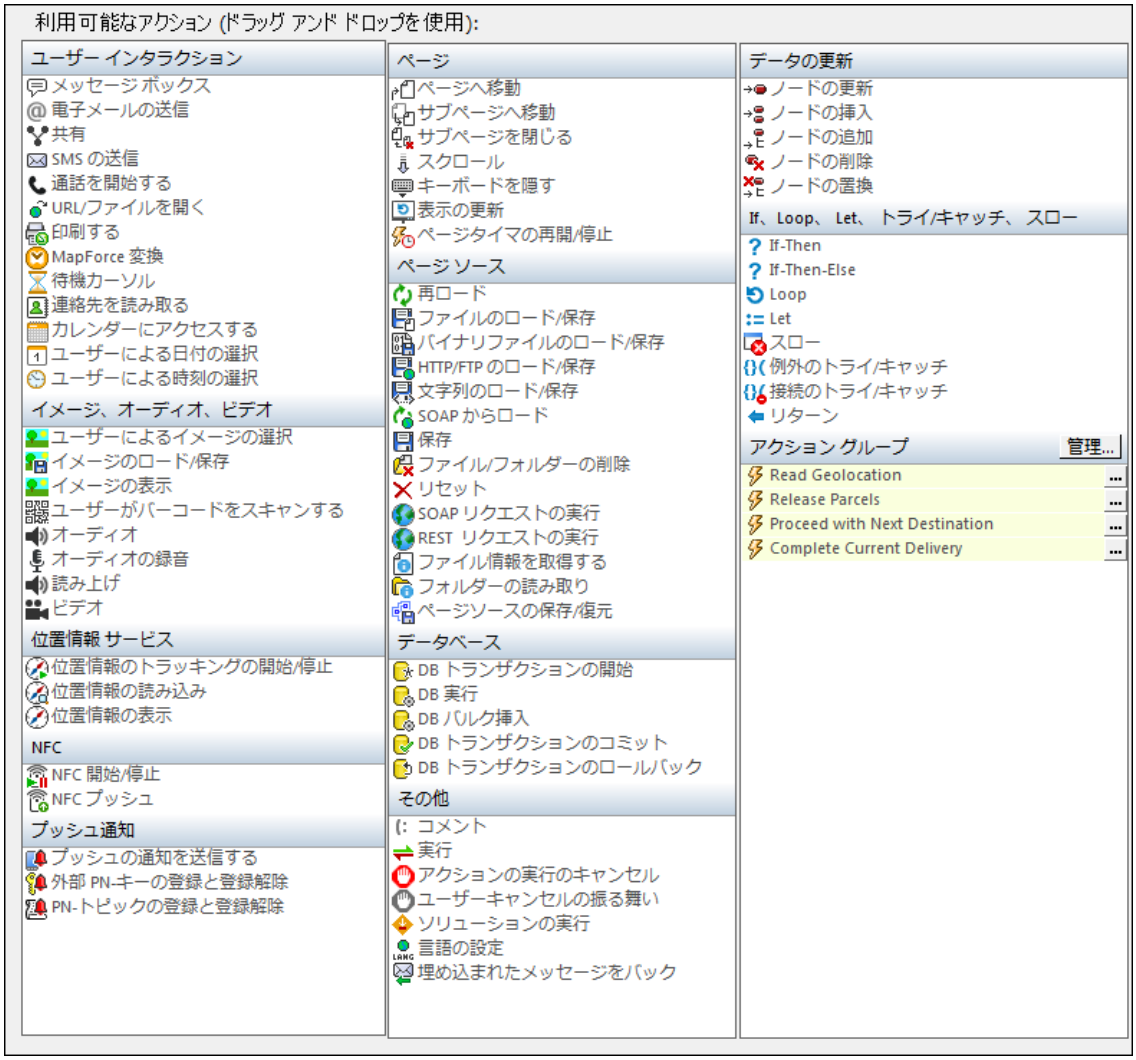
下のスクリーンショットは、クライアントデバイス上で時刻が選択された後のシミュレーション内の対応するページソースノードを表示しています。



10.2 イメージ、オーディオ、ビデオ

アクションダイアログのイメージ グループで次のアクションを使用することができます (下のスクリーンショット):

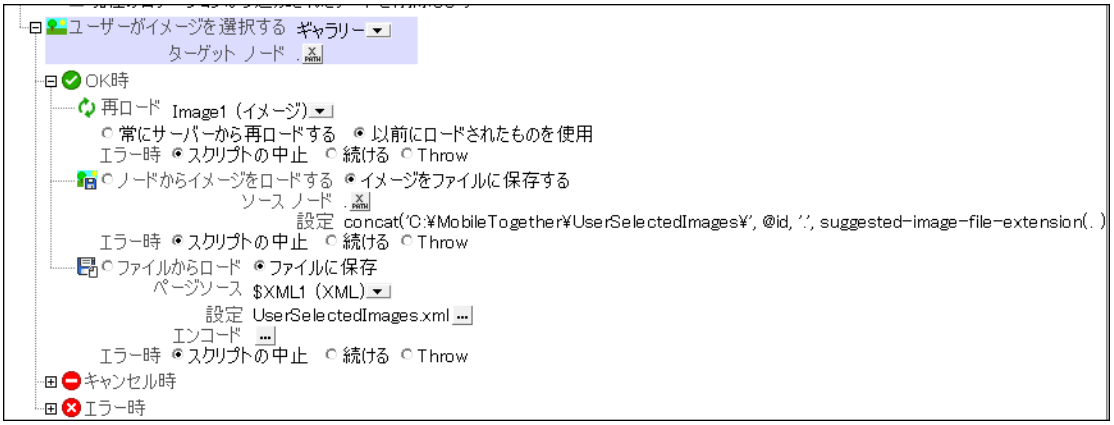
- ユーザーのイメージ選択
- イメージのロード/保存
- イメージの表示
- ユーザーによるバーコードのスキャン
- オーディオ
- オーディオの録音
- 読み上げ
- ビデオ



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

ユーザーのイメージ選択

データソースツリーのノードの保存されるイメージをユーザーは選択することができます (*下のスクリーンショット参照*)。これによりエンドユーザーはデータベースに自動的に保存されるイメージを選択することができます。使用方法の例としては、保険の損害の報告があげられます。ユーザーは、ソリューションを実行して、モバイルデバイスで写真をとることができ、イメージは適切なデータベースに直接アップロードされます。



アクションには次のプロパティがあります:

- *イメージ ソース* [ギャラリー] を選択して、ユーザーがクライアント デバイスのイメージ ギャラリーからイメージを選択します。[カメラ] を選択して、モバイル デバイスのアプリを開始して、カメラで撮られた次の写真をキャプチャします。
- *ターゲット ノード*: イメージがBase64 エンコード データとして保存されるデータソース ノード。

[*ユーザーのイメージ選択*] アクションには3 つの条件があります:

- *OK 時*: イメージが正しくBase64 エンコードデータとしてインポートされた時に実行されるアクションを定義します。典型的な実行されるアクションには次があげられます: (i) 選択されたイメージを表示するイメージの再ロード コントロールは選択されたイメージと共に表示を更新します; (ii) (Base64 エンコード テキストとしてXML ノードに保存されることに対して、) イメージがバイナリイメージ ファイルとして保存される必要がある場合 ファイルにイメージを保存 します; (iii) ページ データソースに新しく追加されたエンコード イメージ データを含むXML データを ファイルに保存 します。
- *キャンセル時*: イメージの選択のプロセスがユーザーによりキャンセルされた場合、ユーザーにより選択されたイメージの準備のための変更をロールバックするために一部のアクションが必要になる可能性があります。
- *エラー時*: イメージがターゲット ノードに正しくインポートされなかった場合とされるアクションを定義します。例えば、選択が失敗したことをユーザーに通知し、およびまたはWeb ブラウザーでトラブルシューティングのページを開くことができます。

このアクションの使用方法の例は、ユーザーにより選択されたイメージを参照してください。

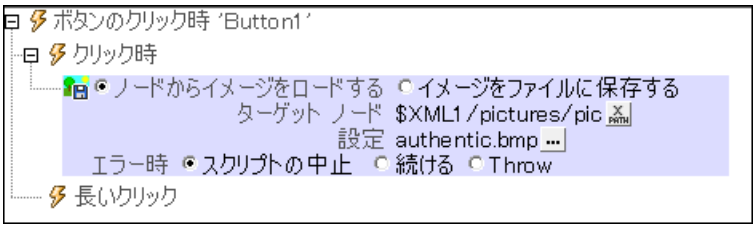
イメージのロード/保存

このアクションで以下を行うことができます:

- データソースノードにイメージファイルをBase64 エンコードイメージとしてロードする。
- サーバー側または外部の場所に、Base64 エンコードイメージをデータソースノード内にイメージファイルとして保存する。

データソースノードにイメージファイルをロードする

イメージファイルは、イメージのロード/保存 アクションの *ノードにイメージをロードする* オプションを使用して、データソースノードにロードすることができます(*下のスクリーンショット参照*)。XPath 式を使用して、イメージデータが保管されるデータソースノードである、ターゲットノードを選択します。ファイルパス フィールド内で、ターゲットノードにロードされるイメージファイルを選択します。イメージファイルは(BMP、EXIF、GIF、JPG、またはPNG を含む) 標準のイメージフォーマットであることができます。イメージファイルデータはBase64 に変換され、Base64 エンコードデータとしてターゲットノード内に保管されます。Base64 エンコードには、元のイメージフォーマットを指定する情報が含まれていることに注意してください。



イメージファイルとしてBase64 エンコードイメージを保存する

データソースノード内に保存されているBase64 イメージデータは、イメージのロード/保存 アクションのイメージをファイルに保存オプションを使用して、イメージファイルとして保存することができます(*下のスクリーンショット参照*)。 Base64 エンコードイメージが存在する、データソースノードの場所を選択します(*下のスクリーンショット参照*)。ファイルが保存されるサーバーまたはクライアントの場所を(ファイルパス フィールドで)選択します。

ファイルパスの入力にAltova XPath 拡張子機能 **`suggested-image-file-extension`** を使用して、イメージのファイルの種類を決めることができます。各イメージは特殊なイメージのフォーマットで、このフォーマットの情報がBase64 エンコード イメージ データに保管されます。**`suggested-image-file-extension`** 機能は拡張子を返します。正しくないファイルの種類をイメージ ファイル名の一部として入力することによりイメージファイルの読み込み不可能になる場合があります。

次のXPath 式は

```
concat('EmployeePhotos/',@name,@surname,'.',suggested-image-file-extension(@photo))
```

以下を評価します:

EmployeePhotos/MaxMuster.png

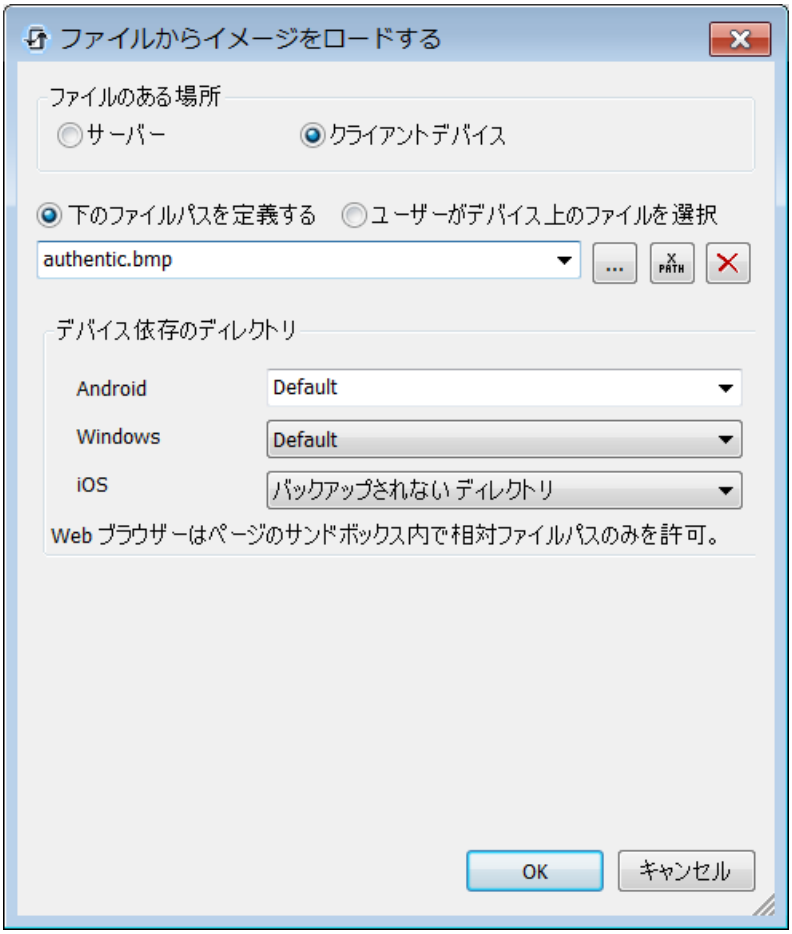
このアクションの使用方法の例は、ユーザーにより選択されたイメージのセクションを参照してください。

イメージファイルのロケーション

イメージのロード/保存 アクションのファイルパスフィールドの**追加ダイアログ**ボタンをクリックすると、(*上のスクリーンショット参照*)、ファイルを指定ダイアログは表示されます。このダイアログ内で、ファイルがサーバーまたはクライアントに存在するかをそれぞれのラジオボタンを選択することにより指定することができます (*下のスクリーンショット参照*)。

ファイルがサーバーに存在する場合

イメージファイルがサーバーに存在する場合、場所を(*絶対/相対パスを使用して*)参照、または、ファイルをグローバルリソース(*ファイルエイリアスまたはフォルダーエイリアス*)を使用して指定することができます。使用するオプションを選択してください。



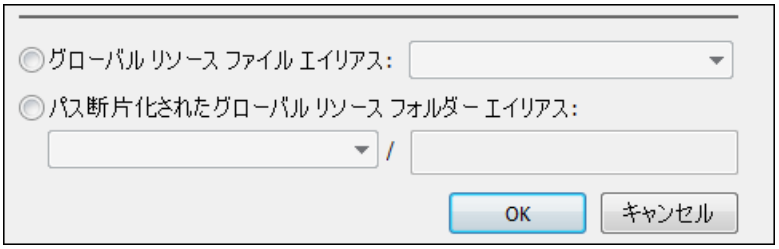
- **絶対/相対 パス**: ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細はプロジェクト ファイルの場所を参照してください。パスを入力またはファイ

ルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対/絶対パスがファイルにアクセスするために(サーバーのデータベース)内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関しては*プロジェクトファイルの場所*のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

- **グローバル リソース ファイル エイリアス**: コンボボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイルリソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova *グローバル リソース*のセクションを参照してください。

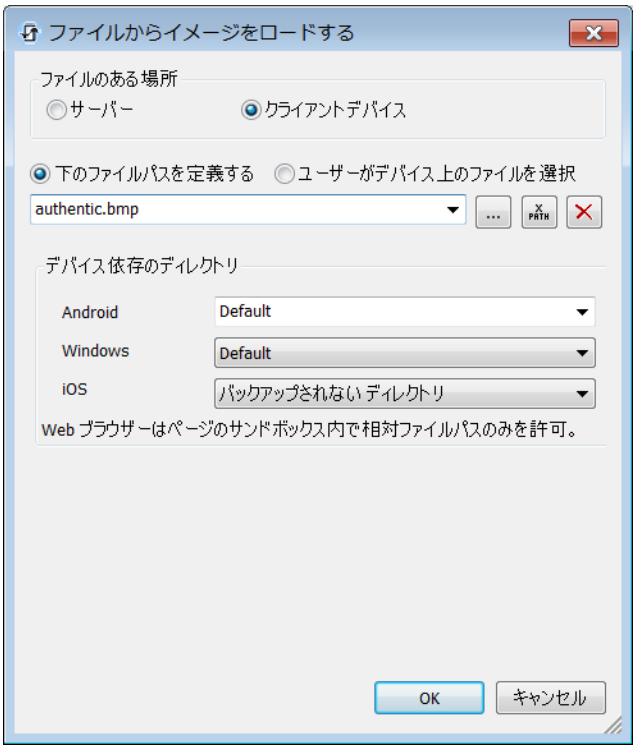
- **パス断片化されたグローバル リソース フォルダ エイリアス**: コンボボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova *グローバル リソース*のセクションを参照してください。

ファイルがクライアントに存在する場合

イメージファイルがクライアントに存在する場合、場所のパスを入力または選択して指定、または、XPath 式 を使用してパスを構築します。現在のエントリを **リセット** ボタンを使用して削除します。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (ソースファイルとターゲットファイルオプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。
- **デバイスに依存するリソース** ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- ***シミュレーションのためのファイルのロケーション:*** ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ（**ツール | オプション**）内で指定されていなければなりません。

エンドユーザーにより（クライアントデバイス上で）ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター:* クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト（例: `txt,html;xml`）、または（ii）各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式（例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス: `'txt'、'html、'xml'`）

- *オプションのデフォルトのファイル:* 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス:* ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、または XPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する:* もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

  メモ: iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法は iCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

エラーの処理

エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続または スローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合（*成功時*）、またはエラーが発生する場合（*エラー時*）。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション の キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

イメージの表示

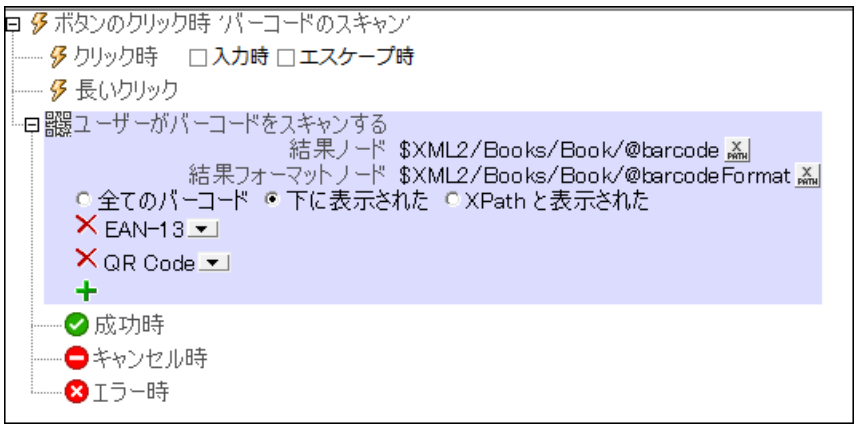
イメージの表示 アクション (下のスクリーンショット) により選択されたイメージが表示されます。エンドユーザーは、イメージを縮小、または、拡大することができ、イメージを上下、および、左右にスクロールすることができます。

以下の方法でイメージの選択を行うことができます:

- データソースノード内に保管されているBase64 イメージ。
- クライアントマシン上のイメージファイル。クライアントマシン上のダイアログボックス内に表示されます。イメージファイルが存在するデバイスのディレクトリを選択し、イメージファイルの名前を入力します。
- イメージを表示するコントロール。イメージ、チャート、と署名フィールド コントロールの使用によりイメージの表示が許可されています。これらのコントロールのインスタンスは、デザイン内で作成され、コンボボックスのドロップダウンリスト内で表示されます。表示するコントロールを選択してください。

ユーザーによるバーコードのスキャン

このアクションがトリガーされると、クライアントデバイスのカメラアプリケーションがローンチされ、エンドユーザーはバーコードをスキャンすることができます。スキャンが完了すると、MobileTogether は、自動的にバーコードを対応するバーコードフォーマットに入力し、2つの個別のXML ノードに入力します。例えば、ISBN バーコードがスキャンされると、ISBN 番号と ISBN バーコードフォーマット (EAN-13) は、アクションの定義内で指定されている2つのノードに保存されます。このバーコードの情報はデザイン内でXML データとして使用することができます。



ユーザーによるバーコードのスキャンアクション (*上のスクリーンショット参照*) 以下のオプションをおこなうことができます:

- *結果ノード:* バーコードデータが保存されている箇所のデータソースノード。
- *結果フォーマットノード:* スキャンされたバーコードが保存されているフォーマットの箇所のデータソースノード。スキャンされたバーコードは自動的に検知され、このノードに保存されます。このノード内に入力されたフォーマット名は、下のサポートされるフォーマットのリスト内と同様に入力されます。
- *スキャンするバーコードフォーマット:* 以下を選択することができます: (i) 全てのバーコード、(ii) エントリを追加してリストするバーコード、各エントリのコンボボックス内のフォーマットを選択しますバーコード、または、(iii) (例: `"Aztec", "Codabar", "Code 39"`) 単一バーコードフォーマットを取る各アイテムが文字列である XPath シーケンスとして入力されるバーコードのリストシーケンスとして返されるバーコードのリスト。このフィルターを XPath 式 を使用して設定した場合、下でサポートされるフォーマットのリストで挙げられているフォーマット名が使用されていることを確認してください。

サポートされている バーコード フォーマット

現在サポートされているバーコードフォーマットは、それぞれのフォーマットに適用される制限と共に下にリストされています。

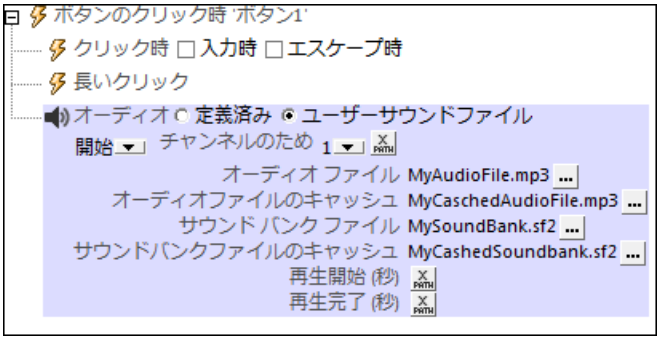
フォーマット名	メモ
Aztec	
Codabar	*iOS では使用不可*
Code 39	
Code 93	
Code 128	
Data Matrix	*iOS: バージョン>= 8*
EAN-8	
EAN-13	
EAN-128	*Windows、iOS では使用不可*

ITF	*iOS: バージョン>= 8*
PDF 417	
QR Code	
RSS-14	*iOS では使用不可*
RSS-Expanded	*iOS では使用不可*
UPC-A	*iOS: EAN 13 として検出されます*
UPC-B	

メモ バーコードは、Web クライアントではサポートされていません。

オーディオ

オーディオアクション (*下のスクリーンショット*) は、定義済みのサウンドまたはオーディオファイルを再生します。アクション内では、これら2つのオプションから選択し、オーディオの再生のパラメーターを定義します。5つの異なるチャンネルのために再生を定義することができます。

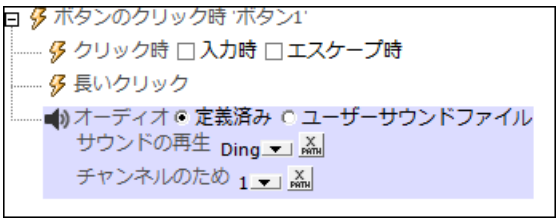


定義済みのオーディオ

クライアントデバイスで使用することのできる、下記の定義済みのサウンドを選択することができます:

```
ClickOffOn, ClickOnOff, Ding, DingDong, ErrorDeepBuzz, ErrorWhoops, Goodbye,
KeyClickTick, KeyClickTock, MessageBounce, MessageXylophone, WhooshDeep,
WhooshExhale, WhooshLong, WhooshQuick, WhooshQuicker
```

定義済みのサウンドの名前をコンボボックス内のリストから選択する、または、これらの定義済みの名前を評価するために XPath 式を入力します。名前は前の段落内の名前に一致する必要があります。



選択されたサウンドファイルを使用する

オーディオアクションにより、選択するサウンドファイルの再生を行うことができます。アクションのダイアログ内では、以下から1つのアクションを選択することができます: *オーディオの開始*、*停止*、*再生*、*ストップ*、*検索 (移動)*。

オーディオは (**1** から**5** の番号を持つ) 5 つのチャンネルを再生することができます。各オーディオアクションは1つの特定のチャンネルのために定義されています。ですから、例えば、次のアクションのシーケンスを定義することができます:

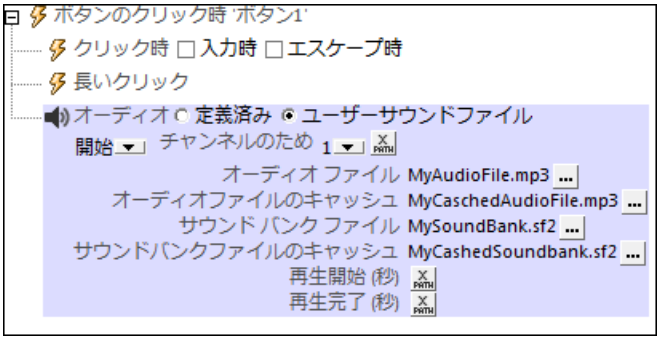
1. チャンネル1 でオーディオの再生を開始します。
2. チャンネル1 でオーディオの再生を一時停止します。
3. チャンネル2 でオーディオの再生を開始します。
4. チャンネル2 でオーディオの再生を停止します。
5. チャンネル1 でオーディオの再生を再開します。

6. チャンネル1 でオーディオの再生を停止します。

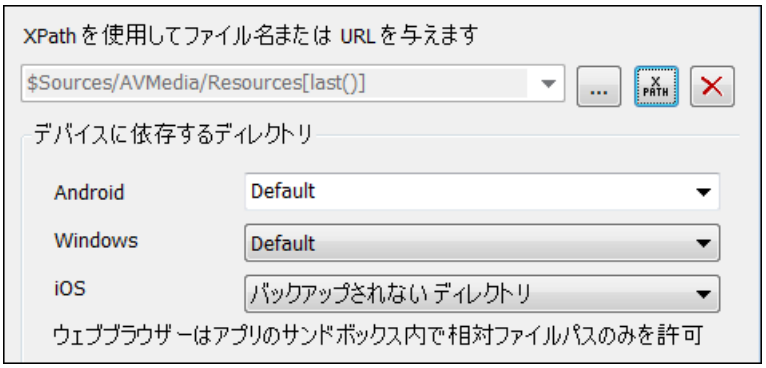
通常、オーディオアクションは、ボタンクリックなどのコントロールイベントで定義されます (下のスクリーンショット参照)。イベントは関連したオーディオアクションをトリガーします。多種のオーディオアクションについて下で説明されています。ソリューション内でオーディオアクションを使用する場合、オーディオ(再生)を参照してください。

オーディオの開始

オーディオアクションの開始は、指定されたオーディオファイルをダウンロードし再生します。下のスクリーンショット内で表示されている設定があります。



- *オーディオファイル:* **追加ダイアログ** ボタンをクリックして、再生されるオーディオファイルを指定します。設定は、URL または相対的ファイルパスであることができます。また、静的な値、または XPath 式を使用して生成 されることができます。相対的なファイルパスはモバイルデバイスに保管されているファイルを探すために使用されます。これらは、デバイスの種類のためのベースディレクトリに相対して解決されます(下のスクリーンショット参照)。オーディオファイルのフォーマットに関する詳細は、オーディオ/ビデオ フォーマットを参照してください。



- *Android:* Android デバイスディレクトリをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。しかしながら、Android デバイスがルートされていない限り (MobileTogether 以外の)他のアプリは、MobileTogether サウンドボックスディレクトリへアクセスすることができません。ですから、他のアプリを使用して MobileTogether サンドボックス内でファイルを開こうと試みると失敗する可能性があります。
- *Windows RT:*Windows Phone または、Windows RT デバイスフォルダーをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。
- *iOS:* MobileTogether は、ユーザーのデバイスに2つのディレクトリを作成します: (i) オペレーティングシステムにより場悪アップされるファイルのためのバックアップディレクトリで、後に使用することができます。例えば、復元時に使用することができます。 (ii) バックアップの必要ないファイルのためのバックアップされないディレクトリであり、再生のため速いバッファーが必要な場合選択されます。バックアップディレクトリまたはバックアップされないディレクトリを必要に応

じて選択してください。

- *Web ブラウザー*: 選択できるオプションはありません。相対的パスはブラウザー内のサンドボックスのコンテキストで解決されます。
- *オーディオファイルのキャッシュ*の値の設定が指定されている場合、ダウンロードされたファイルが指定されたローカルファイル内にキャッシュされます。キャッシュが既存の場合、キャッシュファイルが再生され、ダウンロードは実行されません。拡張子、とフルファイル名を含むキャッシュファイルのフルネームを指定する必要があります。また、フルネームは、拡張子を含め、再生されるソースファイルのファイル名と一致する必要があります。キャッシュファイルがクライアントデバイスに既に存在しているオーディオファイルを指定する場合、利点が無いことに注意してください。
- サウンドバンクファイルはiOS デバイスでMIDI ファイルを再生する際に必要です。サウンドバンクファイルの設定は、サウンドバンクファイルとそのキャッシュの場所を指定します。キャッシュされたサウンドバンクファイルが既存の場合、キャッシュファイルが使用されます。
- オーディオファイルのセグメントを再生する場合、セグメントの開始および終了を秒数で入力してください。フィールドが空の場合、オーディオは最初から終わりまで再生されます。

メモ　Windows Phone では、マルチチャンネルのオーディオおよびビデオの再生はサポートされていません。1つのオーディオ**または**ビデオファイルのみ1度の再生することが可能です。これは最後に再生されたファイルを意味します。

メモ　オーディオ/ビデオファイルは、MobileTogether Designer プロジェクトのサーバーメカニズムにデプロイを使用して MobileTogether Server にデプロイすることはで**きません**。しかしながら、URL を使用してストリームすることはできませんが、サーバーに手動でオーディオ/ビデオファイルをコピーすることができます。MobileTogether Server に存在する、オーディオ/ビデオファイルをストリームするには、以下を行います：(i) バイナリのロード アクションを使用して、バイナリのオーディオ/ビデオデータをデータソースノードにロードします。(ii) バイナリの保存 アクションを使用して、このノード内のデータをクライアント上のファイルに保存します。(iii) オーディオ/ビデオの再生アクション を使用して、クライアントデバイス上に保存されたファイルを再生します。または、オーディオ/ビデオファイルを MobileTogether Server に保存する代わりにWeb サーバーに保存し、URL を使用して、Web サーバーからオーディオ/ビデオファイルをストリームします。

一時停止

このアクションは、特定のオーディオチャンネルのために定義されます。アクションがトリガーされると特定のチャンネルで再生されているオーディオファイルは一時停止されます。通常、このアクションは**一時停止**ボタンで定義されています。このアクションはプロジェクト全体で適用することができ、他のページで再生開始されたオーディオファイルにも適用することができます。

メモ　オーディオストリームがソリューションが中断されている場合に再生されている場合、再生は一時停止されます。再生はソリューションが再開されると継続されます。

再開

このアクションは特定のチャンネルのために定義され、以前に一時停止されている場合、再生の再開をします。通常、このアクションは**オーディオの再開**ボタンで定義されています。このアクションは、再生が開始または一時停止されたページに関わらず、プロジェクト全体の指定されたチャンネルで一時停止されているオーディオの再生に適用することができます。

メモ　オーディオストリームがソリューションが中断されている場合に再生されている場合、再生は一時停止されます。再生はソリューションが再開されると継続されます。

停止

停止アクションがトリガーされると、現在特定のチャンネルで再生されているオーディオは停止されます。通常、停止アクションは**オーディオの停止**ボタンで定義されており、オーディオファイルが他のページで開始されていても、特定のチャンネルで再生されているオーディオファイルに適用されます。

シーク

このアクションは(秒で)指定されたポジションにジャンプして再生します。このアクションは再生が開始されたページに関わらず、現在再生されているオーディオファイルに適用されます。

追加点

オーディオ(再生) のセクションは、MobileTogether のオーディオの再生機能の概要を述べています。また、オーディオ(再生) イベントに関しての情報に関しても説明されています。

オーディオの録音

オーディオの録音アクション(*下のスクリーンショット*) はオーディオの録音を開始および停止します。記録されたファイルはクライアントデバイスに保存されます。

例: *オーディオの録音の開始アクション*がボタンクリックに関連付けされている場合、ボタンをクリックすることにより(デバイスのマイクを使用して) アクション内で指定されたファイルへの録音を開始します(*下のスクリーンショット参照*)。他の録音パレメーターはアクションの設定内で指定されています。他の*オーディオの録音 の停止アクション*がボタンのクリックに関連付けられている場合、このボタンをクリックすることにより録音が開始されたページおよび現在進行中のページの録音が停止されます。

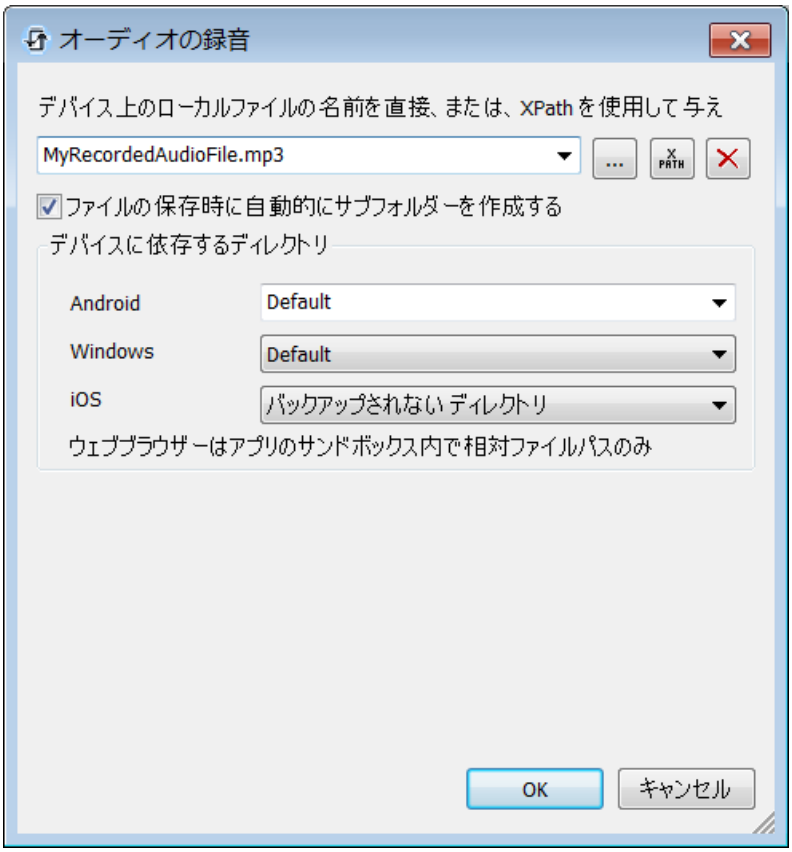
メモ　オーディオの録音中にソリューションが中断された場合、録音は停止されます。

メモ　特にiOS デバイスでの再生の状態に影響をもたらすため、オーディオは、オーディオまたはビデオの再生中に録音することを控えてください。

オーディオの録音開始

オーディオの録音 コンボボックスから開始を選択します (*スクリーンショット参照*) 。

オーディオファイル設定に関しては、録音を保存するクライアントデバイス上のファイルの名前と相対的パスの名前を入力してください。 ファイルパスディレクトリを入力、または、XPath 式を使用して選択することによりパスの名前を入力します。入力するファイルパスは、デバイスの種類のために指定されるベースディレクトリに従い解決されます (*下のスクリーンショット参照*)。パスが存在しないファイル名を含む場合、不足するサブフォルダーを自動的に作成するオプションを選択することができます (*下のスクリーンショット参照*)。オーディオファイルフォーマットに関する詳細に関しては[オーディオ/ビデオフォーマット](#)を参照してください。



- *Android:* Android デバイスディレクトリをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。しかしながら、Android デバイスがルートされていない限り (MobileTogether 以外の)他のアプリは、MobileTogether サンドボックスディレクトリへアクセスすることができません。ですから、他のアプリを使用して MobileTogether サンドボックス内でファイルを開こうと試みると失敗する可能性があります。
- *Windows RT:* Windows Phone または、Windows RT デバイスフォルダーをドロップダウンリストから選択します。デフォルトの設定である、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。
- *iOS:* MobileTogether は、ユーザーのデバイスに2つのディレクトリを作成します: (i) オペレーティングシステムにより場悪アップされるファイルのためのバックアップディレクトリで、後に使用することができます。例えば、復元時に使用することができます。 (ii) バックアップの必要ないファイルのためのバックアップされないディレクトリであり、再生のため速いバッファーが必要な場合選択されます。バックアップディレクトリまたはバックアップされないディレクトリを必要に応じて選択してください。
- *Web ブラウザー:* 選択できるオプションはありません。相対的パスはブラウザー内のサンドボックスのコンテキストで解決されます。

次に録音のパラメーターを設定します (*ページ上のスクリーンショット参照*):

- *自動ファイル拡張子:* 選択されている場合、ファイル拡張子は選択されたフォーマットおよび/またはコーデックに従い自動的に追加されます。XPath 拡張関数 `mt-last-file-extension` を使用して、最後に録音されたオーディオファイルの拡張子を取得することができます。
- *ファイルの最大値 (kB):* デフォルト値の制限はありません。ファイルのサイズはサンプリング速度とエンコードビットレートにより異なります。メモ
- *録音時間の最大値 (秒):* デフォルト値の制限はありません。

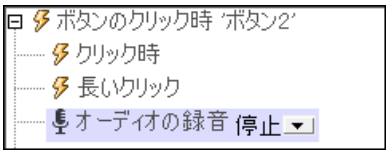
- *サンプリング速度 (Hz):* 更に高いサンプリング速度はオーディオのシグナルを更に正確に伝達します。高いサンプル速度の欠点は大きなファイルのサイズです。参考としては、サンプリング速度を選択する場合、CD-品質のサンプリングは44.1 kHz (44100 Hz)です。しかしながら、サンプリング速度の大部分は使用されるエンコーダーとオーディオコードの基準に左右されます。この設定は空白にしておき、自動的に使用されるデフォルトの設定を使用することが奨励されます。サンプリング速度を指定する場合、関連するオーディオエンコード基準、または、エンコーダーの仕様を確認してから行ってください。
- *エンコードビットレート(kb/s):* 圧縮のレベルのことを意味します。オーディオファイルが圧縮されると、詳細が削除されます。通常、これらの詳細は、人間の周波数範囲(20 Hz - 20 kHz)外であり、高度なスピーカーのみで再生することが可能です。トレードオフはファイルのサイズとオーディオの品質です。128 キロビット毎秒 のビットレートは、MP3 を再生するために合理的な数値です。オーディオのストリーミングは通常 256 キロビット毎秒 です。CD-品質は320 キロビット毎秒です。通常のビットレートの値は以下の通りです: 96 キロビット毎秒 (モバイルの標準の品質)、160 キロビット毎秒 (モバイルでは上質、デスクトップでは通常の品質)、320 キロビット毎秒 (モバイルではとても上質、デスクトップで上質)。この設定は空白にしておき、エンコーダーが自動的に使用されるデフォルトの設定を使用することが奨励されます。ビットレートを指定する場合は、関連するオーディオエンコード基準、または、エンコーダーの仕様を確認してから行ってください。
- *Android 設定:* Android デバイスで使用することのできるオーディオ録音エンコーダー (コーデック) からフォーマットを選択します。デフォルトは、クライアントデバイスのデフォルトのフォーマット、もしくは、コーデックを選択します。`AMR-NB` と`AMR-WB` フォーマットに関しては、AMR-NB とAMR-WB コーデックのみをそれぞれに対して使用します。`AAC` フォーマットはAAC-Low Complexity、High Efficiency-AAC、およびEnhanced Low Delay-AAC コードを使用して生成することができますが、与えられたデバイスで作業するために、これらすべてのコーデックを使用できるとは限りません。ソリューションユーザーのための選択の効果が既知でない場合は、これらをデフォルトに設定することが奨励されます。詳細に関しては[オーディオ/ビデオ フォーマット](#) を参照してください。
- *Windows 設定:* Windows デバイスで使用することのできるオーディオの録音 エンコーダー (コーデック) を選択します。デフォルトではクライアントデバイス上のデフォルトのコーデックを選択します。
- *iOS 設定:*iOS デバイスで使用することのできるオーディオの録音 エンコーダー (コーデック) を選択します。デフォルトではクライアントデバイス上のデフォルトのコーデックを選択します。

オーディオの録音は、以下の場合に停止されます:

- オーディオの録音の停止アクションが実行されます (*下を参照*)。このようなアクションは、ボタンクリックなどのクライアント側のイベントがトリガーされると実行されます。
- エンドユーザーがページを移動した場合、またはページからの移動が実行された場合。
- ソリューションの中断。

オーディオの録音の停止

現在進行中の同じページにあるオーディオの録音アクションにより開始された、オーディオの録音アクションを停止するように設定します (*下のスクリーンショット参照*) 。



メモ　また、ソリューションの中断 または、ユーザーが録音を開始したページから移動すると、録音は中止されます。

追加点

[オーディオの録音](#) のセクションは、MobileTogether のオーディオの録音機能の概要について説明しています。[オーディオの録音](#)で詳細が述べられています。

読み上げ

読み上げアクションが開始に設定されていると、テキスト文字列を読み上げに変換し、再生します。(*下のスクリーンショットを参照してください*)。テキスト文字列は、直接 XPath 式内に入力するか、または、 XPath 式 により選択されるノードから取得されます。下のスクリーンショットは、メインページがロードされ、モバイルデバイスの言語で挨拶が再生されるシナリオを表示しています。挨拶のテキストは、モバイルデバイスの言語の設定に一致する値を持つ `language` 属性を持つ `Greeting` 要素から取得されています。

アクションの言語設定は、モバイルデバイスのデフォルトの言語設定に設定されています。デバイスの言語の設定をーオーバーライドするために使用することができます。例えば、テキスト文字列が特定の言語の場合、変換はモバイルデバイスの言語ではなく、特定の言語で試みられます。XPath 式を使用することにより言語の設定の値を取得することができます。値は言語 -国名のフォーマットを取ります。例: `en-US`, `en-UK`, `de-DE`, `es-US`, `fr-CH`.

メモ 読み上げアクションを1つ以上入力すると、シーケンスの順番の最後の読み上げが使用されます。

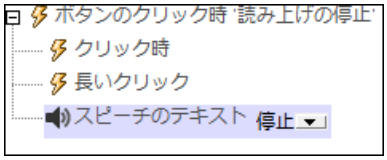
メモ 読み上げの再生は、モバイルデバイスのみで使用でき、MobileTogether Designer ではシミュレートすることはできません。

追加の読み上げアクション

読み上げ機能をデザインの追加 プロジェクトプロパティの読み上げアクション を使用して拡張することができます。 読み上げ機能に関する情報は、以下を参照してください：オーディオ、ビデオ| 読み上げ。

読み上げアクションの停止

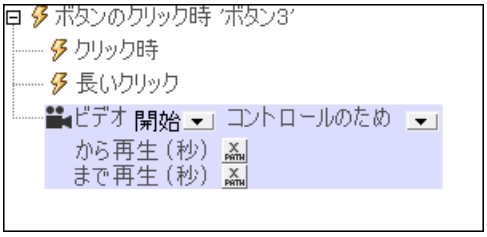
読み上げの再生の実行中、読み上げアクションの停止アクションを使用してアクションを停止することができます。(*下のスクリーンショットを参照してください*)。アクションがトリガーされると、再生中の読み上げが停止されます。特定のテキストを指定する必要はありません。



メモ 1つの読み上げの再生が1度に再生することができます。

ビデオ

ビデオアクション(*下のスクリーンショット*) には、現在のページのビデオコントロール ための次のアクションが存在します: 開始、一時停止、再開、停止、シーク(*ジャンプ* 。



現在のページのすべてのビデオコントロール は、コントロールコンボボックスのドロップダウンリストに名前順にリストされています(*上のスクリーンショット参照*)。アクションを適用するビデオコントロールを選択してください。

次に、実行するアクションを選択します。以下のアクションを使用することができます:

- *開始*: ビデオの一部を再生するには、セグメントの開始および終了を秒数で入力します。このフィールドを空にすると、ビデオは始めから終わりまで再生されます。
- *一時停止*: このアクションがトリガーされるとビデオの再生が一時停止されます。例えば、*一時停止*アクションは **ビデオの一時停止** ボタンにより定義されることができます。
- *再開*: このアクションがトリガーされるとビデオの再生が再開されます。例えば、このアクションはボタンをクリックするイベントのためにトリガーされることができます。
- *停止*: このアクションがトリガーされるとビデオが停止されます。例えば、このアクションは**ビデオの停止** ボタンのアクションとして設定することができます。
- *シーク*: 指定されたポジションに移動します。

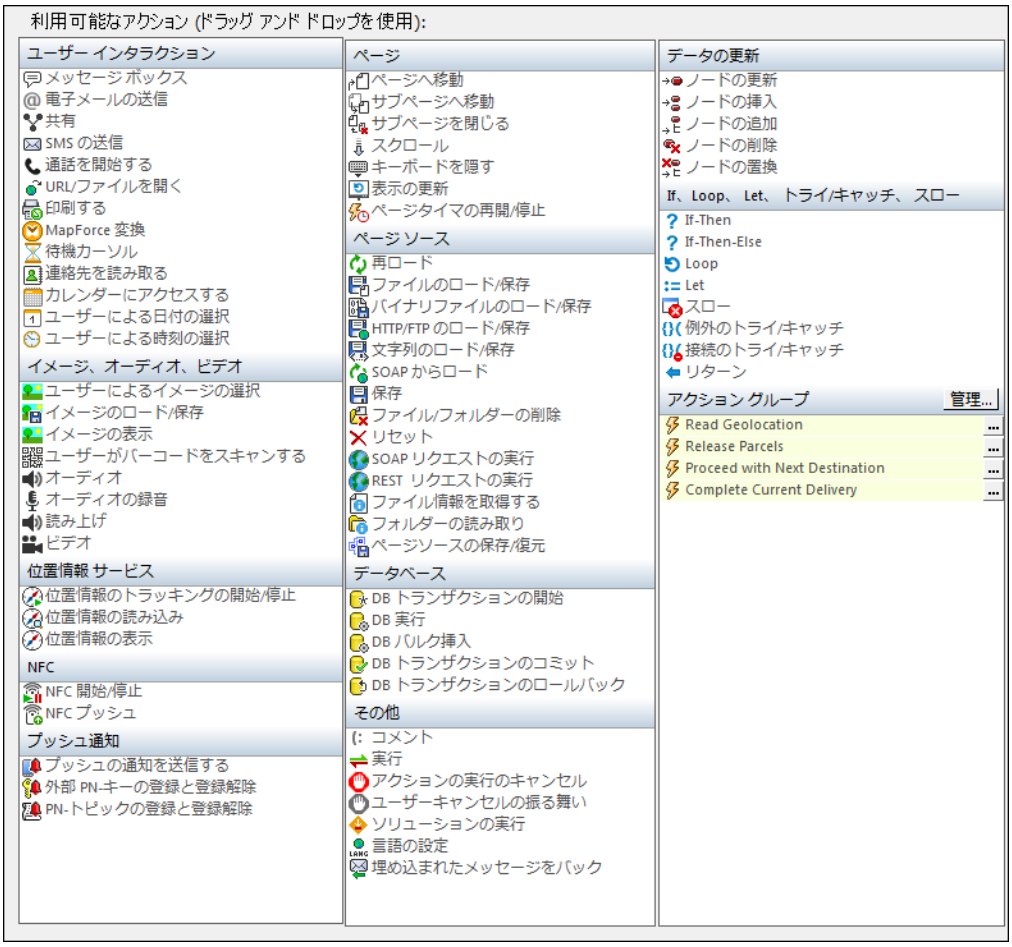
メモ　ビデオコントロールの**ロード時に再生** または **コントロールの表示** プロパティが`false()`に設定されている場合、ビデオの再生は上にリストされるビデオアクションのアクションを使用してのみ管理することができます。

メモ　各 ビデオコントロール には、実行するためにアクションを指定することのできる3つのイベントが存在します: **ビデオの開始時**、**ビデオのエラー時**、**ビデオの完了時**。これらのイベントの定義は、ビデオコントロールの コンテキストメニューの**ビデオコントロール アクション**コマンドを使用して、または、ビデオコントロールの コントロールアクション プロパティの **追加ダイアログ** ボタンをクリックすることにより、アクセスすることができます。

10.3 位置情報サービス

アクションダイアログの位置情報グループで次のアクションを使用することができます(下のスクリーンショット):

- 位置情報トラッキングの開始 停止
- 位置情報データの読み込み
- 位置情報の表示

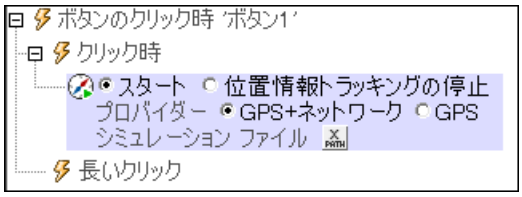


このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

チュートリアル位置情報の共有 はア位置情報のトラッキングの開始 停止 と位置情報の読み込みアクションの使用方法を説明しています。

位置情報トラッキング開始 /停止

位置情報のトラッキング開始 /停止アクション (*下のスクリーンショット参照*) はモバイルデバイスの位置情報のトラッキングを開始または停止します。一度開始されると、トラッキングは停止されるまで継続されます。位置情報データの読み込み アクションが実行されると、その時点での位置情報データがモバイルデバイスから読み込まれ、**$MT_GEOLOCATION** ツリーに入力されます。**$MT_GEOLOCATION** ツリー内のデータは XPath 式によりアクセスすることができます。



チュートリアル 位置情報の共有 は 位置情報のトラッキングの開始 /停止 アクションの使用方法について説明しています。

位置情報トラッキングの開始

モバイルデバイスで位置情報のトラッキングを開始するには、アクションの開始ラジオボタンを選択します (*上のスクリーンショット参照*)。

デザインに最適な位置情報データソースを選択してください:

- *GPS+ ネットワーク:* モバイルデバイスが、GPS により確認できる場合、GPS が位置情報を提供するために使用されます。ネットワークの位置情報は、通常 GPS に比べ正確ではないですが、このオプションは、GPS が作動しない場合 (例えば屋内で)、バックアップ情報ソースとして作動するなど、ネットワークにより提供される位置情報の利点があります。
- *GPS:* GPS は、位置情報データを提供するために使用されます。このオプションの利点は、位置情報データが正確であるという点です。ケッテンは、GPS が特定の場所で使用できない場合、(例えば屋内で) 位置情報データを使用することはできません。

メモ 位置情報のトラッキングの開始 /停止 アクション、または、位置情報データの読み取り アクションがデザインに追加されると、**$MT_GEOLOCATION** ツリーは自動的にページのデータソースに追加されます。

シミュレーションのための位置情報
デザイナー とサーバー シミュレーションのために使用される XML ファイル位置情報 XML ファイルの位置情報を指定することができます。XPath 式は、位置情報ファイルを検索する URL に対して解決されなければなりません。このアクションに対して位置情報ファイルが指定されていない場合、位置情報設定ダイアログ内で定義されているデフォルトの位置情報ファイル が使用されます。

位置情報トラッキングの停止

モバイルデバイスで位置情報のトラッキングを停止するには、アクションの停止ラジオボタンを選択します (*上のスクリーンショット参照*)。

位置情報データの読み込み

位置情報データの読み込みアクションは現在の位置情報データを **$MT_GEOLOCATION** ツリーに入力します。アクションが現在の位置情報を読み込むためには、アクションを実行する前に、モバイルデバイスの位置情報トラッキングが開始されている必要があります。

位置情報のトラッキングの開始/停止 アクション、または、位置情報データの読み取りアクションがデザインに追加されると **$MT_GEOLOCATION** ツリーは自動的にページのデータソースに追加されます。イベントがトリガーされると、2 つのパーツ Location と Address (*下のリスト参照*) から構成される、位置情報のデータの読み込みアクションにより、**$MT_GEOLOCATION** ツリーからデータが取得されます。**Location** 要素は、位置情報の座標を含みます。**Address** 要素は、アドレスに相当するものと、ディレクトリの検索により決定される他の位置情報の座標に関する情報を含みます。郵便番号に相当する情報が存在しない場合は、このパートは記入されません。他の (URL などの) Address の子要素は、関連するデータが存在しない場合は記入されません。

```
$MT_GEOLOCATION
<Root>
  <Location/>
  <Address/>
</Root>
```

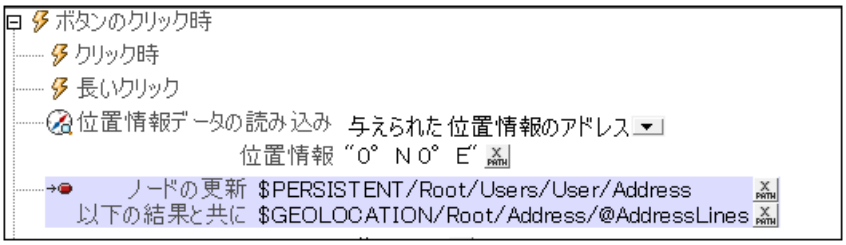
$MT_GEOLOCATION ツリーの詳細な構造

```
$MT_GEOLOCATION
<Root>
  <Location
    Provider=""
    Latitude=""
    Longitude=""
    GeoLocation=""
    Altitude=""
    AccuracyVertical=""
    AccuracyHorizontal=""
    Speed=""
    Time=""
    MagneticHeading=""
  />
  <Address
    Locality=""
    SubLocality=""
    CountryName=""
    CountryCode=""
    PostalCode=""
    AdminArea=""
    SubAdminArea=""
    FeatureName=""
    Thoroughfare=""
    SubThoroughfare=""
    Phone=""
    Url=""
    Premises=""
    AddressLines="">
    <AddressLine></AddressLine>
      ...
    <AddressLine></AddressLine>
  </Address>
</Root>
```

位置情報の取得のオプション

アクション設定のドロップダウンボックスで、以下の 1 つを選択することができます:

- *現在の位置情報*: **$MT_GEOLOCATION** ツリーの Location 要素内にモバイルデバイスの位置情報データを入力します。ツリーの Location 要素の属性のみがデータを含むことになります。ツリーは Address 要素を持ちません。
- *現在の位置情報 + アドレス*: Location と Address 要素ノードの双方にデータを入力します。
- *与えられた位置情報のアドレス*: **$MT_GEOLOCATION** ツリー内に Address 要素のデータを入力します。このデータは、入力された位置情報のための座標に対応します。位置情報のための座標は、下の位置情報入力文字列のフォーマットのセクションで説明されている構文のフォーマットを持つ必要があります。アドレスのデータは、位置情報のディレクトリを検索することで取得されます。
- *与えられたアドレスでの位置情報*: 位置情報の座標は、アドレスのためのフィールドの値として入力された文字列から取得されます。この文字列は、位置情報ディレクトリで検索され、このアドレスの座標を使用することができ **$MT_GEOLOCATION** ツリーの Location 要素はこれらの座標と共に更新されます。アドレスのサブパートの部分をディレクトリ検索のために入力することができます。



位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(**"**) でハイライトされており、エスケープした単位インジケーターは青い(**""**) でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:
33.33N -22°44'66.66"
33.33 22°44'66.66"W
33.33 22.c

- *与えられたアドレスの位置情報:* アドレスオプション内に入力されたアドレスの位置情報を返します。アドレスは文字列として入力されます。例: "Address Line 1、 Address Line 2"。この文字列は位置情報のルックアップとして提出され、返された位置情報コンポーネントは $GEOLOCATION ツリー内に保管されます (*このセクションの始めの*ツリー構造リストを参照してください)。

使用方法

位置情報データを使用するには、最初に位置情報データの読み込みアクションを使用して **$MT_GEOLOCATION** ツリーにデータが入力される必要があります。以下のスクリーンショットは、例えば、Location とAddress 要素のデータを取得する位置情報データアクション、そしてLocation/@Latitude データを使用して他の **$MT_GEOLOCATION** ツリー内のノードの更新を表示しています。

取得された位置情報データの単位とデータ型

異なるモバイルデバイスから取得された位置情報データは **$MT_GEOLOCATION** ツリーに数値として保管されます。これらの数値の単位とデータ型は、下のテーブルに表示されています。

	Android	Web	iOS	Windows Phone	WindowsRT
経度	度 (*double 型*)	実数度 (*double 型*)	度 (*double 型*)	度 (*double 型*)	度 (*double 型*)
緯度	度 (*double 型*)	実数度 (*double 型*)	度 (*double 型*)	度 (*double 型*)	度 (*double 型*)
精度	メートル (*double 型*)	メートル (*double 型*)	メートル (*double 型*)	メートル (*double 型*)	メートル (*double 型*)
高度	WGS 84 メートル以上の楕円外高さ	メートル (*double 型*)	メートル (*double 型*)	メートル (*double 型*)	メートル (*double 型*)

スピード	メートル/秒 (m/s)	メートル/秒 (*double 型*)	メートル/秒 (*double 型*)	メートル/秒 (*double 型*))	メートル/秒 (*double 型*)
時刻	UTC 時刻	DOM TimeStamp (*署名されていない long long*))	NSDate (*TZ に変換することが可能*)	Int64/ `System.Date timeOffset` (UTC)	Long long (UTC)

位置情報データをデザイナーとサーバーシミュレーションのために指定するには、位置情報の設定のセクションを参照してください。

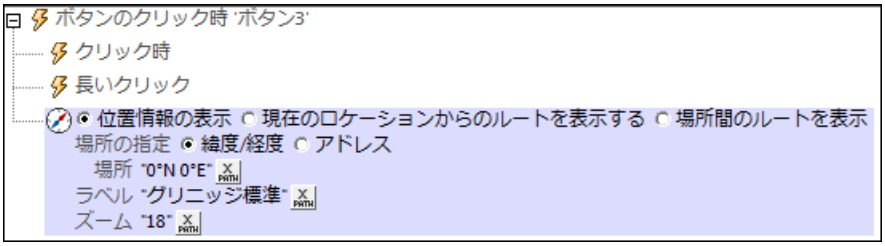
チュートリアル位置情報の共有 は位置情報の読み込み アクションの使用方法について説明しています。

位置情報の表示

タイトルバー位置情報の表示アクションは、モバイルデバイスのマップアプリケーションを開き、指定された情報を表示します。次のサブアクションのうちの1つを指定することができます (*下のスクリーンショット参照*):

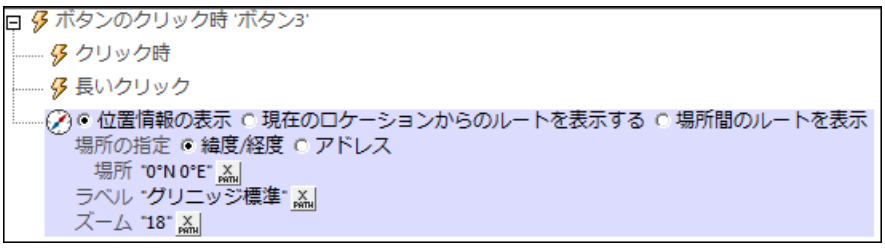
- *位置情報の表示*: マップ上の指定した座標の場所。
- *現在地からのルートを表示する*: 現在の場所から指定された場所へのマップ上のルート。
- *場所間のルートを表示*: 指定された場所間のマップ上のルート。

これらのオプション内の全ての場所は位置情報の座標により与えられます。



位置情報の表示

位置情報の表示 アクションは、モバイルデバイスのマップアプリケーションを開き、指定された位置情報でピンとラベルを配置します。

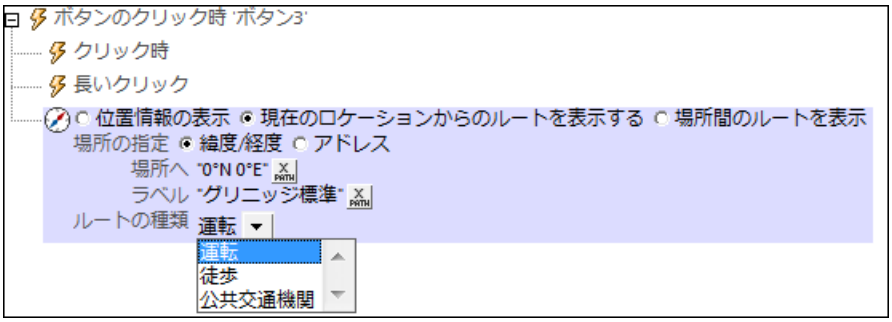


位置情報の表示 アクションには以下の設定があります:

- *場所の指定*: 場所の設定 (次の設定) の入力が緯度/経度 座標または住所かを選択します。上のスクリーンショットは、選択された緯度/経度オプションを表示しています。
- *場所*: マップ内で位置情報が表示されます。緯度/経度座標時が前の*場所の指定*設定で指定されている場合、(上のスクリーンショットに表示されているとおり) 場所の座標はとして入力される必要があります。下の位置情報入力文字列フォーマットのセクションで説明されているとおり、生成された文字列は構文フォーマットの1つが必要です。例: `"Rudolfsplatz 13a, 1010 Vienna"`。
- *ラベル*: マップで表示されている位置情報に添付されてりラベルのテキスト。テキストは `xs:string` を生成する XPath 式として入力されます。
- *ズーム*: マップアプリケーション内で開かれるズーム機能。

現在地からのルートを表示する

現在地からのルートを表示する アクションは、モバイルデバイスのマップアプリケーションを開き、現在の場所から指定された位置情報へのルートを表示します。

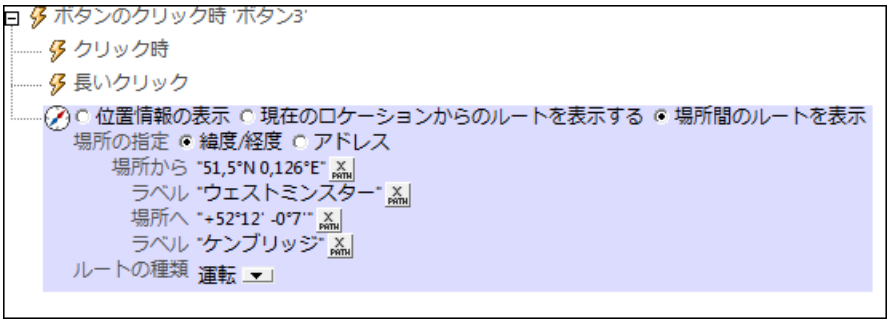


*現在地からのルートを表示*アクションには、以下の設定があります:

- *場所の指定*: 場所の設定 (次の設定) の入力が緯度 経度 座標または住所かを選択します。上のスクリーンショットは、選択された緯度 経度オプションを表示しています。
- *場所へ*: マップ内で目的地が表示されます。緯度 経度座標時が前の*場所の指定*設定で指定されている場合、(上のスクリーンショットに表示されているとおり) 場所の座標はとして入力される必要があります。下の位置情報入力文字列フォーマットのセクションで説明されているとおり、生成された文字列は構文フォーマットの1つが必要です。住所のオプションが選択されると、場所の入力は以下のようになります: `"Rudolfsplatz 13a, 1010 Vienna"`。
- *ラベル*: マップ上で表示されている目的地に添付されるラベルのテキスト。テキストは`xs:string` を生成する XPath 式 として入力されます。
- *ルートの種類*: ルートを使用する移動手段の種類を選択します: (i) 運転 (私的な移動手段)、(ii) 徒歩、(iii) 公共交通機関

場所間のルートを表示

場所間のルートを表示 アクションは、モバイルデバイスマップアプリケーションを開き、指定された2つの位置情報のルートを表示します。



場所間のルートを表示 アクションには、以下の設定があります:

- *場所の指定*: 場所の設定の入力が緯度 経度 座標または住所かを選択します。上のスクリーンショットは、選択された緯度 経度オプションを表示しています。
- *場所から*/*場所へ*: 開始点と目的地がそれぞれマップ上で表示されます。場所の座標はXPath 式を使用して入力されます。緯度 経度 座標構えの場所の設定で示される場合、場所の座標は (上のスクリーンショットに表示されているとおり) XPath 式を使用して入力される必要があります 下の位置情報入力文字列フォーマットのセクションで説明されているとおり、生成された文字列は構文フォーマットの1つが必要です。住所のオプションが選択されると、場所の入力は以下のようになります: `"Rudolfsplatz 13a, 1010 Vienna"`。
- *ラベル*: マップ上で表示されている目的地に添付されるラベルのテキスト。テキストは`xs:string` を生成する

XPath 式 として入力されます。

- *ルートの種類*: ルートを使用する移動手段の種類を選択します: (i) 運転 (私的な移動手段)、(ii) 徒歩、(iii) 公共交通機関

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(**"**) でハイライトされており、エスケープした単位インジケーターは青い(**""**) でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:

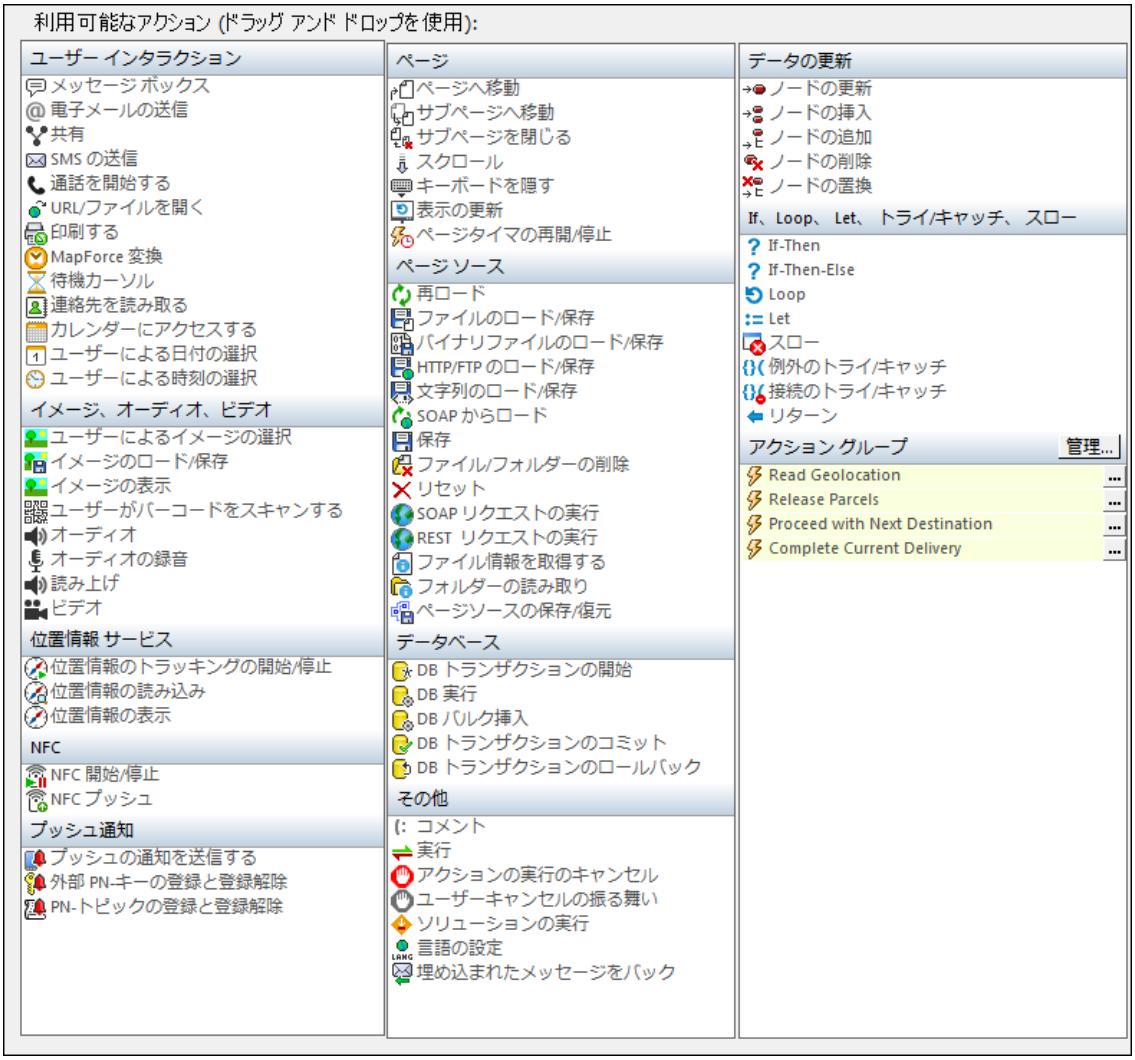
```
33.33N  -22°44'66.66"
33.33  22°44'66.66"W
33.33  22.c
```

位置情報データをデザイナーとサーバーシミュレーションのために指定するには、位置情報の設定のセクションを参照してください。

10.4 NFC

アクションダイアログのNFC グループで次のアクションを使用することができます (下のスクリーンショット):

- NFC の開始 停止
- NFC プッシュ

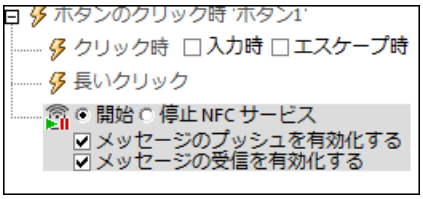


このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

MobileTogether 内でのNFC サポートに関するおおまかな説明はNFC のセクションを参照してください。

NFC の開始 /停止

NFC 開始 /停止 アクション (*下のスクリーンショット*) は、メッセージのプッシュおよび/または受信を開始、または、停止するために使用されます。



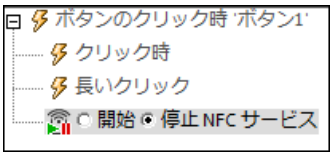
開始 オプションを選択します。アクションがデザインに追加された後、**$MT_NFC** ツリーは、自動的にページソースペインに追加されます (*下のツリー構造 を参照してください*)。

```
<Root>
   <Tag Id=""/>
   <NdefMessage
      CanMakeReadOnly=""
      IsWriteable=""
      MaxSize=""
      Type="">
      <NdefRecord
         Id=""
         TypeNameField=""
         RecordTypeDefinition=""
         Type=""
         Text=""
         Language=""
         URI=""
         Payload=""
         MimeType=""
         ExternalDomain=""
         ExternalPackageName="">
         <NdefRecord />
      </NdefRecord>
      <NdefRecord />
          ...
      <NdefRecord />
   </NdefMessage>
</Root>
```

開始 アクションがトリガーされるとプッシュ、および/または受信が開始されます。アクションを実行するためのステップのシーケンスは、以下のとおりです:

1. NFC がデバイス上で有効化されている必要があります。NFC が有効化されていない場合、開始アクションをトリガーすると、ユーザーに NFC の有効化を問うプロンプトが表示されます。
2. NFC が有効化された後、MobileTogether Client アプリが NFC のために登録されます。
3. この後、NFC タグの検索が自動的に開始され、NFC タグ内の NFC メッセージは自動的に受信されます。NFC プッシュアクションを使用して、プッシュを開始することができます。プッシュは自動的には開始されません。
4. **$MT_NFC** ツリー内に受信されたデータは保管されます。

停止 アクションは全てのメッセージのプッシュと受信を停止します。



メッセージのプッシュと受信を再開するには、開始アクションを再度トリガーしてください。

NFC プッシュ

NFC プッシュアクションは、プッシュするメッセージ、または、ビームするファイルを定義します (*下のスクリーンショット*)。アクションがトリガーされると、指定されたメッセージまたはファイルがNFC を介して送信されます。プッシュアクションが作動するには、デバイスがプッシュアクションを実行する準備を整えるため、NFC 開始アクションが既にトリガーされている必要があります。NFC プッシュ アクションが開始されると、 NFC メッセージまたはファイルを(複数のデバイスに)複数回送信することができます。エンドユーザーは送信デバイスのNFC 範囲内に受信するデバイスを設定するのみです。NFC が停止されると、または、キャンセルオプションを選択して、新規 NFC プッシュ アクションを追加することにより、 NFC プッシュがキャンセルされると継続的な送信は停止されます (*下のスクリーンショットを参照*)。

NFC プッシュアクションは、以下のオプションを提供します(*スクリーンショット参照*):

- *NFC プッシュ メッセージをプッシュします。*
- *Android ビーム*: Android ビームが有効化されているデバイスから他のAndroid ビームが有効化されているデバイスにファイルをビームします (Android デバイスでのみ使用することができます)。

実行するプッシュアクション型のラジオボタンを選択します(*下の説明を参照してください*)。

メモ 各アクション送信されるメッセージ/ファイルに対応しています。しかしながらそのメッセージは、複数のレコードから構成されている可能性があります。例えば、上のスクリーンショット内のメッセージには3つのレコードが含まれています。一件以上のメッセージを送信する場合は、新規 NFC プッシュアクションを各メッセージに追加します。

メッセージをプッシュする

NFC プッシュオプションにより送信するNFC メッセージを定義することができます。コンボボックスで、送信されるメッセージの標準の型を選択します (*下のスクリーンショット参照*) 。これらの型はNFC 技術の仕様内で定義されているとおりです。

テキスト、URI、およびMIME メディア型

`Text`、`URI`、および `MIME` メディアの標準型の定義は分かりやすいです。(スクリーンショットの横の URI オプショ

ンを参照してください)。下のスクリーンショットは、それぞれが異なる標準の型を定義している3つのレコードを持つNFCメッセージを表示しています。メッセージのコンテンツはXPath 文字列であることができ、また、ソースツリーノードからコンテンツを得ることもできます。最後のレコードでは、例えば、メッセージのコンテンツは、XML ページデータソース内のノードから得られて意います。

次の URI オプションを使用することができます:

- *SMS URI:* 送信された SMS は受信するデバイスの SMS アプリ内で開かれます。URI は次のようになります: **"sms:+439991234567?body=MyBody"**
- *電話の URI:* 受信デバイスの電話アプリが開かれ、転送された電話番号に電話がかけられます。サンプル URI: **"tel:+439991234567"**
- *電子メール URI:* 送信された電子メールは、受信デバイスの電子メールアプリ内で開かれます。サンプル URI: **"mailto:name@altova.com?subject=MySubject&body=MyBody"**

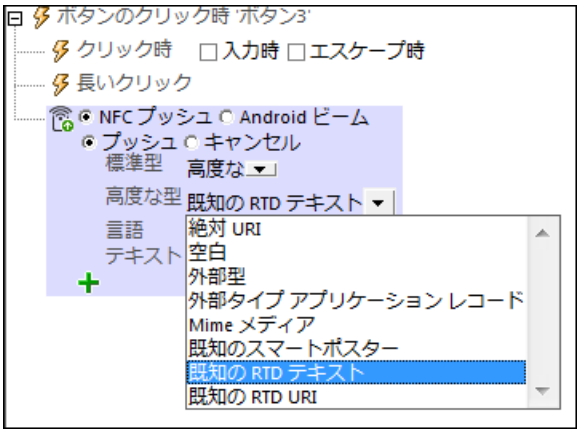
一部のメッセージの型内では、メッセージコンテンツは、保管しNFC メッセージのペイロードに転送するために、明示的に hexBinary に変換される必要があります (*NFC 技術の仕様を参照してください*)。 文字列をhexBinary に変換する場合、XPath 拡張関数 **`mt-string-to-hexBinary`** を使用してください。**`mt-bae64-to-hexBinary`** 関数 を使用してイメージをBase64 からhexBinary に変換することができます。NFC ペイロードがデバイスデバイス上で受け取られた場合、それぞれXPath 拡張関数 **`mt-hexBinary-to-string`** と**`mt-hexBinary-to-base64`** を使用して文字列、または、Base64 イメージに変換することができます (*次を参照してください! NFC タグの検出と読み取り*)。

イメージのプッシュ

NFC プッシュメッセージを使用してイメージを送信する場合、そのイメージに関連するMIME メディア型を使用して行うことができます (*下のスクリーンショット参照*)。 この種類のメッセージを定義する場合、NFC メッセージのペイロード内でイメージが送信できるようにイメージのBase64 フォーマットをhexBinary に変換する必要があります。この変換のためには、XPath 拡張関数 **`mt-base64-to-hexBinary`** を使用します。(デバイスから受信する) hexBinary から Base64 イメージに変換するには、関数 **`mt-hexBinary-to-base64`** を使用してください。

高度な型

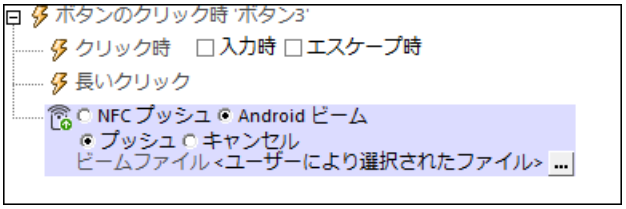
Advanced 型のメッセージのために固有の高度な型が選択される必要があります。(*下のスクリーンショット参照*)。各高度な型のために、対応するテキストを入力し、他の関連する情報を指定してください。



NFC の高度な型の詳細に関しては、次を参照してください：NFC 技術の仕様。

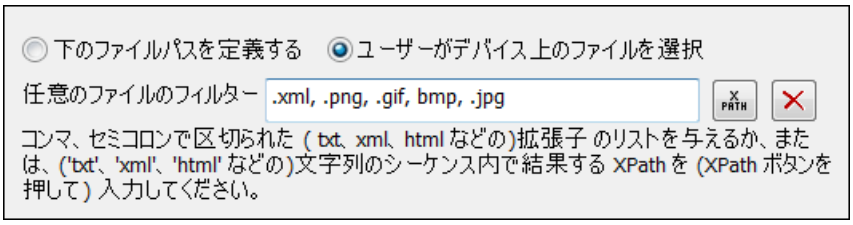
Android ビームを使用しファイルをビームする

Android ビームが2つのAndroid デバイス上で有効化されている場合、ファイルは1つのデバイスから他のデバイスにビームされます。*Android ビーム* と *プッシュ* ラジオボタンを選択し、どのようにユーザーが送信するファイルが選択されるかを定義してください (*下のスクリーンショット参照*)。



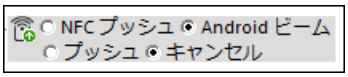
このファイルはクライアントデバイス上に存在する必要があります。ファイルはデザイン内で直接指定、または、ユーザーがファイルを選択して指定することができます (*下のスクリーンショット参照*)。

パスを直接指定し、パスが相対的な場合は、パスはAndroid デバイスのために指定したディレクトリをベースに相対的に解決されます。下のスクリーンショットは、ユーザーがデバイス上のファイルを選択するオプションが選択されているダイアログを表示しています。*任意のファイルフィルター* 設定によりクライアントデバイスの参照ダイアログを使用して開くことのできるファイルの型をフィルターすることができます。ここで定義されたファイル拡張子のみが開くことを許可されます。以下を入力することができます: (i) コンマ、または、セミコロンにより区切られた拡張子のリスト (例: `txt,html;xml`)、または、(ii)各文字列アイテムがファイルの型の拡張子である文字列アイテムのシーケンスを返す XPath 式 (例えば、3つの文字列アイテムを含む1つのシーケンス `'txt','html','xml'`).



プッシュ/ビームのキャンセル

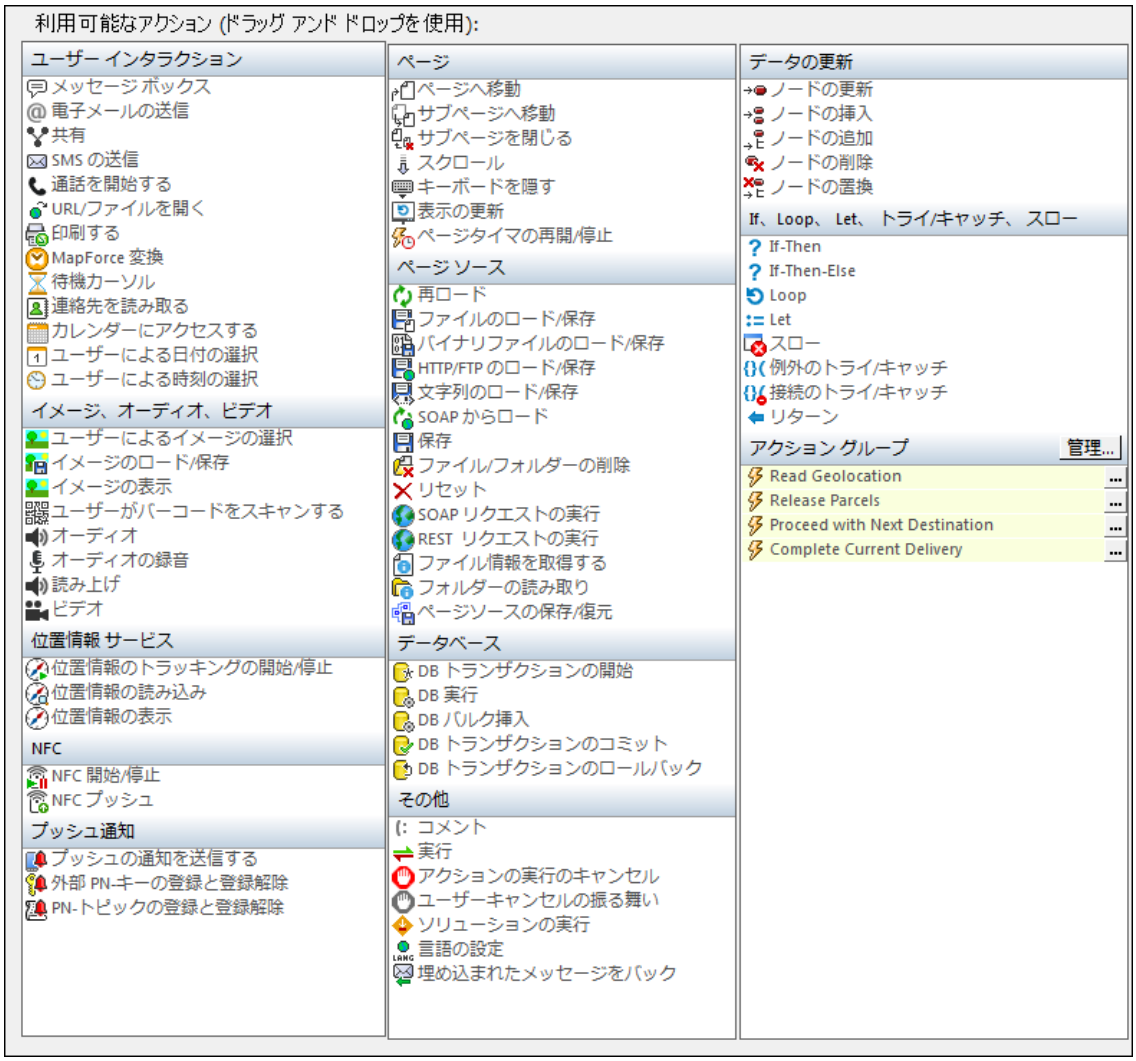
キャンセルアクションにより選択されたプッシュ または ビームアクションをキャンセルすることができます。



10.5 プッシュ通知

アクションダイアログのプッシュ通知グループで次のアクションを使用することができます (下のスクリーンショット):

- プッシュ通知の送信
- 外部プッシュ通知キーの登録(登録の解除)
- 外部プッシュ通知トピックの登録(登録の解除)

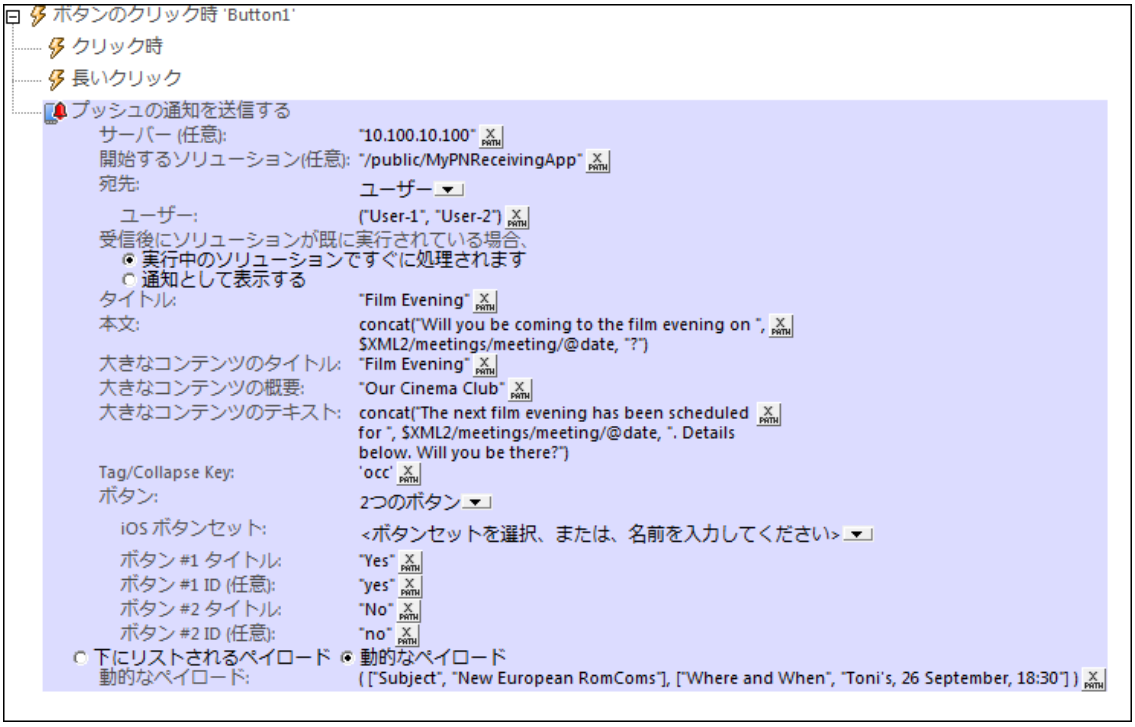


このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

MobileTogether 内でのプッシュ通知の使用に関するおおまかな説明はプッシュ通知 のセクションを参照してください。

プッシュ通知の送信

プッシュ通知の送信アクションがトリガーされると、(*下のスクリーンショット*) プッシュ通知 (PN) が (*下で説明されるとおり*) アクションの設定に送信されます。プッシュ通知は受信先のソリューションに送信されます。受信先のソリューションは次のとおりです: (i) サーバー上にホストされる同じソリューション。または、(ii) (送信元のソリューションと同じ) 同じサーバー、または、他のサーバーにホストされるソリューション。



下にリストされるアクションの設定は、プッシュ通知の異なるパラメーターを定義します:

- *サーバー (任意)*

  このオプションが指定されていない場合、送信元のソリューションと同じサーバー上でホストされる受信先ソリューションにプッシュ通知はターゲットされているものと想定されます。ソリューションの受信先が送信元のサーバーと異なる場合、サーバーをここで指定します。設定のXPath 式は、関連したサーバーのIP アドレスである文字列に対して解決される必要があります (*上のスクリーンショット参照*)。

- *開始するソリューション (任意)*

  このオプションが指定されていない場合、受信先のデバイスでは、プッシュ通知が受信された時、送信元のソリューションとして同じソリューションが開始されます。受信されるソリューションが送信されるソリューションと異なる場合、受信されるソリューションはこの設定で指定されます。設定のXPath 式の設定はサーバー上のソリューションのワークフローのパスである文字列に対して解決されます。上のスクリーンショットでは、例えば、ソリューションの受信先はIP アドレス `10.100.10.100` を持つサーバー上の `public` コンテナー内です。`MyPNReceivingApp` です。

- *送信*

  プッシュ通知の受信先を指定します。次のオプションを作用することができます:

  - *ユーザー*: プッシュ通知ユーザー設定内にリストされるか受信先のソリューションのサーバー上のユーザーに送信

されます。ユーザー設定は入力として単一の文字列のアイテム(例: `'User-1'`)、または、文字列アイテムのシーケンス(例: `('User-1', 'User-2')`) をとります。

- *ソリューションの全てのユーザー:* (受信先のソリューションのサーバー上の) ソリューションの受信先にアクセスすることのできるパーミッションを持つ全てのユーザー(MobileTogether Server の管理者設定 内でワークフローにアクセスするためのユーザーパーミッションは否定されています)。
- *ロール:* ロール設定内で指定されるロールを持つユーザーにプッシュ通知は送信されます。ロール設定は入力として単一の文字列アイテム(例: `'Role-1'`)、または、単一のアイテムのシーケンス(例: `('Role-1', 'Role-2')`)を取ります(ロールの詳細に関しては、次を参照: MobileTogether Server ドキュメント)。
- *外部プッシュ通知キー:* 外部キー設定内で指定されている外部プッシュ通知キーの1つに登録されているデバイスにプッシュ通知は送信されます。外部キー設定は入力として単一の文字列アイテム(例: `'Key-1'`)、または、単一のアイテムのシーケンス(例: `('Key-1', 'Key-2')`)を取ります。詳細に関しては次を参照: 外部プッシュ通知キーの登録 登録の解除)
- *トピック:* プッシュ通知はトピック設定内で指定されているトピックの1つにサブスクライブされているデバイスに送信されます。トピック設定は入力として単一の文字列アイテム(例: `'Topic-1'`)、または、単一のアイテムのシーケンス(例: `('Topic-1', 'Topic-2')`) を取ります。詳細に関しては次を参照: プッシュ通知トピックの登録

プッシュ通知の受信時にソリューションが既に実行されている場合

2つの可能性が挙げられます:

- ソリューションの受信先の **プッシュ通知の受信時** イベントのために定義されているアクションは、すぐに実行することができます。これは通知無しに行われ、プッシュ通知は表示されません。
- ソリューションの受信先の **プッシュ通知の受信時** イベントのために定義されているアクションは、ソリューションのユーザーがプッシュ通知をタップ、または、プッシュ通知内の対応するボタンをタップすると実行されます。

タイトル、本文

プッシュ通知は、タップされると大きな長いメッセージに展開される、短いメッセージにより構成されています。タイトルと本文設定はそれぞれ短いメッセージのタイトルとテキストであるテキスト文字列です。プッシュ通知のためにボタンが指定されていると(*下を参照*)、オペレーティングシステムにより短いメッセージの下に表示されます。

タイトル /概要 /大きなコンテンツのテキスト

プッシュ通知はタップされると長いメッセージを開く短いメッセージにより構成されています。大きいコンテンツタイトル、概要、テキストの設定は、テキスト文字列、長いメッセージのタイトル、概要、および、ボディはテキスト文字列を定義します。プッシュ通知のためにボタンが指定されている場合、(*下を参照してください*)、長いメッセージで表示されます。

キーをタグ付けする /折りたたむ

キーをタグ付けする/折りたたむはプッシュ通知と送信されるテキスト文字列です。文字列に対して解決されなければならない設定のXPath 式により生成されます。キーをタグ付けする/折りたたむが指定されていると、(i) このアクションにより生成されるプッシュ通知は同じキーを持ち、(ii) 受信先のデバイス上でこのキーを持つ全てのプッシュ通知は最後に受信されたプッシュ通知まで折りたたまれます。キーをタグ付け/折りたたむが指定されていないと、このアクションに送信される全てのプッシュ通知が表示されます。複数のプッシュ通知の送信アクションが同じキーをタグ付けする/折りたたむ場合、これらの異なるアクションにより送信されるプッシュ通知は全て同じキーを持ち、折りたたまれます。

プッシュ通知ボタン

プッシュ通知内に表示されるボタンの数量を(0 から3 で)指定します。プッシュ通知内のボタンの目的は、ユーザーが

定義済みのアクションから1つのアクションを選択することです。非 iOS モバイル デバイスの場合、タイトルと個別のプッシュ通知ボタンの任意のID を指定します。iOS デバイスの場合、使用するプッシュ通知 ボタンセット(AppStore App のみで使用できるプッシュ通知ボタン) を選択します。

- *iOS ボタンセット:* iOS デバイス上で表示されるプッシュ通知のためのボタンをMobileTogether ソリューションではなく、AppStore App 内で表示することができます。プッシュ通知の名前を入力または選択します。ここで定義されるプッシュ通知 ボタンセットは、ソリューションの受信先内で(**iOS プッシュ通知ボタンセット** コマンドを使用して)定義されなければなりません。送信されるソリューションが受信されるソリューションと同じ場合、定義されているプッシュ通知ボタンセットは、オプションのコンボボックス内に表示されます。例えば、「2 つのボタン」にボタンオプションが設定されていると、コンボボックス内にプッシュ通知ボタンセットのための2つのボタンが表示されます。送信されるソリューションが受信されるソリューションと異なる場合、プッシュ通知ボタンセットの名前はこの設定内に入力される必要があります。
- *非 iOS デバイスのための個別のプッシュ通知ボタン:* タイトル設定 は、プッシュ通知ボタン内で表示されるテキストです。*ID* 設定はプッシュ通知ボタンの数値をデフォルトの値として取ります。例えば、ボタン #1 には、`"1"` のID が存在します。任意で希望するID を入力することもできます。*ID* 設定の値は、ソリューションの受信先内でユーザーのボタン選択を参照するために使用されます。

メモ　プッシュ通知ボタンはボタンコントロールには関連しないことに注意してください。

ソリューションの受信先内で、ユーザーがプッシュ通知ボタンをタップすると、`$MT_PUSHNOTIFICATION/Root/@button` ノードの値としてボタンのID が**自動的**に入力されます(*下のページソースの構造を参照*)。ボタン #1 をユーザーがタップすると、`$MT_PUSHNOTIFICATION/Root/@button` ノードはボタン #1のID を含むようになります。このノード内の値をテストする条件付きステートメントを使用すると、アクションのセットがユーザーがタップするプッシュ通知ボタンに従い実行されます。ソリューションの受信先の**プッシュ通知の受信時** イベント内で実行するアクションが定義されます。

`$MT_PUSHNOTIFICATION` ページソースの構造:

```
$MT_PUSHNOTIFICATION
Root
|   @button
|
|-- Entry
|       @key
|       @value
```

ペイロード

ペイロードとは、プッシュ通知を使用して伝送されるデータであり、ソリューションの受信先の `$MT_PUSHNOTIFICATION` ページソースに保存されます(*下の構造を参照してください*)。このページソース内のデータをソリューションの受信先のデザイン内で使用できるようになります。

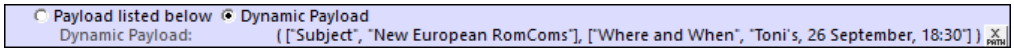
ペイロードはソリューションの受信先の `$MT_PUSHNOTIFICATION` ページソースの `Entry` 要素内のキー値ペアの配列として(インデックス順に)送信されます。

```
$MT_PUSHNOTIFICATION
Root
|   @button
|
|-- Entry
|       @key
|       @value
```

プッシュ通知のペイロードを次のように指定することができます:

- キー値ペアのリストとして (*下のスクリーンショットを参照*)。リスト内の各ペアはプラスのシンボルをクリックして追加され、キーと値のためにXPath 式が入力されます。各 XPath 式は単一の文字列を評価する必要があります。各キーのための文字列値 (または名前) は空であってはならず、一意である必要があります。値の文字列の値は一意である必要は無く、空であることも出来ます。

- 動的に決定されたキー値ペアの配列として。下のスクリーンショットは、XPath 式として直接入力されると配列がどのように表示されるかを示しています。配列は、しかしながら、ノードのセットを反復することによりランタイムに動的に取得することができます。各キーの文字列値は空ではなく、一意である必要があります。



メモ ペイロードキーの名前は `mt_` で開始してはならず、ここで挙げられるパラメーター名であってはなりません。https://firebase.google.com/docs/cloud-messaging/http-server-ref.

外部プッシュ通知キーの登録（登録の解除）

このアクションは、プッシュ通知を受け取るモバイルデバイス上のソリューションで使用されます。テキスト文字列をモバイルデバイス上のソリューションの外部プッシュ通知キーとして登録されます。デバイスが登録されることに注意してください。デバイス上の複数のソリューションが異なる外部プッシュ通知キーに登録されている場合、デバイスは全てのキーに登録されます。

プッシュ通知を送信する場合のプッシュ通知の受信の定義には色々ありますが、外部プッシュ通知キーにプッシュ通知を送信するように指定することができます（*プッシュ通知の送信を参照してください*）。デバイスがターゲット外部プッシュ通知キーの1つに登録されている場合、そのプッシュ通知が受信されます。

メモ　AppStore Apps 内でも外部プッシュ通知キーを使用することができます。

次も参照: ソリューションの受信先

登録

外部プッシュ通知トピックを登録するには、「登録」を選択し、XPath 式を使用してキーを指定してください。式はテキスト文字列に対して解決される必要があります。下のスクリーンショットでは、例えば、ボタンがクリックされると、ページソースノード内にある文字列がそのデバイスの外部プッシュ通知キーとして登録されます。このキーにプッシュ通知は送信され、モバイルデバイスプッシュ通知を受信します。

登録の解除

与えられた外部プッシュ通知キーに向けられているプッシュ通知を受け取ることをユーザーが希望しない場合、*登録の解除*アクションを使用して登録を解除することができます。これは、例えば、ボタンを押すことなどにより行うことができます。

プッシュ通知の登録を解除するには、アクションを追加し、登録の解除を選択します。1つの外部プッシュ通知キーのみが存在するため、キーを指定する必要はありません。

外部プッシュ通知 トピックの登録 登録の解除)

このアクションは、デバイスが1つまたは複数の選択されたトピックに関するプッシュ通知を受け取るためにプッシュ通知を受信するモバイルデバイス上のソリューション内で使用されます。例えば、**news** と**travel** としてマークされているトピックに関するプッシュ通知を受け取るようにデバイスが登録されている場合、**news**、または、**travel** としてマークされているプッシュ通知を受け取ります。送信される際、プッシュ通知はトピックと共にマークされています (次を参照してください: *プッシュ通知の送信*)。例えば、プッシュ通知がトピック**news** と**travel** に送信された場合、これらのトピックどちらか、または両方に登録されているデバイスにより受信されます。

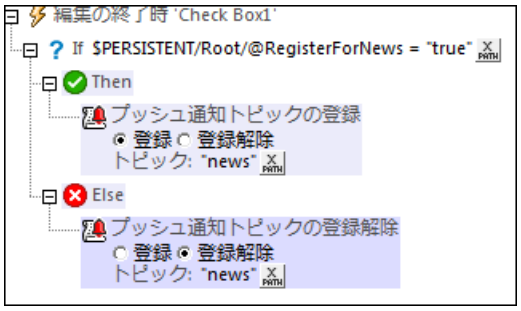
デバイス上のソリューションからアクションをトリガーするために、デバイスが登録されることに注意してください。プッシュ通知を受信するソリューションを使用してプッシュ通知 トピックを登録する場合は理論的に行ってください。

プッシュ通知を送信する場合のプッシュ通知の受信の定義には色々ありますが、プッシュ通知が1つまたは複数の特定のトピックに送信されるかを指定することができます(*プッシュ通知の送信を参照してください*)。

次も参照: ソリューションの受信先

登録

外部プッシュ通知キーを登録するには、「登録」を選択し、XPath 式を使用してトピックを指定してください。式は文字列 (例: **'news'**) 、または、文字列のシーケンス(例: **('news', 'travel')**) に対して解決される必要があります。下のスクリーンショットでは、例えば、プッシュ通知 トピックの登録アクションはページソースノードにリンクされているチェックボックスのために定義されています。このリンケージの結果、
ページソースノードは(チェックボックスがチェックされている場合) **true** の値を持ち、または、(チェックボックスがチェックされていない場合) **false** の値を持ちます。チェックボックスがチェックされているか、いないかにより、デバイスは**news** トピックに登録、または、登録されず、**news** トピックに送信するプッシュ通知を受信、または、受信しません(*スクリーンショットの参照*)。



登録の解除

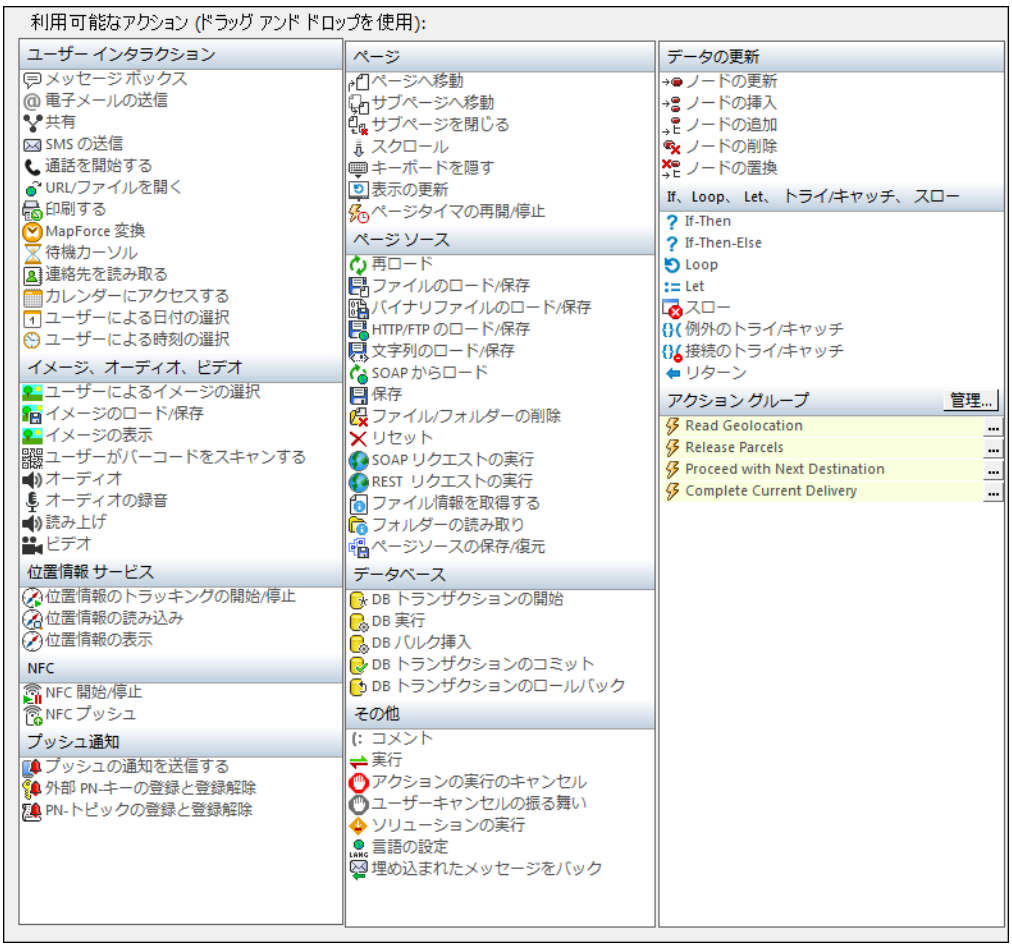
デバイスが登録されているトピックに送信されるプッシュ通知を受け取ることをユーザーが希望しない場合*登録の解除* アクションを使用して登録を解除することができます。これは、例えば、ボタンを押すことなどにより行うことができます。

プッシュ通知の登録を解除するには、アクションを追加し、登録の解除を選択し、登録を解除するデバイスのためにトピックを入力します。複数のトピックの登録を解除するには、トピックを文字列のシーケンスとして入力します (例: **('news', 'travel')**)。

10.6 ページ

アクションダイアログのページ グループで次のアクションを使用することができます:

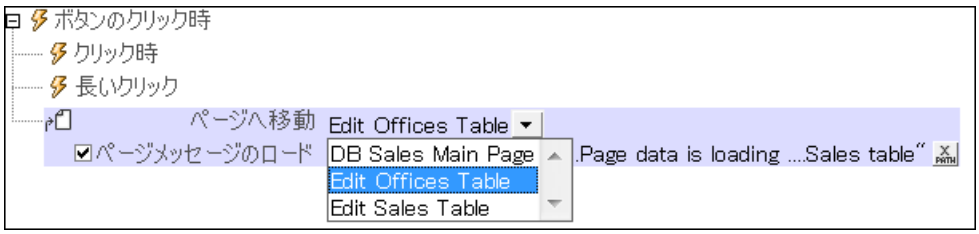
- ページへ移動
- サブページへ移動
- サブページを閉じる
- ソリューションの実行
- 一番下へスクロール
- キーボードを隠す
- 表示の更新
- ページタイマーの再開/停止



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロール アクション コマンドを選択します。ページ イベントおよびコントロール イベントも参照してください。

ページへ移動

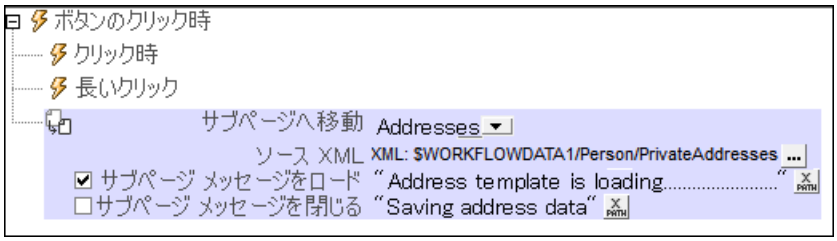
トリガーされるとページへ移動アクションは、指定されたページへ移動します。移動するページはアクションのコンボボックス、または、XPath 条件式を使用して選択することができます(*下のスクリーンショット参照*)。プロジェクトに他のページが存在しない場合、コンボボックスで選択するページはありません。下のスクリーンショットでは、ページへ移動 アクションが2つのサブイベント クリック時 と 長いクリック時 の下に配置されています。これはアクションがボタンがタップ(短いクリック)、または、押されたとき(長いクリック)にトリガーするアクションを定義します。



ページメッセージのロードオプション(*上のスクリーンショット参照*) によりページがロードされるときにメッセージが実行中ダイアログ内に表示されます。メッセージを表示する場合、このオプションをクリックして、XPath 式としてメッセージを定義します。オプションのチェックが解除されている場合、メッセージは表示されません。

サブページへ移動

トリガーされると、サブページへ移動します。プロジェクトの全てのサブページをリストする *サブページへ移動* コンボ ボックスから サブページへ移動を選択します (*下のスクリーンショット参照*)。ラベルのクリック時 イベントがMobileTogether プロジェクト ページの *Orders* のサブページへ移動するように設定されています。または、サブページの名前は、XPath 式を使用して 取得されます。例: `"Orders"` 、または、 `$XML3/DataSources/Subpage/@id="Name"`。



サブページへ移動アクションには以下の設定があります:

- *ソースXML:* サブページのデータが元のページのデータソースに対応するか、またどのように対応するか指定します。この定義はサブページ データマッピング ダイアログに入力され、*ソースXML* フィールドの **[追加ダイアログ]** ボタンをクリックすることでアクセスすることができます。このダイアログの詳細に関しては、"*サブページのソースXML*" " *セクションを参照してください。*
- *サブページ メッセージのロード:* サーバー リクエストにより、サブページが送信される場合、チェックボックスがチェックされていると、サブページのロード中、このメッセージが進行状況ダイアログに表示されます。*上のスクリーンショットを参照してください。*
- *サブページ メッセージを閉じる:* サーバー リクエストにより、サブページが送信される場合、チェックボックスがチェックされていると、サブページを閉じる間、このメッセージが進行状況ダイアログに表示されます。*上のスクリーンショットを参照してください。*
- *モーダル ダイアログとしてサブページを表示する:* このオプションが**チェックされていない**場合、表示内のトップページはサブページにより置き換えられます(ユーザーは **戻る** ボタンをタップしてトップページに戻ることができます)。このオプションが**チェックされている場合、**サブページは、(モーダルダイアログとして既知の) トップページの上の個別のウィンドウ内で表示されます。モーダルウィンドウ外をユーザーがタップすると、[戻る] ボタンのクリック時 ページのために指定されているアクションが実行されます。アクションがこのイベントのために指定されていない場合、サブページは閉じられ、実行はトップページに戻されます。

モーダルダイアログの定義
次の定義をモーダルダイアログに適用することができます:

- モーダル ダイアログ ウィンドウのディメンションを*幅* と*高さ*設定のXPath 式を使用して設定することができます。
- 任意の*高さ*設定が指定されていない場合、高さは、デバイスのビューポイントの高さの最高値までの値がコンテンツをフィットするために必要な高さに調整されます。更なる高さが必要な場合、モーダルダイアログにスクロールバーが自動的に追加されます。
- 入力するディメンションの値は、(ピクセルディメンションのための)数値、または、数値と(ダイアログのディメンションをビューポイントのディメンションのパーセンテージとして指定するための) % サインを含む文字列であることができます。
- 上のスクリーンショットでは、ディメンションは、ランタイムにデバイスから動的に取得された*デバイス固有の幅と高さの値(ビューポイントディメンション)*に設定されています。
- モーダル ダイアログに対して以下を選択することができます: (i) 縦向きと横向きに対して単一のディメンションのセットを使用する(ii) 各向きのための異なるディメンションのセットを使用する(すなわち、縦向きと、横向きにそれぞれ異なるディメンションのセットを使用する)。
- モーダル ダイアログが両方の向きのために単一のディメンションのセットを使用するように定義されている場合、幅と高さのディメンションは、*サブページへ移動*アクションが実行されるページの縦向きから取得されます。このオプションが選択されていると、モーダル ダイアログは、両方の向きで同じ幅と高さの値のセットを使用します。向きが変えられてもモーダルダイアログはサイズを変更しません。
- モーダル ダイアログとして表示されているサブページから、前のモーダルダイアログの上でモーダルダイアログとして開くこ

とのできる他のサブページに移動することができます。

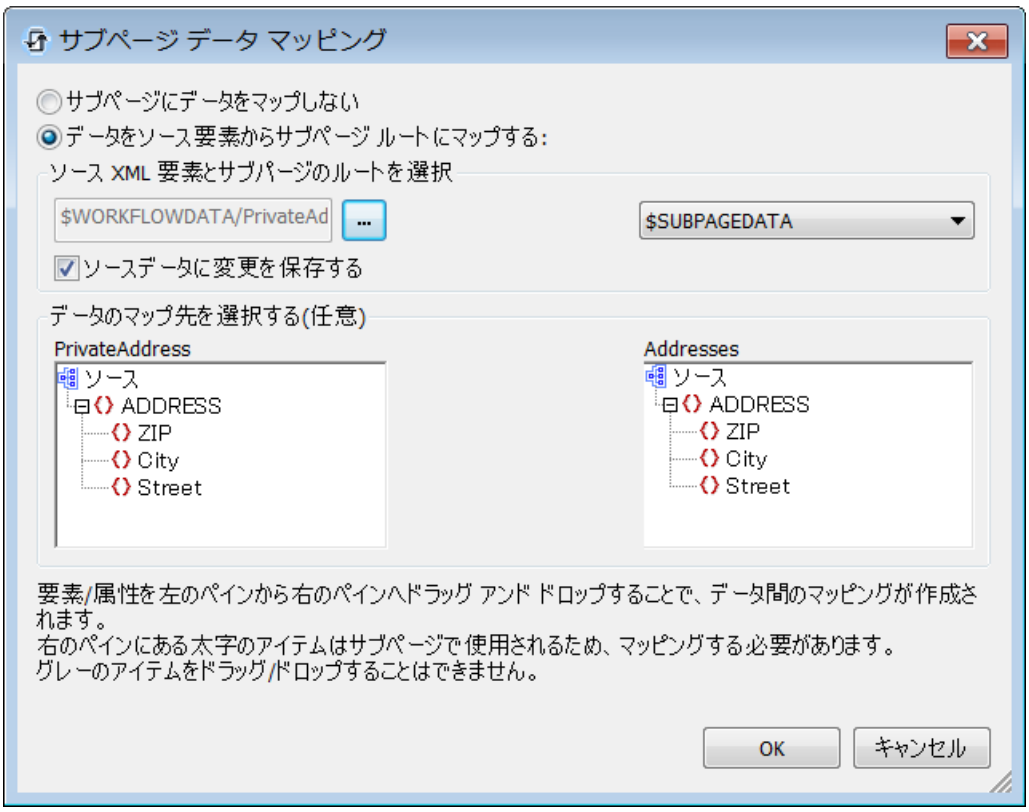
- すべてのモーダルダイアログのすべての相対的なディメンション(パーセンテージの値) は、他のモーダルダイアログから発生する場合も、**デバイスのビューポート**のディメンションに相対的に計算されます。
- モーダル ダイアログからサブページが、モーダルダイアログオプションを選択することなく開かれる場合、新規のサブページがモーダルダイアログと同じディメンションを持ち、オーバーレイされます。これは、新規のサブページが前のサブページにより置き換えられたように表示されます。

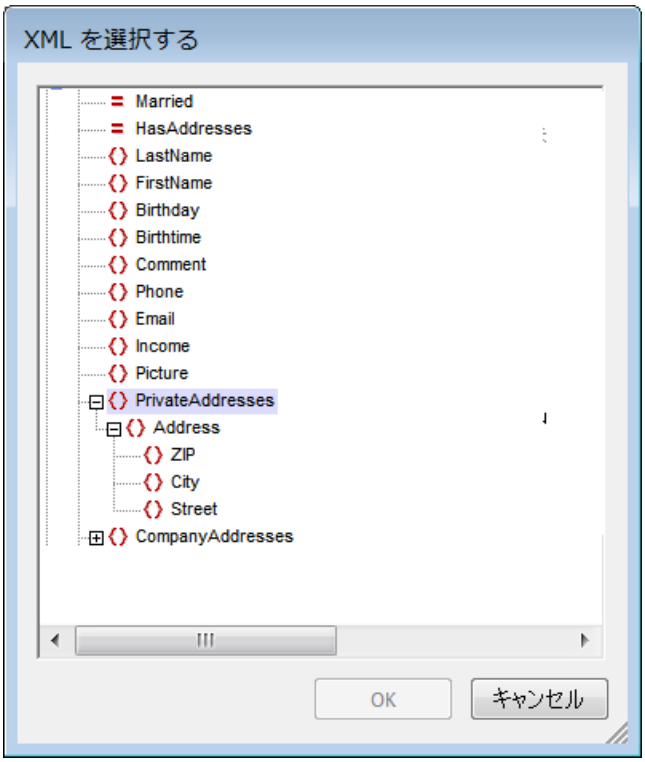
サブページのソース XML

サブページが効率的に使用されるために、各サブページには少なくとも1 つのデータソースが必要です。データソースがない場合、サブページのソースがマップされない可能性があります。マップされると、サブページが閉じられるときに、サブページのデータはトップページのマップされたノードに送信されます。サブページのソースXML の振る舞いは、[サブページ データ マッピング ダイアログ] で定義されます (*左のスクリーンショット*)。このダイアログにアクセスするには、[サブページへ移動アクション] の定義のソース*XML* フィールドの**[追加ダイアログ]** ボタンをクリックします (*上のスクリーンショット参照*)。

以下のサブページのデータ処理には以下の可能性があります:

- サブページがマップされていない。この場合、サブページのデータソースはサブページデータマッピングのメカニズムから独立して処理されています。
- サブページのソースが、要素と属性の名前も同じ、まったく同じの構造のトップページのにマップされている。
- サブページのソースが、異なる構造および/または異なる属性/要素を持つトップページ ソースにマップされている。

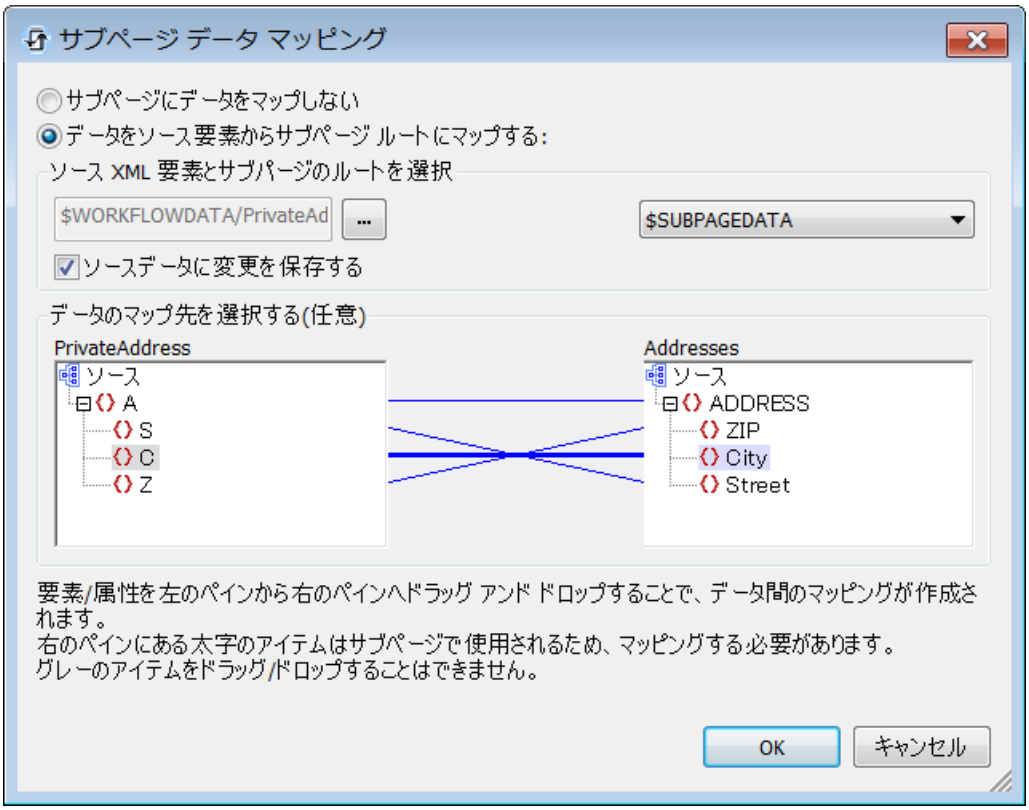




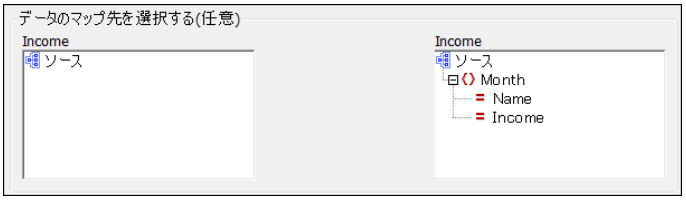
[サブページ データマッピング ダイアログ] (*左上のスクリーンショット*) では、サブページデータソースが右側の [ソースツリーペイン] に表示されており、トップページのデータソースは左側の [ソースツリーペイン] に表示されています。

- (右側) [サブ[ページ] ペイン] の上のコンボ ボックスからサブページデータソースを選択します。このコンボ ボックスはサブページの全てのデータソースをリストしています。サブページの名前は [サブ[ページ] ペイン] の上にリストされています。
- トップページのデータソースがマップされるには、最初に (左側) [トップ[ページ] ペイン] の上の **[追加ダイアログ]** ボタンをクリックします。これにより、トップページの全てのデータソースを含む [XML の選択ダイアログ] が表示されます (*右上のスクリーンショット*)。サブページデータにマップしたいノードを選択します。上のスクリーンショットでは、`PrivateAddresses` ノードが、[XML 選択ダイアログ] で選択されています。この結果、`PrivateAddresses` の子孫は、[トップページ ソースツリー ペイン] に表示され、マップされたノード *PrivateAddresses* はペインの上に表示されます。

上のスクリーンショットで説明されているケースについては、両方のデータ構造は要素の名前も含め同一のため、2 つのデータソース間のマッピングは自動的におこなわれます。下のスクリーンショットでは、左側のペインからノードを右側のペインにドラッグすることにより要素間のマッピングが行われています。これは 2 つのデータ構造が同一ではないためです (最初の要素の S が 3 番目の要素 `Street` に一致しなければなりません)。

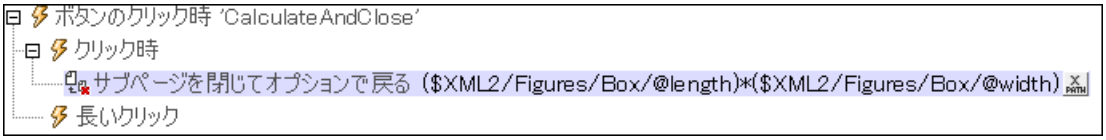


下のスクリーンショットの場合、サブページのデータノード `Income/Months/Month` は、トップページのデータノード `Income` にマップされています(ペインのタイトルに表示されています)。



サブページを閉じる

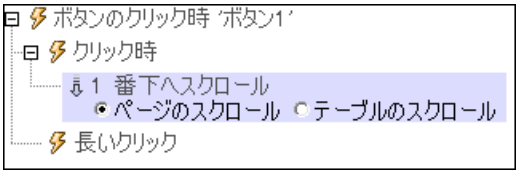
トリガーされると、アクティブなサブページを閉じます。オプションで、サブページが閉じられると、戻り値を保存することができます。戻り値は、指定されたXPath 式により生成されます (*下のスクリーンショット参照*)。値は、Let アクションのサブページ結果として取得されることができます。



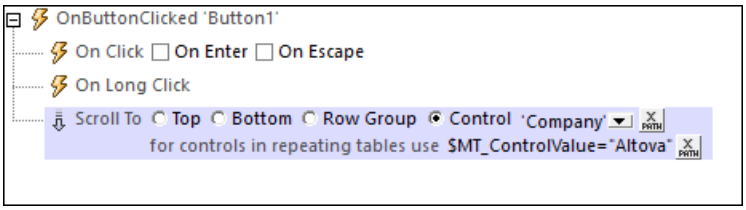
スクロール

アクションがトリガーされると、以下のソリューションが以下の条件でスクロールすることを指定することができます:

- アクティブなページの上、または、下の部分 (*下のスクリーンショット*)。

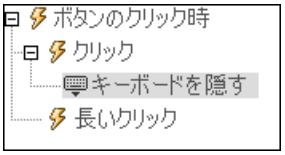


- 選択されたテーブルの上、または、下の部分。上下のスクロールがテーブルのオプションのコンボボックスの選択リストに含まれているテーブルのみで使用することができます。
- テーブルの行グループは、名前別に識別されます。テーブル名は、コンボボックスから選択、または、XPath 式として入力されます。 XPath 条件を任意で指定して、条件が **`true`** を評価する(選択されたテーブルの)最初の行グループを選択します。
- 名前別にコントロールを選択します (*下のスクリーンショットを参照*)。コンボボックスから名前を選択、または、コントロール名を評価するXPath 式を作成します。 繰り返しテーブル構造内でコントロールを選択する場合、コントロールのための検索を絞り込むために、追加の条件をXPath 式を使用して、指定することができます。コントロール情報を返す動的な変数, **`$MT_ControlValue`** などのコントロールの情報を返す動的な変数 が使用され、動的に変更するコンテキスト内でコントロールを検索することができます。



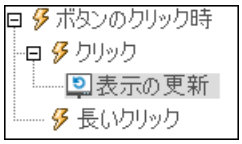
キーボードを隠す

アクションがトリガーされるとモバイルデバイスのキーボードを隠します。テキストフィールドなどの特定のページ要素が、ページに存在またはアクティブな場合、一部のモバイルデバイスは自動的にキーボードを表示します。キーボードを隠すアクションは、デバイス上でページを表示するスクリーンスペースが必要な場合に役に立ちます。



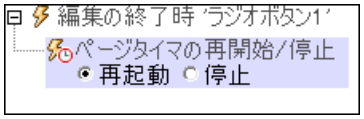
表示の更新

アクションがトリガーされると最新のデータを表示します。アクションがトリガーされると複数のアクションのシーケンスが実行される場合、デフォルトでは、最後のアクションが実行された後に、表示が更新されます。表示の更新アクションはイベントが実行されると表示の更新を有効化します。良い例はループアクション内に含まれるアクションの表示の更新です。表示の更新アクションがループ内に含まれる場合、表示はループの各反復が実行されると更新されます。



ページタイマーの再開 /停止

ページタイマーは、**OnPageRefresh** イベントタブで構成され、ページの更新の間隔の時間を設定するために使用されます。タイマーの初期の更新の間隔はタイマーの構成で指定されています。ページがロードされると、タイマーが開始され、ページが指定された間隔で更新されます。

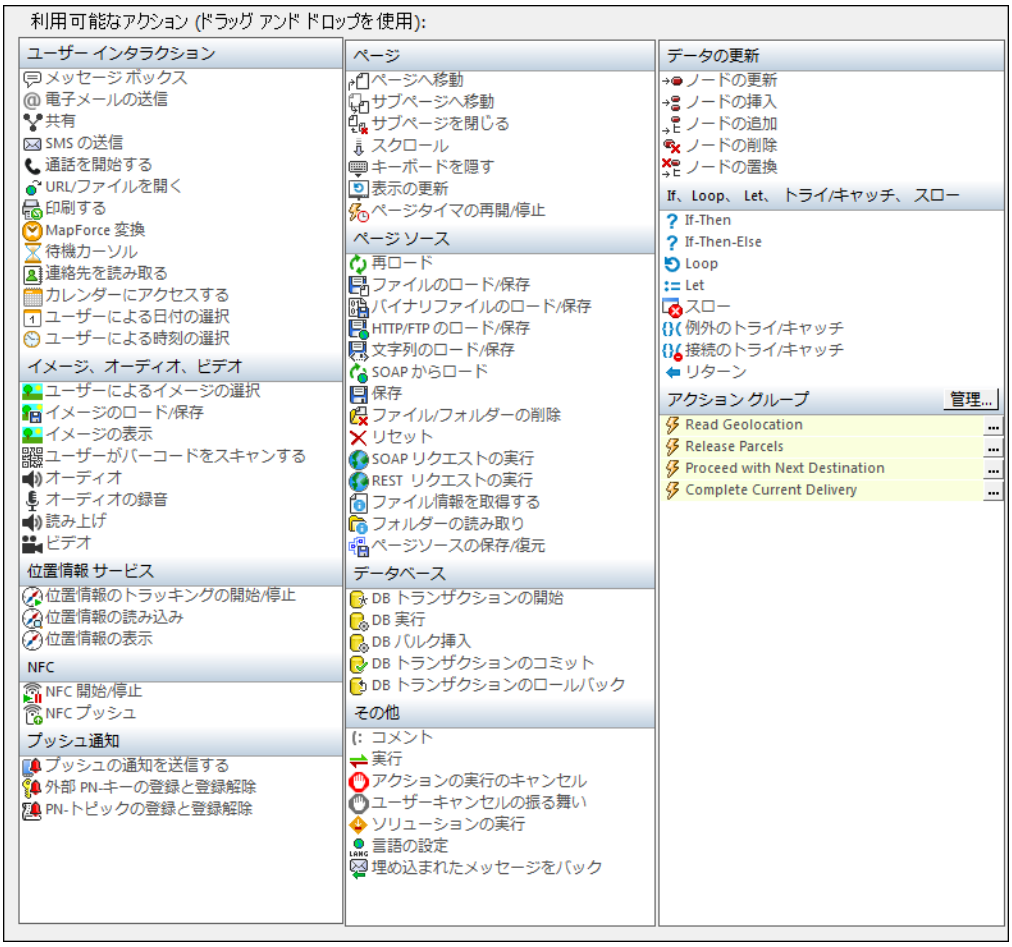


- *ページタイマーの再開:* タイマーの更新 間隔がページの処理中に変更された場合、タイマーは再起動されなければなりません。これは新しい更新間隔と共にタイマーを再初期化するためです。サンプルに関しては、*SOAP リクエストのチュートリアル* を参照してください。
- *ページタイマーの停止:* ページが更新されると*タイマーの更新時アクション* 内で定義されたアクションが実行されます。タイマーが継続して実行される場合、ページは指定された間隔で定期的に更新されます。更新アクションが実行されます。ページの処理中に後でこれらのアクションを停止する場合は、ページタイマーの停止 アクションを使用します。これによりタイマーは停止され、その結果アクションが停止されます。

10.7 ページ ソース

アクションダイアログのページソースグループで次のアクションを使用することができます (下のスクリーンショット):

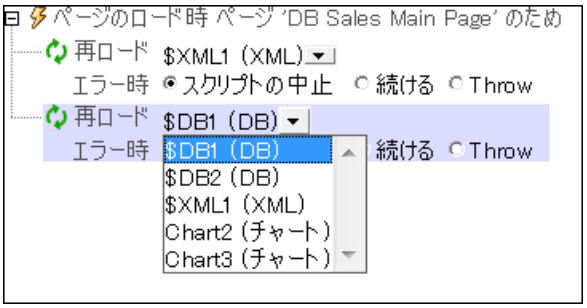
- 再ロード
- ファイルをロード/保存
- バイナリファイルのロード/保存
- HTTP/FTP をロード/保存
- 文字列をロード/保存
- SOAP からロード
- 保存
- ファイル/フォルダーの削除
- リセット
- SOAP リクエストの実行
- REST リクエストの実行
- ファイル情報の取得
- フォルダーの読み取り
- ページソースの保存/復元



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロールアクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

再ロード

*アクションのコンボ ボックス*で指定された外部*ソース*を再ロードします(*下のスクリーンショット参照*)。プロジェクトで参照される外部*ソース*は、コンボ ボ*ックス*のエントリとして使用することができ、XML ファイル、チャート、イメージを含みます。



以下の点に注意してください:

- 上のスクリーンショットは、チャートコントロール全体を含みます。更新が選択されると、チャートイメージはイベントがトリガーされると更新されます。イメージコントロールは、更新アクションによりリンクされたイメージを持つことができます。
- コントロール イベントにより更新を行うこともできます。これは、たとえば、コンボボックスが編集されると、コンボ ボックスの編集の終了時 イベントがイメージコントロールにリンクされたイメージを更新するように設定することができます。クイックスタート(パート1) チュートリアルのこの例を参照してください。
- イベントがトリガーされると、複数のリソースを再ロードするには、上のスクリーンショットで表示されているように複数の再ロードアクションを追加してください。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

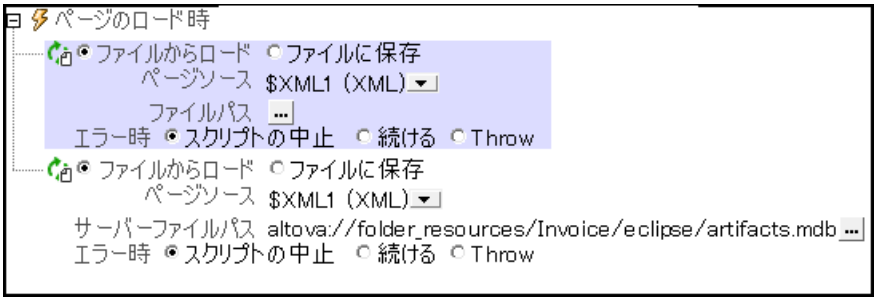
- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

ファイルをロード/保存

アクションを以下のとおり設定することができます: (i) ファイルからロードする、または(ii) ファイルへデータを保存する。実行されるアクションが、ロードアクションまたは保存アクションかを指定するには、適切なラジオボタンを選択してください(*下のスクリーンショット参照*)。

ファイルからロード

各 ファイルからロード アクションでは、アクションコンボボックスからページソースを選択することができます(*下のスクリーンショット参照*)。イベントがトリガーされた際にそのページソースのためにどのデータがロードされるかを指定することができます。



イベントがトリガーされた際に、複数の ページソースからデータをロードするには、上のスクリーンショットに表示されるように、複数の`LoadFromFile` アクションを追加します。

エラーの処理

エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合(*成功時*)、またはエラーが発生する場合(*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションの キャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

ファイルに保存

アクションコンボボックス内で選択されたページソースからのデータをファイル パスフィールド内で指定されたXML ファイルに保存します(*下のスクリーンショット参照*)。XML ファイルのエンコードは、エンコードフィールドで指定されています。複数の データソースのためにデータを保存するには、複数の ファイルに保存 アクションを追加します。

イベントがトリガーされた際に複数のデータソースからデータを保存するには、複数の ファイルに保存 アクションを追加します。他の ファイルに保存 アクションを追加するには、ファイルのロード/保存 アクションをイベントタブにドラッグし、ラジオボタンを ファイルに保存 に設定します。

メモ ファイルに保存 アクションはDB に保存するために使用されます。DB に保存するには、保存 アクションを使用して、ページソースツリーの1つを選択します、または、DB の実行 アクションを使用します。

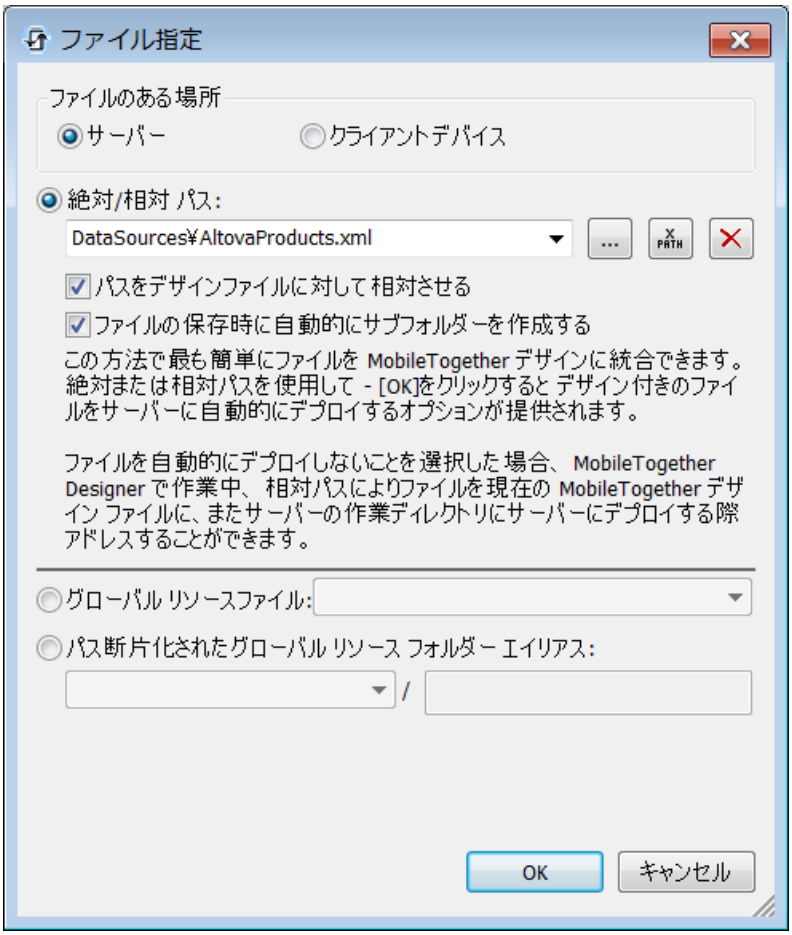
エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

ファイルの場所

ファイルからロード/保存 ファイルアクションのファイル パスフィールドの(上のスクリーンショット参照)**[追加ダイアログ]** ボタンが押されると、ファイルの指定ダイアログが表示されます。このダイアログ内で、ファイルがサーバーまたはクライアント上に存在するかを指定します (*下のスクリーンショット参照*)。

ファイルがサーバーに存在する場合
ファイルがサーバーに存在する場合、ファイルの場所(絶対/*相対* パス)を参照、またはグローバルリソース(*ファイルエイリアスまたはフォルダーエイリアス*)によりファイルを指定します。必要なオプションを選択します。

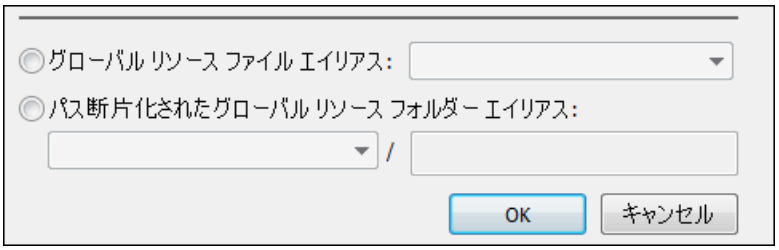


- **絶対 相対 パス:** ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細は*プロジェクト ファイルの場所*を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対/絶対パスがファイルにアクセスするために(サーバーのデータベース) 内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関しては*プロジェクトファイルの場所*のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイル パス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

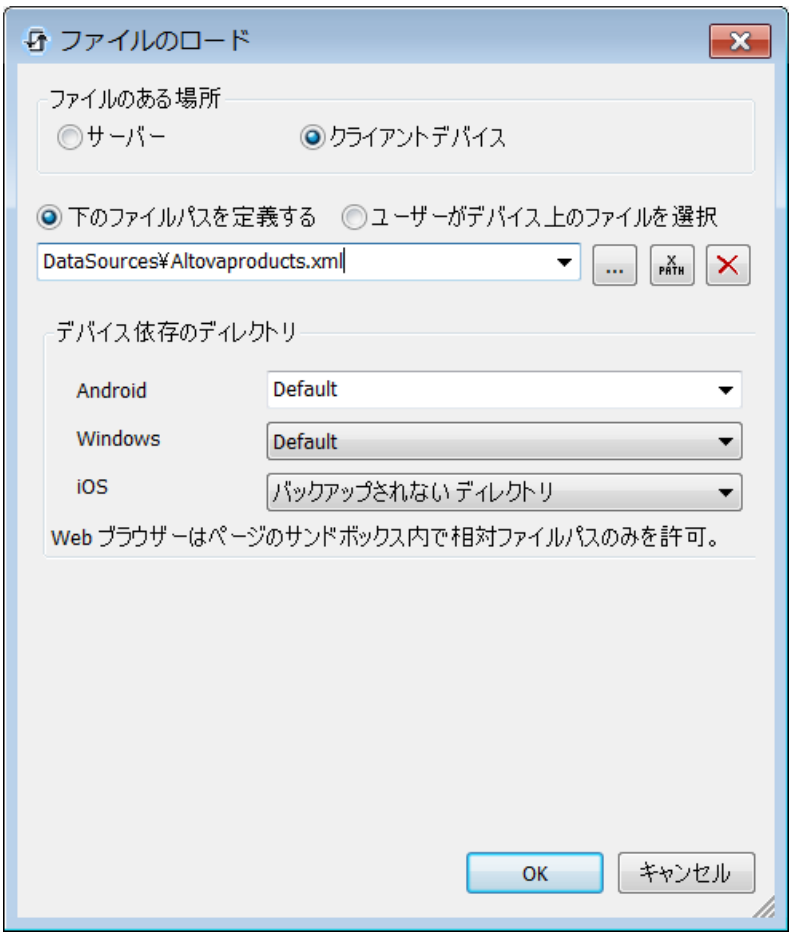
- **グローバル リソース ファイル エイリアス:** コンボ ボックス内で使用可能なファイル エイリアスを選択します。使用可能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイル リソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova *グローバル リソース*のセクションを参照してください。

- **パス断片化されたグローバル リソース フォルダ エイリアス**: コンボ ボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがクライアントに存在する場合
ファイルがクライアントに存在する場合、ファイルの場所を選択することによりパスを指定、または XPath 式を使用して式を作成します。**更新**ボタンを使用して現在のエントリを削除します。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (*ソースファイル* と *ターゲットファイル* オプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。
- **デバイスに依存するリソース:** ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。
- **シミュレーションのためのファイルのロケーション:** ファイルがクライアントに存在するため、シミュレーション中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ (**ツール | オプション**)内で指定されていなければなりません。

エンドユーザーにより(クライアントデバイス上で)ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- *任意のファイルフィルター:* クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト(例: `txt,html;xml`)、または(ii)各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式 (例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`)
- *オプションのデフォルトのファイル:* 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。
- *Web メッセージボックス:* ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、またはXPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。
- *ファイルの保存時に自動的にサブフォルダーを作成する:* もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの

場合使用できません。

メモ iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法はiCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

エラーの処理

エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション の キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

バイナリファイルのロード/保存

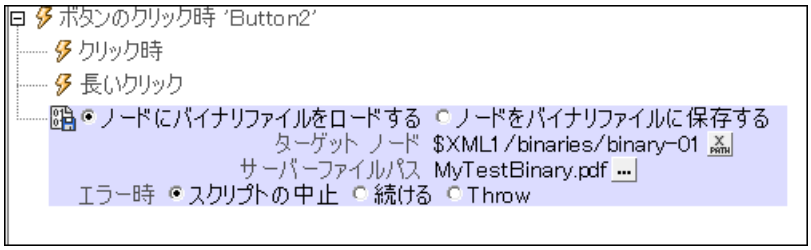
バイナリファイルのロード/保存アクションで以下を行うことができます:

- データソースノードにバイナリファイルをBase64 エンコードイメージとしてロードする。
- サーバー側またはクライアントに、Base64 エンコードイメージをデータソースノード内にバイナリファイルとして保存する。

PDF などのバイナリファイルをXML コンテンツとして保持することが可能になります。

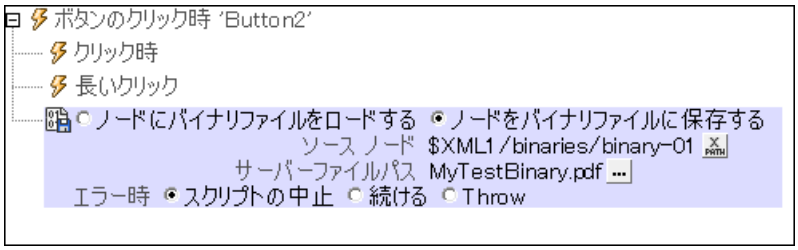
バイナリファイルをデータソースノードにロードする

バイナリファイルのロード/保存アクションのノードにバイナリをロードするオプションを使用して、バイナリファイルをデータソースノードにロードすることができます (*下のスクリーンショットを参照*)。XPath 式を使用して、バイナリデータが保管されるデータソースであるターゲットノードを選択します。ファイルパスフィールド内から、ターゲットノードにロードされるバイナリファイルを選択します。バイナリファイルのデータはBase64 に変換され、ターゲットノード内でBase64 エンコードコンテンツとして保管されます。



Base64 エンコードコンテンツをバイナリファイルとして保存する

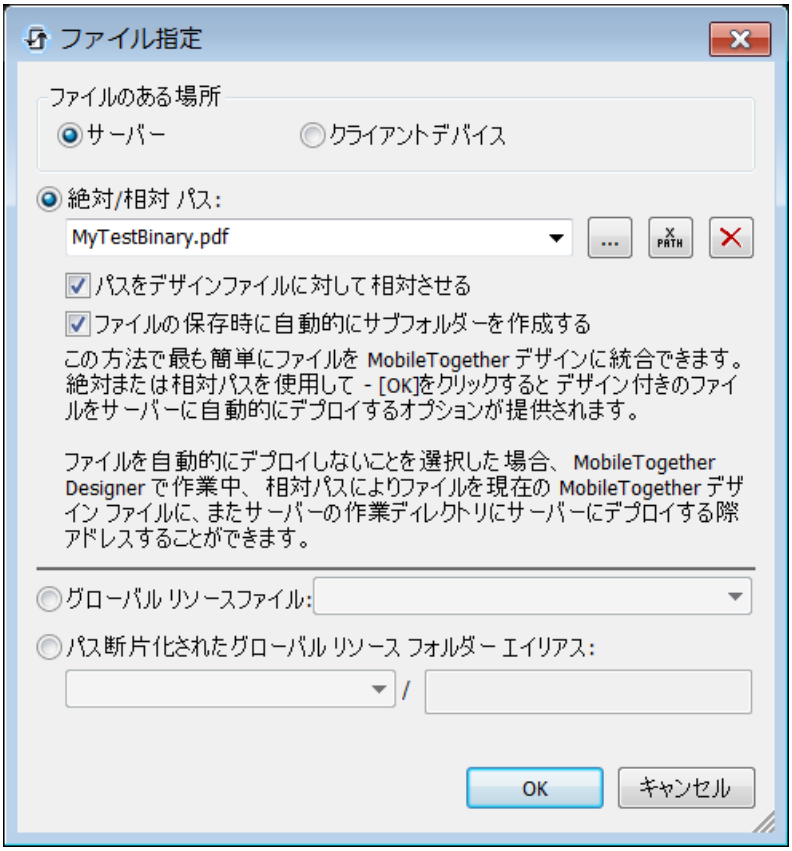
データソースノード内に保存されているBase64 イメージデータは、バイナリファイルのロード/保存 アクションのバイナリをファイルに保存オプションを使用して、バイナリファイルとして保存することができます (*下のスクリーンショット参照*)。 Base64 エンコードコンテンツが存在する、データソースノードの場所を選択します (*下のスクリーンショット参照*)。ファイルが保存されるサーバーまたはクライアントの場所を(ファイルパス フィールドで)選択します 。



バイナリファイルのロケーション

イメージのロード/保存 アクションのファイルパスフィールドの**追加ダイアログ**ボタンをクリックすると、(*上のスクリーンショット参照*)、ファイルを指定ダイアログは表示されます。このダイアログ内で、ファイルがサーバーまたはクライアントに存在するかをそれぞれのラジオボタンを選択することにより指定することができます (*下のスクリーンショット参照*)。

ファイルがサーバーに存在する場合
バイナリファイルがサーバーに存在する場合、場所を(*絶対/相対パスを使用して*)参照、または、ファイルをグローバルリソース(*ファイルエイリアスまたはフォルダーエイリアス*)を使用して指定することができます。使用するオプションを選択してください。



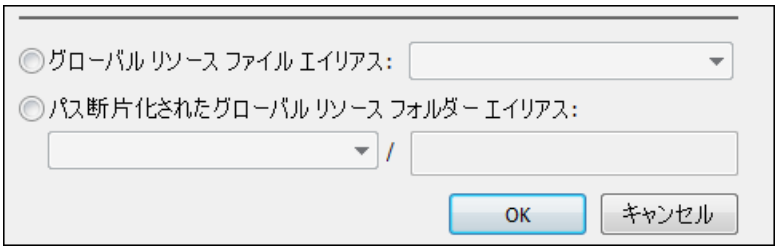
- **絶対/相対 パス:** ファイルを参照するために、絶対および相対パスを入力します。ソリューションがデプロイされると、デフォルト ファイルの場所は、MobileTogether Server 設定で定義されているとおり、*作業ディレクトリ* への参照で解決されていることに注意してください。ファイル指定ダイアログのパスが相対の場合、*作業ディレクトリ* に対応して解決されます。ファイル指定ダイアログのパスが絶対の場合、デフォルト ファイルを含むフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細は*プロジェクト ファイルの場所*を参照してください。パスを入力またはファイルを参照することができます。パスはデザインに対して相対または絶対であることができます。ファイルがデザインファイルと共にサーバーにデプロイされると、ダイアログで指定された相対/絶対パスがファイルにアクセスするために(サーバーのデータベース)内部で使用されます。ファイルがデプロイされていない場合、ファイルは直接サーバーに保管される必要があります。この場合: (i) ファイルの指定ダイアログで相対パスが選択されている場合、ランタイム時に、この相対パスはサーバー上で(MobileTogether Server 設定で定義された) *作業ディレクトリ* を参照して解決されます。(ii) ファイルの指定ダイアログで絶対パスが選択されている場合、サーバー上でのファイルのフォルダーは*作業ディレクトリ* の子孫である必要があります。詳細に関しては*プロジェクトファイルの場所*のセクションを参照してください。ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内で中間フォルダーが存在しない場合、ファイルが保存される時に作成されます。このオプションは保存する場合のみ関連性があります。アクションがファイルのロードのみに制限される場合は、このオプションは、存在しません。

- **グローバル リソース ファイル エイリアス:** コンボボックス内で使用可能なファイル エイリアスを選択します。使用可

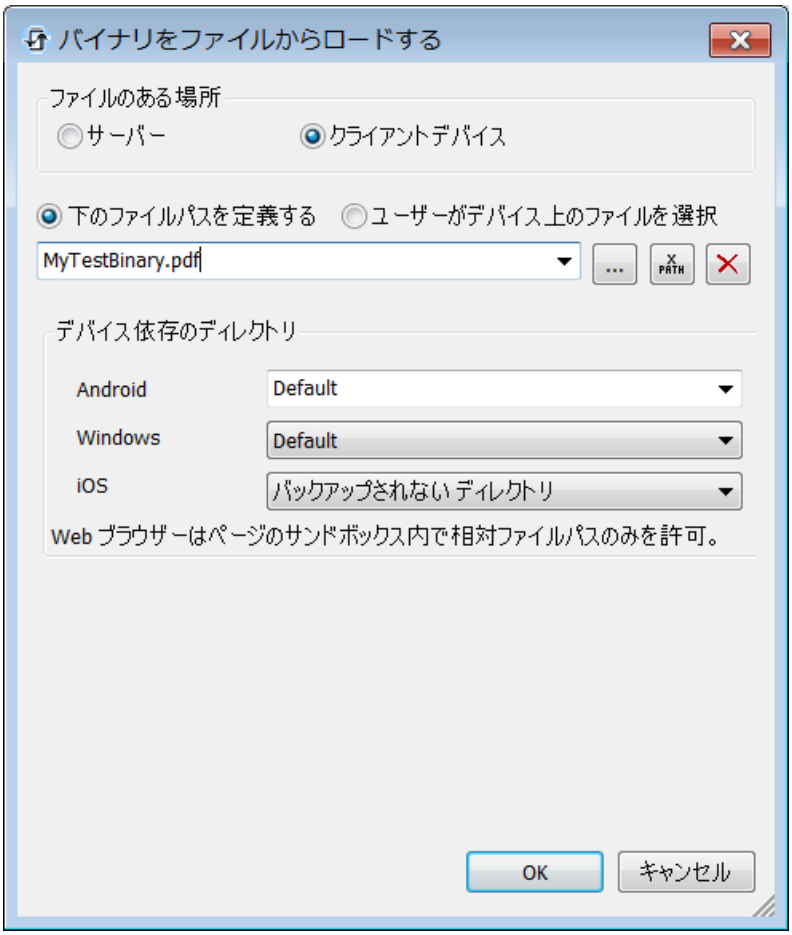
能なファイル エイリアスはグローバル リソース定義ファイルで定義されたものです。各ファイル エイリアスは、異なるファイルリソースを(**[ツール] アクティブな構成]** により選択された) MobileTogether Designer で現在アクティブな構成に応じてマップします。詳細は Altova グローバル リソースのセクションを参照してください。

- **パス断片化されたグローバル リソース フォルダ エイリアス**: コンボ ボックスの使用可能なフォルダー エイリアスからフォルダー エイリアスを選択します (*下のスクリーンショット参照*)。



使用可能なフォルダー エイリアスはグローバル リソース定義ファイルで現在定義されているものです。各フォルダー エイリアスは、(**[ツール] アクティブな構成]** コマンドにより選択された) MobileTogether Designer で現在アクティブな構成に応じて異なるフォルダー リソースをマップします。パス断片化はリソース ファイルへの残りのパスを指定します。詳細は Altova グローバル リソースのセクションを参照してください。

ファイルがサーバーに存在する場合
バイナリファイルがクライアントに存在する場合は、場所を入力またはお選択して行ってください。または、XPath 式を使用してパスを作成することもできます。**更新**ボタンを使用して現在のエントリを削除してください。



デザイナーがロード/保存するファイルを指定することができ、また、エンドユーザーもこれらのファイルを指定することができます。ファイル指定すると、この情報はソリューションに保存され、ファイルはアクションがトリガーされると、ロードまたは保存されます。ユーザーがロードまたは保存されるファイルを選択する場合、アクションがトリガーされると、クライアントデバイスで参照ダイアログが開かれ、ユーザーがロード/保存するファイルを入力または選択することができます。

メモ エンドユーザーがファイルをロードして保存するオプションは以下のアクションに対して使用することができます:印刷 (ソースファイルとターゲットファイルオプションがあります)、ファイルのロード/保存、イメージのロード/保存 、バイナリのロード/保存 。

メモ クライアント上のファイルもモバイルデバイス上のSD カードに保存することができます。

ファイル名は(ソリューションのデザイナーにより)以下で定義されます

- *ファイルの保存のためのデフォルトのファイル拡張子:* ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。
- **ファイルの保存時に自動的にサブフォルダーを作成する** ファイルパス内の中間のフォルダーが、クライアントで不在の場合、ファイルが保存される際に作成されます。このオプションは、保存する際にのみ関連するオプションです。アクションがファイルのロードのみに制限されている場合、不在の場合があります。
- **デバイスに依存するリソース** ファイルが存在するディレクトリを選択します。 Windows Phone/RT とiOS デバイスでは、許可されるディレクトリは既に決まっています。Android デバイスでは、*Android* コンボボックスのドロップダウンリスト内のディレクトリに追加して、ファイルを入力することができます。Android とWindows Phone/RT

で、デフォルトである、デフォルトを選択すると、MobileTogether アプリのサンドボックスディレクトリが選択されます。iOS デバイスでは、MobileTogether は2つのディレクトリを作成します: (i) iCloud に保存される再度ダウンロードすることのできるバックアップディレクトリ、(ii) バックアップを必要としないファイルのためのバックアップしないディレクトリ。必要に応じてバックアップディレクトリまたはバックアップされないディレクトリを選択します。Web ブラウザー内では、ファイルはブラウザーのサンドボックスに相対して検索されます。

- **シミュレーションのためのファイルのロケーション**: ファイルがクライアントに存在するため、シミュレーションに中に使用することはできず、シミュレーション中にクライアントフォルダー内でスタンドインするフォルダーを指定することができます。このスタンドインフォルダーは、もちろん、デザイン内で指定されたファイルと同じ名前を持つ必要があります。このフォルダーはオプションダイアログのシミュレーションタブ (**ツール | オプション**)内で指定されていなければなりません。

エンドユーザーにより(クライアントデバイス上で)ファイル名が定義される場合

- *ファイルの保存のためのデフォルトのファイル拡張子*: ファイルを保存する場合、デフォルトのファイル拡張子を任意で指定することができます。この拡張子はファイル名と共に拡張子が指定されていない場合使用されます。

- *任意のファイルフィルター*: クライアントデバイス上で開かれる参照ダイアログは、定義された拡張子が許可されるように、ロードまたは保存されるファイルの型をフィルターします。以下を入力することが可能です: (i) カンマで区切られた、または、セミコロンで区切られた拡張子のリスト(例: `txt,html;xml`)、または(ii)各文字列アイテムがファイル型の拡張子である文字列アイテムのシーケンスを返す XPath 式 (例えば、ここでは以下の3つの文字列アイテムを含む1つのシーケンス `'txt'、'html、'xml'`)

- *オプションのデフォルトのファイル*: 直接または式を使用して、エンドユーザーを導くためにデフォルトのファイル名を入力します。

- *Web メッセージボックス*: ファイルを開く/保存 ダイアログが開かれる前に、メッセージボックスが表示されます。直接テキストを入力するか、または XPath 式を使用して、メッセージボックスのデフォルトのテキストを上書きするテキストを入力します。

- *ファイルの保存時に自動的にサブフォルダーを作成する*: もし、ファイルパス内の中間フォルダーが不足している場合、ファイルが保存される際に作成されます。オプションは保存時に関係し、アクションがファイルのロードアクションの場合使用できません。

  メモ: iOS デバイスでは、ユーザーがデバイス上のファイルを選択できる唯一の方法は iCloud から、または、iCloud へファイルをインポートまたはエクスポートすることにより行うことが出来ます。バックアップフォルダー、または、バックアップされないフォルダーをユーザーは参照することはできません。

エラーの処理

エラー時: オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション のキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクション のセクションを参照してください。

HTTP/ FTP をロード/保存

アクションを以下に設定することができます: (i) HTTP/FTP ファイルからデータをロード、または(ii) HTTP/FTP を介してファイルにデータを保存します。ロードアクションまたは保存アクションを実行するかは適切ならラジオボタンを選択して指定します (*下のスクリーンショット参照*)。

HTTP/FTP からロード

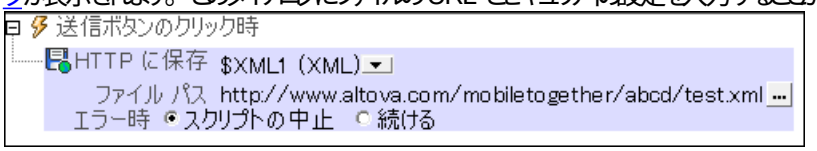
各 HTTP/FTP からロード アクションに対して、使用可能なページソースから1つのデータソースを選択して、データをロードするHTTP/FTP ソースを指定することができます。イベントがトリガーされると、HTTP/FTP ソースからのデータが、ページデータソースにロードされます。複数の データソースのためにデータをロードするには、複数のHTTP/FTP からロード アクションを追加します。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション のキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

HTTP/FTP へ保存

アクションのコンボボックス内で選択されたページソースをXML 、ターゲットHTTP のHTML ファイル、またはアクションの定義の設定 FTP ロケーションに保存します (*下のスクリーンショット参照*)。HTTP/FTP ロケーションのアクセスの詳細を入力するには、 ... をクリックします。これによりHTTP/FTP ソースを選択するための[Web アクセス設定の編集] ダイアログが表示されます。このダイアログにファイルのURL とセキュリティの設定を入力することができます。



複数のページソースからデータを保存、複数の保存先にデータを保存、複数のSaveToHTTP/FTP アクションを追加することができます。他のSaveToHTTP/FTP アクションを追加するには、HTTP/FTP にロード/保存アクションをイベントタブにドラッグし、HTTP/FTP に保存アクションのラジオボタンに設定します。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションの キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

文字列をロード/保存

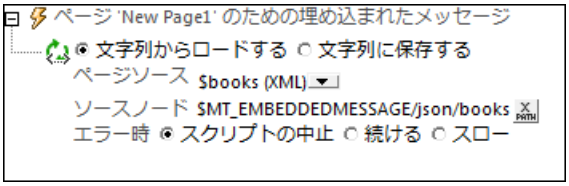
2つのアクション**文字列にロード**と**文字列に保存**はそれぞれ以下を行います:

- JSON または XML データを文字列からJSON/XML ページソースにロードします。
- JSON/XML ページソースを文字列にシリアル化しXPath 式により指定された場所に、シリアル化された文字列を保存します。

シリアル化されたXML データがWeb ページから受信することができ、**$MT_EMBEDDEDMESSAGE** JSON ページソース内で保存される場合、埋め込まれたWeb ページソリューションでこれらの機能はとても役にたちます。文字列からロードするアクションは、XML 文字列を取り、XML ページソースを生成することができます。逆に、文字列に保存アクションを使用して、XML ページソースをシリアル化しページソースノードに保存することができます。

文字列にロード

文字列にロードアクションは、アクションのXPath 式によりターゲットとされた文字列を解析し(*下のスクリーンショットを参照*)、選択されたページソースの構造とデータを生成します(*スクリーンショット参照*)。

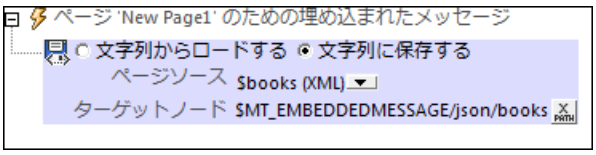


選択されたページソースノードの型(JSON または XML) は文字列のシリアル化に対応する必要があります。ですから、ターゲットとされた文字列がJSON 文字列の場合、選択されたJSON ページソース内でJSON ドキュメントが(ページソースノードの**`json`** 要素内で)生成されます。 シリアル化されたXML 文字列がXML ページソースにロードされると、シリアル化された文字列内のルート要素は、XML ページソースのルート要素として作成されます。

- *シリアル化された文字列について* 上のスクリーンショット内で、XML 文字列は、JSON ページソースノード **`$MT_EMBEDDEDMESSAGE/json/books`** 内に存在します。これは、選択されたページソース内にロードされる文字列です。これが正確に作動するためには、ページソースノードはXML ページソースである必要があります(XPath 式は文字列をノードから取る必要があります。文字列をXPath 式に直接入力することもできます)。
- *JSON/XML ページソースについて* ページソースノードはアクション内で選択される前に作成される必要があります。文字列からロードされたJSON/XML ドキュメントは、ドキュメントノード内でページソース全体として作成されます。ランタイムで、ロードされたドキュメントの構造がページソースノードの構造(デザイン内で定義されたとおり)に一致しない場合、ソリューションは正確に実行されます。これは、デザインが期待されるページソース構造のノード名と作動し、ページソースノードのノードは実際には異なる名前で作成されているからです。

文字列に保存

文字列に保存アクションはページソースオプション内で名前の付けられたページソースノードをシリアル化し、シリアル化された文字列をXPath 式により指定された場所に保存します(*下のスクリーンショット*)。

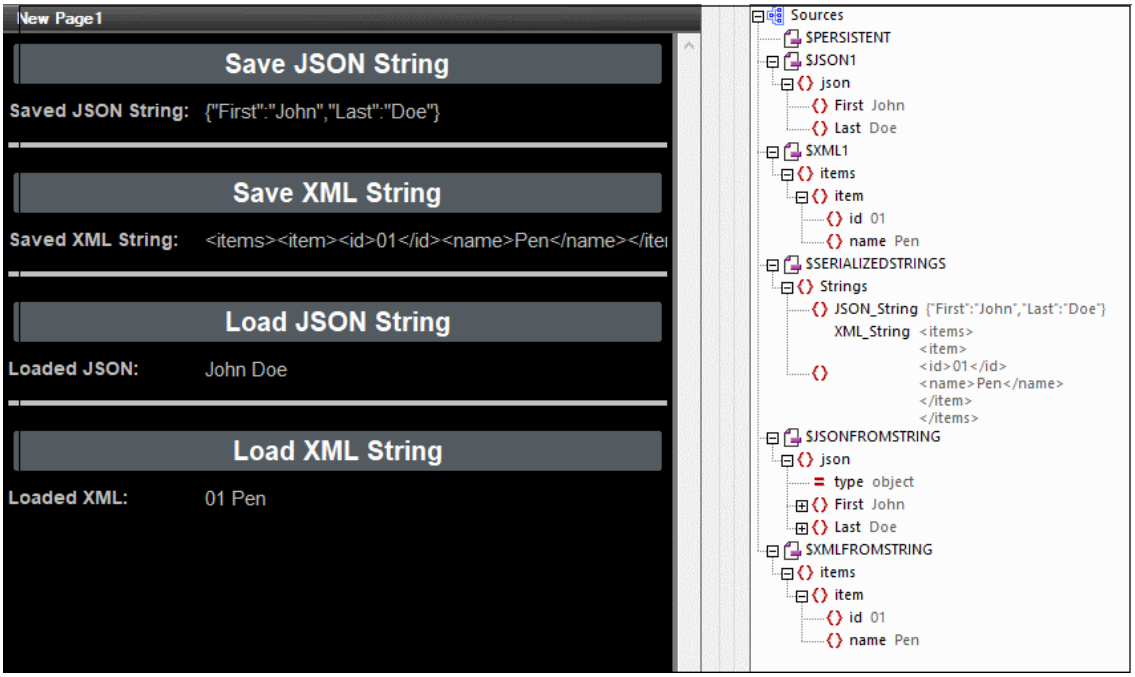


以下の点に注意してください:

- コンボボックスオプションとして選択するために、ページソースノードとその構造は、デザイン時とランタイムの両方でデザイン内に存在する必要があります。
- 型に関わらず、ページソース全体のコンテンツが最初の文字から最後の文字まで文字列にシリアル化されます。
- シリアル化により発生する文字列はXPath 式内で指定されたノードに保存されます。

サンプルファイル

LoadSaveString.mtd という名前のサンプルファイルでは、これらの2つのアクションの使用方法が紹介されています。このファイルはマイドキュメントフォルダー内にあります: `Altova\MobileTogetherDesigner4\MobileTogetherDesignerExamples\Tutorials`. このサンプルには2つのページソース(**$JSON1** と **$XML1**) が含まれています。それぞれには、固定されたJSON とXML データが含まれています(*スクリーンショットを参照*)。文字列に保存アクションを使用することにより、各ページソースを、文字列としてシリアル化することができ、ノードに保存することができます。文字列にコードアクションを使用して、新しいページソース(**$JSONFROMSTRING** と **$XMLFROMSTRING**)にシリアル化されたばかりの文字列をコードするためにこのアクションをリバースすることができます。

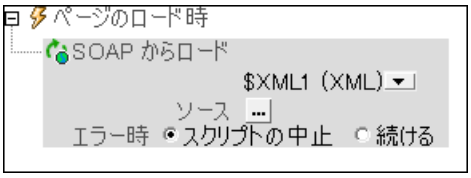


- デザインファイルはJSON ページソース(**$JSON1**) とXML ページソース(**$XML1**) を含み、ファイル内でファイル内で定義されている構造とデータを含みます (*上のシミュレーションのスクリーンショット参照*)。両方のデータ構造は基本的なものです。
- デザインには、各ページソースをシリアル化された文字列として保存するボタン(**Save JSON/XML String**) が含まれており、それぞれは、**$SERIALIZEDSTRINGS** という名前の他のページソースの異なるノード内に存在します。これらのボタンは文字列に保存アクションを使用します。
- 生成されたシリアル化は新規のページソース**$JSONFROMSTRING** と**$XMLFROMSTRING** を作成するために使用されます。これは3番目と4番目のボタン(**「JSON/XML 文字列の保存」**) の文字列にコードアクションをトリガーすることにより行うことができます。

このサンプルは、2つのアクション(文字列に保存 と文字列にコード) を使用して、1つのページソースから他のページソースにシリアル化された文字列を使用して往復します。ページソースの基点とターゲットは同じ構造とデータを保持します。

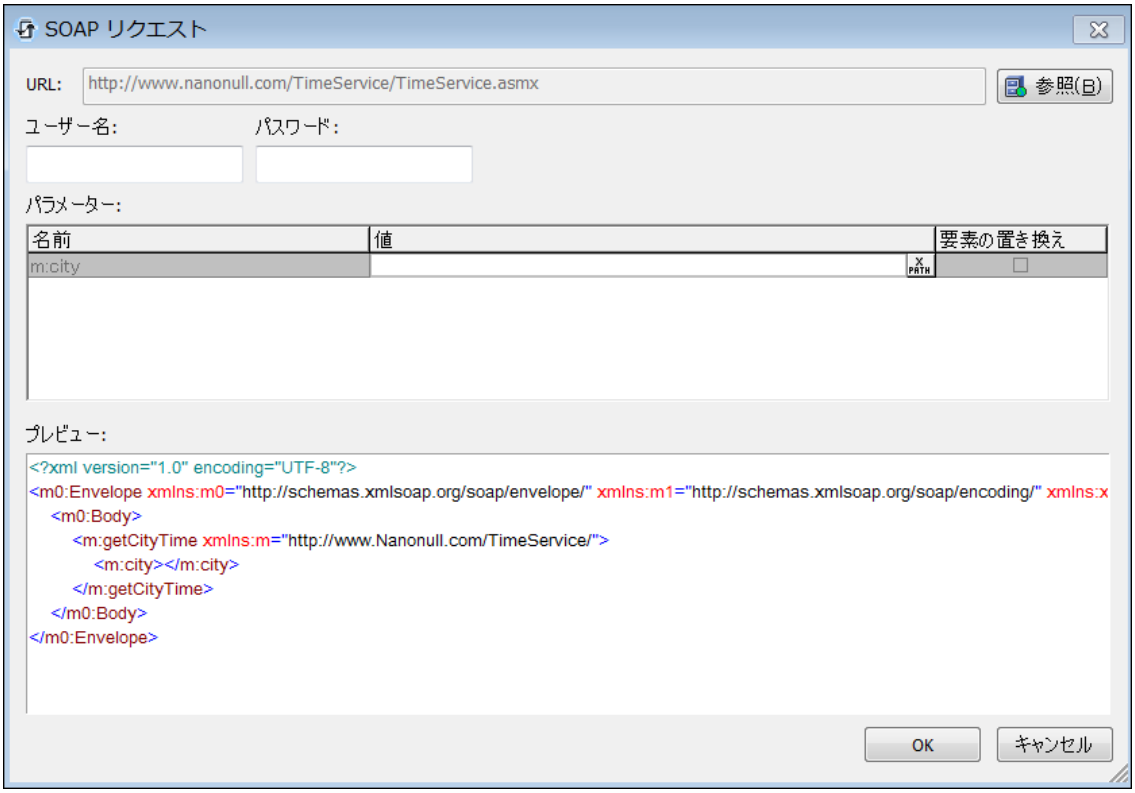
SOAP からロード

各 SOAP からロード アクションのために、WSDL ファイルから生成されたSOAP リクエストからデータをロードすることができます。WSDL ファイルを選択するためには、ソースフィールドの**[追加設定]** ボタンをクリックします (*下のスクリーンショット参照*)。



WSDL ファイルを選択して、SOAP オペレーションを選択します。SOAP リクエストは、自動的にWSDL ファイルから生成され、SOAP リクエスト ダイアログに表示されます。SOAP リクエスト ダイアログ内の **[OK]** をクリックすると、使用することのできるSOAP リクエスト として保存されます。アクションは、ランタイムでSOAP リクエストが送信されるWeb サービスのURL を表示します。(*下のスクリーンショット内のソースフィールドを参照してください*)。

SOAP リクエストを定義後に変更する場合は、ファイル パスフィールドの**[追加ダイアログ]** ボタンをクリックします (*上のスクリーンショット参照*)。これによりSOAP リクエスト ダイアログ (*下のスクリーンショット*) が表示されます。*URL* フィールドの**[参照]** ボタンをクリックして、WSDL ファイルを選択して、SOAP リクエストの定義プロセスを再起動します。



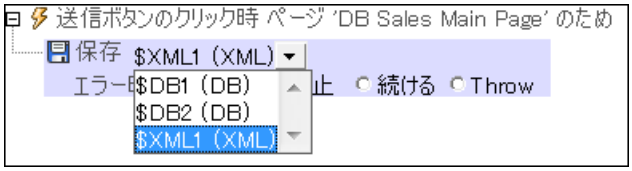
イベントがトリガーされる時に複数のデータソースからデータをロードする場合、複数のSOAP からロード アクションを追加します。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

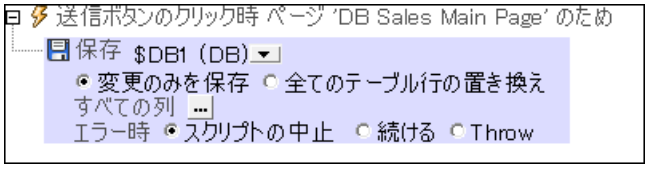
- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションの キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

保存

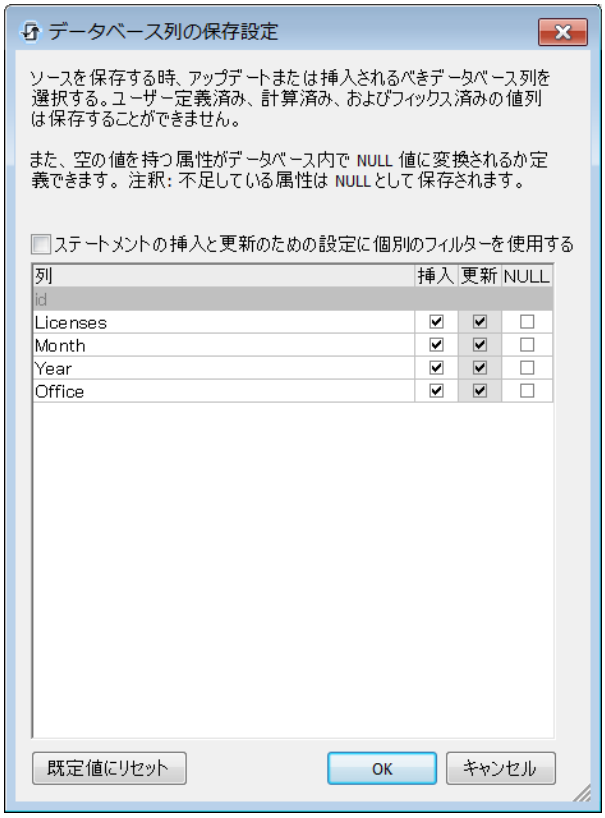
アクションのコンボボックスで選択されたデータソースツリーのデータを、データソースのデフォルトファイルに保存します。データソースは編集可能な XML ファイル、または DB である必要があります。複数のデータソースのデータを保存するには、複数の [保存アクション] を追加します。JSON と読み込まれるページソースは、(データが GUI 内で XML ツリーとして保存されている場合でも、XML としてではなく) JSON として保存されます。ページソースオプションも参照してください。



保存されるデータソースが DB の場合、デフォルトで、すべての編集可能な列が更新されるように選択されます (下のスクリーンショット参照)。変更されたデータのみ (主キーが必要です) または、すべてのテーブル行 (主キーは必要ありません) を保存することを選択することができます。



特定の列を更新する場合は、[...] をクリックします。これにより、データベースの列設定ダイアログを表示します (下のスクリーンショット)。



ダイアログは、DB データソースのための列を表示します。その列が更新され、または、挿入された値を取ることができるかを

指定することができます。(更新は、既存の行要素内の変更されたデータを指します。挿入された値は、新規に追加された行要素を指します)。デフォルトでは、各列の挿入と更新のオプションはペアとして選択されます。ですが、異なるオプションを列の挿入と更新オプションのために指定する場合、挿入慮更新ステートメントのために個別のフィルター設定を使用するチェックボックスをチェックしてください。DB のカラム内の*NULL* チェックボックスをチェックすることで、空の値を持つ属性は、内で NULL 値に変換されることができます。不足する属性は常に NULL として保存できることに注意してください。

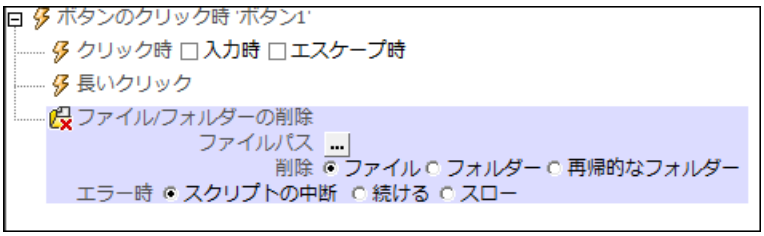
(ユーザーにより定義されているため、固定されているため、または計算されているため) 更新することのできない列は、挿入、更新またはオプションチェックボックスを使用することができません。 上のスクリーンショットでは、固定値を持つため、ID 列は、更新されることができません。更新しない列の選択を解除します。全ての列を更新するために、保存設定をリセットする場合は、**[デフォルトにリセット]** をクリックします。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

ファイル /フォルダーの削除

アクションが実行されると、クライアントまたはサーバー上の選択されたファイル/フォルダーを削除します。1つのアクションに付き 1つのファイル/フォルダーのみを削除することしかできません。複数のファイルまたはフォルダーを削除する場合は、このアクションを繰り返し使用してください。



- *削除の対象:* 削除の対象を選択します: (i) ファイル、(ii) 空のフォルダー(空でない場合はフォルダーは削除されません)、または(iii) 再帰的なフォルダー(サブフォルダーを含むコンテンツ全体が削除されます)。
- *ファイル/フォルダー パス:* 削除されるファイルまたはフォルダーのパス。

クライアントとサーバー上のファイルの場所に関する詳細については、ツリーデータとイメージのロード/保存を参照してください。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

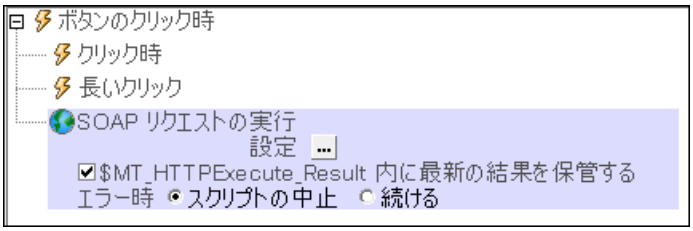
- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション のキャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

リセット

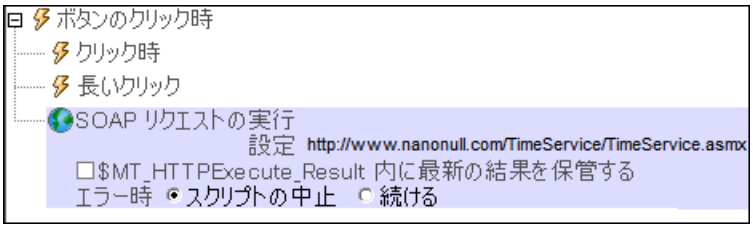
アクションのコンボボックスで定義された、**$PERSISTENT**、**$MT_GEOLOCATION**、**$MT_FILEINFO**、と**$MT_NFC** データソースツリーを含む、[ページ ソース] ペイン] でデフォルトの値に対して選択されたデータソースをリセットします。

SOAP リクエストの実行

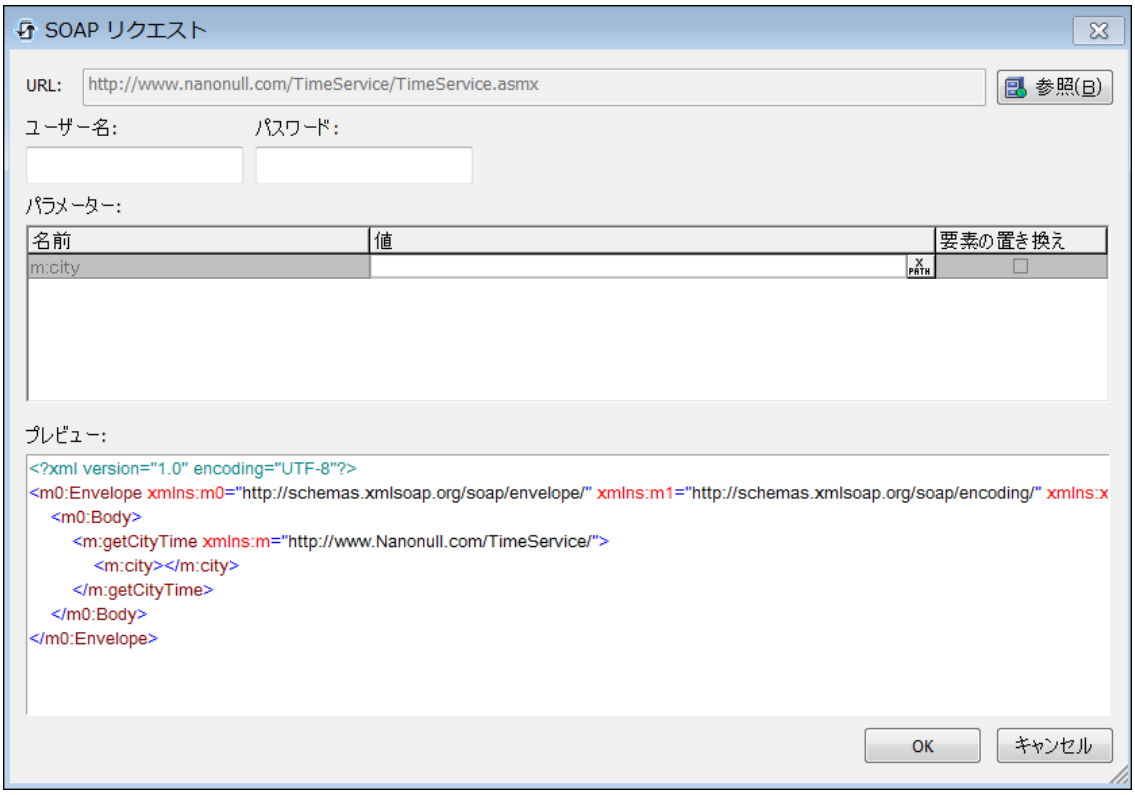
WSDL ファイルから生成されたSOAP リクエストを実行します。WSDL ファイルを選択するには、設定フィールドの**[追加設定]** ボタンをクリックします (*下のスクリーンショット参照*)。



WSDL ファイルを選択して、SOAP オペレーションを選択します。SOAP リクエストは自動的にWSDL ファイルから生成され、SOAP リクエスト ダイアログに表示されます。SOAP リクエスト ダイアログ内の**[OK]** をクリックして、実行するSOAP リクエストとして保存します。ランタイムに送信されるSOAP リクエストのWeb サービスのをURL アクションは表示します (*下のスクリーンショット内の設定フィールドを参照*)。SOAP リクエストへの応答を保管する場合、最新の結果の保管オプションのチェックボックスをチェックします (*下のスクリーンショット参照*)。SOAP 応答は `$MT_HTTPExecute_Result` 変数内に保存されます。この変数を使用して、デザイン内の他の場所のSOAP 応答内のデータにアクセスすることができます。しかしながら、`$MT_HTTPExecute_Result` 変数は、REST リクエストの実行 アクションで使用されているため、変数はこれを使用するすべてのアクションにより生成された最後の結果を含むことに注意してください。



定義後にSOAP リクエストを変更する場合、設定フィールドの**[追加ダイアログ]** ボタンをクリックします(*上のスクリーンショット参照*)。これによりSOAP リクエスト ダイアログが表示されます (*下のスクリーンショット*)。



WSDL ファイルを選択するために*URL* フィールドの**[参照]** ボタンをクリックして、実行するSOAP リクエストの定義プロセスを再起動します。

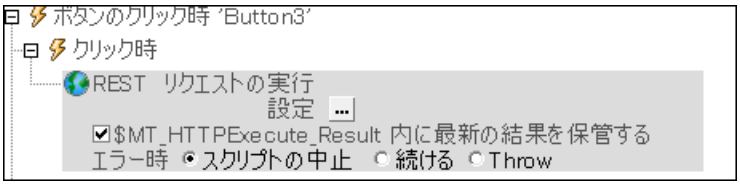
エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

チュートリアル SOAP リクエスト はSOAP リクエストアクションの使用方法を説明しています。

REST リクエストの実行

設定フィールドの**[追加設定]** ボタンをクリックして、(*下のスクリーンショット参照*) [RESTful API リクエスト] ダイアログ内で定義されたREST リクエストを実行します。



REST リクエストを定義した後、アクションの設定フィールド内にリクエストのURL が表示されます。リクエストはランタイムで実行されます。`$MT_HTTPExecute_Result` 変数内でリクエストの結果を保管する場合は、最新の結果を保管オプションをチェックします(*上のスクリーンショット参照*)。`$MT_HTTPExecute_Result` 変数を使用して、デザイン内の結果にアクセスすることができます。しかし、この変数は、SOAP リクエストの実行 アクションで使用することができることに注意してください。ですから、この変数は、この変数を使用するすべてのアクションにより生成された結果を含みます。

定義後にREST リクエストを変更する場合は、設定フィールド上の**[追加]ダイアログ** ボタンをクリックします(*上のスクリーンショット参照*)。これにより新しいリクエストを定義することのできる[RESTful API リクエスト] ダイアログが表示されます。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション の キャッチ 部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

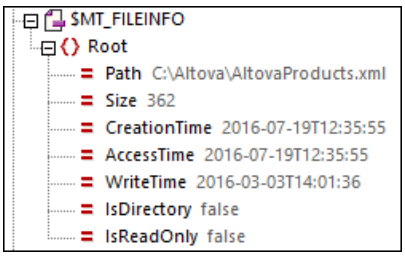
ファイル情報の取得

ファイルの情報を取得するアクションは、指定されたファイルの情報を**$MT_FILEINFO** と呼ばれるデータソースツリーに追加します (*下のスクリーンショットを参照してください*)。**$MT_FILEINFO** という名前のデータソースは1つしか存在しません。しかしながら、ファイルの情報を取得するアクションは、デザイン内の複数の場所で指定されることができます。これらの(アクションの) インスタンスの1つがトリガーされると、**$MT_FILEINFO** ツリーが特定のインスタンスアクション内で指定されたファイルのファイル情報と共に更新されます。

メモ **$MT_FILEINFO** ページソースモデザイン内に作成されます。

$MT_FILEINFO ツリーの構造

$MT_FILEINFO ツリーの構造は下のスクリーンショットで表示されています。ルート要素には、トリガーされるアクション内で指定されているファイルのファイル情報がフィルされている属性が存在します。属性の説明は以下のとおりです。

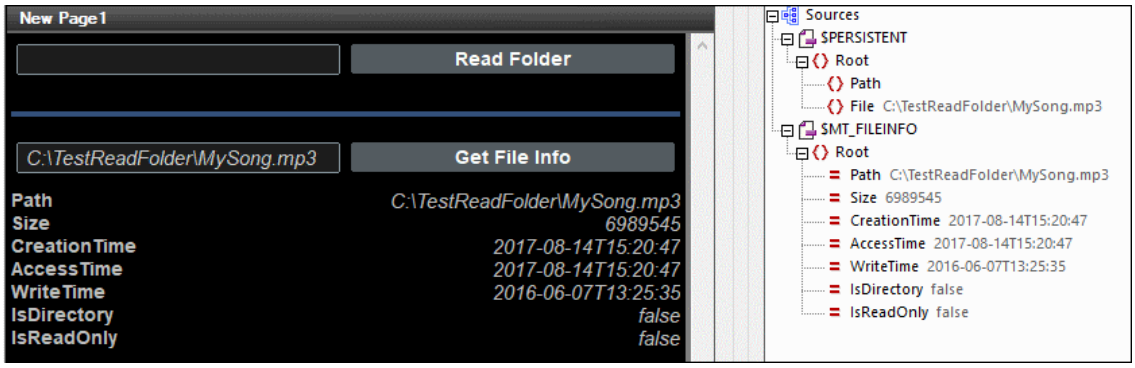


- *Path:* レポートされるファイルのフルパスです。
- *Size:* バイト数で表示されたファイルのサイズ。
- *CreationTime:* ファイルが現在の場所で作成された日時。ファイルが新しい場所にコピーされた場合、ファイルがコピーされた日時が作成の日時として使用されます。このような場合、作成日時はファイルが最後に書き込まれた日時よりも後になる場合があります。
- *AccessTime:* ファイルが最後にアクセスされた日時。
- *WriteTime:* ファイルが最後に書き込まれた日時。
- *IsDirectory:* `true` または `false` の値を取ります。
- *IsReadOnly:* `true` または `false` の値を取ります。

メモ (上で説明されているとおり) ファイルの情報を取得アクションは **$MT_FILEINFO/Root** の**属性** にパスされます。**$MT_FILEINFO** ページソースは、しかしながら、構造内に次の他のノードを有します: **Root** の繰り返し**File** 子。**File** 要素は、他のアクション*フォルダーの読み取り*により取得されたデータを受け取ります。

サンプル

ReadFolderGetFileInfo.mtd という名前のサンプルはファイルの情報の取得アクションの使用方法を表示しています (*シミュレーションのスクリーンショットを参照してください*)。このファイルは次の場所にあります: *マイ ドキュメント フォルダー* `Altova\MobileTogetherDesigner4\MobileTogetherDesignerExamples\Tutorials\Actions`。



このサンプルのしくみは以下のとおりです:

- エンドユーザーは、**「ファイルの情報の取得」**ボタンの左側の編集フィールド内にファイルの名前を入力します。
- **`$PERSISTENT`** ツリーの**`Root/File`** ノードにファイル名が書き込まれます (*スクリーンショットを参照*)。
- **「ファイルの情報の取得」**ボタンには、**`$PERSISTENT/Root/File`** 内に保管されたファイルをターゲットとしたボタンのクリック時 イベントのために設定されたファイル (またはフォルダー) の情報を取得アクションがあります。
- **「ファイルの情報の取得」**をクリックすると、ターゲットとされるファイルの情報が読み取られ、**`$MT_FILEINFO/Root`** の属性の値として**`$MT_FILEINFO`** ページソースにパスされます。
- これらの属性の値は静的テーブルのセル内に表示されます。

フォルダーの読み取り

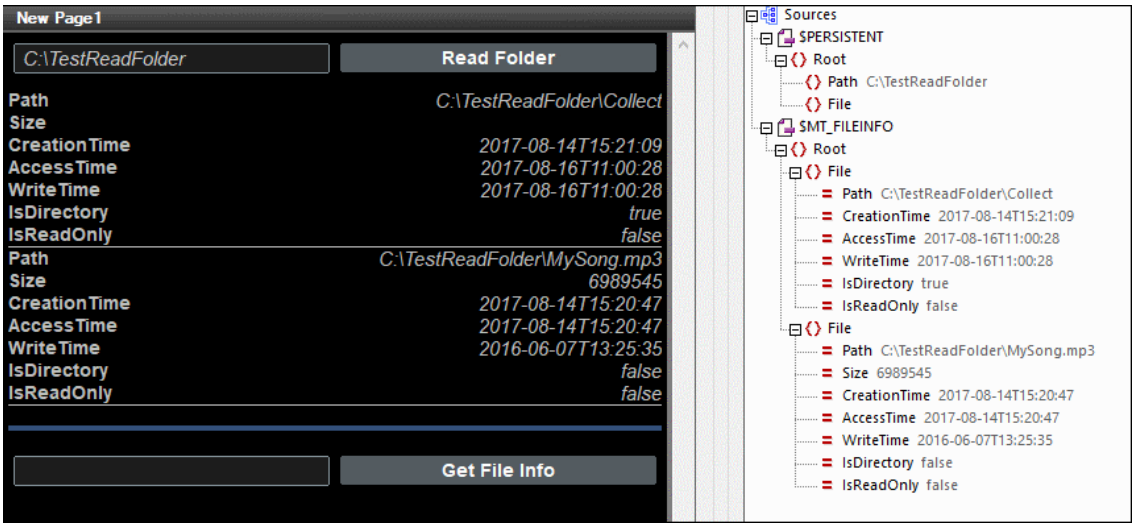
フォルダーの読み取りアクション (*下のスクリーンショットを参照*) がデザインに追加されると、繰り返し **File** 要素のセットとして設定されている構造である **$MT_FILEINFO** ページソースがデザイン内で作成されます。フォルダーの読み取りアクション内では、読み取るフォルダーを指定します (*サンプルとスクリーンショットを参照*)。ランタイムでは、指定されたフォルダー (ファイルとサブフォルダー) のコンテンツが読み取られ、ターゲットフォルダーの各アイテムの詳細が **$MT_FILEINFO** ページソースの対応する **File** 要素に置かれます。

次の点について注意してください:

- フォルダーは XPath 式内でフォルダーを文字列、または、ページソースノードをコケートする式として指定されることができます。指定されたフォルダー内のファイルのみが読み取られます。サブフォルダー内のファイルは**読み取られません**。
- ファイルパターン設定は、フォルダーのファイルをフィルターするためにワイルドカードを使用します。例: **'*.mp3'** は フォルダー内の全ての **.mp3** ファイルの詳細を読み取ります。**'My*.*'** は、**My** から始まる名前を持ち、サフィックスを取る全てのファイルの詳細を読み取ります。**'*'** または **''** はフォルダー全体を読み取ります。フォルダー自身に関する詳細に関しては、ファイル情報の取得 アクション内のフォルダーの名前を送信します。
- 各ページには1つの **$MT_FILEINFO** ページソースが含まれています。ページ上に複数のフォルダーの読み取りアクションが存在する場合、ランタイムでは、 **$MT_FILEINFO** ページソースには、最後に読み取られたアクションにより読み取られたフォルダーに関する情報が含まれます。
- フォルダーの読み取りアクションにより読み取られたデータは、**$MT_FILEINFO/Root** の **File** 子要素にパスされます。**Root** の属性は他のアクション? ファイル情報の取得 アクションにより取得された情報を受け取ります。
- **$MT_FILEINFO** ページソースは、ファイル情報の取得 アクションが追加された際にも作成されることに注意してください。

サンプル

ReadFolderGetFileInfo.mtd という名前のサンプルはフォルダーの読み取りアクションアクションの使用方法を表示します (*シミュレーションのスクリーンショットを参照してください*)。このファイルは次の場所にあります: *マイ ドキュメント フォルダー* `Altova\MobileTogetherDesigner4\MobileTogetherDesignerExamples\Tutorials\actions`.

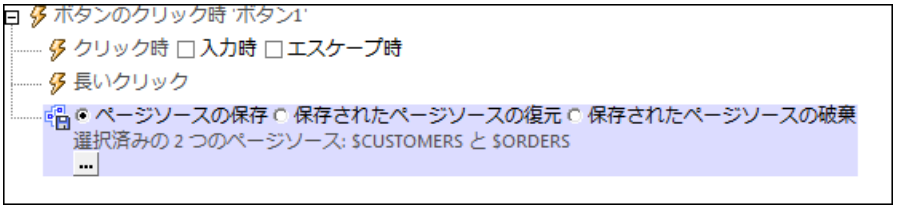


このサンプルのしくみは以下のとおりです:

- エンドユーザーは、**「フォルダーの読み取り」**ボタンの左側の編集フィールド内にファイルの名前を入力します。
- **`$PERSISTENT`** ツリーの **`Path`** ノードにファイル名が書き込まれます (*スクリーンショットを参照*)。
- **「フォルダーの読み取り」**ボタンには、**`$PERSISTENT/Root/Path`** 内に保管されたファイルをターゲットとしたボタンのクリック時 イベントのために設定されたファイル (またはフォルダー) の情報を取得アクションがあります。
- **「フォルダーの読み取り」**をクリックすると、フォルダーのアイテムに関する情報が読み取られ、**`$MT_FILEINFO`** の **`File`** 要素にパスされます。
- 各 **`File`** 要素は、テーブルの繰り返し行としてデザイン内に表示されます。

ページソースの保存 /復元

各 ページソース アクションの保存 /復元は1 つ選択することのできる3 つのサブアクションにより構成されています（*下のスクリーンショット参照*）。



3 つのサブアクションは以下のとおりです:

- *ページソースの保存*: 1つ、または、複数のページソースを保存します。保存するページソースを選択してください。アクションの下にある**追加 ダイアログ** ボタンをクリックして1 つまたは複数のページソースを選択します(*上のスクリーンショットを参照*)。
- *保存されたページソースの復元*: （*ページソースの保存* により）ページソースの内部コピーが保存されると、ページソースを更に変更することができます。*保存されたページソースの復元* アクションは、ページソースを内部で保存されたコピーの最近の状態に復元します。
- *保存されたページソースの破棄*: （*ページソースの保存により*）ページソースの内部コピーが保存され、そのページソースが更に変更されると、*保存されたページソースの破棄* アクションは、最後に保存された内部コピーを破棄します。

メモ ページソースの保存 /復元 アクションは、ファイルに保存されるページソース**ではなく**、内部で保存されているページソースのみに適用されます。

使用方法
ページソースの保存 /復元 アクションにより、ページソースを仮保存、受け入れ、または、1 つまたは複数の条件が満たされることが条件に更に変更することができます。 例えば、サブページに移動する前に、ページソースを内部に保存することができます。サブページ上でユーザーがデータを編集すると、ボタンを押して、確認、または、変更をキャンセルします。ボタンは定義され、保存されたページソースを破棄、または、復元することができます。

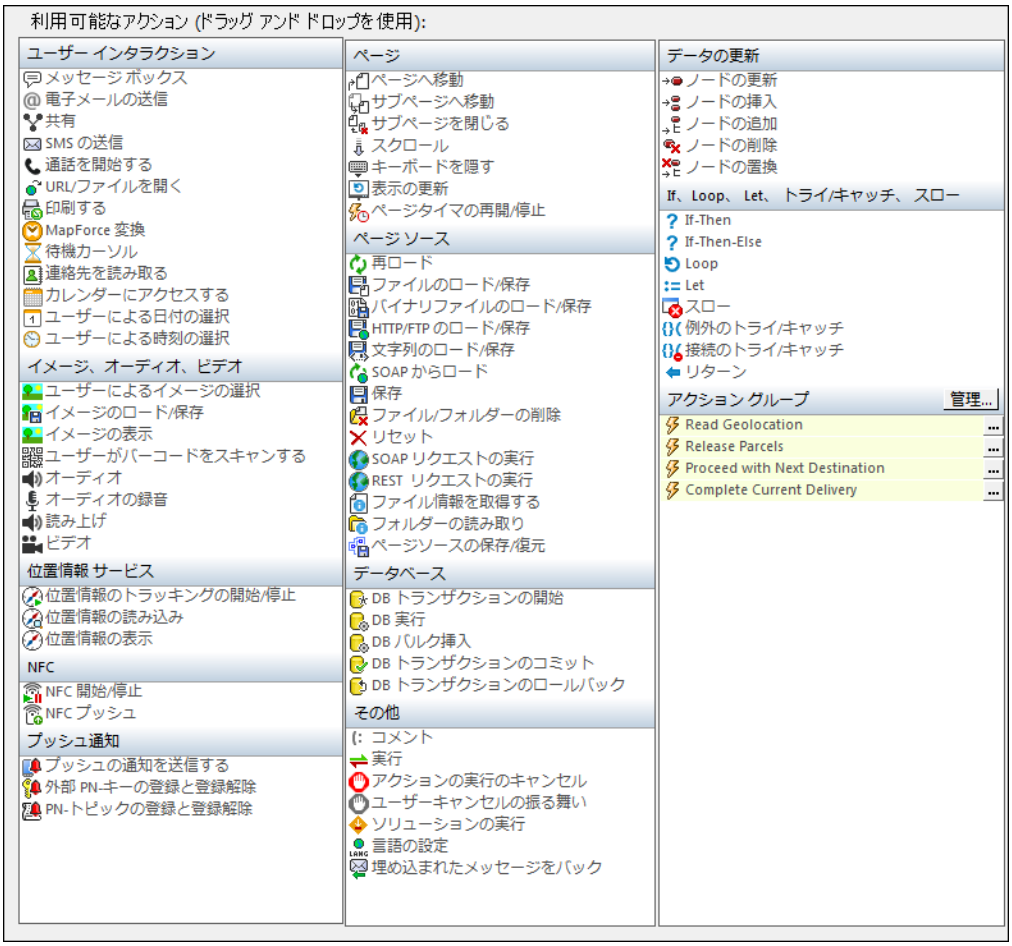
10.8 データベース

アクションダイアログのデータベースグループで次のアクションを使用することができます(下のスクリーンショット):

- DB トランザクションの開始
- DB 実行
- DB バルク挿入
- DB コミット トランザクション
- DB ロールバック トランザクション

メモ これらのアクションは、DB データソース内のデータと相互に作用するために使用されますが、データを表示するためには適していません。DB データソースからデータを表示する場合は、ページデータソースにリンクされているコントロールを(デザインに) 挿入してください。詳細に関しては、コントロールとデータソースに関するセクションを参照してください。チュートリアルは、ページソースデータを表示する実践的な方法について説明しています。

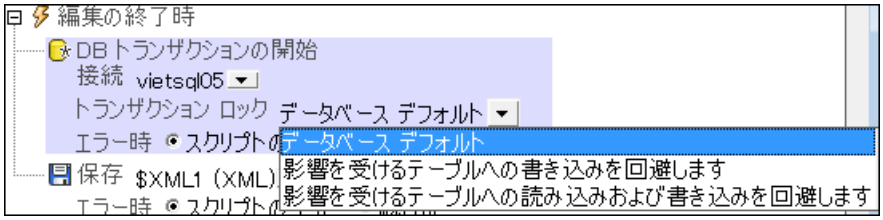
いち



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロール アクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

DB トランザクションの開始

DB トランザクションの開始アクションによりイベントがトリガーされると、*接続*コンボ ボックスにより選択されたデータソースと共にトランザクションが開始されます。このコンボ ボックスはプロジェクトのすべてのデータソースをリストし、DB トランザクションの開始アクション内で具体的に使用されるための追加データベース接続のオプションを与えます。



*トランザクション ロック*オプションは、書き込みアクション中にデータが破損しないようにレベルを指定するために使用することができます。以下のオプションを使用することができます:

- *データベース デフォルト:* DB、サーバー、クライアント上のDB-に関連したデフォルトの設定を収集します。
- *影響を受けるテーブルへの書き込みを回避:* 他の接続を介して、DB に書き込み中の場合、DB に書き込むことはできません。書き込みトランザクションは、他の書き込みトランザクションが完了するまで、遅延され、その他の場合、エラーメッセージが表示されます。
- *影響を受けるテーブルへの読み込みおよび書き込みを回避:* 他の接続を介して、DB に書き込みまたは読み込み中の場合、トランザクションは、他の トランザクションが完了するまで、遅延または、または エラーメッセージが表示されます。

エラーの処理

エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断:* エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続:* アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー:* エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

DB トランザクションについて

トランザクションが必要な各 DB アクセスには、自動的にトランザクションが作成され、後に自動的に閉じられます。これは一部のセットアップでは理想的です。例えば、同時に自動的に更新したい2 つのDB ページソースが存在する場合、両方のテーブルが保存されると、トランザクションがコミットされます。その他の場合はロールバックされます。このようなシチュエーションに対応するために、接続ベースのトランザクションを作成することができます。

トランザクションの開始 が始まると、同じDB 接続に属するすべてのDB オペレーションはこのトランザクションを使用します。

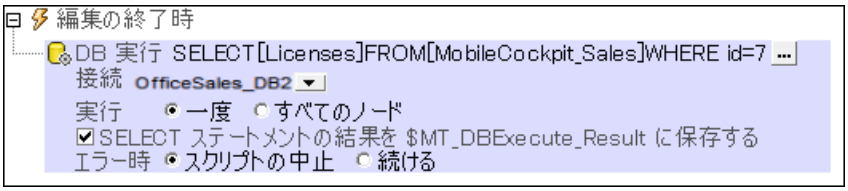
トランザクションのコミット により、トランザクションの外界で変更が可視化されます。変更はロールバックされることができます。この場合、ページソースに保存した場合でも、ロールバックの後で変更は可視化されません!閉じられていない(コ

ミットされたまたはロールバックされた) トランザクションは、アクションツリーのが最後に到達すると自動的にロールバックされます！この効果についてメッセージウィンドウに警告が表示されます。

上記の振る舞いはトランザクションアクションについて参照していますが、振る舞いはトランザクションと同じ接続を使用するDB オペレーションにも適用されます。

DB 実行

イベントがトリガーされると、DB 実行アクションは、*接続*コンボボックスで選択されたデータソースのアクションのSQL ステートメントを実行します。このコンボボックスはプロジェクトのすべてのデータソースをリストし、具体的にDB 実行アクションと使用するための追加データベース接続をセットアップするオプションを与えます。`$MT_DBExecute_Result` に結果を保存するチェックボックスが選択されている場合、結果は `$MT_DBExecute_Result` 変数に保管されます。この変数は実行アクションの結果を提供するためにページの他の箇所のXPath 式に使用することができます。



エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

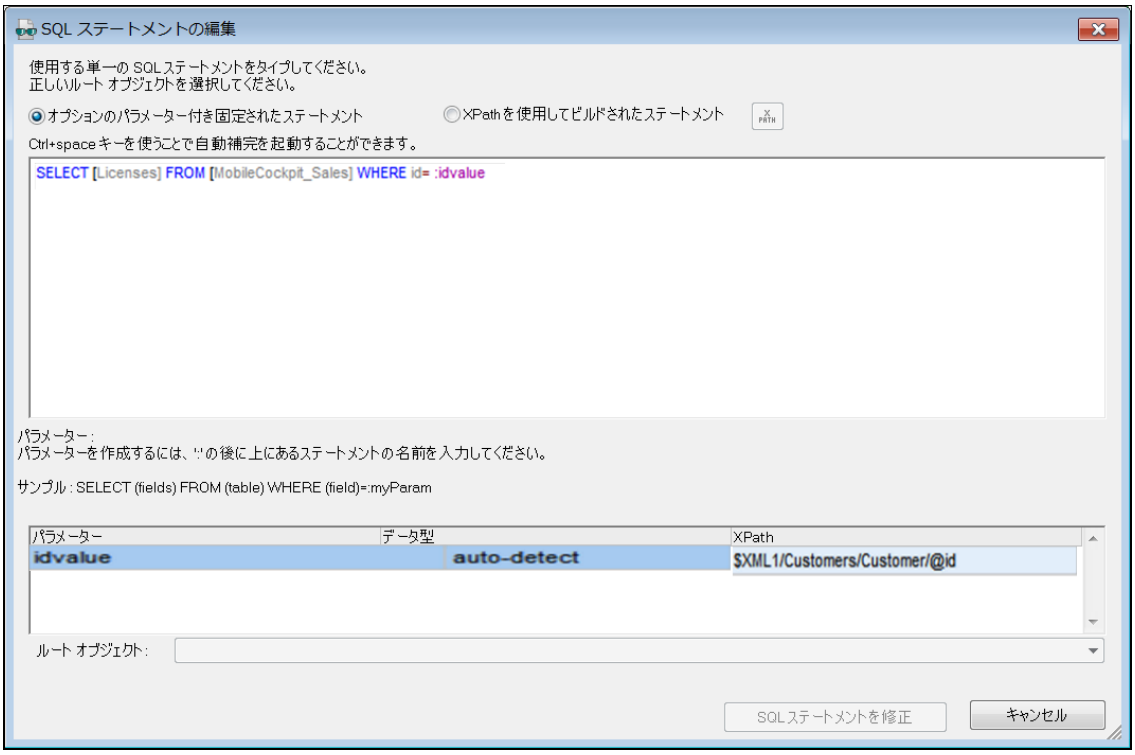
- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

このアクションの使用についての詳細は、ページデザイン| データベース| DB 実行アクションのセクションを参照してください。

メモ これらのアクションは、DB データソース内のデータを相互に作用するために使用されますが、データを表示するためには最適なメカニズムではありません。DB データソースからデータを表示する場合は、ページデータソースにリンクされているコントロールを(デザインに) 挿入してください。詳細に関しては、コントロールとデータソースに関するセクションを参照してください。チュートリアルは、ページソースデータを表示する実践的な方法について説明しています。

SQL ステートメント

SQL ステートメントを入力または編集するには、**[追加ダイアログ]** ボタンをクリックします。これによりSQL ステートメントの編集ダイアログが表示されます。(*下のスクリーンショット*)。ダイアログ下のルート オブジェクトは自動的に選択され、接続コンボボックスの選択をベースにしています。*ルート オブジェクト* フィールドは編集することができません。継続する前に、ルート オブジェクトが正しいか確認をしてください。



オプションのパラメーター付き固定されたステートメント
SQL ステートメントを入力するには、[*オプションのパラメーター付き固定されたステートメント*] を選択し、SQL ステートメントを入力します。ステートメント内でのパラメーターの使用は柔軟性を与えます。例えば、上のスクリーンショットで、WHERE 句に固定値を入力する代わりに、 Maker という名のパラメータが、XML データソースのノードの値を提供するために使用されています。下の一番目のラインでは固定値が使用されています。2番目のラインではパラメーター Maker が使用されています。

```
WHERE Manufacturer= 'BMW'
WHERE Manufacturer= Maker
```

パラメーターを使用するには、使用する箇所の SQL ステートメントのパラメーター名のプレフィックスとしてコロンを使用します (:)。コロンの後に文字を入力すると、パラメーター ペインにパラメーターのエントリが作成されます。次に、パラメーター ペインで、パラメータの値を与えるために XPath 式を入力します。入力できるパラメータの数に制限はありません。

メモ　SQL ステートメントでは、直接 DB をクエリするため、ソースデータベースからの列とテーブル名が使用されます。しかし、パラメーターの XPath 式では、デザインに関連する値内のこれらのツリーが保管されているため、ページソースツリー (Row、RowSet など) のノード名を使用する必要があります。

XPath 使用してビルドされたステートメント
SQL ステートメントをビルドするために XQuery を使用することができます。XPath *を使用してビルドされたステートメントを選択*し必要な SQL ステートメントを生成する XPath 式を入力します。この利点は SQL ステートメントの作成がより柔軟に行えることです。例えば、デザインツリーノードの包含、他の X XPath の構築、エンドユーザーのインプットを使用して計算および SQL ステートメントの部分を生成するなどがあげられます。

XPath 式を使用して SQL ステートメントをビルドするには、*XPath とビルドされたステートメント*を選択します。表示された XPath/XQuery 式の編集ダイアログ に XPath 式を入力して、**[OK]** をクリックします。

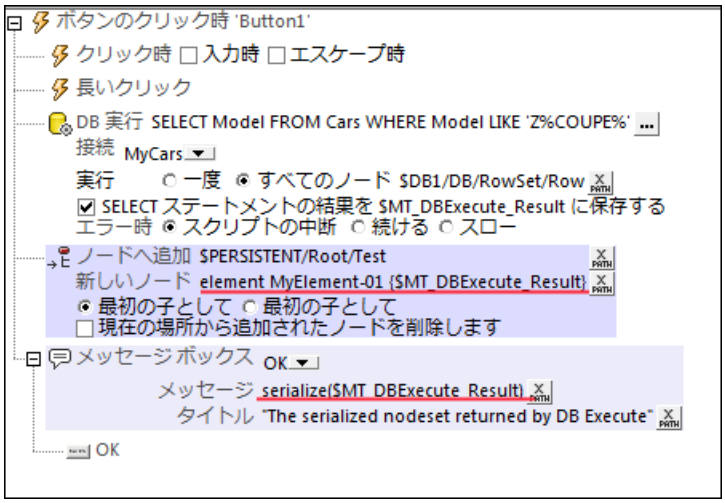
一度またはすべてのノードの実行

SQL ステートメントはデータソースに対して一度、または、カスタムに定義されたノードセットの全てのノードに対して実行されることができます。後のオプションを選択すると、ノードセットを生成するXPath 式を入力する必要があります。 さらに、変数 `$MT_TargetNode` を使用してノードセットの現在のノードの値をクエリすることができます。この変数は、例えば、SQL ステートメントのパラメーターの定義で使用することができます (*パラメーター付きSQL ステートメントを参照*)。

`$MT_DBExecute_Result` 変数

DB 実行アクションの(SQL ステートメント) から返されたノードセットまたは他の値は、MobileTogether Designer のビルドインされた変数 **`$MT_DBExecute_Result`** に保管されます。この変数はプロジェクトの最後のDB 実行アクションの結果を保管し、プロジェクトの他の箇所のXPath 式に使用することができます。

DB 実行アクションがノードセットを返す場合、要素を構成し、構造された要素の子としてノードセットを挿入することができます。 または、**`serialize()`** 関数を使用してノードセットを以下のように **`serialize($MT_DBExecute_Result)`** を使用してシリアル化することができます。2つのXPath 式は、下のスクリーンショット内で赤い色の下線が引かれています。



ノードセット内の特定の要素にアクセスするには、XPath 式を使用してアクセスすることができます。例えば、上のスクリーンショット内のDB 実行アクションを参照してください。**SELECT** ステートメントが次のノードセットを返すと仮定します:

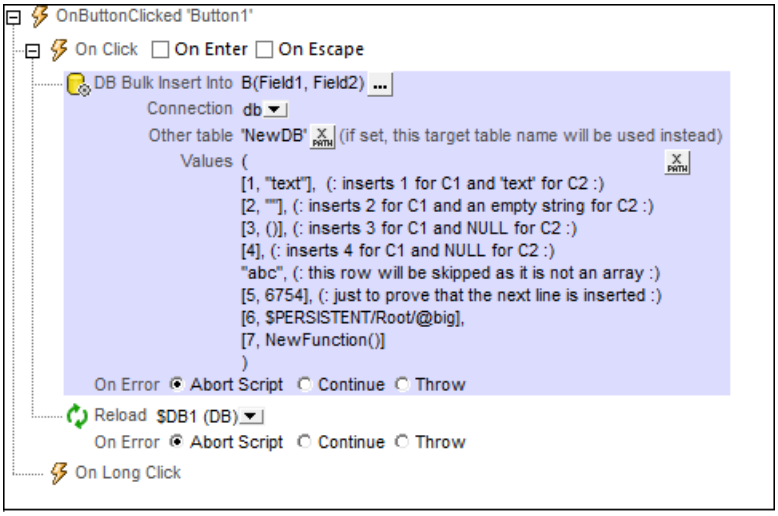
```
<DB>
  <RowSet>
    <Row Model="Z3 COUPE 2014"/>
    <Row Model="Z3 COUPE 2015"/>
    <Row Model="Z3 COUPE 2016"/>
    <Row Model="Z4 30 SICOUPE 2014"/>
```

```
    <Row Model="Z4 COUPE 2015"/>
  </RowSet>
</DB>
```

- 返されたノードセット内で最初の車のモデル名にアクセスします。XPath 式は以下のようになります: **$MT_DBExecute_Result/DB/RowSet/Row[1]/@Model**。
- モデル名内に年度 2016 が記入されている車の行にアクセスするためのXPath 式は以下のようになります: **$MT_DBExecute_Result/DB/RowSet/Row[@Model[contains(. , '2016')]]**。

DB バルク挿入

DB バルク挿入アクションは、*値フィールド*のXPath 式を使用して送信されたデータを新しい行として、*DB バルク挿入設定*で選択されたDB テーブルに挿入付加します (*下のスクリーンショット参照*)。



- *DB バルク挿入*: 挿入するDB テーブルを選択する際、 DB 接続メソッドを指定し、新しい行が挿入されるテーブルを選択します。新しい行は既存のテーブルの行に付加されます。選択されたテーブルは、 *DB バルク挿入* バルク挿入フィールドに列と共にリストされています。例えば、上のスクリーンショットでは、選択されたテーブルの名前は**B** で、**Field 1** と**Field 2** という名前の2つのバルク挿入フィールドがあります。
- *他のテーブル*: *DB バルク挿入* で選択されている以外のテーブルをXPath 式を使用して指定することができます。新しい行もこのテーブルに挿入されます。このテーブルは既存である必要があり、*DB バルク挿入*で選択されたテーブル列と同じ名前を含む必要があります。これらがデフォルトの値またはnullable の場合、追加列を含むこともできます。 上のスクリーンショットでは、新しい行がテーブル**NewDB** に挿入されます。挿入に成功すると、**NewDB** テーブルには、*Field1* と*Field2* という名前を持つ、2つの列があります。データ型も一致する必要があります: 最初の列はデータ型の数値、2番目の列はデータ型の文字列です。提出された値が列のデータ型に一致しない場合、変換が試みられます。
- *値*: *値*バルク挿入フィールドのXPath 式は、各配列が行を表し、配列内の各値が列の値を表す配列のシーケンスを返す必要があります。 上のスクリーンショットでは、各配列は新しいラインに配置されています。異なる方法で配列がインスタンス化されていることに注意してください。

上のスクリーンショットでは、変更されたテーブルを含むDB ページソースを更新する再ロードアクションを使用しました。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ キャッチ アクションの変数内に保管された例外をスローします。 トライ キャッチ アクション のキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、 トライ キャッチ アクション のセクションを参照してください。

DB コミット トランザクション

イベントがトリガーされると、DB コミット トランザクションのアクションは、*接続*コンボ ボックスで選択されたデータ ソースにトランザクションをコミットします。 このコンボ ボックスはプロジェクトのすべてのデータ ソースをリストし、DB トランザクションの開始アクション内で具体的に使用されるための追加データベース接続のオプションを与えます。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

DB トランザクションについて
トランザクションが必要な各 DB アクセスには、自動的にトランザクションが作成され、後に自動的に閉じられます。これは一部のセットアップでは理想的です。例えば、同時に自動的に更新したい2 つのDB ページソースが存在する場合、両方のテーブルが保存されると、トランザクションがコミットされます。その他の場合はロールバックされます。このようなシチュエーションに対応するために、接続ベースのトランザクションを作成することができます。

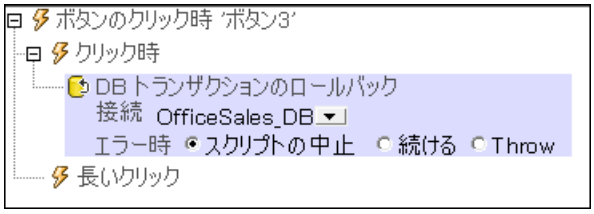
トランザクションの開始 が始まると、同じDB 接続に属するすべてのDB オペレーションはこのトランザクションを使用します。

トランザクションのコミット により、トランザクションの外界で変更が可視化されます。変更はロールバックされることができます。この場合、ページソースに保存した場合でも、ロールバックの後で変更は可視化されません!閉じられていない(コミットされたまたはロールバックされた) トランザクションは、アクションツリーのが最後に到達すると自動的にロールバックされます!この効果についてメッセージ ウィンドウに警告が表示されます。

上記の振る舞いはトランザクション アクションについて参照していますが、振る舞いはトランザクションと同じ接続を使用するDB オペレーションにも適用されます。

DB ロールバック トランザクション

イベントがトリガーされると、ロールバック トランザクションのアクションは接続コンボ ボックスで選択されたデータ ソースのトランザクションをロールバックします。このコンボ ボックスはプロジェクトのすべてのデータ ソースをリストし、使用されるための追加データベース接続のオプションを与えます。



エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合 (*成功時*)、またはエラーが発生する場合 (*エラー時*)。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード保存、バイナリファイルのロード保存、イメージのロード保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクションのキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクションのセクションを参照してください。

DB トランザクションについて
トランザクションが必要な各 DB アクセスには、自動的にトランザクションが作成され、後に自動的に閉じられます。これは一部のセットアップでは理想的です。例えば、同時に自動的に更新したい2 つのDB ページソースが存在する場合、両方のテーブルが保存されると、トランザクションがコミットされます。その他の場合はロールバックされます。このようなシチュエーションに対応するために、接続ベースのトランザクションを作成することができます。

トランザクションの開始 が始まると、同じDB 接続に属するすべてのDB オペレーションはこのトランザクションを使用します。

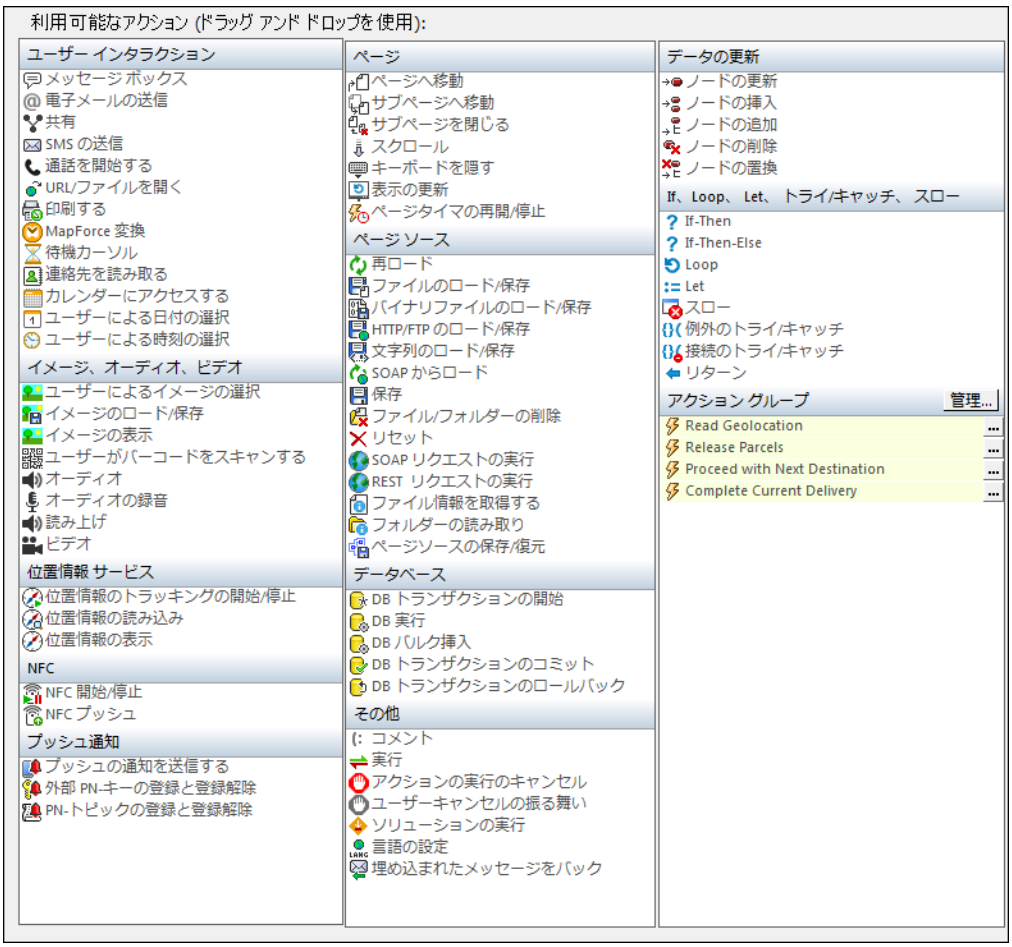
トランザクションのコミット により、トランザクションの外界で変更が可視化されます。変更はロールバックされることができます。この場合、ページソースに保存した場合でも、ロールバックの後で変更は可視化されません!閉じられていない(コミットされたまたはロールバックされた) トランザクションは、アクション ツリーのが最後に到達すると自動的にロールバックされます!この効果についてメッセージ ウィンドウに警告が表示されます。

上記の振る舞いはトランザクション アクションについて参照していますが、振る舞いはトランザクションと同じ接続を使用するDB オペレーションにも適用されます。

10.9 その他

アクションダイアログのその他 グループで次のアクションを使用することができます (下のスクリーンショット):

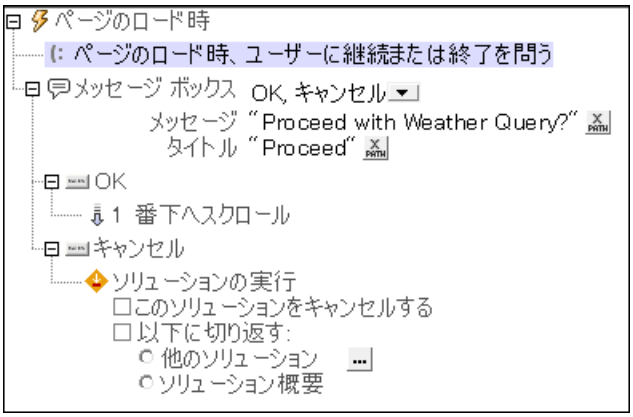
- コメント
- 実行
- アクションの実行のキャンセル
- ユーザーキャンセルの振る舞い
- ソリューションの実行
- 連絡先の読み取り
- 言語の設定
- 埋め込まれたメッセージをバックする



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロール アクション コマンドを選択します。ページイベントおよびコントロール イベントも参照してください。

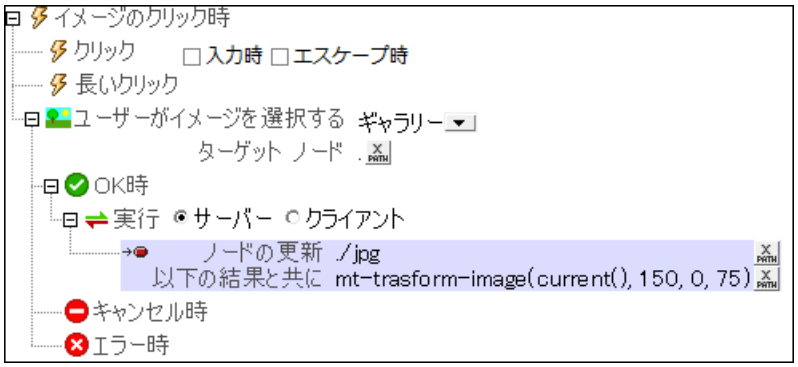
コメント

イベントのアクションの定義にコメントを追加することができます。(*下のスクリーンショット参照*)。イベントのアクションの定義に異なるアクションの説明を挿入するときに役に立ちます。



実行

[実行アクション] (下のスクリーンショット参照) は、サーバーまたはクライアントに対して、明示的にアクションのサブアクションが実行される場所を指定します。



上のスクリーンショットは [実行アクション] の典型的な使用例です:

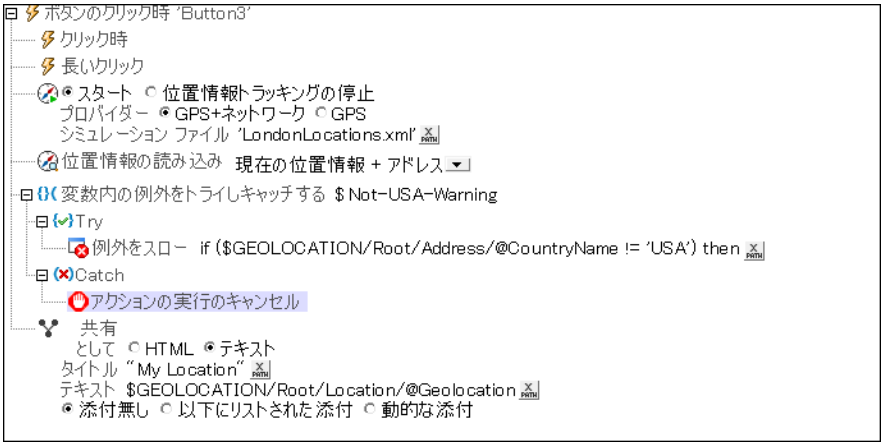
1. [ユーザーのイメージ選択] アクション] はユーザーがギャラリーからイメージを選択して、現在のノードにBase64 として保存するように促します (`//image/base64` にあるとします)。
2. イメージを現在のノードに送信することに成功すると、[OK時] 条件は [実行アクション] を使用して、ユーザーにより選択されたイメージを (Altova XPath 拡張子機能 **`mt-transform-image`**) を使用してサーバー上で変換し、兄弟 **`jpg`** ノードを更新します。ノードはサーバー上でアップデートされ、アクションの処理が完了するとクライアントに送信されます。

クライアントまたはサーバーに対しての変換
明確に指定されていない場合、関数 **`mt-transform-image`** はクライアントに対して実行されます。これはクライアントの一部でメモリに関する問題を起こす可能性があります。変換が開始されると、イメージが Base64 エンコードから、大きくなる可能性のあるBMP フォーマットへとアンパックされます。変換が終了すると、変換されたファイルは元のフォーマットで保管されます。大きな BMP フォーマットは、クライアントの一部でメモリに関する問題を起こす可能性があり、この点に注意する必要があります。

クライアント上でメモリに関する問題を回避するため、明示的に変換をサーバーで実行するように指定します。アクションに対して実行を使用して実行し、サーバーで子アクションが実行されるように指定します。このアクションに対して実行 のすべての子アクションはサーバーで実行されるようになります。ノードの更新 などのアクションを使用して変換の結果と共にノードの更新を行うことができます。変換されたイメージと共にターゲットノードが更新されます。アクションの処理が完了、またはワークフローがクライアントに切り返されるとMobileTogether はクライアントの結果を自動的に送信します。

アクションの実行のキャンセル

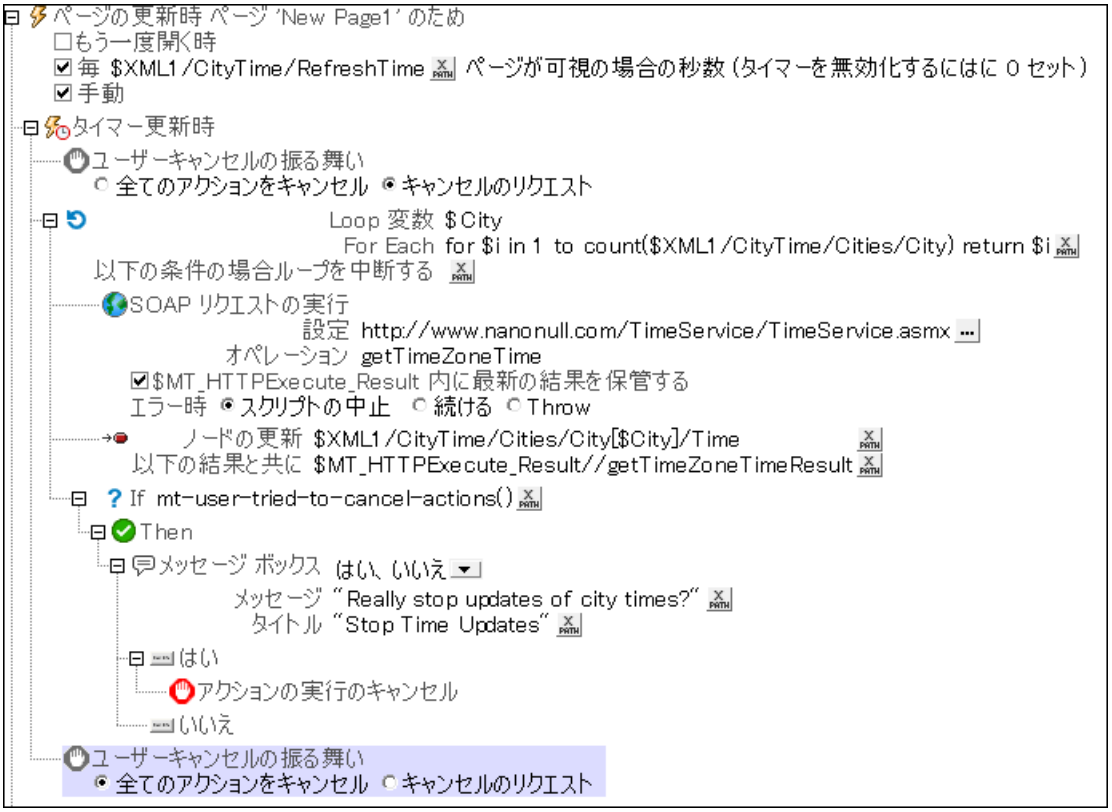
(下のスクリーンショットでハイライトされている)アクションのキャンセルの実行は、イベントのアクションシーケンスの実行をキャンセルします。下のスクリーンショットでは、例えば、トライ/キャッチ アクションの トライ パートで例外がスローされると、トライ/キャッチ アクションの キャッチ パートが実行されます。このパートは、アクションのキャンセルの実行アクションを含んでいるため、アクションのイベントのシーケンスはキャンセルされます。この結果、共有 アクションは実行されません。



ユーザーキャンセルの振る舞い

(*下のスクリーンショットでハイライトされている*)ユーザーのキャンセルの振る舞いにより、ユーザーキャンセルアクションをオーバーライドし、アクションの実行を継続することができます。

- *全てのアクションをキャンセル*: ユーザーが**[戻る]** ボタン(または、長いアクションの実行中に表示される**[キャンセル]** ボタン)を押すと、キャンセルが許可されます。これはデフォルトの振る舞いです。
- *キャンセルのリクエスト*: このオプションによりユーザーのキャンセルのアクションをオーバーライドすることができます。下のスクリーンショット内のサンプルを参照してください。ユーザーがアクションの実行をキャンセルしようと試みると、アクションの実行は中断されません。代わりに `mt-user-tried-to-cancel-actions` 関数が `true` に設定されます。適切なアクションのセットを定義して、この関数の値により取られるアクションを設定します。アクションのキャンセルのリクエストフラグがtrue に設定されると、全てのアクションをキャンセルによりユーザーのキャンセルの振る舞いアクションをリセットすることができます。これにより `mt-user-tried-to-cancel-actions` 関数が `false` のデフォルトの値にリセットされます。



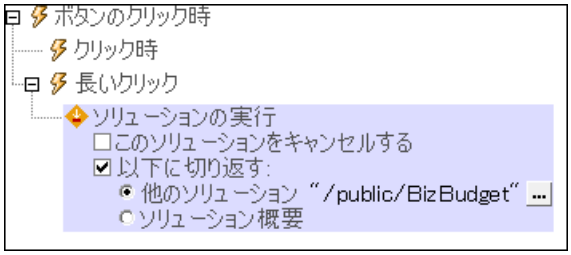
上のスクリーンショットに表示されるサンプル内では、*タイマーの更新時*イベントオプションのために以下が定義されています:

1. ユーザーキャンセルの振る舞いアクションはキャンセルのリクエストに設定されています。この結果、ユーザーが**[戻る]** または**[キャンセル]** ボタンを押すと、`mt-user-tried-to-cancel-actions` 関数は `true` に設定されます。
2. *ループアクション*が開始されます。各反復により、`//City` 要素がSOAP リクエストによりそれぞれの都市の現在の時刻に更新されます。複数の `//City` 要素が存在し、更新に時間がかかる場合、エンドユーザーは**[戻る]** または**[キャンセル]** ボタンを押して更新をキャンセルすることができます。
3. エンドユーザーがキャンセルを試みる間の反復の最後で、`mt-user-tried-to-cancel-actions` 関数が評価されます。この時点での値は `true` (*上記ポイント1 参照*) であるため、表示されるメッセージボックスはユーザーにキャンセルの続行を問います。反応により、アクションはキャンセルされるか、または、続行されます。
4. ループが完了すると、ユーザーキャンセルの振る舞いアクションは、明示的に全てのアクションをキャンセルのデフォルトの値に設定されます。このステップはユーザーキャンセルの振る舞いが(確認無しにキャンセルが実行される)デフォルト

の値に設定される場合は必須のステップです。それ以外の場合、ユーザーキャンセルの振る舞いアクションは**全てのアクションを**キャンセルのリクエストの値を明示的に変更されるまで保有します。

ソリューションの実行

イベントがトリガーされると、ソリューションの実行アクションによりソリューションのキャンセルまたは、中断 (すなわち、背景で実行されている) の状態を選択することができます。(キャンセルまたは中断) 両方のケースで、以下を行うことができます: (i) 他のソリューションに切り替える、(ii) ソリューションの概要に切り替える、(MobileTogether Client のソリューションページ)、(iii) ソリューションまたはソリューションの概要に切り替えない。これらの複数の可能性に関しては下のテーブルにリストされています。



切り替え...	ソリューションのキャンセル	ソリューションの中断
他のソリューション	ソリューションはメッセージ無しに閉じられ、指定されたソリューションが開かれます。	現在のソリューションは、背景で "最小化 "され実行され、指定されたソリューションが開かれます。
ソリューションの概要	ソリューションはメッセージ無しに閉じられ、表示は、MobileTogether Client の*ソリューションの概要*ページに切り替えられます。	現在のソリューションは、背景で "最小化 "されて実行され、表示は、MobileTogether Client の*ソリューションの概要*ページに切り替えられます。
チェックボックスのチェック無し	ソリューションはメッセージ無しに閉じられ、表示はソリューションが開始された場所に戻ります (*下のノート参照*)。	現在のソリューションは、背景で "最小化 "されて実行され、 表示はソリューションが開始された場所に戻ります (*下のノート参照*)。

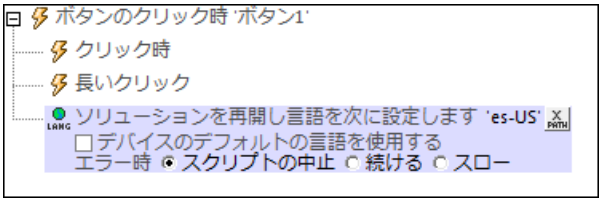
下のポイントに注意してください:

- Web クライアントは中断されたソリューションをサポートしません。アクティブなソリューションのみがサポートされます。
- 中断されたソリューションが "最小化 "され MobileTogether Client アプリケーションの実行中 (背景で実行中) のページのアイコンとして表示され、このアイコンをタップすることで開くことができます。
- ソリューションが背景で "最小化 "されると、中断されているソリューション内でアクションは実行されません。例えば、タイマーは実行されず、位置情報は使用されず、オーディオの再生は一時停止されます。ソリューションが再び開かれると、ページの更新時イベントが実行され、一時停止されていたオーディオの再生が再開されます。*他のソリューションに切り替え時のプロジェクトプロパティを参照してください。*
- *他のソリューションへの切り替え*: 現在のソリューションはキャンセルされず、"最小化 "されて背景で実行されます。開かれるソリューションは、XPath 式の編集をクリックして、ソリューションの場所を (つまり引用符内で) 文字列として入力することができます。ロケーションは、現在のソリューションと同じサーバーである必要があり、ロケーション文字列は、ターゲットソリューションがデプロイされた際に入力された文字列と同じである必要があります。(上のスクリーンショットを参照してください)。ターゲットソリューションが既に (最小化されて) 実行されている場合、ソリューションは最小化された箇所で開かれ継続されることができます。
- 切り替えのオプションがチェックボックスのチェック無しの場合、表示は、以下のとおりです: (i) ショートカットを使用して開始された場合はホーム画像に戻ります、または (ii) *ソリューションの概要* ページに戻ります。iOS では、*ソリューションの概要* ページ常に戻ります。i
- 平行して実行中のソリューションの **[戻る]** ボタンをタップすると、ソリューションは中断されます。(i) *Android*、*Windows App* および *Windows*: ショートカットからソリューションが開始されるとホーム画像に戻ります。それ以外の場合は、*ソリューションの概要* に戻ります。どちらの場合にも、戻るアクションはユーザーの入力無しに直接実行されます。 (ii) *iOS (非パラレルソリューション)*: ユーザーはソリューションを終了するかを割れます。終了が選択されると、*ソリューションの概要* ページに戻ります。

言語の設定

ソリューションがローカライズされている場合、モバイルデバイスの言語の設定により、異なる言語でソリューションを表示することができます。 例えば、ソリューションが英語のテキスト文字列でデザインされている場合、このソリューションのデフォルトの言語は英語になります。これらの文字列がスペイン語にローカライズされた場合（すなわち、スペイン語に翻訳された場合）、ソリューションは英語のデバイス上では英語で、スペイン語のデバイスではスペイン語で表示されます。選択するに当たって重要な点は、モバイルデバイスの言語設定がソリューションのローカライズ言語のコード名の1つに対応している必要がある点です。

言語の選択アクションにより、デバイスの言語を変更することなく、ソリューションをユーザーにより選択された言語で再開することができます。例えば、ユーザーはボタンを押して特定の言語に変更することができます（*US Spanish でソリューションを再開させる* 下のスクリーンショットを参照してください）。
または、ユーザーはコンボボックスのドロップダウンリストから言語を選択することができます。



`language-country` コード（例えば、**es-US** または **fr-CH**）を入力、または、`language` コード（例えば、**es** または **fr**）を入力します。（特定の言語を選択するために）アクションがトリガーされると、ソリューションの言語は以下のカスケード法則に従い決定されます。

1. ソリューションの対応するローカライズされた **language-country**（**es-US** または **fr-CH**）文字列は存在する場所で使用されます。
2. 文字列のための、`language-country` ローカリゼーション（**es-US** または **fr-CH**）が存在しない場合、ローカライズされた **language**（**es** または **fr**）文字列 が存在する場合は使用されます。
3. 文字列のための、`language-country` ローカリゼーション（**es-US** または **fr-CH**）または `language` ローカリゼーション（**es** または **fr**）が存在しない場合、デフォルトの言語 が使用されます。

または、モバイルデバイスのデフォルトの言語を使用することもできます、この場合、デバイスの言語の設定が使用されるローカライズ言語を決定します。上記のカスケードの法則が適用されます。

ローカライズされた文字列が定義されている言語でシミュレーションを確認するには、シミュレーション言語 **[プロジェクト | シミュレーション言語]** コマンド を使用して、シミュレーションする言語を設定して、シミュレーションの実行を行います。

エラーの処理
エラー時 オプションはエラーが発生すると何が行われるかを決定する事ができます:

- *スクリプトの中断*: エラーの発生後、トリガーされたイベントの後全てのアクションは終了されます。これがエラーが発生した時のデフォルトのアクションです。エラーに関わらず継続する場合は、継続またはスローオプションを選択します。
- *継続*: アクションは終了されません。その代わりに、以下のイベントの際に何が起こるかを選択する事ができます: エラーが発生しない場合（*成功時*）、またはエラーが発生する場合（*エラー時*）。例えば、ページのロードが成功したかを示すメッセージを表示することができます。アクション ファイルのロード/保存、バイナリファイルのロード/保存、イメージのロード/保存 アクションのためには、次の追加オプションがあります: ユーザーがダイアログをキャンセルすると実行されるキャンセル時。
- *スロー*: エラーが検出されると、このオプションは トライ/キャッチ アクションの変数内に保管された例外をスローします。トライ/キャッチ アクション のキャッチ部分は、エラーが発生した時、どのアクションが起こるかを指定します。エラーが発生しない場合は、次のアクションが処理されます。詳細に関しては、トライ/キャッチ アクション のセクションを参照してください。

埋め込まれたメッセージをバックする

埋め込まれたメッセージをバックするアクションは、現在のソリューションをロードしたIFrame へ(シリアル化されたJSON) 文字列を送信します。文字列は **メッセージ** イベントとして送信され、埋め込まれるHTML ページによりIFrame から JavaScript の **addEventListener()** メソッドを使用して **message** 型イベントをリッスンするためにピックアップされます。

アクションは、入力 XPath 式として評価する(シリアル化されたJSON) 文字列を取ります。全ての文字列は受け入れられますが、HTML ページを受け取るために使用される文字列のために、文字列はJSON 文字列である必要があります(これはWeb サーバー間で使用されるJSON は共通のデータ交換書式であるからです)。

メッセージ文字列を与えるXPath 式は以下の1つです:

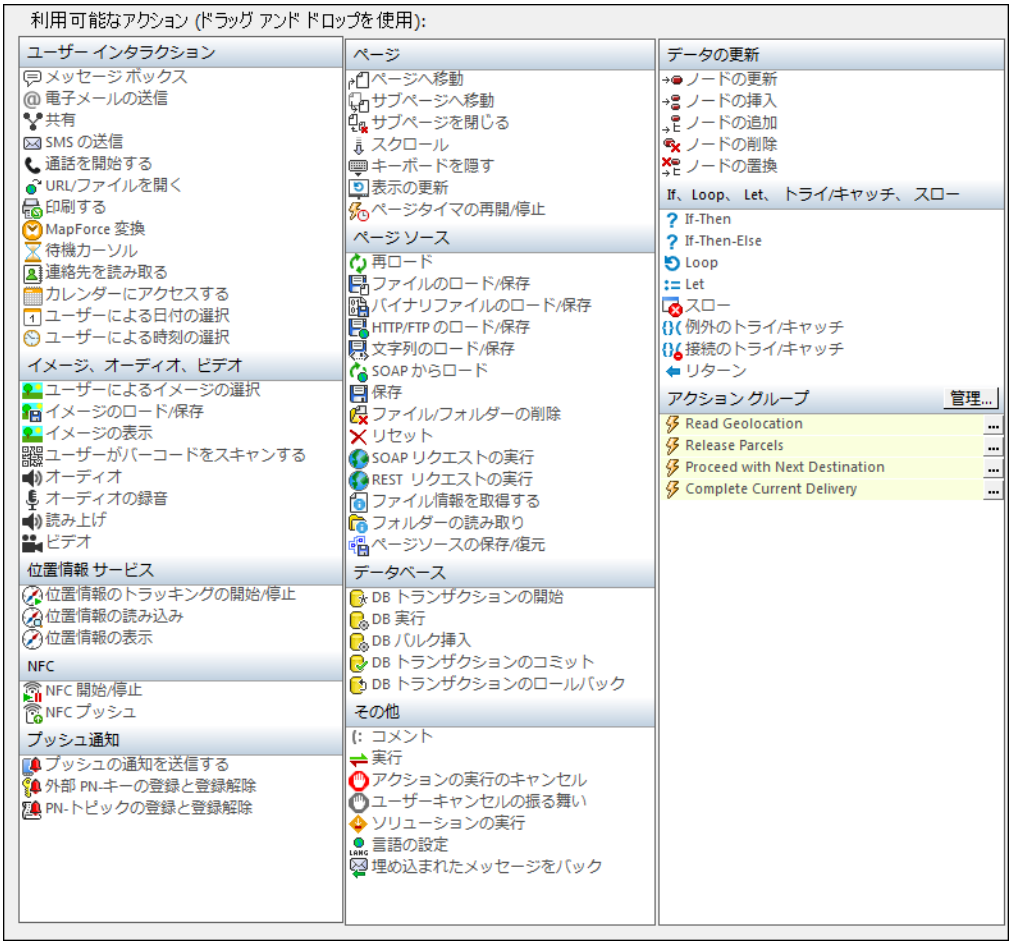
- **$MT_EMBEDDEDMESSAGE** は処理し転送するためのJSON データを含むページソースノードツリーです。このツリーのルート要素は、常に **json** という名前が与えられます。**$MT_EMBEDDEDMESSAGE** ツリー全体が返されると、上のスクリーンショット内で、シリアル化されたJSON 文字列はトップレベルプロパティとして **json** プロパティを持ちます。または、メッセージを **$MT_EMBEDDEDMESSAGE** ページソースのフラグメントとなるように定義することができます。例えば、**$MT_EMBEDDEDMESSAGE/json**。この場合、メッセージのシリアル化されたJSON 文字列は、ページソースノードの **json** ノードのコンテンツになります。
- JSON データ構造、または、文字列を評価するノードは、JSON データ構造を持ちます。JSON データ構造である文字列のサンプルは次のとおりです: **'{ "books": { "author": "Mary Shelley", "title": "Frankenstein" }}'**。シリアル化されたJSON 文字列内では、**books** プロパティは含まれている構造です。

次も参照してください: リッスン: ソリューションからWeb ページへ。

10.10 データの更新

アクションダイアログのデータの更新 グループで次のアクションを使用することができます (下のスクリーンショット):

- ノードの更新
- ノードの挿入
- ノードの追加
- ノードの削除
- ノードの追加の削除



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロール アクション コマンドを選択します。ページ イベントおよびコントロール イベントも参照してください。

ノードの更新

ノードの更新 アクションは指定された値をもつ1 つ以上の指定されたノードを更新します。ノードの更新と値の更新は XPath 式により指定されます。下のスクリーンショットでは、コンテキスト要素の`@Updated` 属性ノードが現在の日付と共に更新されています (XPath 関数 `current-date-no-TZ()`の評価の結果)。

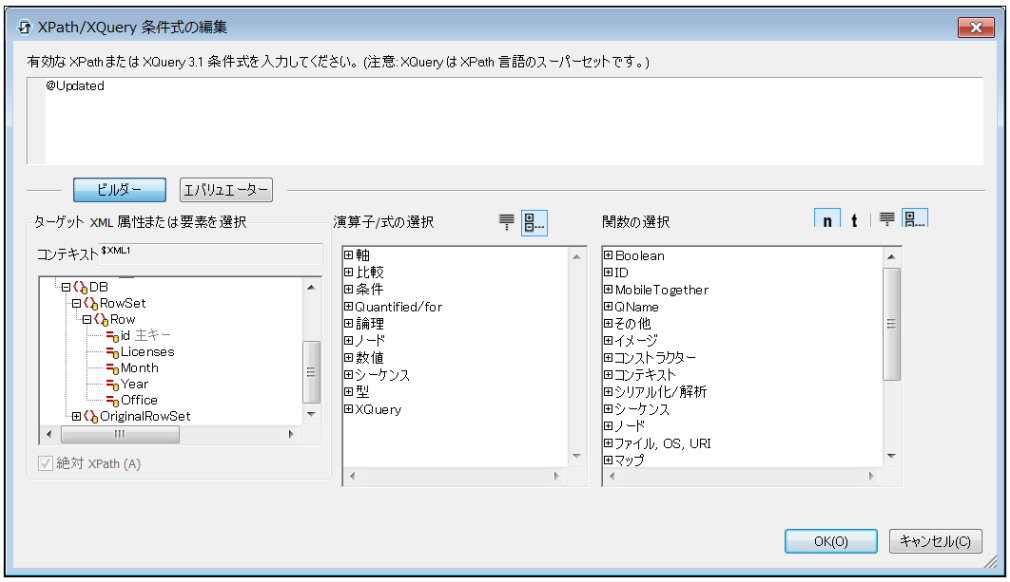
ノードの更新および値の更新を指定する時に注意点があります:

- 相対パスに対するコンテキスト ノードの重要性
- 更新のターゲットノードは**`$MT_TargetNode`** 変数により参照される場合があります。
- ソースノードが混在コンテンツ(テキストと要素) を持つ要素である場合、混在コンテンツ要素のテキストのみが更新に使用されます。子孫要素のテキスト コンテンツは無視されます。

相対パスに対するコンテキスト ノードの重要性

更新されるノードが相対パスとして指定される場合、コンテキストノードの子孫であるノードのみが更新されます。兄弟要素の子孫を更新するには、絶対ロケーターパスが必要です。この点に関しては下で説明されています。

- 下のスクリーンショットでは、(コンテキスト フィールドで指定されている) コンテキスト ノードはDB のRow 要素です。コンテキスト ノードは(アクションが定義される) コントロール内のノードで ロケートされます (またはコントロールが関連付けられます)。
- 更新される`@Updated` 属性は、ですから、その特定のRow 要素の`@Updated` 属性です。これは、`Row` 要素のコントロール イベントがトリガーされると、この場合、現在の日付と共に、**`Row` 要素のみの`@Updated` 属性** が更新されます。



- XPath 式が、@Updated ノードがルートノードから開始されるようにアドレスするよう変更されると、(以下のスクリーンショットのように: $DB2/DB/RowSet/Row/@Updated) **すべての Row 要素の @Updated ノード** が更新されます。

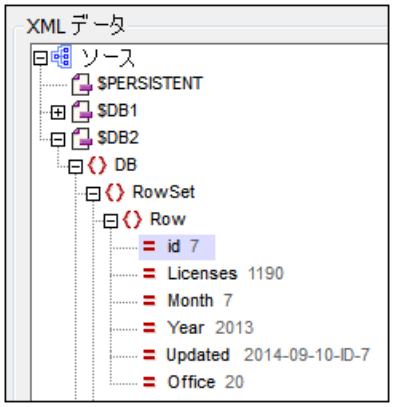
- 上のスクリーンショットで、XPath 式によりデータソースのすべてのノードにアクセスすることができることを確認してください。

$MT_TargetNode を使用して特定のターゲットノードを更新する

更新のためのターゲットノードが定義されると、**$MT_TargetNode** 変数と共にこのノードを参照することができます。例えば

```
ノードの更新 : @Updated
値の更新: concat(current-date-no-TZ(), '-ID-', $MT_TargetNode/../@id)
```

値の更新は、現在の日付にサフィックスされた現在の Row 要素の @id 属性の値を持つことが下のスクリーンショットに表示されています。



ノードの更新がすべてのRow要素の@Updated 属性で、値の更新のXPath 式は、前の例と同じです:

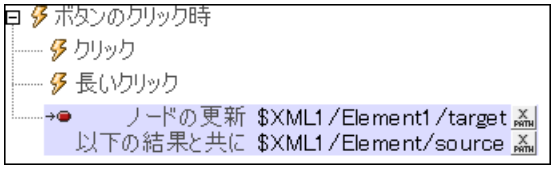
```
ノードの更新 : $DB2/DB/RowSet/Row/@Updated
値の更新: concat(current-date-no-TZ(), '-ID-', $MT_TargetNode/../@id)
```

各 Row 要素の@Updated 属性は、現在の日付にサフィックスされた自身のRow 要素 @id 属性値 を持つ値をもちます。

混在コンテンツを持つソース要素

混在したコンテンツ(テキストと要素)付きの要素がXPath ロケーター式によりロケートされる場合、混在したコンテンツ要素のテキストコンテンツのみが返されます。子孫要素のテキストコンテンツは無視されます。

ノードの更新アクションの例により最もよく説明されています。下のスクリーンショットで定義されるノードの更新アクションを熟考してください。



XML ツリーが次の構造とコンテンツを持つ場合:

```
<Element1>
   <source>AAA
      <subsource>BBB</subsource>
  </source>
  <target></target>
</Element1>
```

子要素 subsource のコンテンツを無視する一方、混在コンテンツ要素 source のテキストコンテンツと共に、target 要素が更新されます。target という名のノードは、<target>AAA</target>に更新されます。

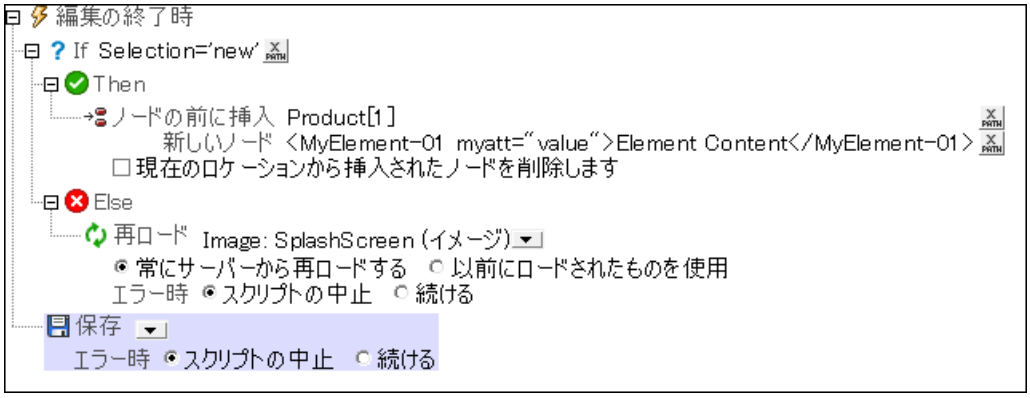
メモ 子孫ノードのテキスト コンテンツを含みたい場合、空の文字列付きの**concat** 関数を第 2 番目の引数として使用します。上のXML 例を使用することにより、たとえば、式 **concat($XML1/Element1/source, '')** は **"AAABBB"** を返します。

メモ XPath 対応メソッドのシリアル化を使用するチャート: 混在コンテンツ要素がXPath ロケーター式を使用してロケ

ートされる場合、子孫要素のテキスト コンテンツもシリアル化されます。

ノードの挿入

[ノードの挿入アクション] は1 つ以上の新しいノードをXPath 式により、*ノードの前に挿入*の設定のために選択されたノードの前に挿入します。挿入されたノードは単一ノード、ノードのシーケンス、ツリー全体のフラグメントであることができます。これらの挿入されたノードは、XQuery のXML コンストラクターを使用して構築されます。次の7 つすべてのXML ノードの型はこのXQuery 構文を用いて構築されることができます: 要素、属性、テキスト、ドキュメント、コメント、処理命令、および名前空間。



メモ [ノードの挿入] と[ノードの追加] の違いは、[ノードの挿入] はノードを選択されたノードの前に追加し、[ノードの追加] はノードを選択されたノードの(最初または最後の) 子ノードとして追加します。

挿入されたノードの場所

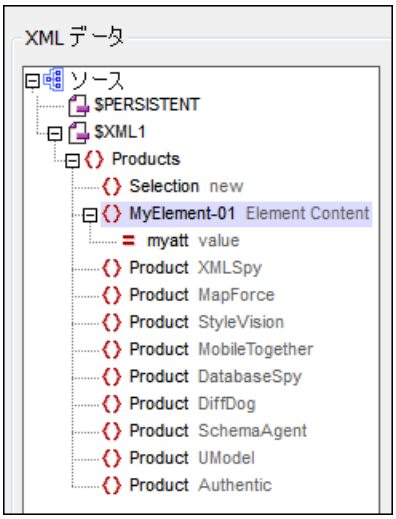
新しいノードは、この設定のXPath 式 (*ノードの前に挿入*) により返されたノードの前に挿入されます。上のスクリーンショットでは、新しいノードが、(XPath 式 `Product[1]`で選択された) 最初の`Product` 要素 の前に挿入されています。コンテキストノードは`Product` の親で`Products` という名のノードです。もし、述語 `[1]` が使用されていない場合、`Products` のすべての子 `Product` は、XPath 式により返され、新しいノードが各 `Product` 要素の前に挿入されます。

新規ノード

上のスクリーンショットに表示されるように、新規ノードは直接 XML コンストラクタとして挿入されることができます:

```
<MyElement-01 myatt="value">Element Content</MyElement-01>
```

これにより、下のスクリーンショットに表示されるように、`MyElement-01` 要素を 最初の`Product` 要素の前に挿入します。



XPath ロケーター式を使用して、ページのデータソースからノード（とその全ての子孫）を挿入することができます。例えば

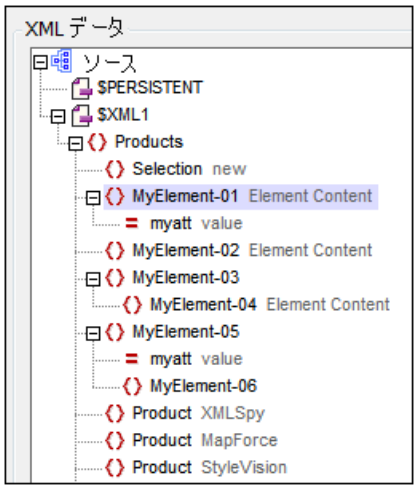
```
$XML2/Row
```

XQuery の計算されたノードコンストラクタも使用することができます。例えば

```
element MyElement-01 {xs:string("Element Content")}
attribute myatt{"value"}
```

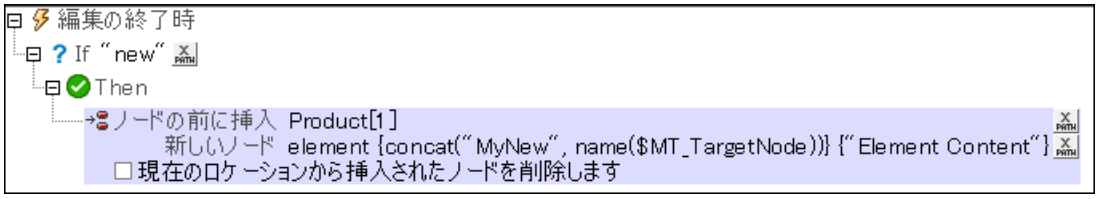
下のXPath 式は、下のスクリーンショットに表示されるように、最初のProduct 要素の上に挿入されている出力を作成します。

```
<MyElement-01 myatt="value">Element Content</MyElement-01>,
element MyElement-02 {"Element Content"},
element MyElement-03 {element MyElement-04 {"Element Content"}},
element MyElement-05{attribute myatt{"value"}, element MyElement-06{}}
```



$MT_TargetNode 変数

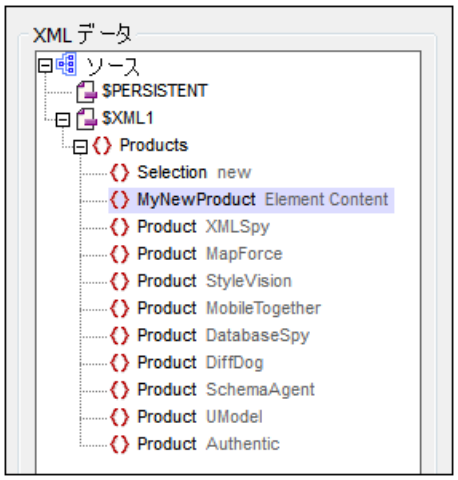
子ノードの挿入の前にターゲットされた、ノードの挿入の定義のノードは、自動的にMobileTogether Designer に内蔵された**$MT_TargetNode** 変数に保存されます。この変数は、下のスクリーンショットで表示されるように、定義の第 2 の XPath 式に使用することができます。



第 2 のXPath 式は新しいノードの名前の一部としてターゲット ノード($MT_TargetNode) と同じ名前を使用して、新しいノードを作成します。

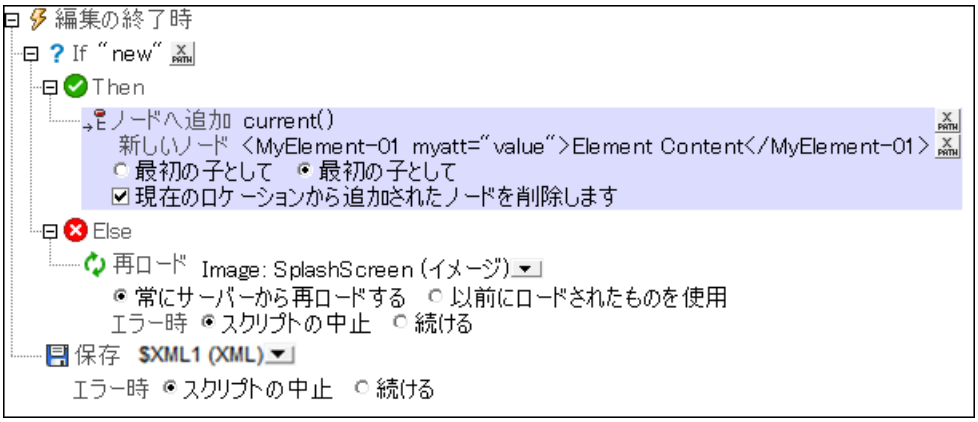
```
element {concat("MyNew",name($MT_TargetNode))} {"Element Content"}
```

上に定義される[ノードの挿入 アクション] の結果は、下のスクリーンショットに表示されています。



ノードの追加

[ノードの追加アクション] は1つ以上の新しいノードを、*ノードの追加*設定のXPath 式で選択されたノードの最初または最後の子 (*ノードのセット*) として、追加します。追加されたノードは単一ノード、ノードのシーケンス、ツリー全体のフラグメントであることができます。これらの追加されたノードは、XQuery のXML コンストラクターを使用して構築されます。次の7 つすべてのXML ノードの型はこのXQuery 構文を用いて構築されることができます: 要素、属性、テキスト、ドキュメント、コメント、処理命令、および名前空間。



メモ　[ノードの挿入] と[ノードの追加] の違いは、[ノードの挿入] はノードを選択されたノードの前に追加し、[ノードの追加] はノードを選択されたノードの(最初または最後の) 子ノードとして追加します。

追加されたノードの場所

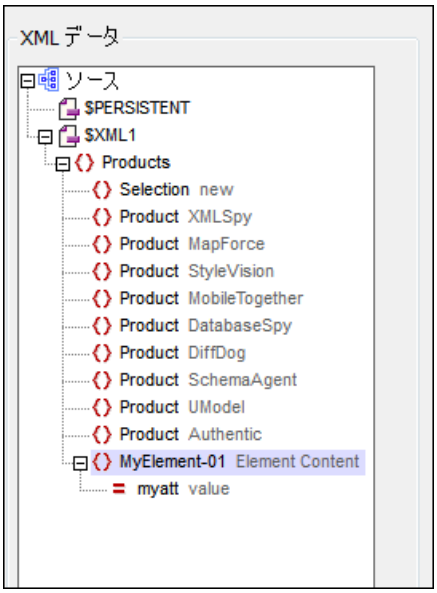
新しいノードは、この設定のXPath 式 (*ノードの前に挿入*) により返されたノードに最初または最後の子ノードとして追加されます。上のスクリーンショットでは、新しいノードはコンテキストノード、(XPath 式 `current()` により選択された) `Products` 要素 の最後の子ノードとして追加されています。新しいノードが最初または最後の子ノードとして追加されるかは、アクションの定義の適切なラジオボタンを選択してください。

新規ノード

上のスクリーンショットに表示されるように、新規ノードは直接 XML コンストラクタとして挿入されることができます:

```
<MyElement-01 myatt="value">Element Content</MyElement-01>
```

これにより、下のスクリーンショットに表示されるように、`MyElement-01` 要素を 最後の`Product` 要素の後に挿入します。



XPath ロケーター式を使用して、ページのデータソースからノード(とその全ての子孫) を挿入することができます。例えば

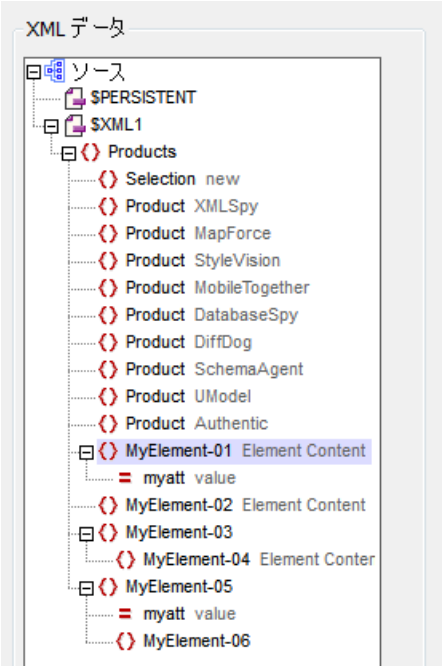
```
$XML2/Row
```

XQuery の計算されたノードコンストラクタも使用することができます。例えば

```
element MyElement-01 {xs:string("Element Content")}
attribute myatt{"value"}
```

下のXPath 式は、下のスクリーンショットに表示されるとおり、Products 要素の最後の子ノードとしてとして追加された出力、すなわち、最後の現在の子ノード(最後のProduct ノード) を作成します。

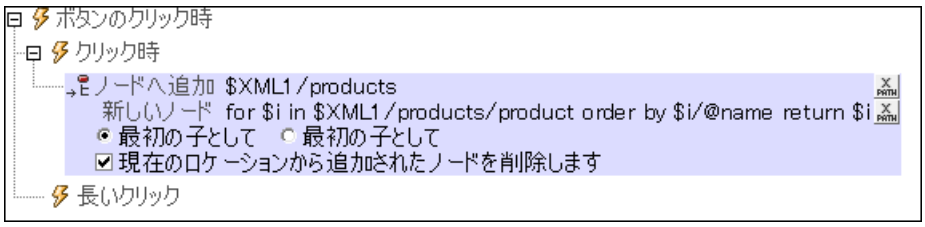
```
<MyElement-01 myatt="value">Element Content</MyElement-01>,
element MyElement-02 {"Element Content"},
element MyElement-03 {element MyElement-04 {"Element Content"}},
element MyElement-05{attribute myatt{"value"}, element MyElement-06{}}
```



追加されたノードを元の場所から削除する

追加されたノードがプロジェクトのデータソースから取得された場合、追加されたノードの削除を現在の場所のチェックボックスを選択してノードを元の場所から削除することができます。新しいノードが直接構成される場合、つまりプロジェクトのデータソースを参照しないで構成される場合、このオプションの選択はデータベースに影響をもたらしません。

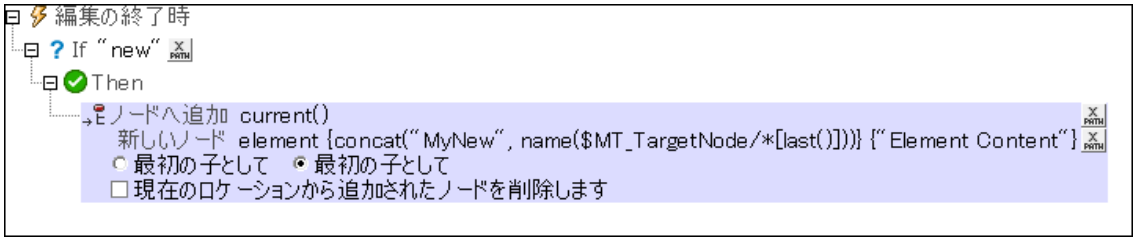
現在の場所から追加されたノードを削除するオプションの使用のよい例は、ノードの保管です。以下の構造を持つツリーがあるとします: **`$XML1/products/product/@name`**。ノードを`@name` の値をもとに並べ替えるとします。下のスクリーンショットに表示されるように、ノードの追加定義を使用して行うことができます。



- `$XML1/products` ノードの最後の子として新しいノードを追加します。
- XPath 式を使用して新しいノードが生成されます: **`for $i in $XML1/products/product order by $i/@name return $i`**。**`order by`** 句は、**`product`** アイテムのシーケンスを反復する前に並べ替えます。
- 現在の場所から追加されたノードを削除するオプションは、もとの並べ替えられていない製品のシーケンスを削除します。追加され、並べ替えられたシーケンスは維持されます。

$MT_TargetNode 変数

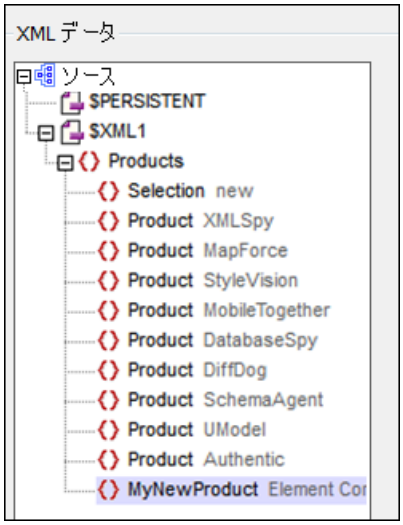
子ノードの追加の前にターゲットされた、ノードの追加の定義のノードは、自動的にMobileTogether Designer に内蔵された**$MT_TargetNode** 変数に保存されます。この変数は、下のスクリーンショットで表示されるように、定義の第 2 の XPath 式に使用することができます。



第 2 のXPath 式は、ターゲットノード($MT_TargetNode) を使用して、ターゲット ノードの最後の子要素を検索し、その子要素の名前を使用して新しい要素の名前を作成するために使用します。

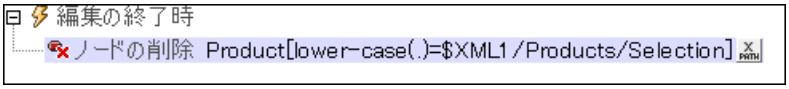
```
element {concat("MyNew",name($MT_TargetNode/*[last()]))}{"Element Content"}
```

(ターゲットノードが$XML1/Products の場合、) 上に定義される[ノードの追加 アクション] の結果は、下のスクリーンショットに表示されています。

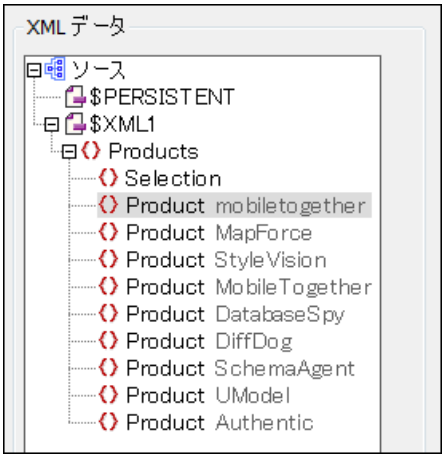


ノードの削除

ノードの削除アクションはアクションのXPath 式で選択されたノードを削除します。



上のスクリーンショットの定義は、コンボボックスの [編集の完了時] イベントで`$XML1Products/Selection` 要素を更新します。コンテキスト ノードは`Products` です。アクションは、小文字に変換されると`Selection` 要素の(小文字の) コンテンツに一致する、コンテンツを持つ子 `Product` 要素を削除します。下のスクリーンショットでは、`mobiletogether` がコンボボックスで選択されており、Selection 要素の値になります。テキスト`"MobileTogether"` を含む`Product` 要素が削除されます。



ノードの置換

アクションがトリガーされると、以下が発生します:

1. サブノードXPath 式により指定されている**ノードセット**は、(ターゲットノード設定内で設定される)**ターゲットノード**から削除されます。
2. サブノードXPath 式により指定されている**ノードセット**は、(ソースノード設定内で設定される)**ソースノード**からターゲットノードにコピーされます。ノードセットはターゲットノードの最初の子、または、最後の子として追加することができます。

サブノード設定のXPath 式はソースとターゲットのために個別に評価されます: コンテキスト内では、それぞれ ソース ノードとして指定されているノード、および、ターゲットノードとして指定されているノードです。

ノードの置換 アクションはアクション ノードの削除 と ノードの追加の2つのアクションの組合せです。

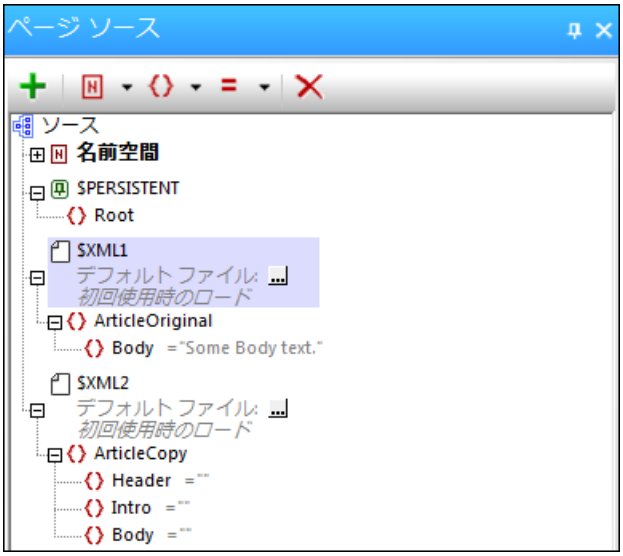
サンプル

以下のサンプルはノードの置換 アクションの作動をデモします。

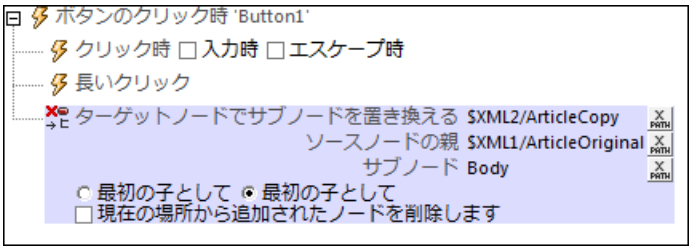
サンプルの置換

下に表示されているページソースペイン内で表示されている2つのXML ツリーの構造を考慮してください。

- **`$XML1/ArticleOriginal`** には、**`Body`** という名前の子要素が存在します。
- **`$XML2/ArticleCopy`** には、**`Header`**、**`Intro`**、**`Body`** の順序でという名前の子要素が存在します。



下のスクリーンショットで示されるように、置換えノードアクションを使用して**`ArticleCopy`** の**`Body`** 要素を ArticleOriginal の**`Body`** 要素と置き換えることができます。



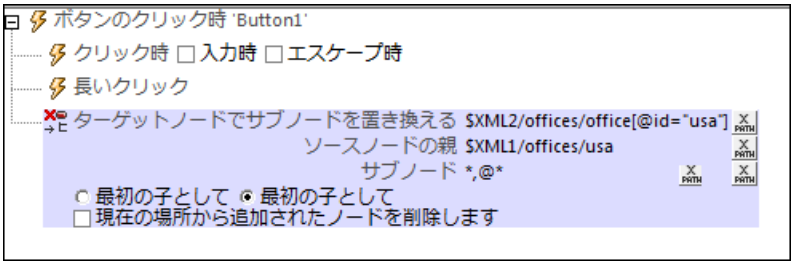
アクションは**Body** という名前のサブノードをと**ArticleOriginal** い名前のソースノードから最後の子として、**ArticleCopy** という名前のターゲットノードに追加します。下のスクリーンショットは、アクション実行後のシミュレーション中のページソースを表示しています。**ArticleCopy** の**Body** 要素は**ArticleOriginal** の**Body** 要素と置き換えられます。

Header と**Intro** は影響を受けていませんが、**Body** は置換えられていません。**ArticleCopy** に**Body** 要素が存在しない場合、新規の**Body** 要素が追加されるはずです。

(サブノード設定内で設定済みの) XPath 式 **Body** は、ソースノードとターゲットノードXPath 式のコンテキストで評価されます (i) コピーするソースノードセットをロケート、(ii) 削除するノードと追加するノードをロケート。

サブツリー全体の置換

下のスクリーンショットは、(ソース) ノードから(ターゲット) ノードへノードセット全体を追加するために必要な設定を表示しています。 すべての子孫要素 of the ソースノード**$XML1/office/usa** の全ての子要素は、属性と共にターゲットノード**$XML2/offices/office[@id="usa"]**に追加されます。



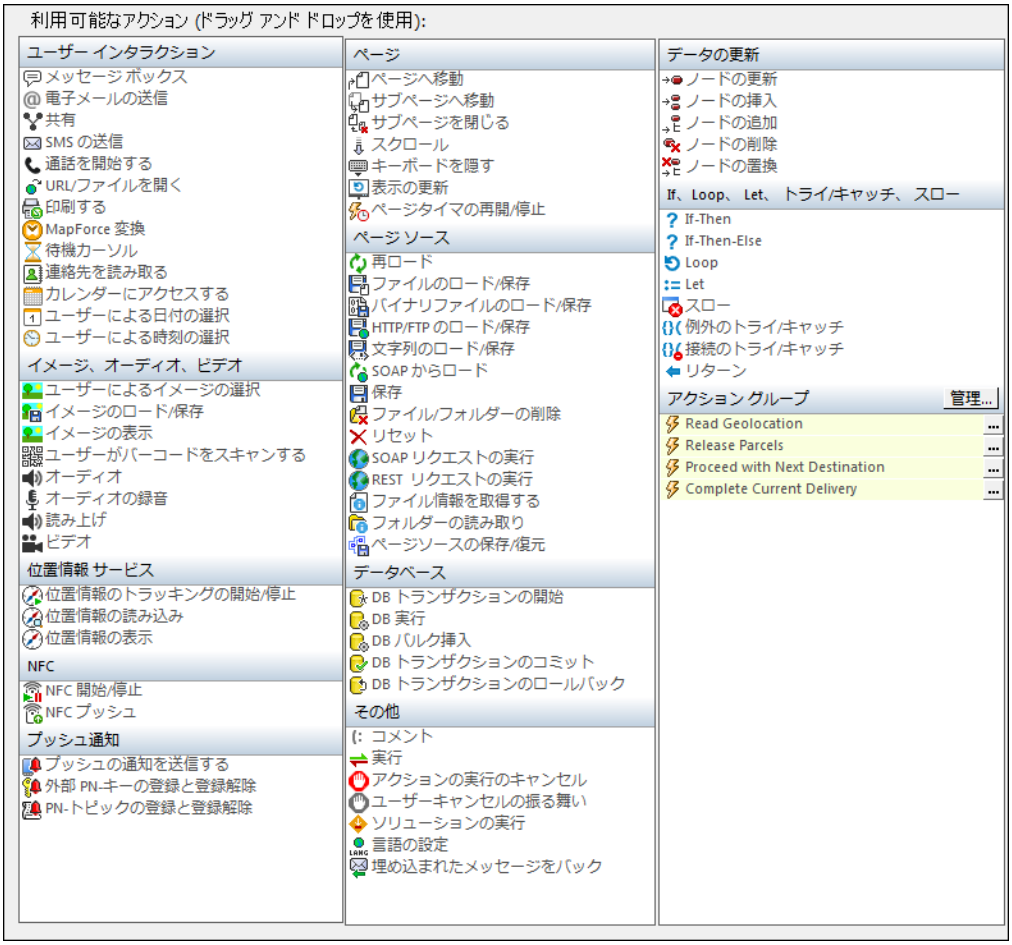
アクションのシーケンスは、以下のとおりです:

1. サブノードXPath 式 `*, @*` は、ターゲットノード**$XML2/offices/office[@id="usa"]** の全ての子要素をロケートし、**削除します。**
2. サブノードXPath 式 `*, @*` は、ソースノード**$XML1/office/usa** のすべての子孫要素をロケートし、単一のノードセットとして、ターゲットノード**$XML2/offices/office[@id="usa"]**のボディにコピーします。ターゲットノードの子要素の全ては削除され、新規のサブノードがターゲットノードの前の子孫を置き換えます。

10.11 If、Loop、Let、トライ /キャッチ、スロー

If、Loop のアクションダイアログの条件グループで次のアクションを使用することができます(下のスクリーンショット):

- If-Then
- If-Then-Else
- Loop
- Let
- スロー
- 例外のトライ/キャッチ
- サーバー接続エラーのトライ/キャッチ
- リターン



このグループのアクションはページ イベントとコントロール イベントに使用することができます。アクション ダイアログ(上のスクリーンショット) に素早くアクセスするには、ページまたはコントロールを右クリックして、ページ/コントロール アクション コマンドを選択します。ページ イベントおよびコントロール イベントも参照してください。

If- Then

IF 条件式が true と評価した場合実行されるアクションを設定します。実行されるアクションは THEN 句の子としてドロップされます。

上記の定義の条件は、現在の要素が、Altova の値を持つ company という名前の属性を持つかテストします。true の場合、Altova の Web サイトの URL が開かれます。

If- Then- Else

IF 条件式のtrue とfalse の評価に実行されるアクションを設定します。 true の評価の実行されるアクションは、THEN 句の子としてドロップされます。false の評価の実行されるアクションは、ELSE 句の子としてドロップされます。

上記の定義の条件式は、現在の要素がAltova の値を持つcompany という名前の属性を持つかをテストします。true の場合、Altova のWeb サイトのURL が開かれます。false の場合、ソリューションを終了します。

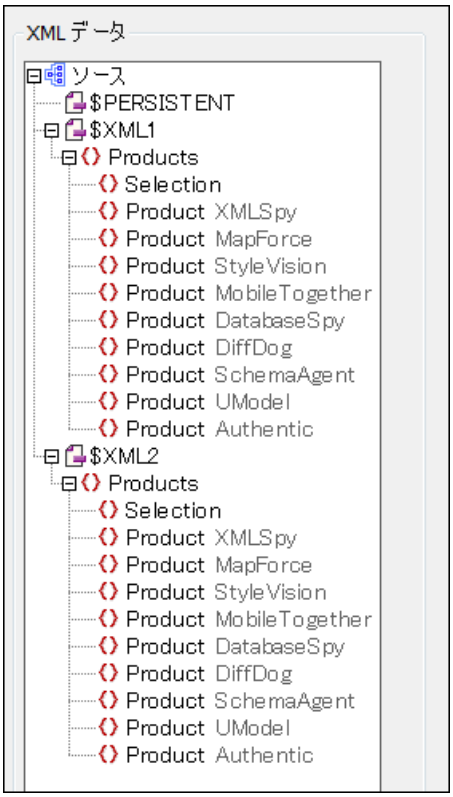
Loop

ループアクション (下のスクリーンショットを参照) は、[For Each] と[以下の条件の場合ループを中断する] 設定を使用して定義するアイテムのシーケンスを反復します。ループ内では、各反復中に実行されるアクションのセットを定義することができます。例えば、下のスクリーンショットの各反復では、[ノードの追加] アクションが実行されます。

[For Each] と[ループを中断する] 設定のXPath式により反復されるループのシーケンスは定義されます。注意する重要なポイントは以下の通りです。

- *For Each:* XPath 式 (例えば `1` から `7`) 内で名前を付けられた、または、) XML ツリー内のツリー要素から取得されたシーケンスであることができます。(例えば `$XML1/Products/Product` は`$XML1` ツリー内の全ての`Product` 要素を選択します。ループの中断設定がなされている場合、すべての反復が完了すると、ループを終了します。
- *Loop 変数:* ループ変数は、現在反復されているシーケンスのアイテムを持つ変数です。ループ変数は、サインの後の最初のクリックの後入力される名前で識別されます。上のスクリーンショットでは、ループ変数の名前は**`MyLoop`** です。この変数は、名前 (**`$MyLoop`**) の前に**`$`** サインを持ち、他のXPath 変数と同様参照されます。変数は、ループ内が対象範囲であり、これは、ループの範囲外のXPath 式内の変数を参照することができないことを意味します。上のスクリーンショットでは、ループ変数は、ノードへ追加内の[新しいノード] 設定で内で参照されています。これは、ノードへ追加アクションがループ内で作成されており、変数はこのポイントではスコープの範囲内であり、妥当な参照です。
- *ループを中断する* このXPath 式は、各反復の前に評価されます。式が`true()` を評価する場合、ループが実行されます。上のスクリーンショットでは、XPath 式 `Product=''` では、最初の空の`Product` 要素が発生した場合、ループが実行されます。(コンテキストノードは`Products` です。ですから、相対ロケーターパスは有効です。) [ループを中断する] 設定は、上のスクリーンショットで表示されるように、[For Each] 設定と共に使用することができます。または、個別に設定することができます。
- ***重要なポイント:*** ループの実行中、反復されるシーケンスへの変更は、許可されていません。上のスクリーンショットでは、ノードへ追加 アクションが他のXML ツリー (`$XML2`) に追加されるとともに、新しいノードが追加されています。反復されているシーケンスは、`$XML1` ツリー内にあり、変更することはできません。しかしながら、XPath の理論に沿った反復するシーケンスを変更する洗練された方法があります。ノードを直接反復するのではなく、ノードシーケンスに関連した複数のシーケンスを反復するのです。例えば、上のスクリーンショット内の`Product` ノードのシーケンスを反復する代わりに、ノードシーケンスに関連した数の範囲を反復します。*For Each* 設定のXPath 式は、**`$XML1/Products/Product`** から**`for $i in 1 to count($XML1/Products/Product) return $i`** に変更することができます。これは、反復されているシーケンスの数です。この結果、`Product` ノードはループ内で自由に変更することができます。ループ内の現在の`Product` ノードは、次のXPath 式によりアクセスすることができます: **`$XML1/Products/Product[$i]`** 。

上のスクリーンショットで定義されているアクションは、重複するツリーフラグメントを作成します。ページがロードされると、ループアクションが`$XML1/Products/Product` 要素に対して反復します。各反復の間、現在の`Product` ノードは、`MyLoop` 変数に保管されます。(`$MyLoop` 変数内の) `Product` ノードは、`$XML2/Products` ノードの最後の子として追加されます。ループは、最後の`Product` 要素が`$XML1/Products` から`$XML2/Products`へコピーされるまで継続されます。 *下のスクリーンショットを参照。*


XML データ
ソース
$PERSISTENT
$XML1
Products
Selection
Product XMLSpy
Product MapForce
Product StyleVision
Product MobileTogether
Product DatabaseSpy
Product DiffDog
Product SchemaAgent
Product UModel
Product Authentic
$XML2
Products
Selection
Product XMLSpy
Product MapForce
Product StyleVision
Product MobileTogether
Product DatabaseSpy
Product DiffDog
Product SchemaAgent
Product UModel
Product Authentic

Let

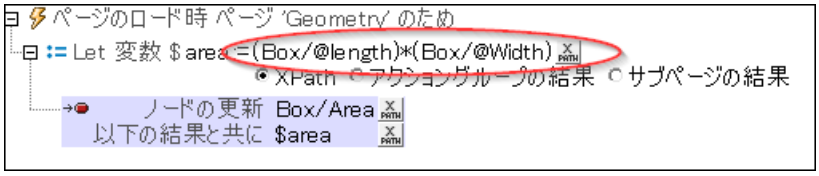
Let アクション(*下のスクリーンショット*) は、以下を使用して変数を定義します: (i) XPath 式、(ii) *アクショングループの結果*または(iii) *サブページの結果*。

- 変数の名前を入力するには、$ サインの右をダブルクリックして、(*下のスクリーンショットでは緑の枠で囲まれている*)変数の名前を入力します .
- XPath 式、*アクショングループの結果*または、サブページの結果を使用して変数の値を設定するかを選択します。
- Let アクションの1つまたは複数の子アクションを定義します。例えば、下のスクリーンショットで表示されているLet アクションは、Let アクション内で定義されている変数の値を使用するノードを更新する ノードの更新 アクションを含みます。

メモ Let アクション内で定義されている変数は、そのLet アクション内でのスコープにあります。これは、Let アクションの子アクションでのみ使用できることを意味します。

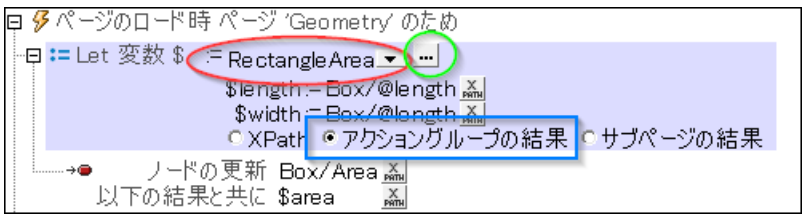
XPath オプション

XPath オプションによりXML ツリーノードを使用して静的な値または動的な値を入力することができます。例えば、下のスクリーンショット内では、(*赤い枠で囲まれているとおり*) 2つのノードが乗算されています。結果の値は、変数 ($area) の値となります。変数はXML ツリーノードのコンテンツを更新するために使用されました。

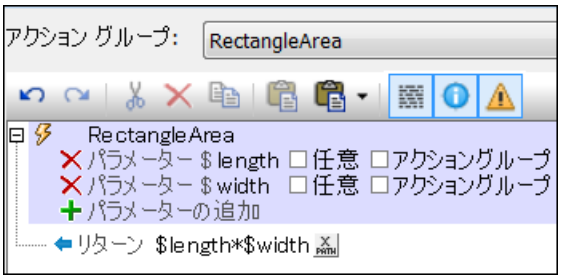


アクショングループの結果 オプション

アクショングループの結果 オプション(*下のスクリーンショット*) は、*アクショングループの結果となる変数の値を設定します。*下のスクリーンショットでは、変数に`$area` という名前を与え、(*のスクリーンショットで赤枠で囲まれている*) `RectangleArea` という名前のアクショングループの結果を値となるように設定しました。(全ての既存のアクショングループは、変数のコンボボックス内での選択のために使用することができます)。アクショングループを編集するには、(緑の枠で囲まれている) **[編集]** ボタンをクリックします。2つのパラメーターの値が動的に定義されているため、これらのXML ツリーノードから値を得ます。



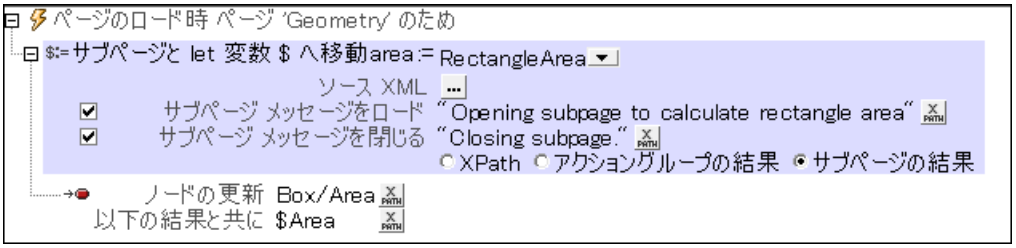
アクショングループは、 リターン アクションを使用して結果を返します (*下のスクリーンショット参照*)。下のスクリーンショットでは、例えば、2つのパラメーター ($length と$width) を定義します、Return アクションでは、2つのパラメーターの値を乗算します。パラメーターの値は、ランタイムでLet アクション内で定義されたXML ツリーノードからLet アクションのパラメーターとして取得されます (*上のスクリーンショット参照*) 。



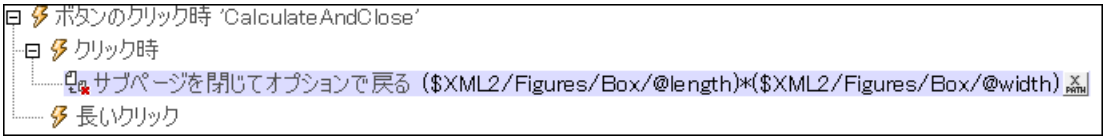
サブページ結果オプション

サブページ結果オプション (*下のスクリーンショット*) は、変数の値をサブページの結果として設定します。これにより他のページで計算が行われることを許可します。サブページが閉じられると、結果はオプションで返されます。この結果は、アクション内で定義された変数の値として使用されるサブページの結果です。例えば、下のスクリーンショットでは、Let アクションは、`$area` という名前の変数を定義し、(*下で赤い枠で囲まれている*) `RectangleArea` という名前のサブページの結果を自身の値として設定します .

サブページの結果オプションのプロパティは、サブページへ移動 アクションと同じです。

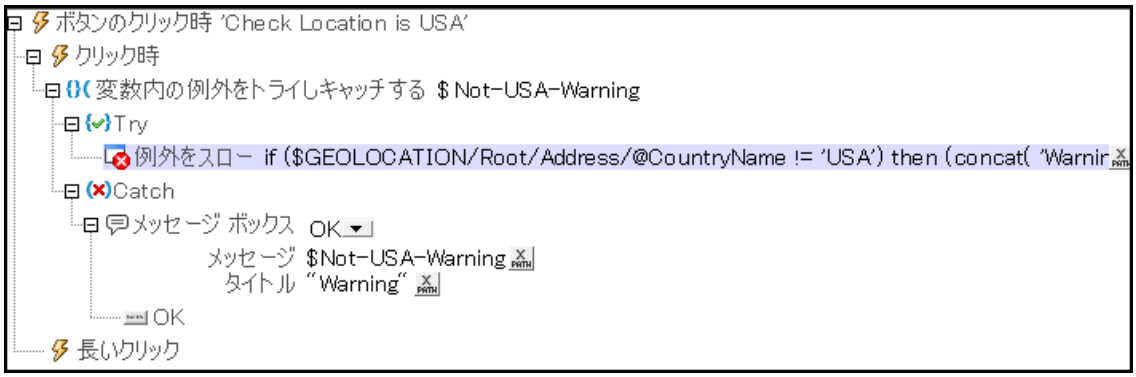


ランタイムでは、Let アクションが実行されると、サブページが開かれ、デザイン内で定義されたとおりに処理されます。サブページは、サブページを閉じる アクションが実行されると閉じられます。このアクションは、XPath 式により計算される戻り値をもっています
(*下のスクリーンショット参照*)。この戻り値はLet アクションにパスされ、Let アクション内で定義された変数の値となります。



スロー

スロー アクションは、トライ/キャッチ アクションの部分で使用されます (下のスクリーンショット参照)。このアクションは XPath 式を評価します。評価の結果がからのシーケンスではない場合、例外がスローされ、例外は トライ/キャッチ アクションの変数に保管されます。下のスクリーンショットでは、この変数は `$Not-USA-Warning` という名前を持ちます。



上のスクリーンショットで表示されている通り、デバイスの位置情報が米国でない場合、例外がスローされます。XPath 式は、以下の通りです:

```
if ($MT_GEOLOCATION/Root/Address/@CountryName != 'USA')
then (concat( 'Warning:Device location is outside the US:',$MT_GEOLOCATION/Root/Address/
@CountryName))
else ()
```

この式は以下のように動作します

- **`if`** 句が **`$MT_GEOLOCATION/Root/Address/@CountryName`** ノードの値が `'USA'` であるか (否か) をチェックします。
- **`then`** 句は、国名が `USA` で**ない**場合処理されます。この句は文字列を生成します。
- **`else`** 句は、国名が `USA` で**ある**場合、処理されます。これは空のシーケンスを生成します。

位置情報の国が `USA` でない場合、条件は `true` になり、式は **`then`** 句により生成された文字列を評価します。この結果は空のシーケンスではないため、例外がスローされ、生成された文字列が トライ/キャッチ 変数 `$Not-USA-Warning` 内に保管されます。この文字列はメッセージボックスに表示されます (上のスクリーンショット参照)。

位置情報の国が `USA` の場合、条件は `false` になり、(**`else`** 句により生成された) 式は空のシーケンスを評価します。結果が空のシーケンスのため、例外はスローされません。ですから、トライ/キャッチ アクションのキャッチ 部分は実行されません。

メモ シーケンスが文字列アイテム `('')` を含む場合、シーケンスは空では**ありません**。

チュートリアル位置情報の共有 はトライ/キャッチ とスロー アクションの使用法について説明しています。

例外の トライ /キャッチ

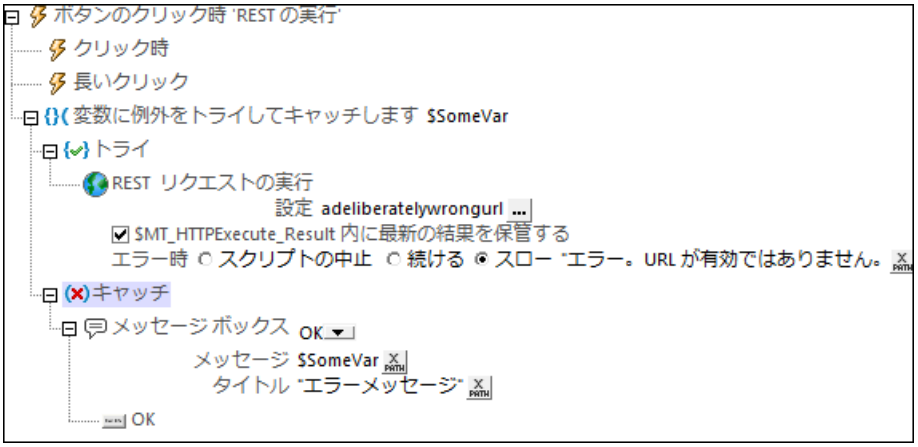
例外のトライ /キャッチ アクションには2つのパートがあります (*下のスクリーンショットでハイライトされています*):

- *トライ*: トライするアクションを定義します。
  - 条件 は スロー アクションのXPath 式内で定義されています。(*スロー アクション の使用方法のサンプルとして位置情報の共有 チュートリアル を参照してください*)。
  - アクション実行中に、アクション が定義されている場合、(下のスクリーンショット内でREST リクエストの実行 アクションなど) およびエラーが検出された場合、以下の **オプション** から選択することができます: (i) アクションの中断、(ii) エラーを無視して継続、または(iii) トライ /キャッチ 変数に保管された例外をスローする。これが *スロー* **オプション** です。(2番目のオプションを) 継続することを選択しても スロー アクションを使用してエラーをスローすることができます。
  - (条件のために定義されている) スロー アクション または(アクションのために定義されている) スロー オプション は、トライ /キャッチ アクションの変数内に保管されている例外をスローします。
- *キャッチ*: 例外がスローされた場合のみ実行されるアクションを定義します (*下のスクリーンショットの詳細を参照してください*)。アクションが定義されていない場合、トライ /キャッチ アクションに続くアクションが実行されます。

メモ 例外は、以下の2つの方法でスローされます:(条件のために定義される) スロー アクション または(アクションのために定義される) スロー オプション を使用。

メモ トライ /キャッチ アクションの変数内で例外がスローされると、キャッチ の部分が実行されます。

メモ (*スロー アクション /オプション* で) アクションの トライ の部分内で例外がスローされない場合、キャッチ の部分は実行**されません**。



上のスクリーンショットの トライ /キャッチ アクション内では、以下を行いました

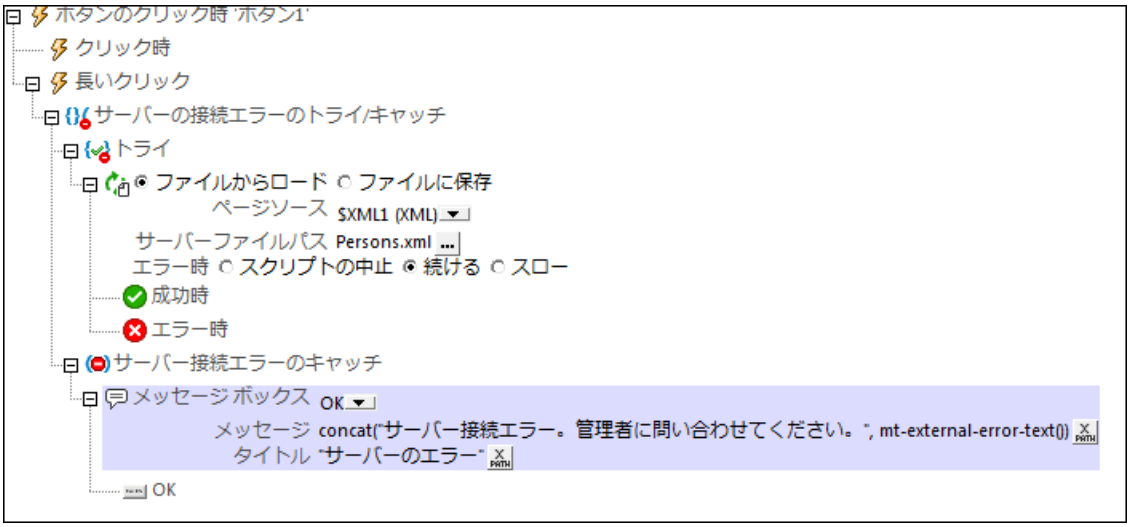
1. トライ /キャッチ 変数に `$SomeVar` (`$` シンボルの右をダブルクリックして名前を入力することにより) の名前を与えました。
2. アクションの トライ の部分で、REST リクエストの実行を設定しました。
3. アクションの トライ の部分で、REST リクエストの実行のエラー時のサブアクションのために スロー オプションを選択し、オプションのXPath 式として例外メッセージを入力しました。この結果、エラーが検出されると例外がスローされ、`$SomeVar` 変数に例外メッセージが保管されます。
4. アクションの キャッチ 部分で、変数に保管されている `$SomeVar` メッセージを表示するメッセージボックス アクションを定義しました。

メモ 上記の スロー **オプション** に加え、スロー **アクション** も使用する事ができます。エラー時のサブアクションの *スロー* オプションを使用する代わりに、(*上のステップ2 で説明されているとおり*) 継続オプションを使用して、継続オプションのエラー時のサブアクションに スロー アクションを挿入することができます。

チュートリアル位置情報の共有 はトライ/キャッチ とスロー アクションの使用法について説明しています。

サーバー接続エラーの トライ /キャッチ

このアクションのトライの部分は、サーバーに接続するように試みるアクションで使用されます(下のスクリーンショット参照) 。接続が作成されない場合、アクションのキャッチ部分がトリガーされます。接続が作成されても、(ファイルが存在しないなどの)例外が存在する場合、これらの例外はオプション スロー オプションを使用して例外を生成することができます (下のスクリーンショット参照)。

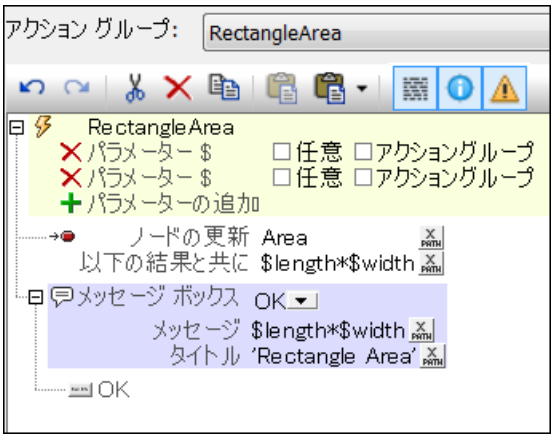


メモ MobileTogether XPath 拡張関数 `mt-external-error-text()` をサーバーエラーの *キャッチ* の部分内で使用し、サーバー接続エラーに関する情報を表示します。

メモ サーバー接続エラーが発生すると、存在する次のアクションの最初のアクションがトリガーされます: (i) サーバー接続エラーのトライ /キャッチ アクション、(ii) **サーバー接続のエラー** イベントのためのアクション(iii) エラーに関する MobileTogether メッセージとワークフローの再開。

リターン

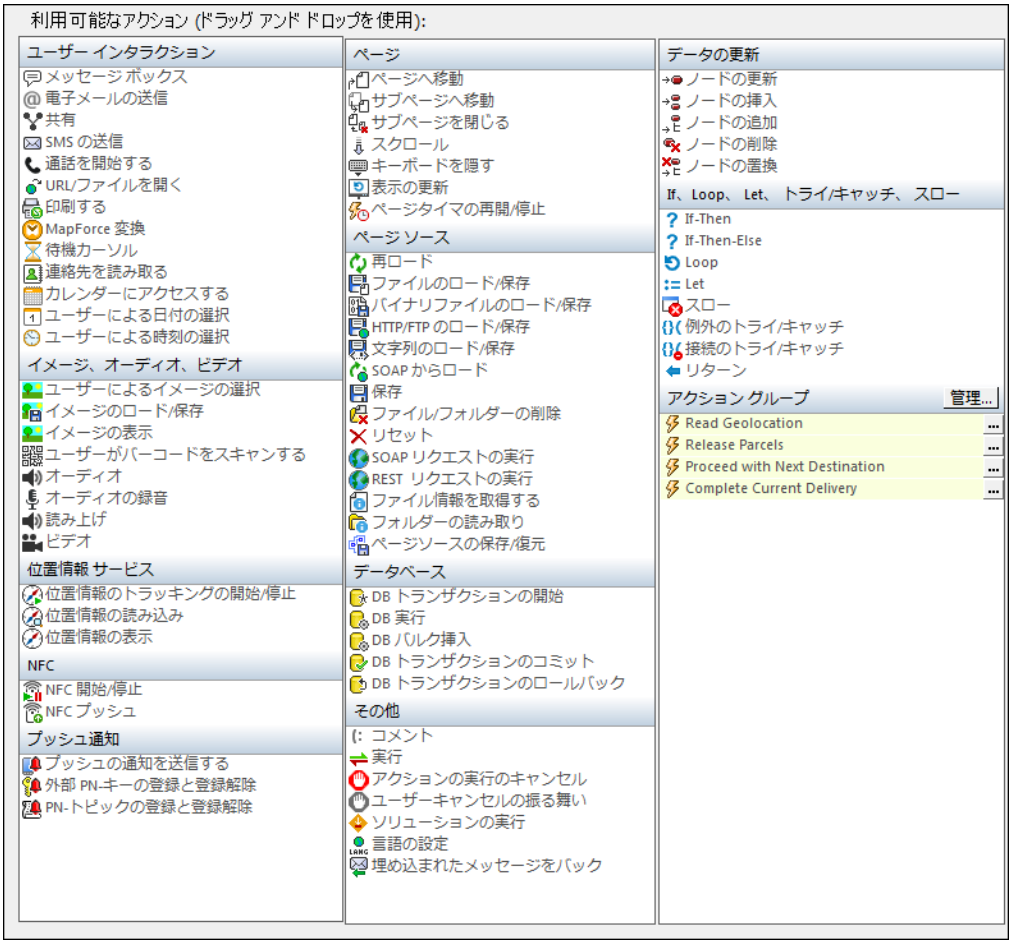
リターンアクション内でXPath 式は、アクショングループにより返された値を定義します。この値は、アクショングループの結果と呼ばれ、Let アクション内で定義された変数の値を設定するために使用されます。下のスクリーンショットでは、例えば、リターンアクションは、`RectangleArea` という名前のアクショングループにより返された値を定義します。Return アクションのXPath 式は、アクショングループ内で宣言されたパラメーターを含むことができます。パラメーターの値はアクショングループを呼び出す Let アクションによりランタイムに与えられます リターン アクションの使用方法に関しての詳細は *アクショングループを使用して変数の値を設定する* のセクションを参照してください。



10.12 アクショングループ

アクショングループは実行されるアクションのシーケンスを定義します。ワークフロー内の異なる箇所で同じシーケンスのアクションが実行される場合、アクションのシーケンスを含むアクショングループを作成してワークフローの異なる箇所で実行するほうが効率的です。アクショングループをパラメーター化することもできます。

プロジェクト全体のためにアクショングループは作成され、作成することのできるアクショングループの数に制限はありません。アクショングループの作成後、すべてのページイベントまたはプロジェクトのコントロールイベントで使用することができます。イベント内でのアクショングループの使用回数に制限はありません。



このセクションは以下について説明しています:

- アクショングループの追加、削除、および編集方法
- アクションの再使用のためのアクショングループの使用方法
- アクショングループ内でのパラメーターの使用方法
- アクショングループをアクショングループ内でパラメーターの値として使用する方法
- Let アクションを使用して定義した変数の値を与えるためのアクショングループの使用方法

このセクションは、アクショングループの作成と編集とアクショングループの使用方法について説明しています。更に、Let アクションを使用し定義した変数の値を提供するためにアクショングループを使用する方法の詳細があります。

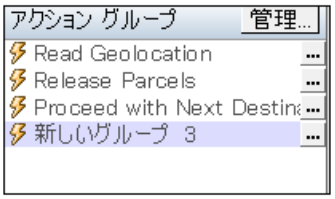
アクショングループの管理

アクショングループは、プロジェクト全体のために作成することができ、プロジェクトの全てのページイベントとコントロールイベントのために使用することができます。プロジェクトの全てのアクショングループは[アクション] ダイアログ内のアクショングループペイン内に表示され、下で説明される[グループアクションの管理] ダイアログ内で管理されます。

[グループアクションの管理] ダイアログ ダイアログのアクセス方法

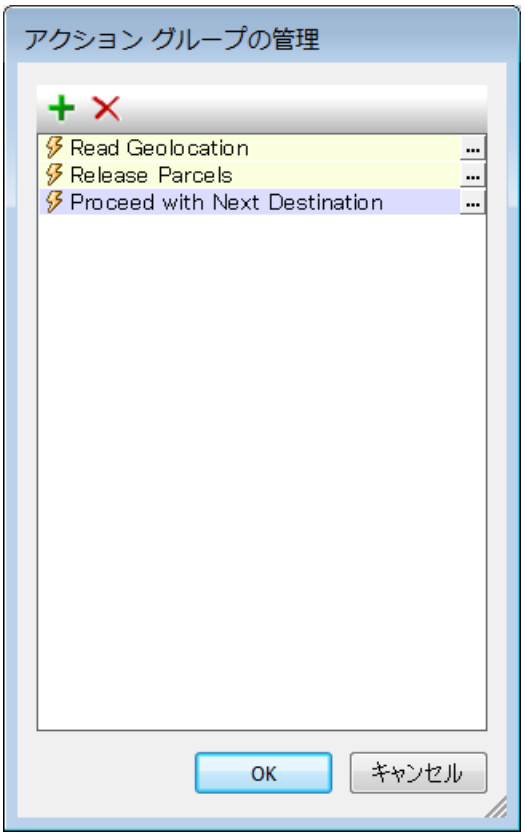
[グループアクションの管理] ダイアログにアクセスするには、以下を行います:

1. [アクション] ダイアログを開きます: (i) コントロールイベント または ページイベント から、(ii) **[プロジェクト] アクショングループ** をクリックして。
2. [アクション] ダイアログ 内で、アクショングループ ペイン内の**[管理]** をクリックします (*下のスクリーンショット参照*)。



[グループアクションの管理] ダイアログ

[グループアクションの管理] ダイアログ (*下のスクリーンショット*) でアクショングループを追加または削除することができます。



アクショングループの追加
ダイアログのツールバーの**[追加]** をクリックします。 [アクショングループ] ペイン内のグループのリストに新しいアクショングループが追加されます(*上のスクリーンショット参照*)。デフォルトの名前をダブルクリックして、名前を編集して、**[OK]** をクリックします。

アクショングループの削除
削除するアクショングループを選択して、ダイアログのツールバー内の**[削除]** をクリックします

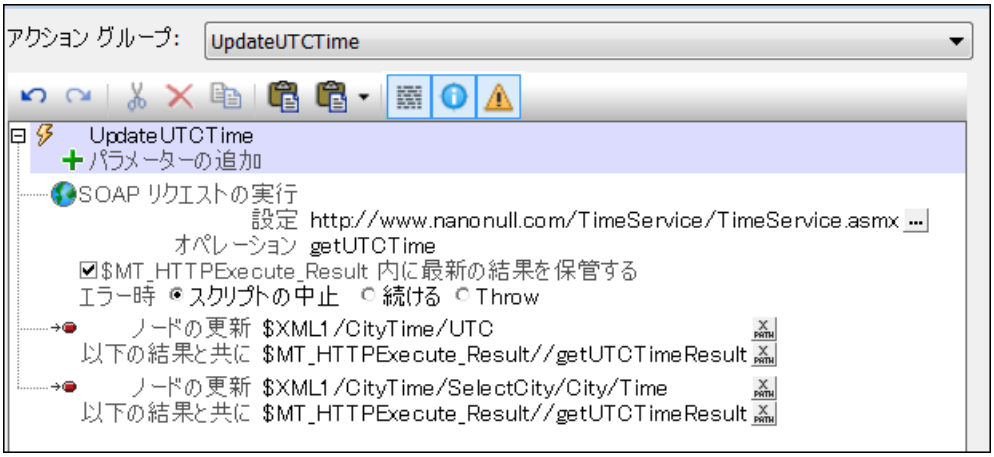
アクショングループにアクセスし編集する
アクショングループには、以下の方法でアクセスすることができます:

- 編集する場合は、[アクショングループの管理] ダイアログ内の**[編集]** ボタンをクリックします。
- 編集するには、[アクション] ダイアログのアクショングループペインで(*下のスクリーンショット*) 編集するアクショングループの**[編集]** ボタンをクリックします。または、アクショングループエントリをダブルクリックすることができます。

アクションを再使用するためのアクショングループ

アクショングループは、アクションのセットを再使用するために使用することができます。 この目的のためにアクショングループを設定するには、以下を行います:

1. 編集するためにアクショングループにアクセスします。
2. アクション および/または アクショングループを使用することのできるアクションペインからドラッグアンドドロップします。例えば、下のスクリーンショットでは、*UpdateUTCTime* アクショングループのために3つのアクションが定義されています。



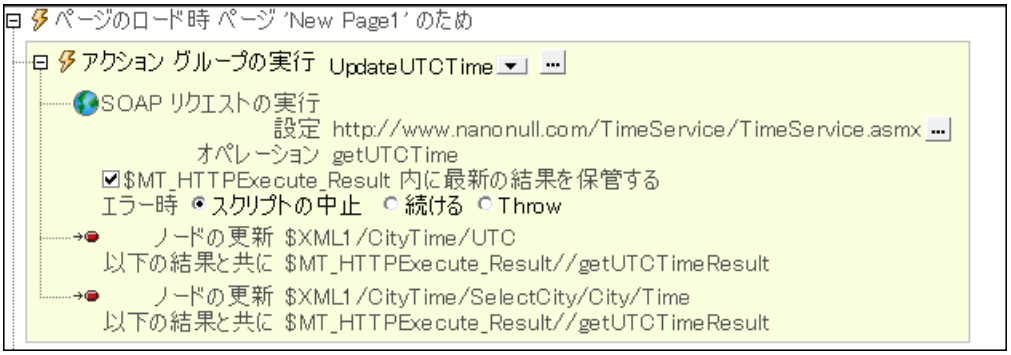
3. **[OK]** をクリックして完了します。

アクション内で定義されたアクションのセットは、ワークフロー内の全ての箇所で使用することができます。

メモ　アクショングループの定義を編集する場合は、アクショングループ内で行います: ウィンドウの上の部分のコンボボックスから編集するアクショングループを選択します。

アクショングループの使用

アクショングループは他のアクションと同様に使用することができます。(アクショングループペインから) ドラッグして、イベントのアクションの定義にドロップします。ドロップされると、アクショングループはアクショングループの実行アクションとして表示されます。下のスクリーンショットは展開されたフォームでアクショングループの実行アクションを表示しています。 このアクショングループ内のアクションは、ページのロード時イベントがトリガーされた時に実行されます。



以下の点に注意してください:

- イベントがトリガーされると、アクショングループ内の全てのアクションが指定された順序で実行されます。

- アクショングループは、ワークフローの複数の時点で同じ組み合わせのアクションを実行するために使用することができます。
- アクショングループの実行 アクションにはプロジェクト内で定義された全てのアクショングループを選択することのできるコンボボックスがあります。
- アクショングループの **[編集]** ボタンをクリックして、現在選択されているアクショングループを編集することができます。

パラメーターを持つアクショングループ

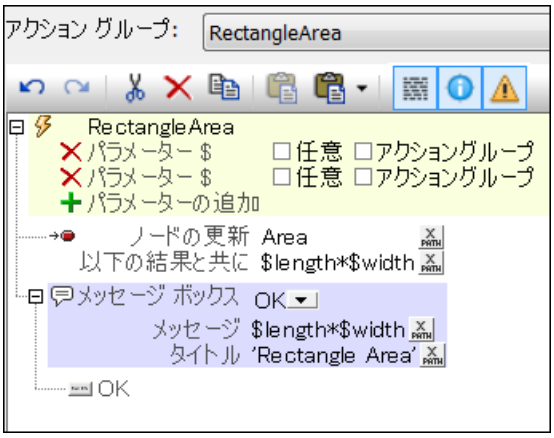
アクショングループ内のパラメーターは、以下の通り動作します:

- アクショングループ内でパラメーターを宣言し、これらのパラメーターを使用するアクションを(アクショングループ内で)定義します。
- アクショングループのアクションの実行によりアクショングループが呼び出されると、パラメーターの値はパラメーターにパスされます。

アクショングループ内のパラメーターを宣言し、これらのパラメーターを使用するアクションを定義する

アクショングループ内で、必要とする結果を生成するために必要なパラメーターを宣言します。これは、**[パラメーターの追加]** アイコンをクリックして行います(*下のスクリーンショット参照*)。パラメーターが追加された後、パラメーターの **$** シンボルの右側をダブルクリックし、パラメーターの名前を入力します。

下のスクリーンショットでは、RectangleArea という名前のアクショングループが **$length** と **$width** という名前の2つのパラメーターを宣言しています。パラメーターは任意としてマークされていません。これは、アクショングループがランタイムで呼び出されると、これらのパラメーターの値を提出する必要があり、それ以外の場合はエラーが報告されます。



アクショングループは2つのアクションより構成されています:

- Area ノードを更新するための値を生成する2つのパラメーター **$length** と **$width** を乗算するノードの更新 アクション。
- 2つのパラメーター **$length** と **$width**. を乗算し取得された値を含むメッセージボックスを表示するメッセージボックス アクション。

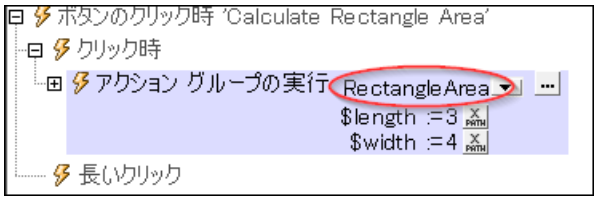
アクショングループ内でパラメーターを宣言し、これらのパラメータを使用するアクションを定義することに注意してください。パラメーターの値は、アクショングループのアクションの実行により、ランタイムで提供されます。

パラメーターの値を提供する

アクショングループのパラメーターの値は、アクショングループのアクションの実行により提供されます。アクショングループ アクションの実行は、イベントのために定義され、イベントがトリガーされると処理されます。ですから、与えられたイベントがトリガーされると、パラメーターの値がアクショングループにパスされることができます。アクショングループは、これらの値を定義されたアクションを実行するために使用します。

アクショングループ アクションの実行を作成するには、([アクショングループ] ペインから) 対応するアクショングループをイベント

のアクションの定義にドラッグします。ドロップされる際に、アクショングループは実行するアクショングループとして表示されます。下のスクリーンショットでは、ボタンのクリック時イベントのアクショングループアクションのための実行を定義しました。クリック時イベントの下の`RectangleArea` アクショングループをドラッグして行いました。現在選択されている実行するアクショングループは、(下のスクリーンショットで赤枠で囲まれている) アクショングループの実行コンボボックスにより変更されることができます
.



パラメーターが現在選択されているアクショングループ内でパラメーターが宣言されている場合、これらのパラメーターはアクショングループアクションの実行内で表示されています。これはランタイムにパラメーターに値をパスする箇所です。上のスクリーンショットでは、`RectangleArea` アクショングループには次の二つのパラメータがあることが表示されています: **`$length`** と **`$width`**。簡単な統計の値 (**3** と**4**) を、2つのパラメーターの値を生成するために使用したXPath 式に入力しました。XML ツリーソースから動的に、または、複雑なXPath 計算を指定して値を取得することもできます。アクショングループ内で、パラメーターがオプションではなく、必須として宣言されている場合、赤い文字で表示され、値は与えられません。

ランタイムでは、以下が発生します:

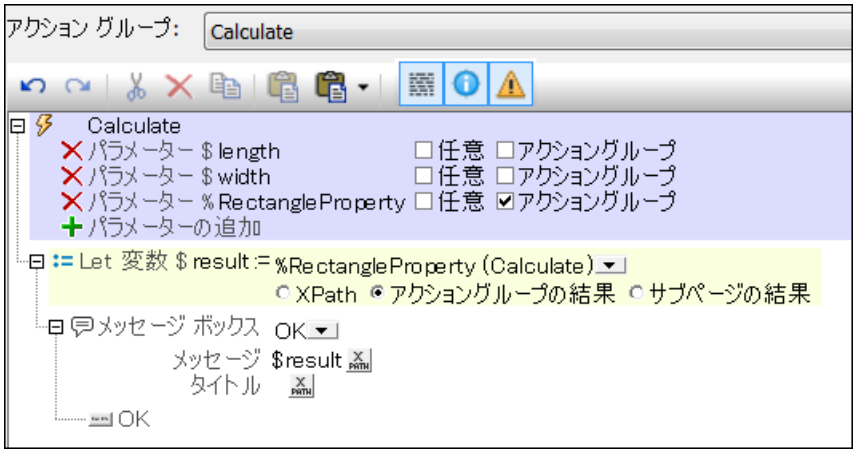
1. イベントがトリガーされると、(アクショングループアクションの実行で指定されているとおり) パラメーターの値は、アクショングループ内の対応するパラメーターにパスされます。
2. アクショングループのアクションが処理されます。これらがアクショングループの宣言されたパラメーターを使用する箇所で、与えられたパラメーターの値が代替されます。

アクショングループパラメーターを持つアクショングループ

アクショングループは、以下の2つの種類のパラメーターを使用することができます:

- パラメーターを持つアクショングループ内で説明されている*単純型パラメーター*。MobileTogether Designer 内では、名前の前のドルシンボル**$** で示されています。
- アクショングループを自身の値として取る*アクショングループパラメーター*。MobileTogether Designer 内では、名前の前のパーセントシンボル**%** で示されています。

下のスクリーンショットでは、**`$length`** と**`$width`** は、単純型パラメーターですが、**`%RectangleProperty`** は、アクショングループパラメーターです。アクショングループパラメーターとしてパラメーターを作成するには、最初に単純型パラメーターとして作成し、アクショングループチェックボックスをチェックします (*下のスクリーンショット参照*)。



アクショングループ パラメーターの動作方法

アクショングループ パラメーターは、自身の値としてパラメーターを宣言しないパラメーターを取ります。これにより、呼び出されたパラメーターはMobileTogether タスクを実行する関数のように扱われます。アクショングループ パラメーターは広い意味で、単純型パラメーターと同じように動作します。

- アクショングループ パラメーターをパラメーター内で宣言します。含まれるパラメーターはこれらのアクショングループパラメーターを使用するアクションを定義することができます。または、含まれるパラメーターをパラメーターの実行アクションを使用して直接アクショングループ パラメーターを処理することもできます (*アクショングループ パラメーターの処理*のセクション参照)。
- アクショングループ パラメーターの値は、含まれるパラメーターを呼び出すパラメーターの実行アクションにより提供されます。許可される値は(プロジェクト内の) パラメーターを持たないパラメータです。

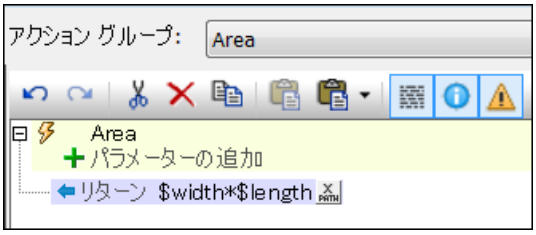
使用方法のサンプル

アクショングループ パラメーターの使用方法のサンプルが以下に表示されています。三角形の3つのプロパティ(面積、対角線、周囲) を計算し表示するために作成されました:

- **`Calculate`** (*上のスクリーンショット参照*): 2つの単純型のパラメーター (**`$length`** および**`$width`**) とアクショングループ パラメーター (**`%RectangleProperty`**) を宣言します。アクショングループ パラメーターは、パラメータが定義されていない3つのパラメータの値をとることができます。**`Calculate`** パラメーター内では、Let アクションが定義されています。このアクションは、**`$result`** と呼ばれる、変数を、**`%RectangleProperty`** パラメーターの値

であるパラメーターのパラメーターの結果をとるために定義します。前のアクションで定義された**$result** パラメーターの実行アクションの値を表示するメッセージボックスアクションが次です。

- **Area** (下のスクリーンショット): このパラメーターは、2つの単純型のパラメーター**$length** と**$width** の結果を含む単一のReturn アクションを含んでいます。このパラメーターは、パラメーターを含みません。ですから、**%RectangleProperty** の値となることができます。**%RectangleProperty** が処理されると、**Calculate** パラメーター内の**$result** 変数は、**Area** パラメーターの戻り値を受け取ります。



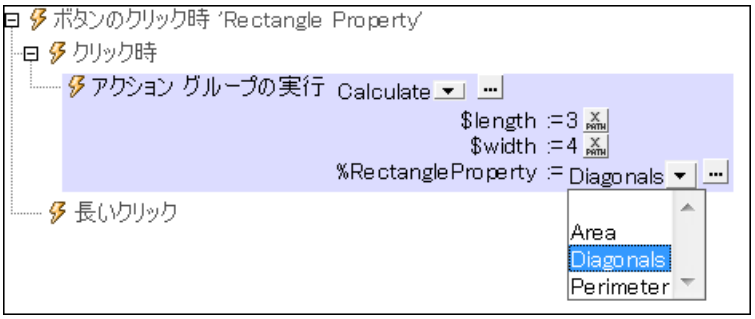
- **Diagonals**: このパラメーターは、**Area** パラメーターに類似しています。正方形の対角線の長さを返す、単一のリターンアクションを含んでいます。(2つの単純型のパラメーター**$length** と**$width**の正方形の合計の平方根。*XPath:* math:sqrt($width*$width + $length*$length))。このパラメーターが**%RectangleProperty** の値である場合、返される値は、**Calculate** パラメーター内の**$result** 変数の値として設定されます。
- **Perimeter**: このパラメーターは、**Area** と**Diagonals** パラメーターに類似しています。これは、長方形の周囲 (*XPath:* 2*$width + 2*$length) を返す単一のリターンアクションを含んでいます。このパラメーターが**%RectangleProperty** の値の場合、**Calculate** パラメーター内の**$result** 変数の値が返される値として設定されます。

上記の4つのパラメーターの詳細に関して、以下の点に注意してください:

- **Calculate** パラメーターは、3つのパラメーターを含みます: 2つの単純型のパラメーターと1つのアクショングループパラメーター。
- 他の3つのパラメーター(**Area**、**Diagonals**、**Perimeter**) は、パラメーターを宣言しません。
- 3つのパラメーター(**Area**、**Diagonals**、**Perimeter**) のそれぞれは、**Calculate** 内で宣言されている単純型のパラメーターから値を計算するリターンアクションを持ちます。返された値は、これらのパラメーターのパラメーターの結果です。
- 3つのパラメーター(**Area**、**Diagonals**、**Perimeter**) のそれぞれは、アクショングループパラメーター**%RectangleProperty** の値として設定することができ、処理される際、対応するパラメーターの結果を返します。
- **Calculate** パラメーター内で処理されるアクショングループパラメーター**%RectangleProperty** 。メッセージボックス内で表示される**$result** 変数の値を提供します。(上の**Calculate** パラメーター スクリーンショット参照)。

アクショングループパラメーターの値の提供

パラメーターは、**Calculate** パラメーター内で宣言されているため、これらの値は(パラメーターの実行アクションを使用して) **Calculate** パラメーターが呼び出されるとパスされます。ですから、例えば、ボタンのクリック時 イベントを下のスクリーンショットに表示されるように定義することができます。



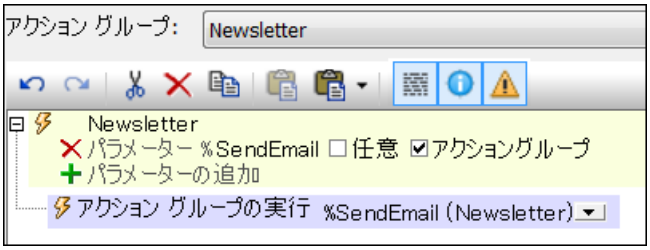
(**Calculate** パラメーターをパラメーターペインにドラッグすることにより) パラメーターの実行アクションをイベントペイン内に作成します。**Calculate** パラメーターのパラメーターは、自動的に表示されます。**%RectangleProperty** アクショングループパラメーターに関しては、パラメーターの値とするパラメーターを選択します(*上のスクリーンショット参照*)。ですから、**Calculate** パラメーターが実行されると、呼び出しで指定されている3つのパラメーターの値にパスされます。**Calculate** パラメーター内のLet アクションが実行されると、(*下のスクリーンショット参照*)、**$result** 変数が、**%RectangleProperty** の値として指定されているパラメーターのパラメーターの結果として設定されます。

メモ　上のサンプルは、意図的に単純になるように作成されており、その目的はアクショングループパラメータの背景にあるメカニズムを説明することです。しかし、アクショングループのパラメーターは動的なコンテンツを使用し、複雑なアクションを実行するために最適です。

アクショングループ パラメーターの処理

アクショングループパラメーターは、パラメーターを自身の値として取ります。このパラメーターはパラメーター値として2つの有効な方法で処理することができます:

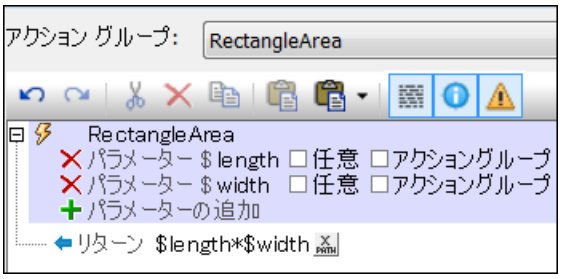
- Let アクションを使用して定義される変数の値を設定するために使用することのできる? パラメーターの結果 ? を提供します。この使用方法は上のサンプルで説明されています。
- ノードの更新、電子メールの送信、パラメーターの結果が存在しない場合など、特定のMobileTogether タスクを実行することができます。これは、含まれるアクショングループ内でをアクショングループパラメーターのためのアクショングループの実行アクションを定義することで指定されます(*下のスクリーンショット参照*)。これは、アクショングループのパラメーターをアクショングループペインからイベントの定義にドラッグすることで行うことができます。



両方のアクションのセットを組み合わせることも可能です。

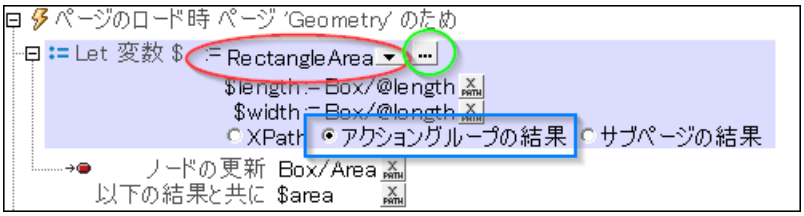
変数とアクショングループの結果

アクショングループに値であるアクショングループの結果を返すようにセットアップすることができます。(例えば、下のスクリーンショットでは、2つのパラメーターを宣言し、リターンアクション内で、乗算してアクショングループの結果を作成しました)。 Let アクションが定義されると、アクションぐうrぷの結果に変数を割り当て、Let アクションの子アクション内で変数を使用しすることができます。



これを行うためのステップは、以下のとおりです:

1. アクショングループ内で、必要とされる結果を生成するために必要なパラメーターを宣言します。**[パラメーターの追加]** アイコンをクリックしてこれを行います (*上のスクリーンショット参照*)。新しいパラメーターが追加された後、パラメーターの**$** シンボルをダブルクリックして、パラメーターの名前を入力します。上のスクリーンショットでは、**$length** と**$width** という名前の2つのパラメーターを宣言するアクショングループにRectangleAreaという名前をつけました。以下に注意してください (i) パラメーターは宣言されましたが、値は定義されていません。(ii) パラメーターはアクショングループのスコープ内にあり、外部で使用することはできません。パラメータが任意で定義されている場合、変数の定義内で使用されていなくてもエラーにはなりません (*下のステップ3 参照*)。
2. リターンアクションを追加します。XPath 式を使用して結果を定義します。アクショングループの結果をLet アクションが使用することができます。上のスクリーンショット内では、**$length** と**$width** パラメーターの値を乗算する式を定義しました。リターンアクションは、アクショングループ内に存在することに注意してください。このため、パラメーターもスコープ内にあります。
3. Let アクションにより定義された変数がアクショングループの値を持つように宣言するには、Let アクションを以下のように定義します (*下のスクリーンショット参照*): (i) 変数の**$** シンボルの右側をダブルクリックして、変数の名前を入力します。(ii) (*下のスクリーンショットで青い枠で囲まれている*) アクショングループの結果ラジオボタンを選択します。(iii) (*赤い枠で囲まれている*) コンボボックスの上のから変数の値として使用するアクショングループを選択します。(iv) 選択されたアクショングループのパラメーターがリストされています (必須の場合は赤、任意の場合は黒で表示されます)。値を生成するXPath 式を入力します。ランタイムでは、これらの値はアクショングループのパラメーターにパスされ、アクショングループの結果を計算するために使用されます。



上のスクリーンショットでは、 Let アクション内で、変数に**$area** という名前を与え、RectangleArea アクショングループに値を与える変数を選択しました。(Let アクション内の) パラメーターの値には、2つのXML ツリーノードを選択し、(RectangleArea アクショングループ内で宣言されている) **$length** と**$width** パラメーターパラメーターの値を与えるようにしました。Let アクションが実行されると、パラメーターの値は、Return アクションがパラメータの値を計算に使用する箇所で、アクショングループにパスされます。結果は返され、Let アクション内で定義された変数の値になります。上のサンプルでは、**$length** と**$width** パラメーターの値は、互いを乗算し、 Let アクションの**$area** 変数の結果を返すRectangleArea アクショングループにパスされます。
4. Let アクション内で定義されている変数は、Let アクションの子アクションとして使用することができます。例えば、**$area** 変数 を使用して**Box/Area** ノードを更新することができます。

詳細に関しては、Let アクションを参照してください。

チャプター 11

デザインオブジェクト/機能

11 デザインオブジェクト/機能

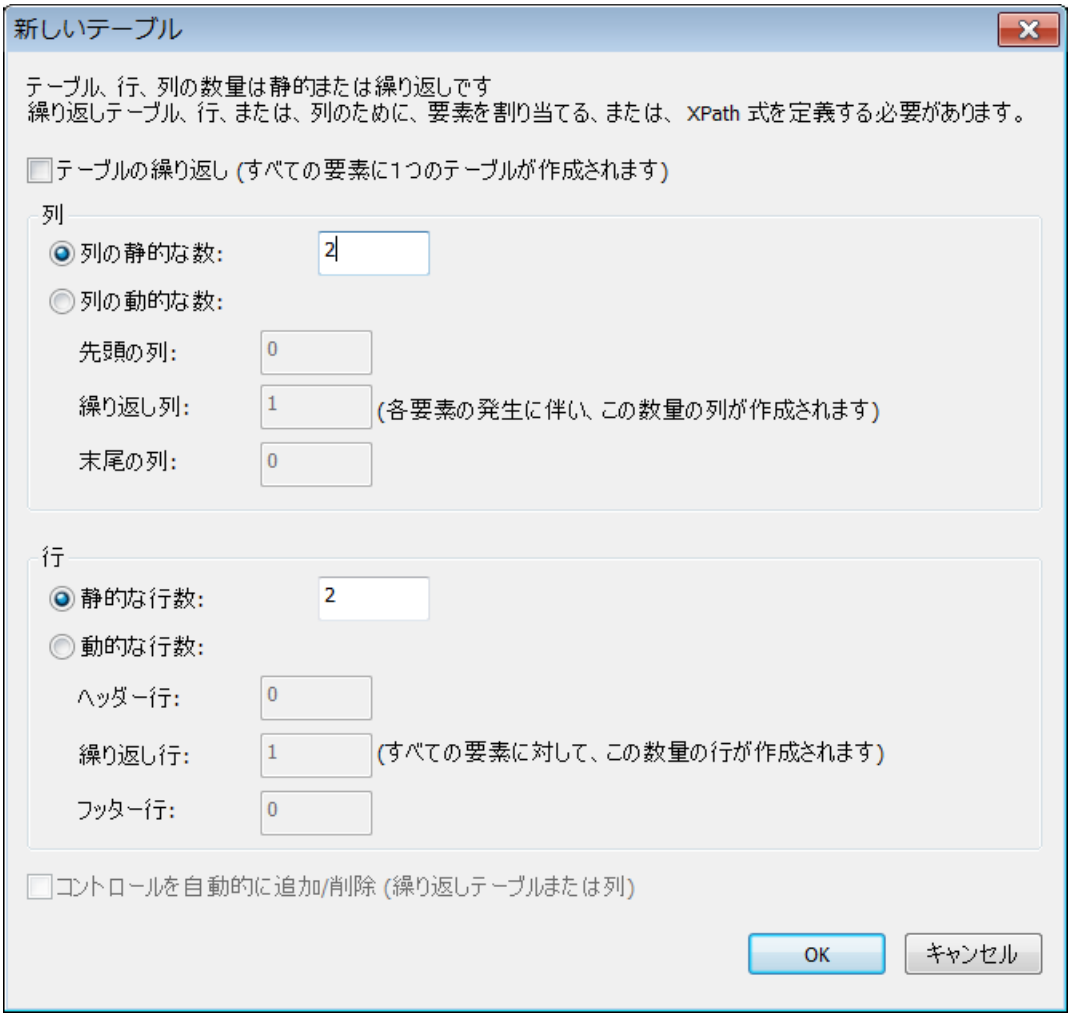
このセクションでは、ページに配置することのできるデザインオブジェクトの範囲について説明されています。コントロールの大半はページおよび、または、プロジェクトのために定義することのできる高度な機能としてページに配置することができます。

- テーブル: デザイン内で使用することのできる異なる種類のテーブルの使用方法について説明しています。
- イメージ: MobileTogether のパワフルなイメージの処理機能について説明しています。
- オーディオ、ビデオ: MobileTogether のオーディオおよびビデオ機能の概要について説明しています。
- NFC: NFC メッセージの送信と受信方法、および更なる処理のためのセットアップについて説明しています。
- プッシュ通知: ソリューション内でのプッシュ通知の送信と受信のセットアップについて説明しています。
- チャート: デザイン内で構成されるチャートについて説明しています。
- スタイルシート: プロジェクト、ページ、テーブル、およびコントロールレベルグローバルスタイルがどのように適用されるかについて説明しています。
- リッチテキスト: リッチテキストコントロール とリッチテキストスタイルシート がソリューション内でリッチテキストを表示し、リッチテキストを編集するために使用されるかについて説明しています。
- ソリューションへのハイパーリンク (i) 他のソリューション、または (ii) 電子メール内のリンクからのソリューションの開始のためのメカニズムについて説明しています。

(C) 2018 Altova GmbH

11.1 テーブル

テーブルコントロールをテーブル内に箇所にドロップすることによりテーブルを挿入することができます。各 テーブルコントロールに対して、テーブルが1つ作成されます。コントロールをドロップすると、新しいテーブルダイアログが表示されます (*下のスクリーンショット参照*) 。テーブルの型を指定し、テーブルにでッだー、または/およびフッターの追加を指定することができます。



テーブルの型

テーブルの型は、テーブルの作成時に指定されることができます。テーブルの作成後にテーブルの型を変更することもできます。

テーブルの型には4種類あります:

- 静的テーブル: 整理された行と列でデータを表示する際に役に立ちます。新しいテーブルダイアログ内では、行と列の数を指定することができます。構造は後で変更することができます。
- 繰り返しテーブル: データソース要素に関連した全ての発生が行として作成され、各行はデザイン内で作成されたとおりのテーブルにより構成されています。繰り返しテーブルは、動的な (繰り返し) 列を含むことができます(*このリストの4つ目のリスト*)。

- 動的な行が存在するテーブル: 繰り返しテーブルと同様、データソース要素に関連した全ての発生が行として作成されます。繰り返しテーブルとの違いは、各要素の発生は、テーブルとしてではなく行として作成されます。 動的な行を動的な (繰り返し) 列から構成することができます。
- 動的な列が存在するテーブル: 2種類存在します: (i) 行が静的な箇所、および (ii) 行が動的な箇所。動的な行と動的な例は、それぞれデータソースからの異なる繰り返し要素に関連しています。

テーブルの構成とプロパティ

テーブルの内部構成は、後に修正することが可能です。テーブル全体およびこの列、行、セルのためのテーブルは対応するテーブルコンポーネントを選択して、[スタイル & プロパティ] ペイン内でプロパティに割り当て指定します。

詳細に関しては、以下を参照してください: テーブルプロパティとテーブルのコンテキストメニュー。以下のセクションも参照してください: テーブルコントロール。

このセクション

このセクションは以下のように整理されています:

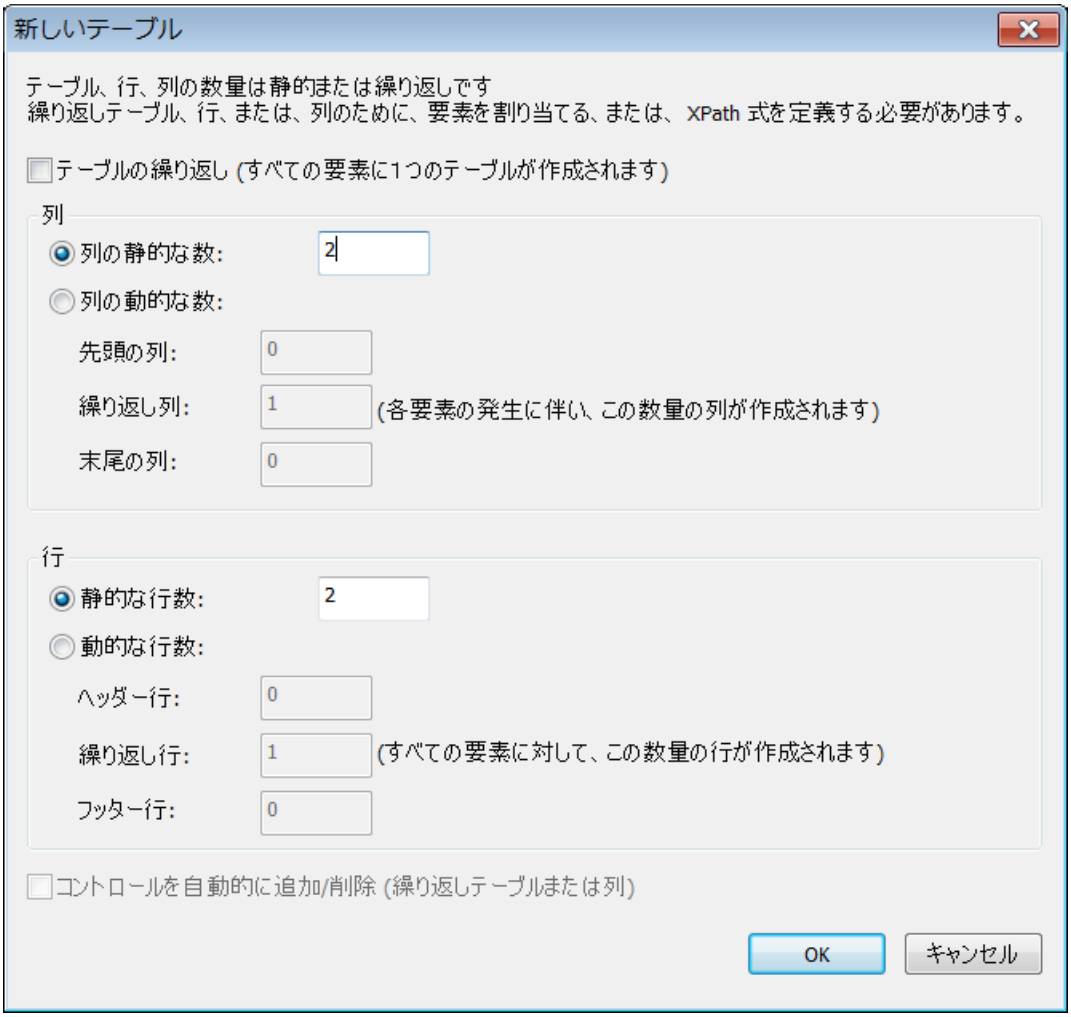
- 静的テーブル
- 繰り返しテーブル
- 動的な行
- 動的な列
- テーブルプロパティ
- テーブルのコンテキストメニュー

静的なテーブル

静的なテーブルは、データを整理された行と列で表示するためにとても役にたちます。

テーブルの作成と構成

静的テーブルはきれいにならべられた行と列のデータを表示するために役に立ちます。テーブルは、デザインに[テーブル] コントロールがドロップされたときに、静的 テーブルと定義することができます。コントロールがデザインにドロップされると表示される [新しいテーブル ダイアログ] (下のスクリーンショット) は下のスクリーンショットに表示されているとおり、的な列と行の数を指定することができます。**[OK]** をクリックすると、指定された数の行と列が挿入されます。テーブルのセルにコンテンツを追加し、テーブルと各列、行、およびセルのスタイル プロパティを指定することができます。



テーブルのコンテキストメニュー で使用可能なコマンドを編成することができます。

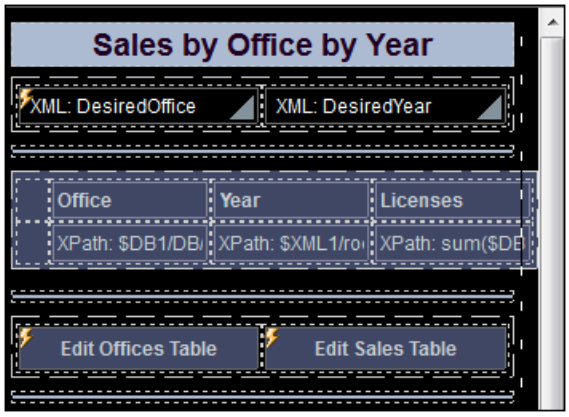
[テーブル フォーマット] プロパティは [スタイル & プロパティ] ペイン で使用することができます。

テーブルのコンテンツ

静的テーブルのセルは次を含むことができます

- 静的なテキスト
- データソースからのノード
- ページ コントロール(データベースノードへのリンク有 無)
- ネストされたテーブル

セルのコンテンツは通常[ページ] コントロールです。下のスクリーンショットはデザイン内の3 つの静的テーブルを表示しています。



これらのテーブルの、2 つを除く すべてのセルは各 1 つずつコントロールを含んでいます。

- 最初のテーブルの2 つのセルにはコンボ ボックスがあります。
- 2 番目のテーブルには(間隔をあけるための) 2 つの空のセル、および[ラベル] コントロールを1 つずつもつセルが6 つあります。
- 3番目のテーブルには2 つのボタンがあります。(コントロールの稲妻のシンボルは、定義されたアクションがあることを示します)。

繰り返しテーブル

繰り返しテーブルは以下のように作動します:

- データソース内の繰り返し要素は、繰り返しテーブルと関連付けられています。(*下のサンプルを参照してください*).
- テーブルに含むことのできる行と列の数量に制限はありません。
- 繰り返し要素の発生の都度、デザインしたテーブルは繰り返されます。ですから、テーブルデザインに2つの行が含まれている場合、繰り返し要素が発生する都度、2つの行を含むテーブルが作成されます。
- テーブルのコンテンツは、動的であることができます。テーブル内のは、XPath 式には、コンテキストノードとして関連付けられた繰り返し要素が存在します。要素が発生する都度、特定の発生はその特定のテーブルのためのコンテキストノードとなります(*下のサンプルを参照してください*)。
- 追加/削除コントロールをエンドユーザーが新規の行を追加して、ここの行を削除することを有効化するテーブルに追加することができます。例えば、ユーザーが行を追加すると、関連付けられたテーブル要素の新規の発生がデータソースのツリーに追加されます。これらの変更は、データソースに保存され、ユーザーがデータソースを変更することを有効化します。

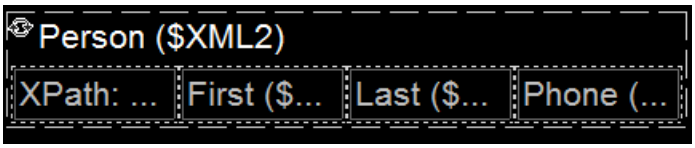
繰り返しテーブルは、テーブルコントロールがデザインにドロップされるときに定義されます。

サンプル

データソース内の**Person** 要素には、**First**、**Last**、および**Phone** 子要素が含まれています。**Person** 要素は(インスタンス内で)複数回発生することができます。以下のようなテーブルを作成すると想定します:

	<First>	<Last>	<Phone>
<Person>			
<Person>			
<Person>			

Person 要素 は繰り返されるため、**Person** 要素と関連付けられた繰り返しテーブルを作成し指定することができます。新しいテーブルダイアログ内にはテーブルには1つの行と4つの列を持つテーブルが作成されます(*下を参照*)。このテーブルコンテキストノードは、**Person** です。各列内でページソースリンクを子ノードに相対的にすることができます(*下のスクリーンショット参照*)。**Person** 要素に番号を付けるXPath 式が最初の列に含まれます 例: **count(preceding-sibling::*)+1**。デザインは、以下のようになります:



生成されたテーブルは、以下のようになります:

1	Basil	Brown	1234567
2	Mary	Gold	4567890
3	Daisy	White	7890123

メモ XPath/XQuery 式によりデータストリームを生成することができ、データソースとして使用することができます。しかし

ながら、この種のデータソースは、現在で使用するために作成され、ドキュメントの他の箇所で使用するためのデータソースとしてはアクセスすることはできません。

繰り返しテーブルと動的な行が存在するテーブルの違い

繰り返しテーブルは *動的行の存在するテーブル* と以下の点で異なります:

- 繰り返しテーブルでは、繰り返しデータ構造と**テーブル全体**が関連付けられています。新規のテーブルは、発生する都度作成されます。
- *動的行の存在するテーブル*では、繰り返し構造と**テーブル行グループ**が関連付けられている。

この差異はデザインに以下の2つの影響を与えます:

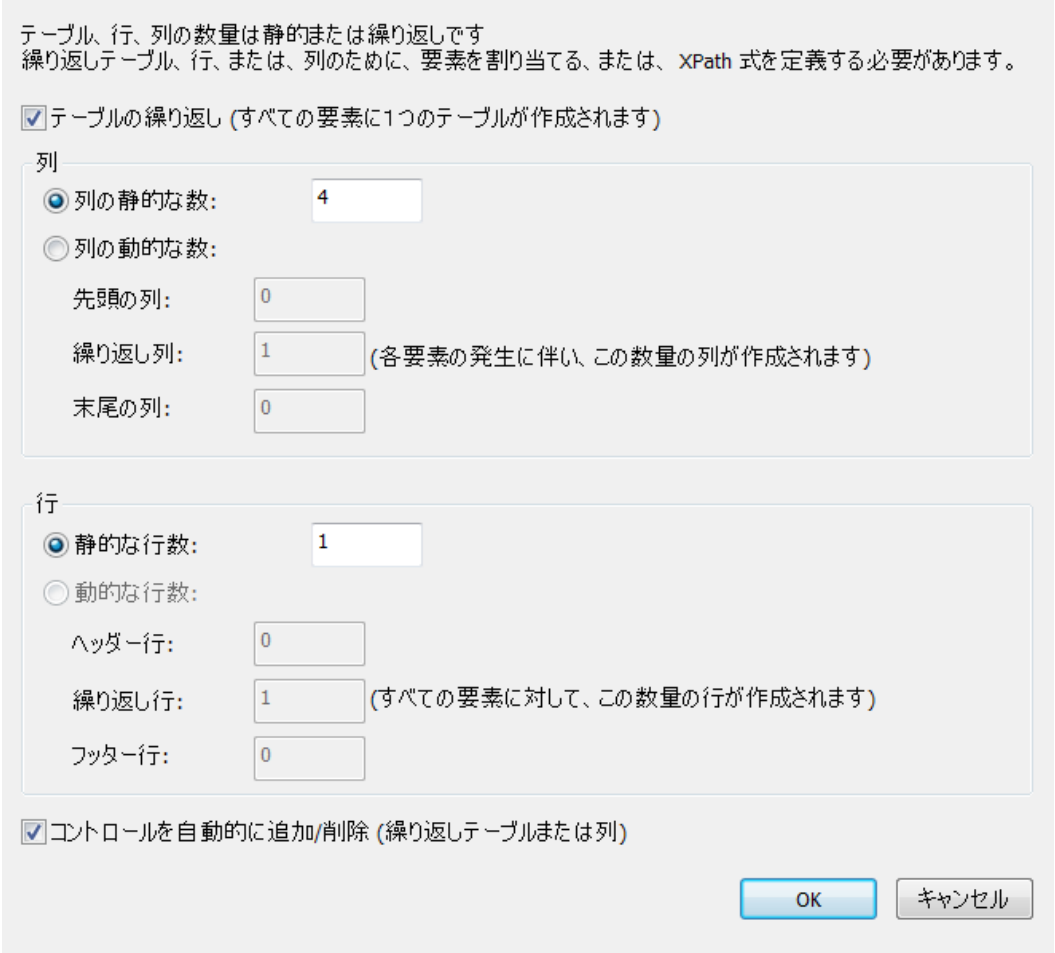
- *動的行の存在するテーブル* は、テーブル全体に対して適用することのできるヘッダー および/または フッターを持つことができます。これはテーブルのための ヘッダー/フッターです。ヘッダー/フッターが繰り返しテーブルのために必要な場合、繰り返しテーブルの外部に手動で追加することができます。繰り返しテーブル内に追加された場合、これらは各要素が発生する都度、繰り返されます。
- テーブルは通常デバイス上で上下にスペースが加えられて表示されます。繰り返しテーブルは、繰り返されたテーブルのペアの間に上下のスペースが加えられ表示されます。

テーブルが作成された後、静的テーブルを繰り返しテーブルに変更するには、繰り返しプロパティをtrue に変更します。そして、テーブルをデータソースからの繰り返し要素と関連付けます。

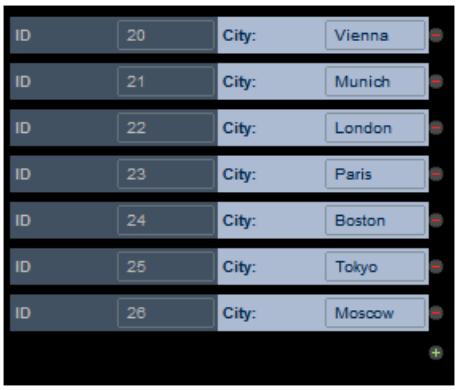
繰り返しテーブルの作成

繰り返しテーブルを次のようにセットアップします:

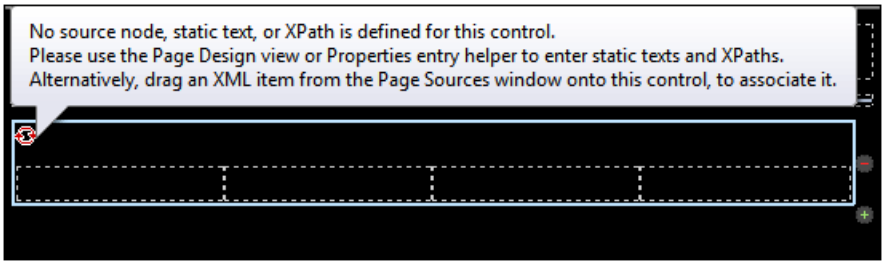
1. テーブルコントロールがデザインにドロップされた時に表示される[新しいテーブル] ダイアログで[テーブルの繰り返し]をチェックしてテーブルを繰り返します (*下のスクリーンショット*)。繰り返しデータの行の全てのインスタンスに対してテーブルが繰り返されることに注意してください。



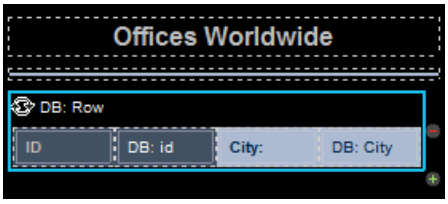
1. 繰り返しテーブルの持つ静的な列と行の数量を指定します。テーブルのコンテキスト メニューを使用して後から列と行を繰り返しテーブルに追加することができます。
2. 自動追加 削除 コントロールが追加されるか指定します。追加されると、ソリューションの各繰り返しテーブルは、**[削除]** ボタンを、繰り返しテーブル構造全体は、構造に繰り返しテーブル 行) を追加する**[追加]** ボタンを持ちます (*シミュレートされたソリューションのスクリーンショットを参照*)。



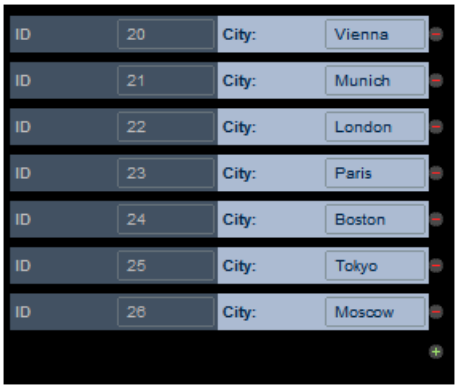
3. [新しいテーブル] ダイアログで **[OK]** をクリックすると、デザインにテーブルが追加されます。繰り返しテーブルはデータソースの繰り返し要素に関連付けられる必要があります (*下のスクリーンショット参照*)。



4. 繰り返し要素を繰り返しテーブルに関連付けるには、要素を[ページ ソース] ペインからドラッグしてテーブルにドロップします。
5. テーブルのセルにコンテンツを追加することができます。 XPath 式のためのコンテキストノードは動的テーブルに関連した要素ノードです (*前のステップ参照*)。 コンテキストノードを使用するには、テーブルセル内のXPath 式はコンテキストノードに対して相対的であるべきです。 セルのコンテンツは (静的または動的な) ネストされたテーブル、または (データソース ノードへのリンクがあるない) ページコントロール、またはページソースノードであることができます。ページソースノードがセルにドロップされると、そのセル内のデータは編集可能になります。下のスクリーンショットでは、(左から右に) 4 つのコントロールが追加されました： ラベル、編集フィールド、ラベル、編集フィールド。



この繰り返しテーブルは次の繰り返し構造をMobileTogether ソリューション内に作成します。



テーブルのコンテキストメニュー で使用可能なコマンドを編成することができます。[テーブル フォーマット] プロパティは [スタイル & プロパティ] ペイン で使用することができます。以下も参照してください： 動的な行 および 動的な列。

動的な行

動的な行は、以下のように作動します:

- 動的な行は、データソース内の繰り返し要素と関連付けられています。
- テーブルが表示されると、行数は、繰り返し要素の発生数と動的に一致します。各テーブル行は繰り返し要素の発生と対応します。
- 動的な行を定義すると、各要素の発生のために繰り返される行数を指定することができます。ですから、2つの行を含む各動的な(または繰り返し)行グループを指定することができます。この場合、2つの行の行グループが各要素の発生のために生成されます。
- 動的な行(または行グループ)のためのコンテキストノードは、特定の要素の発生になります。
- エンドユーザーが新規の行を追加、または、削除することを有効化する追加/削除コントロールを追加することができます。例えば、ユーザーが行を追加すると、データソースのツリーに行要素に関連した新規の発生が追加されます。これらの変更は、データソースに保存することができ、エンドユーザーはデータソースを変更することができます。

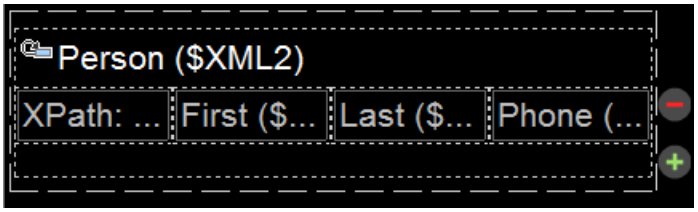
テーブルコントロールがデザインにドロップされると動的な行が存在するテーブルを定義することができます。

サンプル

データソース内の**Person** 要素には繰り返し構成(例えば、**First**、**Last**、と**Phone** 要素)が含まれます。**Person** 要素は複数回発生することができます。動的な行(すなわち、行グループ)が**Person** 要素と関連付けられていると、**Person** 要素内で存在する数と同様の数の行(グループ)が存在するテーブルが生成されます(*下のテーブル参照*)。データソース内の**Person** 要素の数量が変更されると、テーブル内の行数も自動的に変更されます。

	<First>	<Last>	<Phone>
<Person>			
<Person>			
<Person>			

下のスクリーンショット内のデザインには、単一の動的な行(グループ)を含むテーブルが表示されています。行グループは**Person** 要素に関連付けられており、1つの行と4つの列により構成されています。



行グループと関連付けられている要素の発生のために、行グループ全体が生成されます。行グループ内のXPath 式は、コンテキストノードとしての関連付けられた要素の現在の発生により解決されます。

メモ XPath/XQuery 式によりデータストリームを生成することができ、データソースとして使用することができます。しかしながら、この種のデータソースは、現在で使用するために作成され、ドキュメントの他の箇所で使用するためのデータソースとしてはアクセスすることはできません。

繰り返しテーブルと動的な行が存在するテーブルの違い

繰り返しテーブルは *動的行の存在するテーブル* と以下の点で異なります:

- 繰り返しテーブルでは、繰り返しデータ構造と**テーブル全体**が関連付けられています。新規のテーブルは、発生する都度作成されます。
- *動的行の存在するテーブル*では、繰り返し構造と**テーブル行グループ**が関連付けられている。

この差異はデザインに以下の2つの影響を与えます:

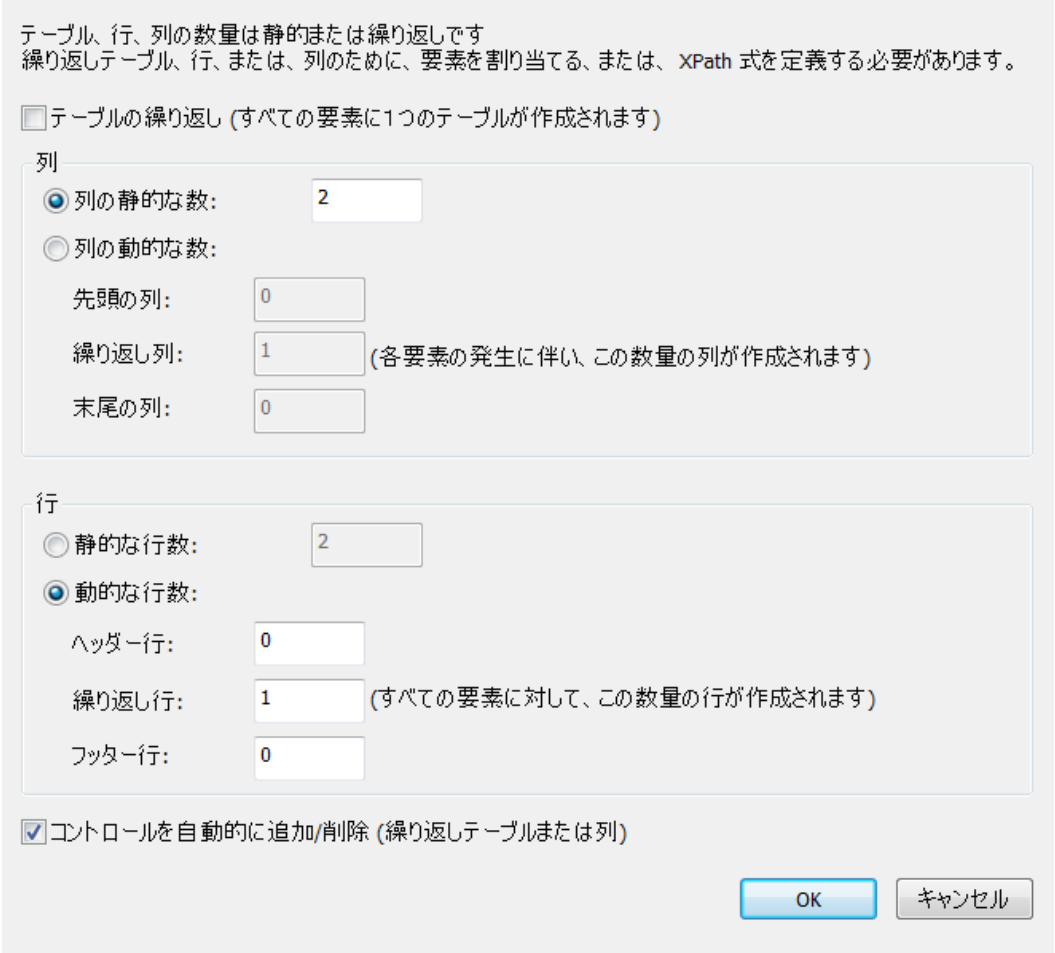
- *動的行の存在するテーブル* は、テーブル全体に対して適用することのできるヘッダー および/または フッターを持つことができます。これはテーブルのための ヘッダー/フッターです。ヘッダー/フッターが繰り返しテーブルのために必要な場合、繰り返しテーブルの外部に手動で追加することができます。繰り返しテーブル内に追加された場合、これらは各要素が発生する都度、繰り返されます。
- テーブルは通常デバイス上で上下にスペースが加えられて表示されます。繰り返しテーブル は、繰り返されたテーブルのペアの間に上下のスペースが加えられ表示されます。

繰り返しテーブルを *動的な行が存在するテーブル* に変換するには、動的な行に変換するテーブル行を右クリックし、**動的な または繰り返しテーブル | この行を繰り返し行に変換する** を選択します。

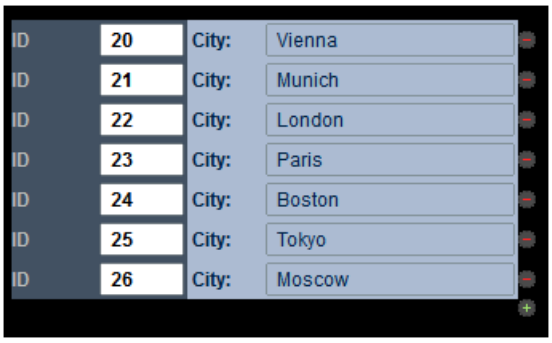
動的な行が存在するテーブルの作成

動的な行が存在するテーブルを以下のようにセットアップします:

1. In 新しいテーブルダイアログ内では、コントロールがドロップされると、テーブルの繰り返しのチェックが解除されていることを確認してください(下の画面スクリーンショット)。そして、*動的な行数*を選択します。動的なテーブル行グループを含むテーブルが作成されます。

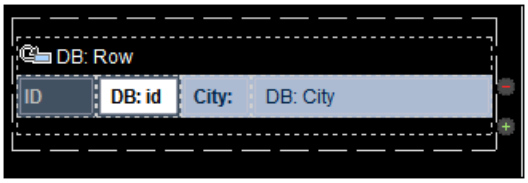


2. テーブル内の列数と行グループの行数を指定します (行グループは行により構成されています)。また、テーブルにヘッダー および/または フッターが存在するかを指定することもできます。
3. 自動コントロールの追加/削除を指定することができます。追加されると、ソリューション内の各繰り返し行に**削除**ボタンが追加され、テーブルには行を追加する**追加**ボタンが追加されます (シミュレートされたソリューションのスクリーンショットを参照してください)。

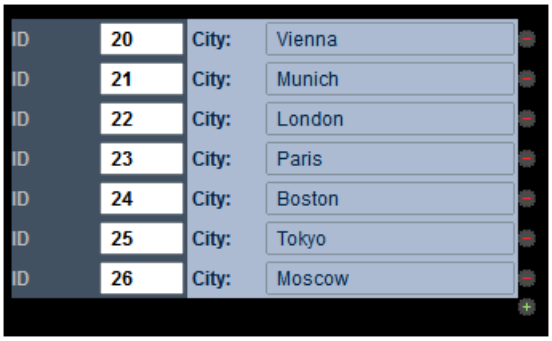


4. 新しいテーブルダイアログ内で、**OK** をクリックすると、テーブルがデザインに追加されます。
5. 行グループはデータソースからの繰り返し要素に関連付けられている必要があります。行グループを繰り返し要素に関連付けるには、ページソースペインから要素をドラッグして、テーブルにドロップします。この要素の各発生はテーブル内に行 (グループ) を生成します。現在の発生は指揮グループ内の XPath 式のコンテキストノードとなります。
6. テーブルのセルにコンテンツを追加することができるようになります。テーブルセル内の XPath 式のためのコンテキストノードは動的な行に関連付けられた要素ノードです (*前のステップを参照してください*)。コンテキストノードを使用す

るには、in テーブル内のXPath 式セルはコンテキストノードに対して相対的である必要があります。セルのコンテンツはネストされた(静的な または動的な)テーブル、または、(データソースノードにリンクを有する、または、有さない) ページコントロール、または ページソースノードであることができます。ページソースノードがセルにドロップされると、そのセル内のデータを編集することができます。下のスクリーンショットでは、4つのコントロール (左から右): ラベル、編集フィールド、ラベル、編集フィールド) が追加されている状態が表示されています。



この動的な行は、MobileTogether ソリューション内で次の構造を作成します。



動的な列のスパンに関しては、次を参照してください：行 列のジョインとスパン。

テーブルの再構成コマンドは、テーブルのコンテキストメニュー内で使用することができます。テーブルフォーマットプロパティは、[スタイル & プロパティ]ペイン内で使用することができます。繰り返しテーブル と動的な列を参照してください。

動的な列

テーブル行内で、列が繰り返されると、これらの繰り返しは、動的な列機能により表示されることができます。列はデータソースの繰り返し要素と関連付けられています。テーブルが表示されると、列数は関連した要素の発生数と動的に対応します。動的な列は、2つの種類の行コンテキスト内で発生することができます：

- *静的な行 ,動的な列:* このシチュエーションでは、テーブルは上下ではなく左右に広がります (*下のテーブルと動的な行が存在するテーブルを比較してください*)。この場合、最初の列が「ヘッダー」としての役割を果たし、行の名前を含むことができます。(最初の列は、テーブルが最初に作成された時点で追加することができますが、列は静的であるため、後に作成することも可能です)。

	<Person>	<Person>	<Person>
<First>			
<Last>			
<Phone>			

- *動的な行、動的な列:* テーブルは上下(追加行)、および左右(追加列)に広げることができます。このような種類のテーブルを作成するには、データソース内の列要素の発生回数は、全ての行で同じ数量である必要があります。例えば、下のテーブル内で、(テーブル内の行に対応する) 4つの**week** 要素それぞれは、3つの**day** 要素 (テーブルの列) を含んでいます。**week** 要素が3つ以外の**day** 要素を含んでいる場合、テーブルを正確に作成することはできません。以下の点に注意してください！ (i) データソースとテーブルデザイン内の列要素は行要素内で発生します。(ii) 行要素の名前は同じで、列要素の名前も同じです。しかしながら、上記の2つの点はこの種類の動的なテーブルを作成するために必ずとも必要ではありません: (i) 列要素は、行要素外で発生することができ、(ii) 行 / 列は異なる名前を持つことができます。

	<day>	<day>	<day>
<week>			
<week>			
<week>			
<week>			

メモ XPath/XQuery 式によりデータストリームを生成することができ、データソースとして使用することができます。しかしながら、この種のデータソースは、現在で使用するために作成され、ドキュメントの他の箇所で使用するためのデータソースとしてはアクセスすることはできません。

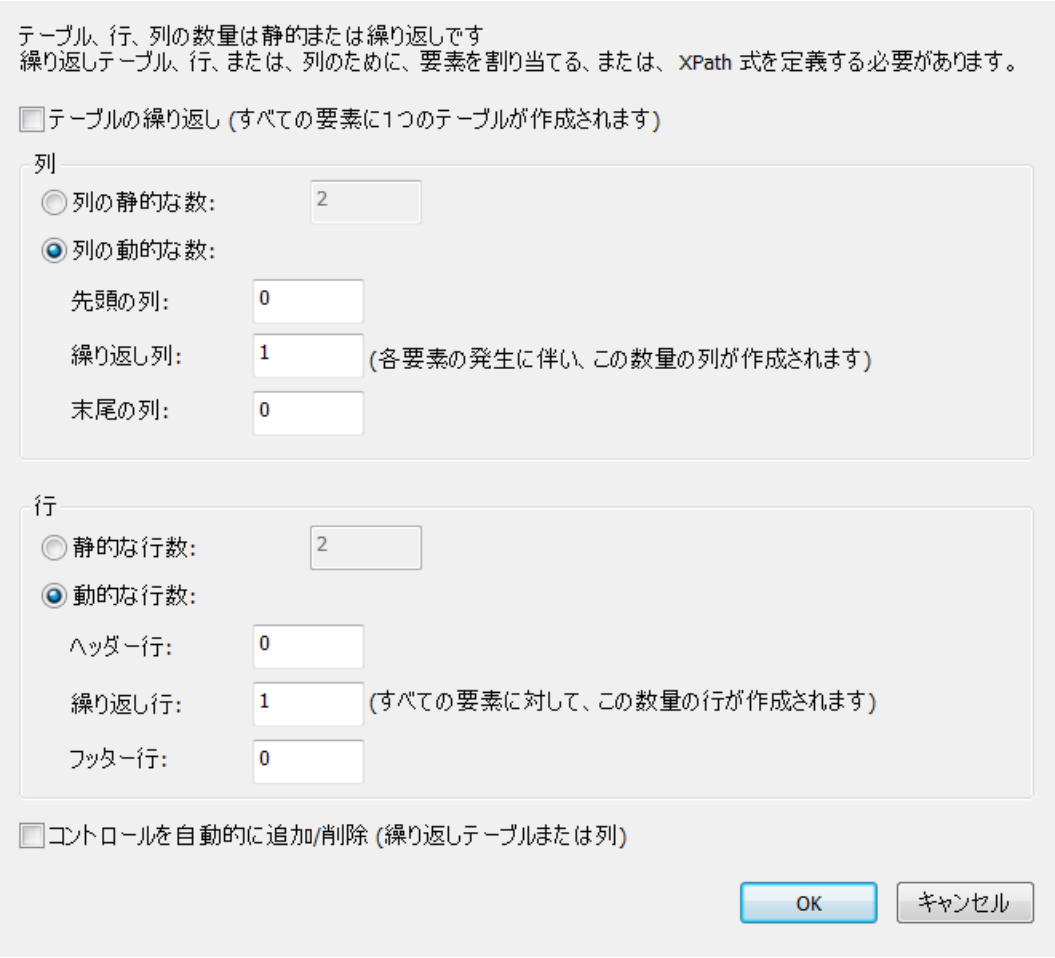
サンプル: 週のための動的な行内の日のための 動的な列

下のスクリーンショットは、それぞれが7つの**day** 要素を含む4つの**week** 要素を含む**calendar** 要素 を表示しています。**week** 要素と動的な列 (**week** 要素内の) の**day** 要素 のために動的な行が存在するテーブルを作成することができます。データ構造に注意してください (i) **day** 要素は、**week** 要素内に存在し、(ii) **week** 要素内の**day** 要素数量は同じです: 7つ。

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <calendar>
3       <week id="W1">
4           <day id="D1"/>
5           <day id="D2"/>
6           <day id="D3"/>
7           <day id="D4"/>
8           <day id="D5"/>
9           <day id="D6"/>
10          <day id="D7"/>
11      </week>
12      <week id="W2">
13          <day id="D1"/>
14          <day id="D2"/>
15          <day id="D3"/>
16          <day id="D4"/>
17          <day id="D5"/>
18          <day id="D6"/>
19          <day id="D7"/>
20      </week>
21      <week id="W3">...</week>
30      <week id="W4">...</week>
39  </calendar>
40
```

動的な行内に動的な列を作成する

テーブルコントロールをデザインにドラッグしドロップします。新しいテーブルダイアログ内で、動的な行と動的な列を選択します (*下のスクリーンショットを参照*) 。行と列にそれぞれ対応するように各要素の発生のために繰り返される行と列の数量を選択します。最初と最後の列はヘッダー/フッター行に対応します。動的な列ではなく動的な行ためのコントロールを追加/削除することができます。



テーブルはデザイン内に作成されます。下のスクリーンショットでは、動的な行と列が存在するテーブルのデザインが表示されています。以下の点に注意してください:

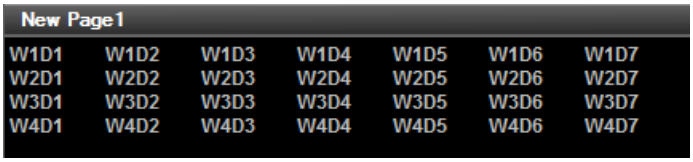
- テーブルの動的な行と列を表示しているフィールドは行と列を表示するアイコンにより示されています。
- これらのフィールドは、テーブルの行と列のためのデータを提供するデータソースノードと関連付けられている必要があります。列に関連付けられる要素は、データソース内では、行に関連する要素内に含まれている必要があります。デザイン内では、しかしながら、行フィールドは列フィールド内に配置されていることに注意してください。

- データソースノードと行と列フィールドを関連付けるために使用されるXPath 式にも注意してください。行フィールドのために要素を選択する式は、対応する要素の全てのインスタンスを選択する必要があります。上のスクリーンショット内のXPath 式は、**calendar** 要素の全ての**week** 子要素を選択します: **$XML1/calendar/week**。列フィールドのためのXPath 式はしかしながら、現在の行内の列要素のみを選択する必要があります。 ですから、**$XML1/calendar/week/day** などのXPath 式は全ての**week** 要素の全ての**day** 子要素を選択す

るため、使用することはできません。列フィールドのためのXPath コンテキストノードも行に関連付けられている要素であることに注意してください。このサンプルでは、列フィールドのコンテキストノードは、**week** です。この結果、**day** のXPath 式は、現在の**week** 要素の**day** 子要素を選択します。

- デザイン内では、行取れるフィールドのインターセクションには単一のセルが存在します。このセルのコンテキストノードは、行フィールドに対応する要素です (このサンプルでは、 **week** 要素です)。このセル内のXPath 式は、このコンテキストで構成されている必要があります。テーブルが作成されると、各「行 要素」(このサンプルでは**week** 要素)内で、各列のためにセルが作成されます。XPath 式は、現在の行のコンテキスト内の各列のセルのために評価されます。列フィールドに対応する要素のコンテンツに達成するために、特別な変数を使用することができます: **$MT_TableColumnContex** 、は、ランタイムで、(このサンプルでは現在の**week** 要素内の現在の**day** 要素)現在の列に対応する要素を含みます。上のスクリーンショットでは、行と列フィールドのインターセクションにあるデザイン内のセルはラベルコントロールを含んでいます。このコントロールには、XPath 式により提供されたテキストが存在します: **concat(@id, $MT_TableColumnContext/@id)**。**concat()** 関数は2つの文字列を結合します: 現在の**@id** により取得された現在の週数 のID と **$MT_TableColumnContext/@id** により取得された現在の週数内の現在の日数。XPath 式のコンテキストノードは、(行と関連付けられた) **week** 要素 **@id** であり、現在の**week/@id** 属性の値を提供しますが、**$MT_TableColumnContext/@id** 式は、現在の**week** 要素内の現在の**day/@id** 属性のコンテンツを取得します。

上のスクリーンショットで示されているテーブルのデザインの出力は以下のようになります:

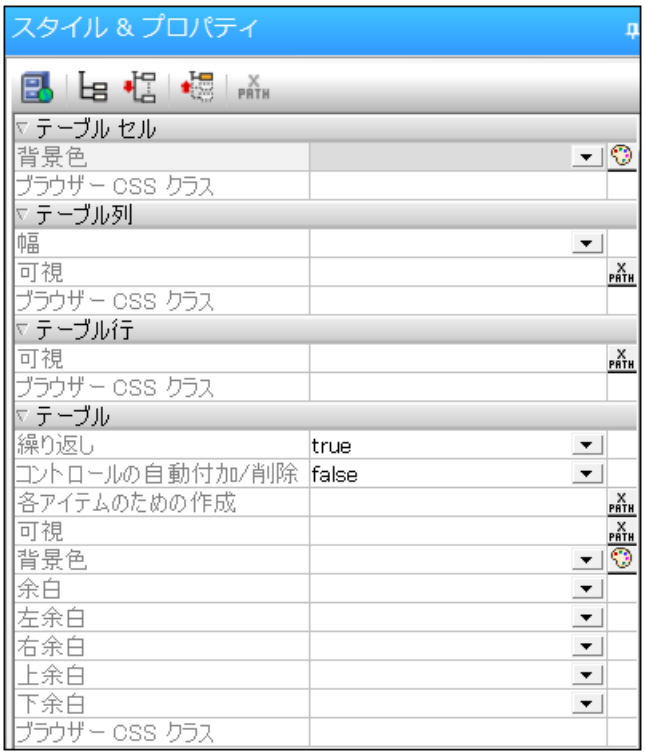


- データソース内の各 **week** 要素は行 (**W1** から**W4**)で表示されています。
- week 内の各 **day** 要素は、対応する列 (**D1** から**D7**)内で表示されています。
- 2つのID の連結は、**W1D1** から**W4D7** への28個のセルのために、個別に行われます。

動的な列に関する詳細は、次を参照してください: 行 列のジョインとスパン。

テーブルプロパティ

特定のプロパティをテーブルのセル、列、行、およびテーブル全体に設定することができます。下左のスクリーンショットでは、静的テーブルと繰り返しテーブルのプロパティを表示しています。下右のスクリーンショットは動的なテーブルのプロパティを表示しています。動的テーブルには、追加のテーブル行グループのプロパティのセットがあります。これらすべてのプロパティについては[コントロール] テーブル] のセクションで説明されています。



プロパティ: 繰り返し

繰り返し プロパティは、静的なテーブルを繰り返すために使用され、テーブルが繰り返し または静的 かを定義します。動的なテーブルに対しては、使用することができません。プロパティの値は、`true` または`false` です。

このプロパティの値は、繰り返し、または、静的なテーブルが作成される際に自動的に割り当てられます。繰り返しテーブルには`true` の**繰り返し** プロパティが、静的なテーブルには、`false` の**繰り返し** プロパティがあります。テーブルが特定の型 (繰り返し、または、静的) として作成されると、型はテーブルの**繰り返し** プロパティの値を変更することによりおこなわれます。

プロパティ: 各アイテムのために作成

[各アイテムのための作成] プロパティはテーブル(繰り返しテーブル)およびテーブル行グループ(動的なテーブル)に対して使用することができます。繰り返しテーブルまたは繰り返しテーブル行グループが何度作成されるか指定します。この数は、プロパティのXPath 式により返されたシーケンスのアイテムの数量と等しいです:

- データソースツリーからのノード。これは繰り返しテーブル(またはテーブル行グループ) のノードをテーブルにドラッグ アンド ドロップして関連付けられる、データソース ノードとの関連付けの代替です。この種類のXPath 式は、ノードの選択に対して更なる柔軟性を与えます。例えば、XPath式 `$XML1/Offices/Office[@location='US']` は属性 `@location='US'` を持つ `Office` ノードのシーケンスを返します。US フィルターは、テーブルに`Office` ノードをドロップする代替メソッドを使用して適用することができません。

しかし、このフィルターは、[各アイテムのために作成] プロパティで達成することができます。

- データソースツリーに関連しないアイテム。例えば、2014 年10月の月の場合、1 to subsequence(age-details(xs:date("2014-01-01")),2,1) 式は、2014 年 01 月 01 日から2014年 10 月 のある日付の間の月の数である9 つのアイテムから成るシーケンスを返します。これは基本的な式が 1 to X であるからです。そして X は、(subsequence 関数によると) age-details 関数により返された3つのアイテムのシーケンスの第2アイテムです。この後の関数が、入力の日時 (2014 年 01 月 01 日) に対応して、3つのアイテムのシーケンスの年、月、日として、(2014 年 10 月の)現在の日付の年齢を(0 年 9 ヶ月 xx 日として) 返します。 3つのアイテムのシーケンスの第2番目のアイテムは、年齢内の月数で、この場合 9 です。返されたシーケンスは9つのアイテム(1 から 9 の範囲)を含み、テーブルは9回作成されます。

メモ XPath 式の結果をプレビューしたい場合、MobileTogether Designer に内蔵されたシミュレーション **[プロジェクト | ワークフローのシミュレーション]** を実行します。表示されたダイアログのをクリックして **[XPath 評価]** および **[エバリュエーター]** をクリックします。

行 /列 の下のジョインとスパン

複数の行、または、複数の列をジョインするには、デザイン内でスパンする行または列を選択し、適切な**ジョイン**コマンドをコンテキストメニュー、テーブルメニュー、またはアプリケーションツールバーを使用します。 選択された行 列が、選択された行 列にジョインされます。ジョインされた行または列は、それぞれ、(動的な行 または列のために作成された)行グループまたは列グループ内に存在します。ジョインはグループの各インスタンス内で発生し、ジョインされた行 列は各グループ内で表示されます。

動的な列の場合、スパンと呼ばれる列のマージの型を使用することができます: デザイン内の列グループが1つまたは複数の列により構成されているかに関わらず、すべての列グループの列は、単一の列にスパンされます。これを行うには、([スタイル & プロパティ]ペイン内にある) 列グループのスパン プロパティを true にセットします。 このプロパティは **true** または **false** の値を取ります。デフォルトは、**false** です。プロパティが **true** に設定されている場合、出力内では、列グループの全ての列はスパンされ、ひとつの列として表示されます。

下のテーブルは、2つの列から構成される列グループにより構成されるデザインを示しています。列グループは要素 **ノード** と関連付けられています。

*繰り返し要素 **ノード** に対応するデザイン内の列グループ。**スパンされていません。***

デザイン内の列 1	デザイン内の列 2

出力内では、列グループの各インスタンスの **ノード** ために繰り返されます。この結果、下に示されるように、各 **ノード** 要素のために2つの列が作成されます。

ノード[1] *列 1*	**ノード[1]** *列 2*	**ノード[2]** *列 1*	**ノード[2]** *列 2*	...	**ノード[n]** *列 1*	**ノード[n]** *列 2*

動的列グループがスパンされていると、列グループのスパン プロパティを true に設定すると、デザイン内では両方の列がジョインされているような効果が与えられます (*下のテーブル参照*)。結果する列のプロパティとコンテンツは、最初の列になります。

*繰り返し要素 **ノード** に対応するデザイン内の列グループ。**スパンされています。***

デザイン内の列 1 は、列 2 を計算します

出力では、列グループが**ノード**の全てのインスタンスにスパンされます。この結果、下に示されるように、**ノード** の全てのインスタンスのために1つの列が存在するようになります。列のコンテンツが動的に**ノード** 要素を検索するXPath 式 により選択された場合、エラーが返されます。

ノード[1 **から** n] *列 1 は**ノード**のすべてのインスタンスをカバーするようにスパンします*

動的な列がスパンされると、変換は以下の2つのステップで発生すると考えることができます: (i) **ジョイン**コマンド が使用されたと同様、デザイン内で列グループ内の(静的を含む全ての型の)全ての列が1つの列にジョインされます。(ii) 出力内では、繰り返し要素のインスタンスが単一の列として作成されます。(i) デザイン内でスパンされた動的な列内で検索された(ii) 出力列に対応するこの要素インスタンスを検索しようと試みたXPath 式はエラーを返します:

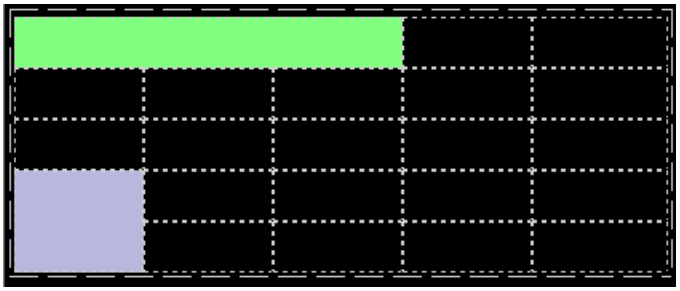
下のスクリーンショットは、行グループ内で作成された動的な列の簡単なサンプルを表示しています。デザイン内の列グループには、**day** 要素と関連付けられている単一の列グループが含まれており、この列グループは、(**day** 要素の親であるデータソース内の) **week** 要素に関連付けられている 繰り返し テーブル内に存在します。**week** 要素は繰り返されるため、各 **week** 要素のために新しいテーブルが作成されます。データソース内で、**week** 要素の**day** 子要素が複数存在する場合、デザイン内で列グループの動的な列はスパンされず、(各 **week** のために) **day** 子要素内に存在する数と同時数の列を持つテーブルがこのデザインから生成されます。しかしながら、列グループのスパン プロパティを`true` に設定すると、生成されたテーブル内の列はスパンされ、テーブルには1つの列のみが存在するようになります。

行グループ関しては、次を参照してください:動的な列.

列 /行 可視性

列 または行 の可視性は、列または行を選択して、設定することができます。また、**可視** プロパティを**true** または **false** に設定することができます。デフォルトの値は**true** です。

列または行がスパンされている場合、スパンされている列/行の可視性は、**もし**最初の列/行の可視性が**true** に設定されている場合、個々に指定されることができます。最初の列/行の可視性が**false** に設定されている場合、スパンされているすべての列/行には**false** の値はが与えられます



上のスクリーンショット: 最初の行内のスパンされた列(緑色):

- 行 1 (緑)内の3つのスパンされた列は、列 1 と考えられます。行 1 内の次の列は、列 4 と考えられます。行 1 内には、列 2 または列 3 は存在しません。
- 行 2 と5 からそれぞれ最初の3つの列を選択し、可視性を`true/false` に設定した場合、他の列の可視性は影響を受けません。
- 行 1 内の(スパンされた) 列 1 を選択し、可視性を`true/false` に設定した場合、列 1 の可視性のみが変更されます。(行 2 から 5 内の) 列 2 と列 3 は影響されません。

上のスクリーンショット: 最初の列内のスパンされた行 (青色):

- 列 1 (青)内のスパンされた行は、行 4 と考えられます。In 列 1 内には、行 5 は存在しません。
- 列2 と5 からをそれぞれ2つの行(行 -4 または 行 -5) を選択し可視性を`true/false` に設定した場合、他の行は影響されません。
- 列 1 内の(スパンされた) 行 4を選択し、可視性を`true/false` に設定した場合、行 4 の可視性のみが変更されます。 (列 2 から 5 内の) 行 5 は影響されません。

スクロールすることのできるテーブル

テーブルが長く、幅が広い場合は、上下または左右にスクロールできるように設定する事ができます。このイベントでは、テーブルの一部のみが表示されます。残りはスクロールする事ができ、表示されません。

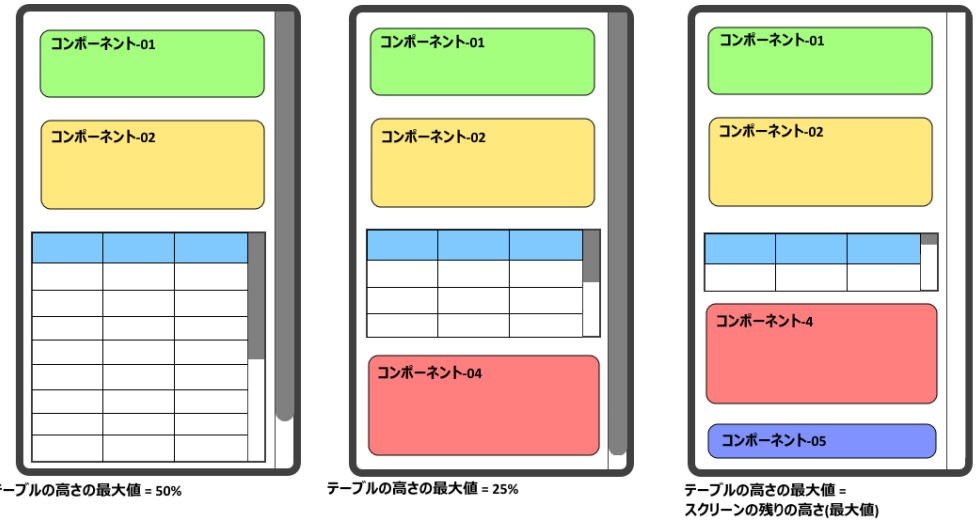
上下にスクロール

`Maximal Table Height` プロパティは、デバイス上でのテーブルの高さを、ピクセルまたはデバイスのスクリーンの高さに相対して指定します。例えば、`50%` を選択すると、デバイスのスクリーンの高さの半分の高さのテーブルが表示されます(下の左端のイメージを参照してください)。テーブルが上下に長く、割り当てられたスクリーンのスペースに入りきらない場合、テーブルには上下のスクロールバーが表示され、エンドユーザーはテーブルの残りを、割り当てられたスクリーンのスペースの内外でスクロールする事ができます。(このサンプルではスクリーンの高さの50%)。テーブルの上でデザインコンポーネントが起こる場合、これら全てのコンポーネントはテーブルの上に表示されます。テーブル自身は、このプロパティにより指定された絶対または相対的な高さを持ちます。

メモ テーブルとページには個別のスクロールバーがあります (*下のスクリーンショットのイメージを参照してください*)。MobileTogether Designer シミュレーターでは、スクロールホイールを使用して行下にスクロールし、クリックアンドドラッグを使用して左右にスクロールします。

メモ **Android 4.x** デバイスでは、2つ以上の(スクロールすることのできる、または、スクロールすることのできない) テーブルがページに存在する場合、1つのテーブルがスクロールすることのテーブルである場合、上下のスクロールを使用することはできません。

メモ スクロールすることのできるテーブルのチュートリアルのサンプルを参照してください。



テーブルの高さの最大値 プロパティは(ピクセルまたはパーセントの値の他に)他の2つの値を取ることができます:

- *スクリーンの高さの残り(最大値)*: テーブルの高さは、ページの残りが表示されるようにできるだけ最小化されます。上のイメージでは、右端のスクリーンは、このプロパティの値を与えられたテーブルを表示しています: テーブルの高さは、縮小されページの5つのコンポーネントがすべて表示されています。この場合、ページを表示するため、ページのスクロールバーはゼロに縮小されています。
- *スクリーンの高さの残り(常に)*: このオプションによりスクリーンの高さ全体を使用して、ページを表示することができます。テーブルにページを埋めるだけの上下の余白がない場合、追加スペースがテーブルの下に追加され、ページの最後のコンポーネントがスクリーンの下のすぐ上に表示されます。この設定によりページの最後のコンテンツをスクリーンの一番下の場所に配置することができます。

上下にスクロール プロパティは、**テーブルの高さの最大値** プロパティのために値が設定されると使用することができます。**テーブル幅の最大値** は設定されません。 **上下にスクロール** プロパティは1つまたは2つの値をとることができます:

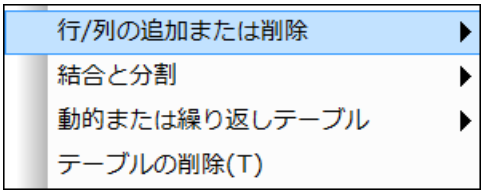
- *テーブル全体*: テーブル全体は、**テーブルの高さの最大値** プロパティを使用してテーブルに割り当てられたテーブルの高さをの内外をスクロールします。*テーブル全体* は、デフォルトの値です。
- *ヘッダーとフッターを除く行*: テーブルのヘッダーとフッタはビュー内では固定されています。テーブルのボディ行は残りのテーブルの高さ内でスクロールします。

繰り返し行グループがテーブル内に存在し、スクロールすることのできるテーブルの**テーブルの高さの最大値**のために値が設定されると**行グループチャンクサイズ** プロパティを使用することができるようになります。これにより、1度にロードする行グループの数を指定することができます。ユーザーが下にスクロールし、最後にロードされたチャンクの最後の行グループに到達すると、次のチャンクがロードされます。このプロパティのデフォルトの値は存在しません。

左右にスクロール
テーブル幅の最大値 プロパティは、テーブルの幅: (i) ピクセル値で、(ii) デバイスのスクリーン幅に相対するか、または (iii) 列のために最適化された値 (`wrap_content`) を指定します。デフォルトは、`wrap_content` です。プロパティのコンボボックスのドロップダウンリストから値を選択します。テーブルの幅がスクリーンの幅よりも大きい場合は、テーブルは左右のスクロールバーで表示されます。

テーブルのコンテキストメニュー

すべてのテーブルの(静的、繰り返し、動的な行 および/または 動的な列 が存在するテーブル) コンテキストメニューには同じコマンドがあります (*下のスクリーンショット参照*)。これらのコマンドにより、テーブル作成後のテーブルの構造を変更することができます。



行 /列の追加と削除

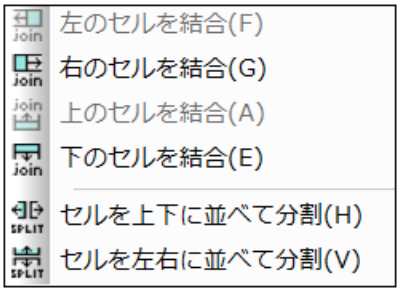
行列の追加または削除コマンドの上をポイントすると、現在選択されたセルに関連した行/列を挿入/追加することのできるコマンド付きサブメニューが表示されます (*下のスクリーンショット*) 。この方法で追加される行と列は静的です。これは、例えば、1つの静的な行がデザインに追加されると、出力内に静的な行が結果として表示されます。もちろん、繰り返し構造内に行が追加された場合、静的な行も繰り返されます。

現在選択されている行/列 を削除することもできます。

これらのコマンドは、テーブルの行または列(静的、繰り返し、動的な行 および/または 動的な列 が存在するテーブル)が選択されると使用することができます。

結合と分割

結合と分割コマンドの上をポイントすると、現在選択されたセルと隣接したセルを結合することのできるコマンド付きサブメニューが表示されます (*下のスクリーンショット*) 。現在選択されているセルを左右に並べて、または上下に並べて分割することができます。これらのコマンドはすべてのテーブルの行または列 (静的、繰り返し、動的な行 および/または 動的な列 が存在するテーブル)が選択されると使用することができます。



動的または繰り返しテーブル

動的または繰り返しテーブルコマンドの上をポイントすると、以下を行うことのできるコマンド付きサブメニューが表示されます (*下のスクリーンショット*)

サブメニューには、以下をおこなうことのできるコマンドが含まれています:

- 表示されるテーブル(繰り返しテーブルと動的な行 が存在するテーブル)に自動的に追加/削除コントロールが追加されることを指定します。
- (動的な行 が存在するテーブル)にヘッダー/フッター を挿入または削除します。(動的な列が存在するテーブル)に行頭/文末列を挿入または削除します。
- 選択された行が動的な繰り返し行**ではない**場合、繰り返し行に変換します。
- 選択された行が動的な繰り返し行**である**場合、静的な行に変換します。

- 選択された列が動的な繰り返し列**ではない**場合、繰り返し列に変換します。
- 選択されたれ地が動的な繰り返し列**である**場合、静的な列に変換します。
- 繰り返しと静的 のテーブルの型を変換します(静的 テーブルの行と列をそれぞれ、繰り返し行と繰り返し列に変換することができることに注意してください)。

テーブルの削除

選択されたテーブルを削除します。

11.2 イメージ

デザイナーとエンドユーザー両方がイメージをデザインに追加することができ、イメージはURL を使用して、またはBase64 エンコード テキストとしてXML ファイルに保管することができます。イメージ機能のベースは、イメージをデザインに置き、イメージの基本プロパティを設定するイメージ コントロールにより提供されます。主要なメカニズムと使用方法は、このセクションのサブセクションに説明されています。

- イメージ ソースは、表示するイメージを選択するイメージ コントロールのプロパティです。このセクションでは使用することのできる2 つの種類のイメージ ソースについて説明されています: (i) URL によりロケートされたイメージ ファイル および (ii) Base64 エンコード文字列としてのイメージ。
- Base64 エンコード イメージ では、Base64 エンコード イメージをどのようにしてデザインで使用するかについて説明されています。
- エクスチェンジャブル イメージ ファイル フォーマット (Exif) は、イメージ付きメタデータの保管のためのフォーマットです。このセクションでは、個別のExif データが取得されどのようにしてデザイン内で使用できるかが説明されています。
- エンドユーザーにより選択されたイメージ では、MobileTogether ソリューションのエンド ユーザーが選択するイメージがデータベースに保管されるメカニズムについて説明されています。これらのイメージはイメージ ファイルまたはBase64 エンコード文字列として保管されることができます。
- イメージの変換 では、(例えば、イメージのサイズ変更や回転など) どのようにしてBase64 イメージを変換するかについて、また (クライアントでのExif データおよびメモ) 損失などの問題など) 変換に関連する問題について説明されています。
- データベースのイメージ では、どのようにしてイメージがデータベースで保管されるかについて説明されています。

これらのメカニズムは、パワフルなイメージ アクション とイメージに関連したAltova XPath 拡張機能により有効化されています。

イメージ ソース

以下の種類のイメージソースをページデザインで使用することができます:

- PNG、BMP、JPG などの、イメージファイルのよく使う形式。イメージファイルの URL を参照するバイナリファイルソース追記イメージ。
- イメージのテキストエンコードである Base 64 エンコード文字列。Base 64 エンコードを持つイメージは、XPath 式を介して Base 64 エンコード文字列にアクセスします。XPath 式は、通常 Base64 文字列を含むノードを返します。MobileTogether は、Base64 文字列を読み込み、文字列からエンコードされたイメージを生成します。

デザインにイメージを挿入する

デザインにイメージを挿入するには、以下を行います:

1. [イメージ] コントロールをデザインにドロップします。
2. [スタイル & プロパティ] ペインで、イメージ プロパティである **[イメージ ソースの種類]** を、挿入されるイメージの種類に一致させるために URL または Base64 に設定します。このプロパティのデフォルトの設定は URL です。
3. **[イメージ ソース]** プロパティのイメージを指定します。イメージファイルが参照される場合は、URL を指定します。Base 64 エンコード イメージが参照される場合、**[イメージ ソース]** の XPath 式を使用して、Base64 文字列を与えるか、または Base64 文字列を含むノードを与えます。両方のソースの種類 (URL または Base64) には、プロパティの値を指定する代替方法があることに注意してください! (i) **[イメージ ソース]** プロパティを選択して、[ページ ソース] ペインの XPath ツールバーボタンをクリックし、URL または Base64 文字列を評価する XPath 式を入力します; (ii) URL または Base64 文字列を含む XML ノードを [ページ ソース] ペインから [イメージ] コントロールにドロップします。

メモ イメージソースが変更するたびに、(例えばユーザーの選択により)、イメージの [再ロード] アクションが新しいイメージを表示するために必要になります。

URL を介してイメージファイルを挿入する

ブラウズしてイメージファイルを挿入、またはグローバルリソースから選択します。詳細は、**[イメージ ソース]** プロパティを参照してください。URL を使用して挿入する例は、クイック スタート チュートリアルを参照してください。

> **デザインファイルで URL をソースにしたイメージを埋め込む**
> イメージが (Base 64 エンコード イメージとしてではなく) URL から提供される場合、イメージはデザインファイルに埋め込まれることができます。[イメージ] コントロールの **[イメージを埋め込む]** プロパティを使用します。このプロパティが true と設定される場合、イメージは Base64 エンコードに変換され、デザインファイルに埋め込まれます。

Base 64 エンコード イメージの挿入

イメージが Base64 テキストとしてエンコードされると、XML 要素ノードのテキストコンテンツとして保管されることができます。この結果、データ送信しやすくなり、そのメタデータを簡単に解析および取得することができます。下のリストでは、Base 64 エンコード イメージが <png> 要素のコンテンツです。

```
<images><png>iVBORw0KGgoAAAANSUhEU...</png></images>
```

Base 64 エンコード イメージを挿入するには、**[イメージ ソース]** プロパティの XPath 式はイメージの Base 64 エンコー

ドのテキスト文字列を解決しなければなりません。イメージのBase 64 エンコードを含むXML ノードを[ページソースペイン から[イメージ] コントロールへドロップすることができます。

Base 64 エンコードイメージの使用方法の例に関しては、次のセクションのBase64 エンコードイメージを参照してください。

Base64 エンコードイメージ

イメージがBase64 テキストとしてエンコードされる場合、XML 要素ノードのテキストコンテンツとして保管されることができます。下のリストでは、Base 64 エンコードイメージは<png> 要素のコンテンツです。

```
<images><png>iVBORw0KGgoAAAANSUhEU...</png></images>
```

Base 64 エンコードイメージを挿入するには、イメージコントロールの **[イメージ ソース]** プロパティのXPath 式がイメージのBase64 エンコードテキスト文字列を解決する必要があります。イメージのBase64 エンコード文字列を含むXML ノードを[ページソース] ペインから[イメージコントロール] へドロップすることもできます。下の例はBase64 イメージがデザインでどのように使用できるかを説明しています。

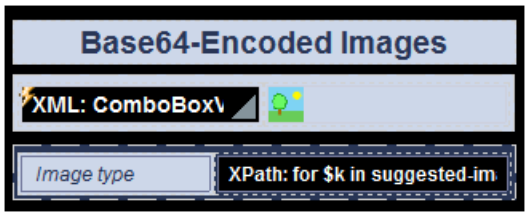
メモ イメージソースが変更するたびに、(例えばユーザーの選択により)、イメージの[再ロード] アクションが新しいイメージを表示するために必要になります。

ファイルの例: Base64Images.mtd

デザインファイル**`Base64Images.mtd`** は マイドキュメント MobileTogether フォルダーにあります: **`MobileTogetherDesignerExamples\Tutorials\Images`**。 MobileTogether Designer でこのファイルを開くことができ、シミュレータ**[F5]** を実行してデザインの定義を確認することができます。

デザインファイルは、(`Tutorials\Images` フォルダにも存在する) XML ファイル**`Base64Images.xml`** に保管される、Base 64 エンコードイメージを使用します。XML ファイルの構造は下のスクリーンショットに表示されています。`images` 要素は、5 つの子要素を持ち、各子要素は、Base64 テキスト文字列として保管される異なるフォーマットのイメージを持ちます。`$PERSISTENT` ツリーは、一時的なユーザー選択(`ComboBoxValue`) と選択されたイメージのExif データが存在する場合、そのデータを保存するために使用されます。

デザイン(*下のスクリーンショット*) には、ページタイトルのラベルと2 つのテーブルがあります。2 番目のテーブルにはラベルと編集フィールドがあります。



コンボボックスからイメージの種類を選択します (*下のシミュレーターを参照*)。コンボボックスは(XML ファイルから) 表示するBase64 イメージを選択するために使用されています。

注意する重要な点:

- ページのロード時イベントは jpg の値を使用して $PERSISTENT/ComboBoxValue ノードを初期化します。
- ([ページ ソース] ペインからノードコンボ ボックスにドロップして) コンボ ボックスをノード $PERSISTENT/ComboBoxValue に関連付けます。 この関連付けは、現在のノードの値がコンボ ボックスに表示されており、コンボ ボックスの選択が自動的にノードを更新します。
- 値の簡単なリスト付きコンボ ボックスのドロップダウンリストが作成されます。
- [イメージ] コントロールの **[イメージソースの種類]** プロパティは Base64 に設定されています。
- [イメージ] コントロールの **[イメージ ソース]** プロパティは次の XPath 式に設定されています: $XML1/images/element()[local-name() eq $PERSISTENT/Root/ComboBoxValue]。これは、ノード $PERSISTENT/ComboBoxValue のコンテンツと等しいイメージ要素の子要素を選択します。 つまり、ノードのコンテンツに一致する名前を持つ XML ファイルの Base 64 エンコード イメージ要素を選択します。
- エンドユーザーがコンボ ボックスからアイテムを選択すると、そのアイテムの値がノード $PERSISTENT/ComboBoxValue に入力されます。このノードの値は XML ファイル内の正確な Base64 イメージ要素を選択するために使用されます。例えば、png がコンボ ボックスで選択されている場合、png が $PERSISTENT/ComboBoxValue ノードの値として入力されます。**[イメージ ソース]** プロパティの XPath 式は、XML ファイルの png 要素を選択し、コンテンツ (Base 64 エンコード PNG イメージ) をイメージとして表示します。
- もう一つ重要なアクションがあります。コンボボックスで新しい値が選択されると、イメージが再ロードされることを指定しなければなりません イメージが再ロードされると、$PERSISTENT/ComboBoxValue の値を読み込み、XML ファイルから対応するイメージを取得します。
- 2 番目のテーブルの、イメージの種類は XPath 拡張子関数 **suggested-image-file-extension** を使用して Base 64 エンコード テキスト文字列 から取得されたものです。この関数は文字列 (Base64 イメージ) を引数として使用し、ファイル拡張子の情報を文字列から取得します。Base64 文字列に使用可能なファイル拡張子に情報が存在しない場合、空の文字列が返されます。使用される XPath 式は

  ```
  for $k
  in suggested-image-file-extension($XML1/images/element()[local-name() eq $PERSISTENT/Root/
  ComboBoxValue])
  return if ($k != '') then $k else "Data not available"
  ```

  式は、**suggested-image-file-extension** 関数により返されたファイル拡張子変数 ($k) を作成します。変数が空でない場合、コンテンツは表示されます。その他の場合は適切なメッセージが表示されます。

次のセクション エクスチェンジャブル イメージ ファイル フォーマット (Exif) では、Exif データに関連するデザインの残りの部分について説明されています。

エクスチェンジャブル イメージ ファイル フォーマット (Exif)

エクスチェンジャブル イメージ ファイル フォーマット (Exif) は、デジタルカメラとスマートフォンのカメラで使用されるイメージフォーマットを定義する基準です。Exif 基準のメタデータのタグは、イメージの日時と位置情報からカメラ設定とイメージ構成の詳細など広範囲の情報を格納します。Exif イメージが Base64 エンコードに変換されると、イメージのメタデータもまた Base64 に変換され、使用するために取得することができます。

メモ すべてのデジタルカメラまたはスマートフォンが Exif データを提供するわけではありません。

MobileTogether Designer の Exif 機能

MobileTogether Designer には次の Exif に関連する機能が搭載されています:

- [ユーザーがイメージを選択] アクションは、エンドユーザーのクライアント デバイスでカメラ アプリケーションを開始するオプションを提供します。撮影された写真は Base64 エンコード イメージとして XML ノードに保存されます。カメラ アプリケーションが Exif フォーマットを使用する場合、もまた Base64 エンコード イメージに保存されます。このデータは XML ノードからの即時取得の際に使用することができます。
- **`image-exif-data`** と呼ばれる Altova XPath 拡張関数は、takes a Base64-エンコード JPEG イメージを引数として取り、文字列内に含まれる Exif メタデータを属性の値ペアとして返します (詳細に関しては、**`image-exif-data`** 関数の説明を参照してください。イメージの次元を検索するには、MobileTogether XPath 拡張関数 **`mt-image-width-and height`** を使用してください)。
- **`suggested-image-file-extension`** と呼ばれる Altova XPath 拡張機能は、Base64 文字列を引数とし、(`jpg`、`png`、`bmp` などの) イメージ ファイルの拡張子を返します。これは、イメージ フォーマットを自動的に検索、修正し、適切なファイル拡張子と共にファイルを保存する際に役に立ちます。
- イメージをファイルにコード保管アクションは、Base64 エンコード イメージの (`jpg`、`png`、`bmp` などの) バイナリ イメージ フォーマットでの保存を有効化します。Exif データは Base64 エンコード テキストに保存されます。

下の例はどの様に Exif データが Base64 エンコード イメージから取得される、またこのデータがどのようにソリューションで使用されるかをかを説明しています。

メモ Exif データはイメージがサイズ変更または回転されると失われます。

ファイルの例: Base64Images.mtd

デザインファイル **`Base64Images.mtd`** は マイドキュメント MobileTogether フォルダにあります: **`MobileTogetherDesignerExamples\Tutorials\Images`**。 MobileTogether Designer でファイルを開き、シミュレーター **[F5]** を実行し、デザインの定義を確認します。デザインのデフォルトのファイルは Exif メタデータ付きの 1 つのイメージを含みます。

基本デザイン

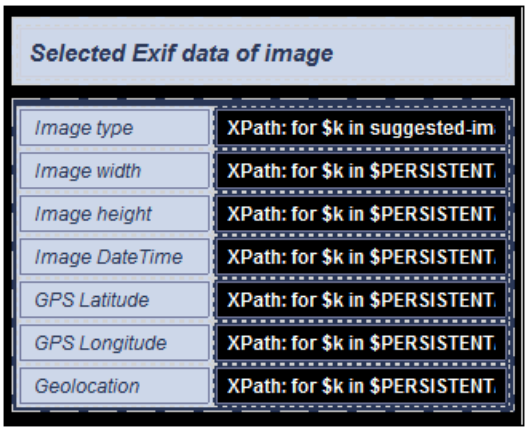
デザインファイルは (チュートリアル フォルダーにも保存されている) XML ファイル **`Base64Images.xml`** に保存されている、Base64 エンコード イメージを使用します。XML ファイルの構造は下のスクリーンショットに表示されています。`images` 要素は、Base64 テキスト文字列として保管されている異なるフォーマットのイメージを持つ 6 つの子要素を持ちます。Exif メタデータ (`exif` 要素) を持つ 1 つのイメージがあります。`$PERSISTENT` ツリーは、一時的にユーザーの選択 (`ComboBoxValue`) と Exif メタデータ (`ExifData`) を保存するために使用されます。

デザインのトップの部分 (*下のスクリーンショット*) にはページ タイトルのラベルおよび 2 つのテーブルがあります。ページのこの部分のデザインは、前のセクション Base64 エンコード イメージで説明されています。目的はエンド ユーザーがコンボ ボックスからイメージの種類を選択することです。この選択は、コンボ ボックスの右のセルに表示される XML ファイルの Base64 エンコード イメージの選択を決定します。

ユーザーがコンボボックスの **exif** アイテムを選択すると、XML ファイルの `exif` 要素の Base64 エンコード イメージが表示されます。Exif メタデータは 2 つのテーブルで表示されています (*「イメージの選択されたExif データ」と「選択されたイメージのExif メタデータ」; 下のシミュレーターのスクリーンショットを参照してください*)。シミュレーターでは、XML データペインの `$PERSISTENT` ツリーを展開すると、(*下のスクリーンショット参照*)、Base64 文字列から取得された Exif データを確認することができます。2 つの Exif データの表示テーブルは下に説明されています。デザインの他の箇所の詳細に関しては、前のセクション Base64 エンコード イメージを参照してください。

イメージの選択された Exif データ

- 選択された Exif データは、2 つの列と複数の行からなる静的テーブルとして表示されます (*下のスクリーンショット*)。
- 最初の列はラベルを含みます; 2 番目の列は、各自が 1 つの Exif *メタデータを*返す XPath 式を持つ編集ボ*ックス*を含みます。



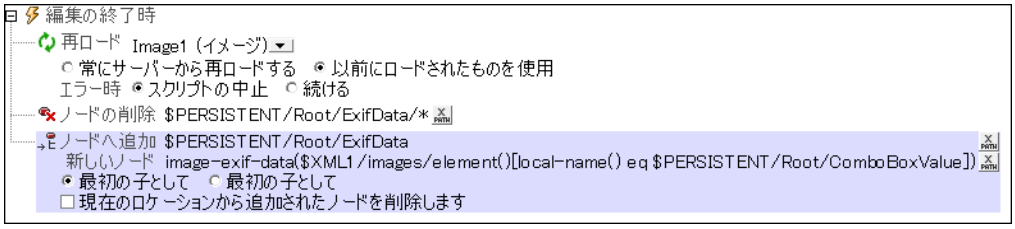
- *イメージの種類* の情報は、Altova XPath 拡張関数 **suggested-image-file-extension** を使用して、Base64 エンコードテキスト文字列から取得することができます。この関数は、文字列 (Base64 イメージ) を引数とし、文字列からファイル拡張の情報を取得します。ファイル拡張の情報が Base64 文字列に存在しない場合、関数は空の文字列を返します。使用される XPath 式は次のとおりです:

```
for $k in suggested-image-file-extension($XML1/images/element()[local-name() eq
$PERSISTENT/Root/ComboBoxValue])
return if ($k != '') then $k else "Data not available"
```

  関数が空ではないまたは空の文字列を返すかにより、式が代替の返される文字列を提供します。空ではない文字列が返されると、文字列が表示されます; 空の文字列が返されると、適切なメッセージが返されます。
- (最初の行以外の) テーブルのその他の XPath 式は、1 つの Exif メタデータを得るために Altova XPath 拡張関数 **image-exif-data** を使用します。この関数は文字列 (Base64 イメージ) を引数として、Exif メタデータを持つ属性付き (**Exif という名の**) 要素ノードを返します。各属性 値ペアは 1 つの Exif メタデータタグに相対します。下の式では、**image-exif-data** 関数は、複数の属性付き **Exif** 要素を返します。この式から取得するメタデータの情報は、イメージの幅です。この情報は `Exif` 要素の `@PixelXDimension` 属性ノードに保管されています。

```
for $k in $PERSISTENT/Root/ExifData/Exif
return if ($k/@PixelXDimension != '') then $k/@PixelXDimension else "Data not available"
```

  式は **Exif/@PixelXDimension** ノードが空でないか確認します。空でない場合、文字列が表示されます; その他の場合は、適切なメッセージが表示されます。
- **image-exif-data** 関数に関しての詳細は、Altova 拡張機能のセクションを参照してください。
- *テーブル内最後の位置情報* の値に注意してください。これは、Altova により作成された `Exif/@Geolocation` 属性を介して得られたものです。
- `$PERSISTENT/Root/ExifData` ノードは、子ノードを **image-exif-data** 関数の結果を含むものに追加して、Exif データと共に作成されます。これは、表示するイメージを選択するコンボボックスの [ノードの追加] アクションを指定することにより行われます。(*下のスクリーンショット参照*)。アクションはコンボボックスの編集の終了時 イベントによりトリガーされます。

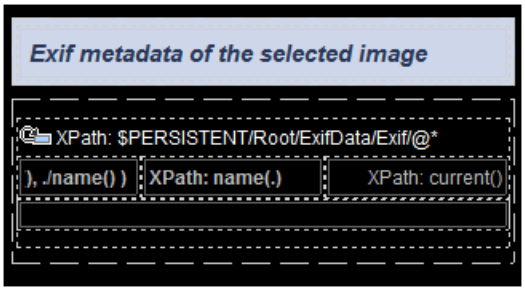


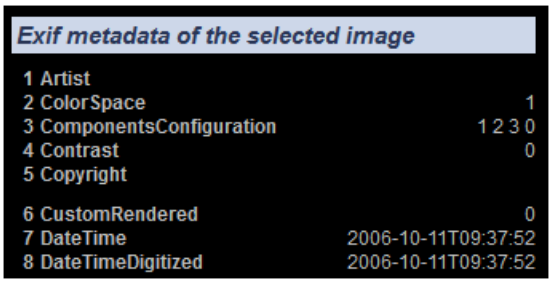
以下の点に注意してください:

(i) [ノードの追加] アクションのXPath ロケーター式は、$PERSISTENT/Root/ComboBoxValue 内の文字列と同じ名前のXML ファイル内のノードをロケートします。
(ii) $PERSISTENT/Root/ExifData ノードは、(Exif ノード内の) Exif データがExifData に追加される前に削除されます。
(iii) $PERSISTENT ツリー内のExif ノードを削除するページアクションが設定されます。これにより、初期のイメージ(jpg) とExifData 内の古いExif データの間の不調和を避けることができます。

イメージのすべての Exif データ

- 繰り返し行のあるテーブル(*左下のデザインのスクリーンショット; 右下のシミュレーター ビュー*) は、**image-exif-data** 関数から返されたすべての属性値ペアを表示するために使用されています。
- 繰り返し行は**image-exif-data** 関数より返された**Exif** 要素ノードすべての属性を選択するXPath 式と共に指定されます:
  $PERSISTENT/Root/ExifData/Exif/@*

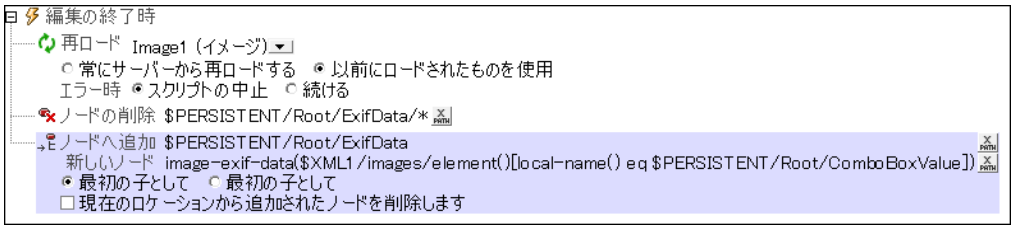




- テーブルの最初の列は、現在の属性のインデックスの位置を含みます: index-of(./@*/name(), /name())
- 2 番目の列は現在の属性の名前を含みます: name(.)
- 3 番目の列は現在の属性の値を含みます: current()
- すべてのイメージが同じExif メタデータを持つわけではありません。メタデータの一部は、不完全な場合があります; 他の場合、追加
- メタデータがある可能性もあります; また他の場合、メタデータは基準外の、ベンダ固有のタグにタグ付けされている場合もあります。結果的には、使用可能な メタデータと、どの属性名の下で使用可能かを確認することは重要です。この情報をもって特定の属性の値を取得することができます。
- 返された属性の名前を把握すると、以下のように**image-exif-data** 関数を使用して値にアクセスすることができます: **image-exif-data(Base64String)/@WantedAttribute**。関数は**Exif** 要素を返すことに注意してください。

Exif データを使用して $PERSISTENT ツリーを作成する

- **image-exif-data** 関数により返されたすべての属性 値ペアを一覧することはとても役に立ちます。属性 値ペアを表示するには、この出力を手軽に一時的 $PERSISTENT ツリーに保管することができます。
- デザインの例では、$PERSISTENT/Root/ExifData ノードが、子ノードを、**image-exif-data** 関数の結果を含むExifData ノードに追加することにより、Exif データと共に作成されています。
- これは、表示するイメージを選択するコンボ ボックスのを[ノードの追加] アクションを指定しておこないます(下のスクリーンショット参照)。[ノードの追加] アクションはコンボ ボックスの編集の終了時 イベントによりトリガーされます。



- [ノードの追加] アクション のXPath ロケーター式は、$PERSISTENT/Root/ComboBoxValue の文字列と同じ名前を持つXML ファイルのノードをロケートします。
- $PERSISTENT/Root/ExifData ノードは、**image-exif-data** 関数から返されたExif データが ExifData ノードに追加される前に、から削除されます。
- 返される属性の名前を把握することにより、以下のような **image-exif-data** 関数を使用して属性の値にアクセスすることができます: **image-exif-data(Base64String)/@WantedAttribute**。関数は **Exif** 要素を返すことに注意してください。

エンドユーザーにより選択されたイメージ

[ユーザーがイメージを選択] アクションでは、エンドユーザーがデータソースにイメージを保存することを選択することのできるソリューションをデザインすることができます。エンドユーザーが選択するイメージは既にギャラリー(フォルダ)に存在するイメージまたは、モバイルデバイスのカメラアプリケーションを使用してユーザーが撮るイメージです。後者の場合、[ユーザーがイメージを選択] アクションによりカメラアプリケーションが自動的に開かれ、にユーザーが撮ったイメージを指定されたデータソースノード保存します。(ギャラリーとカメラ) 両方の場合ともイメージはXML ノードにBase64 エンコードイメージとして追加されます。

2 番目のアクション[イメージをファイルに保存] は、データソースノードのイメージをイメージファイルに(適切なイメージファイル拡張子と共に) 保存します。

ファイルの例 **UserSelectedImages.mtd** には、モバイルデバイスのギャラリーからエンドユーザーがイメージを選択することのできるデザインがあります。このイメージは、データソースのXML ノードに、Base64 エンコード文字列として自動的に保存されます。Base64 エンコードイメージは、イメージとして、デザイナーにより選択された(また、デザインにより定義された)場所に自動的に保存されます。

メモ イデザインでイメージが表示される場合、メージソースが変更するたびに、(例えばユーザーの選択により)、イメージの[再ロード] アクションが新しいイメージを表示するために必要になります。

ファイルの例: UserSelectedImages.mtd

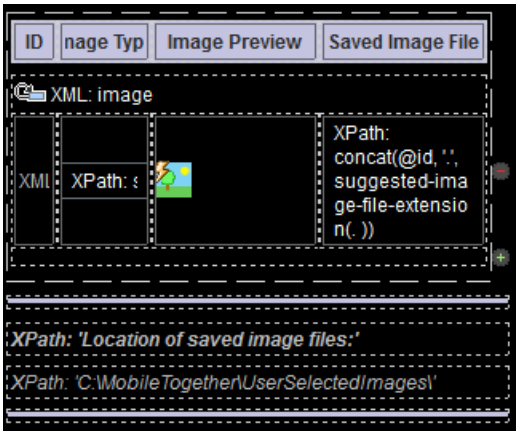
デザインファイル**UserSelectedmages.mtd** は、*マイドキュメント* MobileTogether フォルダーにあります: **MobileTogetherDesignerExamples\Tutorials\。**MobileTogether Designer でファイルを開き、シミュレーター**[F5]** を実行し、デザインの定義を確認します。以下を行ってください:

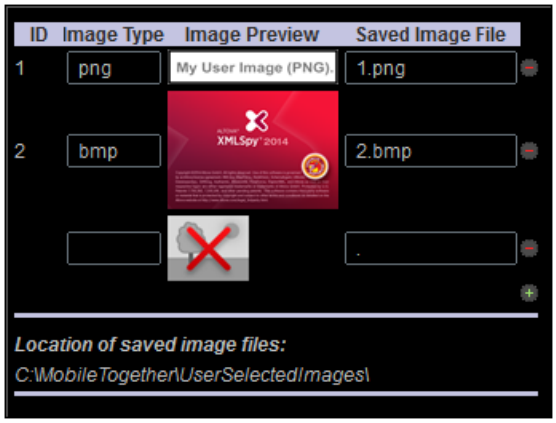
- このフォルダーはユーザーが選択するイメージが保存される場所とデザインで定義されているため、フォルダー**C:\MobileTogether\UserSelectedImages** を作成します。または、他の場所をイメージのクリック時 イベントの*イメージをファイルにコード保管アクション* アクションの保存場所として定義することもできます。
- MobileTogether Server 設定ページで、*サーバー側ソリューション作業ディレクトリ* をデザインのデフォルトファイル`UserSelectedImages.xml` の祖先ディレクトリとして設定します。これにより、デフォルトファイルがイメージのクリック時 イベントの[ファイルへ保存] アクションがトリガーされると更新されます。作業ディレクトリがデザインで参照されるすべてのファイルのベースとなるため、作業ディレクトリを**C:\MobileTogether** と設定し、デフォルトファイルを保存することを奨励します。このような方法で、イメージフォルダーとデフォルトフォルダーが**C:\MobileTogether** の同じベースURI で相対します。

デザインは、(チュートリアル フォルダーにもある) **UserSelectedImages.xml** と呼ばれるXML ファイルの単一のデータソースがあります。XML ドキュメントの構成は、下のスクリーンショットに表示されています。`images` 要素は、複数の**image** 子要素をもつことができます。各、`image` 要素は、`id` 属性を持ち、`image` 要素のコンテンツは、ユーザーにより選択されるイメージのBase64 エンコードとなります。ユーザーにより選択されたすべての新しいイメージは、自動的に`image` 要素に追加されます。

ノード**$PERSISTENT/Root/@ImagePath** は、イメージが保存されるフォルダーのパスを保管するために作成されます (*下のスクリーンショット参照*)。**C:\MobileTogether\UserSelectedImages** のデフォルトの値として設定されています。イメージが保存されるフォルダーの場所を変更する場合、パスを直接デザイン内のページソースペインで変更する事ができます: ダブルクリックし、名前を与え編集してください。

デザイン(*左下のスクリーンショット*) は、ページ タイトルを表示するラベルと動的 テーブルにより構成されています。動的なテーブルには、ノード**$XML1/images/image** に関連したヘッダー行と繰り返し行があります。これは、`image` 各要素に対し行が繰り返されることを意味します。また、各 `image` 要素が自身の行に作成されることを意味します。右下のスクリーンショットは、シミュレータで実行されたソリューションを表示します。説明は下に述べられています。



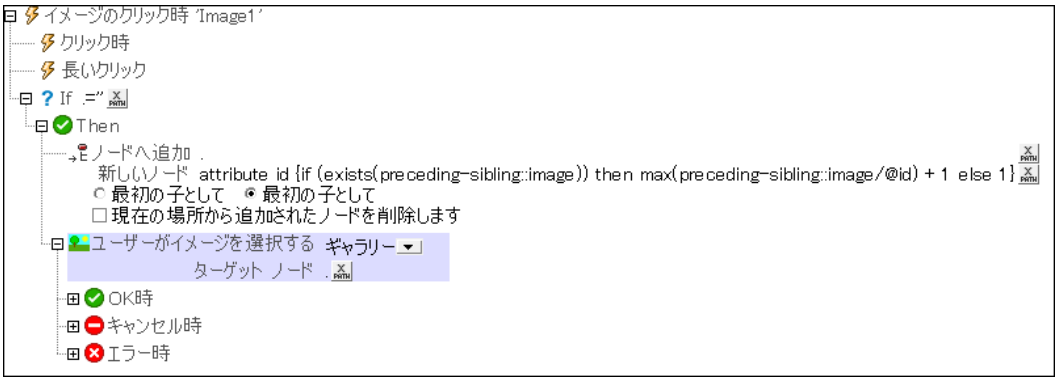


デザインに関して以下の点に注意してください:

- 動的テーブルには (テーブルのコンテキストメニューで追加することのできる) **[追加 削除]** ボタンがあります。これによりユーザーは、新しい`image`-行を追加し、`image`-行を削除することができます。
- 各`image`-行には4 つの列があります: *ID*; *Image Type*; *Image Preview*; および*保存されたイメージファイル*(の名前)。
- 新しい`image`-行が追加されると、イメージのプレースホルダが作成されクリックすることができます(*右上のスクリーンショット参照*)。
- イメージのプレースホルダがクリックされると、`@id` 属性を`image` 要素に追加することができます。
- `@id` 属性の値は、既存のイメージID より1 つ大きく計算されます。これにより各 ID 値の一意性を保つことができます。これにより、各 ID 値の一意性が保障されます。前に**`image`** 要素が存在しない場合、追加されるイメージは、最初の**`image`** 要素となり、**`1`** の値が要素の`@id` 属性に割り当てられます。XPath 式は、コンテキストノードとして、`image` 要素とともに定義されます: `attribute id {if (exists(preceding-sibling::image)) then max(preceding-sibling::image/@id) + 1 else 1}`
- XPath 式は、コンテキストノード `max(preceding-sibling::image/@id) + 1` としての`image` 要素とともに定義されます。
- *ID* 列は、**`$XML1/images/image/@id`** ノードと関連する[ラベル] コントロールを持ちます。(ノードをコントロールにドロップすることで作成された) この関連は、現在の行のID セルの現在の**`image`** 要素の**`@id`** 属性の値を表示する効果をもちます。
- *イメージの種類の列* にはテキスト文字列からファイルタイプの情報を取得するXPath 式付き[編集フィールド] コントロールがあります。XPath 式は、現在のノード(現在の**`image`** 要素) を関数 **`suggested-image-file-extension`** の引数として提出します。 関数は、ファイルタイプの情報のBase64 エンコード文字列を解析し、ファイルの拡張子を文字列として返します。
- *Image Preview* 列は、[イメージ] コントロールを含みます。コントロールには、[イメージソースの種類] プロパティがあり、`base64` に設定されています。[イメージ ソース] プロパティはXPath 式`current()`に設定されています。現在の`image` 要素は現在のノードです。イメージソースの種類 の設定は、イメージ要素のコンテンツが(URL ではなく) Base64 テキストとして読み込まれるかを決定します。
- [イメージ] コントロールには、イメージのクリック時 イベントのために定義された複数のアクションがあります
- 4 番目の列の*Saved イメージファイル*は、ディスクに保存されたイメージ ファイルに名前を与えます。Altova XPath 拡張子関数 **`suggested-image-file-extension`** を使用して、イメージ ファイル拡張子を与えます。

イメージのクリック時 イベントのアクション

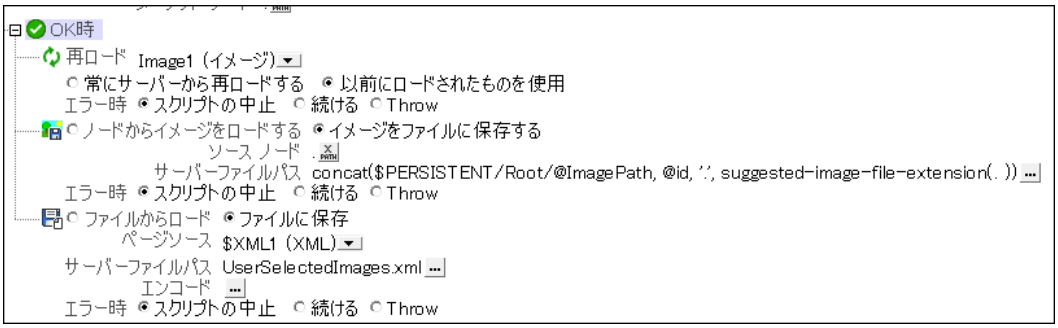
- [イメージ] コントロールの イメージのクリック時 イベントは、下のスクリーンショットに表示されるアクションに割り当てられました。



- 現在のノード(image 要素) を指定するIf 条件が空の場合、イメージがクリックされると、新しい**id** 属性が作成され、現在空のimage 要素の子として追加されます。(ユーザーがテーブル行を追加すると空のimage 要素が追加されます。(*上のシミュレーターのスクリーンショットを参照してください*)。) id 属性はXPath 式を介して値を計算することを割り当てられます: if (exists(preceding-sibling::image)) then max(preceding-sibling::image/@id) + 1 else 1. この式は、既存のイメージID よりも1 つ大きい値を返し、各 ID 値の一意性を保ちます。イメージが存在しない場合、新しく追加されたイメージには**1** のID 値のが与えられます。
- [ユーザーがイメージを選択] アクションは、モバイルデバイスのフォルダー(ギャラリー) からイメージが選択されることを指定します。これにより、ユーザーは、イメージがクリックされるとイメージを参照することができます。アクションのターゲットノードはBase64 エンコードイメージが保存される場所です。例では、ターゲットノードは、Image 要素である現在のノードです。
- [ユーザーがイメージを選択] アクションには3 つの条件があります:OK 時 、キャンセル時、および エラー時。これらについては下で個別に説明されています。

OK 時: イメージの再ロード + イメージをロード、ファイルに保存 + ファイルに保存

- **OK 時** の条件は、指定されたデータソース ノードへのイメージのインポートに成功すると3 つのアクションを実行します(*下のスクリーンショット参照*): (i) イメージの[再ロード] アクション; (ii) データソース ノードからイメージ ファイルへイメージを保存する[ファイルにイメージを保存] アクション; (iii) ソース ツリー(クライアント/サーバー上) のデータをデータソース ファイルに保存する[ファイルに保存] アクション。

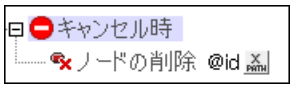


- [再ロード] アクションは、[イメージ] コントロールのために設定されます。これにより、[イメージ] コントロールのイメージ ソース プロパティで指定されたイメージが再ロードされます。イメージ ソース プロパティの値は、現在のノードに設定されているため、また、現在のノードは、ユーザのにより選択されたイメージのターゲット ノードであるImage 要素であるため、現在の行のためのイメージのプレビューのセルは、ユーザーのイメージと共に再ロードされます。
- [イメージをファイルに保存] アクション(*下のスクリーンショット*) は、データソース ノードのイメージをイメージ ファイルに保存します。ソース ノードは(image 要素である) 現在のノードに設定されています。バイナリイメージ ファイルはこのノードのBase64 データから生成されます。

- 設定オプションは、バイナリのイメージファイルが保存される場所を指定します。XPath 式は、イメージが保存される場所とイメージのファイル名を生成します。イメージフォルダーへのパスを持つ **$PERSISTENT** ツリー内のノードを指定します。**@id** 属性はファイル名を与え、Altova XPath 拡張関数 **suggested-image-file-extension** は、ファイルの拡張子を与えます。
- [ファイルに保存] アクション は、サーバー上のデータソースツリーのデータを指定されたデータソースファイルに保存します。

キャンセル時: ノードの削除

ユーザーがイメージの選択プロセスをキャンセルする場合、@id ノードは[ノードの削除] アクションから削除されます。注意：@id ノードは、(イメージのプレースホルダをクリックすることにより) イメージ選択プロセスが開始されると作成されます (*上の "ファイルの例 "のセクションを参照してください*)



エラー時: メッセージボックス + ノードの削除

指定されたXML ノードに、イメージをBase64 データとしてインポート中にエラーが発生した場合、エラー時 条件式に定義されたアクションが実行されます。エラーメッセージが表示され、@id ノードが削除されます。@id ノードは、イメージの選択プロセスが開始されると作成されます (*上の "ファイルの例 "のセクションを参照してください*)。

イメージの変換

Base64 エンコードイメージは、Altova XPath 拡張機能 **mt-transform-image** により変換 (サイズ変更、回転、画質 /ファイルのサイズに合わせて変更) されることができます:

```
mt-transform-image(Base64Image as Base64BinaryString, Size as item()+,
Rotation as xs:integer, Quality as xs:integer) as Base64BinaryString
```

関数は Base64 エンコードイメージを最初の引数として返し、変換された Base64 エンコードイメージを返します。第 2、第 3、第 4 引数は変換されたイメージパラメータです: サイズ、回転、画質。関数と使用法の例の詳細に関しては、XPath/ XQuery 関数: イメージに関連したのセクションを参照してください。

以下の点に注意してください:

- 変換の入力イメージは Base64 エンコードイメージであり? イメージファイルではありません。
- Base64 エンコードの Exif データは、変換されたイメージ内で失われます。
- クライアントで変換が行われた場合、クライアントでメモリの問題が生じるかもしれません。*以下を参照してください。*

クライアントまたはサーバーに対しての変換

明確に指定されていない場合、関数 **mt-transform-image** はクライアントに対して実行されます。これはクライアントの一部でメモリに関する問題を起こす可能性があります。変換が開始されると、イメージが Base64 エンコードから、大きくなる可能性のある BMP フォーマットへとアンパックされます。変換が終了すると、変換されたファイルは元のフォーマットで保管されます。大きな BMP フォーマットは、クライアントの一部でメモリに関する問題を起こす可能性があり、この点に注意する必要があります。

クライアント上でメモリに関する問題を回避するため、明示的に変換をサーバーで実行するように指定します。アクションに対して実行を使用して実行し、サーバーで子アクションが実行されるように指定します。このアクションに対して実行 のすべての子アクションはサーバーで実行されるようになります。ノードの更新 などのアクションを使用して変換の結果と共にノードの更新を行うことができます。変換されたイメージと共にターゲットノードが更新されます。アクションの処理が完了、またはワークフローがクライアントに切り返されると MobileTogether はクライアントの結果を自動的に送信します。

データベースのイメージ

データベースのイメージは、バイナリデータがエンコードされることができるテキストフォーマットである、Base64 フォーマットで保管することができます。

11.3 オーディオ、ビデオ

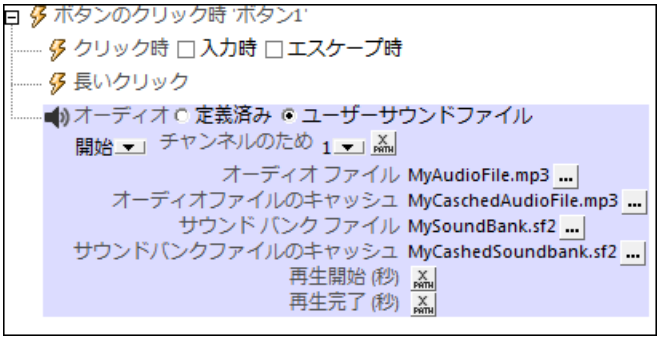
このセクションはMobileTogether のオーディオとビデオ機能の概要について説明します。このセクションは以下のセクションから構成されています:

- オーディオの再生
- オーディオの録音
- ビデオの再生
- オーディオ/ビデオのフォーマット

オーディオの再生

MobileTogether のオーディオの再生機能を使用することにより、(クライアント上で使用できる)定義済みのオーディオサウンド、または、モバイルデバイス、または、遠隔の場所にあるオーディオファイルを再生することができます。 オーディオは5つのチャンネル(1 から5)で再生することができます。各オーディオアクションは、1つの特定のチャンネルのために定義されています。ですから、例えば、次のアクションのシーケンスを定義することができます

オーディオの再生は、オーディオ(再生) アクションにより管理することができます。各オーディオアクションは以下のうちの1つを定義することができます: (i) 特定のチャンネルの特定のファイルを再生する(*下のスクリーンショット参照*)、(ii) 特定のチャンネルの再生を一時停止する、(iii) 特定のチャンネルの再生を再開する、(iv) 特定のチャンネルの再生を停止する、および(v) 特定のチャンネルのオーディオファイルの再生内の特定の場所にジャンプする。詳細に関しては、オーディオアクションの詳細を参照してください。通常、オーディオアクションは、ボタンクリックなどのコントロールイベントで定義されます。イベントは関連したオーディオアクションをトリガーします。



オーディオの再生 機能は以下のように動作します:

- オーディオの開始アクションがトリガーされると、選択された定義済みのサウンド、または、オーディオファイルが指定されたチャンネルで再生されます。
- (クライアントデバイスで使用することができる) 次の定義済みのサウンドから選択することができます:

```
ClickOffOn, ClickOnOff, Ding, DingDong, ErrorDeepBuzz, ErrorWhoops, Goodbye,
KeyClickTick, KeyClickTock, MessageBounce, MessageXylophone, WhooshDeep,
WhooshExhale, WhooshLong, WhooshQuick, WhooshQuicker
```

- アクション内で名前の付けられたオーディオファイルが、クライアント上に存在する場合、指定されたチャンネルで再生されます。ファイルが遠隔のサーバーにファイルが存在する場合、ファイルはクライアントにダウンロードされます。ローカルキャッシュファイルが設定内で指定されている場合、ダウンロードされたデータはローカルファイルに保存されます。指定されたキャッシュファイルが既存の場合、キャッシュファイルが再生され、ダウンロードはおこなわれません。
- MIDI ファイルの再生は、Web ブラウザー以外のすべてのクライアントデバイスでサポートされています。iOS デバイスでは、MIDI ファイルの再生は、サウンドバンクファイルを必要とします。このファイルのロケーションは開始アクション内に入力されなければなりません(*上のスクリーンショットを参照*)。
- 開始アクションでは、オーディオファイル全体が再生されるか、または、一部分のみが再生されるかを指定することができます。セグメントは開始と終了時間により定義されます(*上のスクリーンショットを参照*)。
- 各開始アクションは1 から5 の番号を付けられているチャンネルに割り当てられています。ですから、同時に5つのオーディオストリームを実行することができます。各オーディオアクションは特定のチャンネルのために定義されており、アクションの設定はそのチャンネルに適用されます。
- 一時停止、再開、および停止アクションは簡単なアクションで、ボタンなどのコントロールで通常定義されます。これらのアクションのそれぞれは特定のチャンネルのために定義されます。特定のチャンネルで再生されているオーディオファイルのためのアクションを実行します。
- シークアクションは指定されるチャンネルで再生されているオーディオファイルに適用され、ファイル内の特定のポジションに移動することができます。

メモ オーディオストリームがソリューションが中断されている場合に再生されている場合、再生は一時停止されます。再生

はソリューションが再開されると継続されます。

メモ Windows Phone では、マルチチャンネルのオーディオおよびビデオの再生はサポートされていません。1つのオーディオ**または**ビデオファイルのみ1度の再生することが可能です。これは最後に再生されたファイルを意味します。

メモ オーディオ/ビデオファイルは、MobileTogether Designer プロジェクトのサーバーメカニズムにデプロイを使用して MobileTogether Server にデプロイすることはで**きません**。しかしながら、URL を使用してストリームすることはできませんが、サーバーに手動でオーディオ/ビデオファイルをコピーすることができます。MobileTogether Server に存在する、オーディオ/ビデオファイルをストリームするには、以下を行います：(i) バイナリのロード アクションを使用して、バイナリのオーディオ/ビデオデータをデータソースノードにロードします。(ii) バイナリの保存 アクションを使用して、このノード内のデータをクライアント上のファイルに保存します。(iii) オーディオ/ビデオの再生アクション を使用して、クライアントデバイス上に保存されたファイルを再生します。または、オーディオ/ビデオファイルを MobileTogether Server に保存する代わりにWeb サーバーに保存し、URL を使用して、Web サーバーからオーディオ/ビデオファイルをストリームします。

オーディオの再生イベント

オーディオの再生イベントはプロジェクト全体のために定義することができます。3つの定義済みのオーディオの再生イベント(下のリスト)が存在します。MobileTogether の使用することのアクションはこれらのイベントのために定義することができます。これらのイベントアクションが定義されるダイアログにアクセスするには、**オーディオアクション** のプロジェクトプロパティの**追加ダイアログ**ボタンをクリックします。これらの各イベントはプロジェクト全体に適用することができ、各イベントはチャンネルのオーディオによりトリガーされることができます。`$MT_AudioChannel` 動的な変数はイベントによりトリガーされたチャンネルを含んでいます。例えば、ユーザーがチャンネル2で実行されるオーディオファイルを再生すると、**オーディオの開始時** イベントアクションのXPath 式内の`$MT_AudioChannel` 変数を使用することができます。アクションは、例えば、チャンネル2で実行されているオーディオファイルについての情報を表示することができます。

- **ビデオの開始時**: イベントが発生する前に(すなわち、ビデオの再生が開始する前に)、ビデオファイルの詳細を使用することはできない場合は、ビデオの高さ 幅、時間、および現在の位置を取得する関数を呼び出してはなりません(下を参照) 。この時点では、`mt-audio-is-playing` 関数が有効な情報を返します。このイベントは、例えば、(ノードの更新アクション などを仕使用して)ビデオの再生の再生をXML ツリーノードにログするために使用できます。
- **ビデオのエラー時**: 可能性のあるエラー: ファイルが見つからないエラー、ファイルフォーマットのエラー、またはダウンロード/再生の中断エラーなど。MobileTogether XPath 実行関数 `mt-external-error` を使用してエラーに関する情報を取得することができます。アクションがイベントのために定義されている場合、これらのアクションは実行されます。それ以外の場合は、エラーはメッセージボックスに表示されます。
- **ビデオの完了時**: ビデオの再生は(停止アクションが実行されることなく)ファイルまたは指定されてセグメントの再生が最後に到達すると完了したものと考えられます。これらのアクションはビデオが中断(プロジェクトプロパティ*他のソリューションに切り替える*)または一時停止されると**実行されません**。

オーディオの再生に関連した MobileTogether XPath 拡張関数

以下のオーディオの再生に関連したMobileTogether XPath 拡張関数 を使用することができます:

- **mt-audio-get-current-position(*ChannelNumber*** を`xs:integer` として) を `xs:decimal`として
- **mt-audio-get-duration(*ChannelNumber*** を`xs:integer` として) を`xs:decimal`として
- **mt-audio-is-playing(*ChannelNumber*** を`xs:integer` として) を`xs:boolean` として

XPath 式内のこれらの関数をデザイン内で使用することができます。例えば、オーディオの再生のポジションを秒数でユーザーに表示するなど。オーディオの再生が開始する前は、オーディオファイルの詳細を確認することができません。この結果、ファイルの長さ また、現在の再生箇所に関する情報を確認することができません。関連する機能は再生が開始してからのみ使

用してください。

オーディオの録音

MobileTogether のオーディオの録音機能を使用することにより、クライアントデバイスのオーディオの録音アプリからオーディオが録音されクライアントデバイス上のファイルに保存されることができます。

オーディオの録音は、オーディオの録音 (開始) アクションを使用して開始され、オーディオの録音 (停止) アクションを使用して停止されます(*下のスクリーンショット参照*)。詳細に関しては、オーディオの録音アクションの詳細 を参照してください。通常各オーディオの録音アクションは、ボタンクリックなどのコントロールイベントに割り当てられます。イベントが発生すると、オーディオの録音アクションがトリガーされます。例えば、1つのボタンが録音の開始に使用され、他のボタンが録音の停止に使用されます。エンドユーザーが録音が開始されたページから移動した場合、または、ソリューションが中断された場合オーディオの録音は停止されます。

オーディオの録音機能は、以下のように動作します:

- オーディオの録音の開始 アクションがトリガーされると、デバイスの録音アプリが開始され、アクション内のファイル名にオーディオの録音が開始されます (*上のスクリーンショット参照*)。このファイルはクライアントデバイス上に存在する必要があります。
- 各オペレーティングシステムのために使用するエンコーダー (コーデック) を指定することができます。この設定を空白にすると、各デバイスのデフォルトのコーデックが使用されます。Android システムに対しては、録音されるファイルのファイルフォーマットを指定することができます。
- 録音されるファイルの録音時間の長さとファイルサイズを指定することができます。これらの制限のうちの1つに到達すると、録音は停止され、完了したものと解釈されます。
- 録音のサンプリング速度とエンコードビット速度を指定することができます。これらの設定を空白にすると、デフォルト設定の録音コーデックが使用されます。ユーザー自身の設定を指定する場合は、オーディオのエンコード基準またはエンコーダー仕様を確認してから行ってください。
- オーディオの録音の停止アクションはそのページで開始されたオーディオの録音アクションを停止します。

メモ オーディオの録音中にエンドユーザーがページを移動すると、または、ソリューションが中断される場合、録音は停止されます。2番目の録音アクションが最初のアクションの実行中に開始されると、最初の録音アクションは停止されます。これは、最初の録音アクションが中断されたものとして解釈されるためです。

メモ 特にiOS デバイスでの再生の状態に影響をもたらすため、オーディオは、オーディオまたはビデオの再生中に録音することを控えてください。

オーディオの録音イベント

オーディオの録音は各ページで定義されます。以下の2つのイベントを使用することができます: **オーディオの録音エラー時** と **オーディオの録音の完了時** 。これらのイベントのために定義されたアクションは、**ページのすべてのオーディオの録音に適用**

されます。これらのイベントは(i) オーディオの録音 アクションプロパティの**追加ダイアログ**ボタンをクリック または(ii) **デザインを右クリックして、ページオーディオのアクション**を選択することで実行することができます。各イベントのために、実行されるアクションを左側のアクションペインからイベントタブにドラッグアンドドロップすることにより定義することができます。

- **オーディオの録音エラー時**: 可能なエラーは以下の通りです: ファイルの不足、ファイルフォーマットエラー、録音の中断など。MobileTogether XPath 拡張関数 `mt-external-error` を使用してエラーに関する情報を取得することができます。イベントのためにアクションが定義されている場合、これらのアクションが実行されます。それ以外の場合はエラーがメッセージボックスに表示されます。
- **オーディオの録音の完了時**: オーディオの録音はファイルのサイズの最大値 (**ファイルサイズの最大値**)、または録音時間の最大値(**録音時間の最大値**)に到達すると終了します (*上のスクリーンショットを参照*)。

オーディオの録音に関連した MobileTogether XPath 拡張関数

次のオーディオの録音に関連したMobileTogether XPath 拡張関数 を使用することができます:

- **`mt-audio-is-recording()`** を`xs:boolean` として

XPath 式内のこの関数を使用することができます。例えば、アクションが現在録音中かどうかの条件に従い処理を指定することができます。

読み上げ

読み上げ機能は、テキスト文字列をオーディオに変換し、再生します。再生するテキスト文字列は、読み上げアクション の開始設定でXPath 式を使用して、直接指定することができます(*下のスクリーンショットを参照してください*)。デバイスの言語の設定をーオーバーライドするために使用することができます。詳細に関しては、次を参照してください読み上げアクション。

アクションの停止モードは実行されると、現在再生されている読み上げを停止します。

メモ 読み上げの再生は、モバイルデバイスのみで使用でき、MobileTogether Designer ではシミュレートすることはできません。

読み上げイベント

プロジェクトプロパティの読み上げアクションは次のイベント上で設定されるアクションを有効化します: 読み上げの開始時、読み上げのエラー発生時、読み上げの完了時。これらのイベントはイベントがトリガーされるとこれらの箇所で取られる更なるアクションを有効化します。

読み上げに関連するXPath

次の読み上げに関連したMobileTogether XPath 拡張関数 を使用することができます:

- **`mt-text-to-speech-is-language-available(`*`language`*`)`** を`xs:boolean` として
- **`mt-text-to-speech-is-speaking()`** を`xs:boolean` として

XPath 式内でこれらの関数を使用して、定義される条件が満たされているかをテストします。デザインにより取られるアクションは、ですから、これらの環境変数により条件付けられる必要があります。

ビデオの再生

MobileTogether のビデオの再生機能は、(i) リモートビデオファイルをクライアントデバイスにダウンロードし、(ii) 再生することを有効化します。ビデオの再生は以下の2つのステップで定義することができます:

1. ビデオコントロールはページ上の確認ウィンドウをセットするため、および、ダウンロードするビデオのURL を指定するために使用されます。詳細に関してはビデオコントロールのを参照してください。
2. ビデオアクションは以下の実行される再生アクションを指定します: *ビデオの開始*、*一時停止*、*再開*、*停止*、*シーク(ジャンプ)*。

ビデオウィンドウとビデオファイルの設定

ページに複数のビデオコントロールを挿入することができます。各 ビデオコントロールは、名前で識別され、URL を使用してビデオソースを割り当てられます。ビデオコントロールの名前は、ビデオアクションでビデオコントロールが実行されるアクション使用されます。

次のビデオコントロール プロパティはコントロールの主な属性を定義するために使用されます:

- **ロード時に再生**: ページがロードされるとすぐにビデオが再生されるかを指定します。再生が後に再生される場合、ビデオの開始アクション(例えば、ボタン)を使用します。
- **ビデオソース**: 再生するリモートまたはローカルのファイルを指定します。
- **キャッシュされたビデオソース**: キャッシュされたビデオファイルが保存されるクライアントデバイス上のURL です。この場所にキャッシュファイルが存在しない場合、ビデオソースファイルが再生のためにダウンロードされると作成されます。キャッシュファイルが既存の場合は、キャッシュファイルが再生されると、新規のダウンロードは行われません。
- **コントロールの表示**: コントロール内でビデオの再生ボタンが表示されるかを決定します。これはエンドユーザーが再生アクションを管理することを許可します。例えば、ビデオの開始、一時停止、再生、および停止など。このプロパティが `false` に設定されていると、再生アクションはビデオアクションを使用して提供されます。Windows Phone ではビデオコントロールボタンはサポートされていないことに注意してください。
- **初期の幅**: コントロールの初期の幅を設定します。ビデオが開始すると、コントロールはコントロールの **コントロールの幅** プロパティが `wrap_content` に設定されていると実際の幅を調節します。**コントロールの幅** プロパティが `fill_parent` に設定されていると、(親の) 幅が使用され高さのみが調節されます。
- **初期の高さ**: コントロールの初期の高さを設定します。ビデオが開始すると、コントロールは実際の高さに高さを調整します。

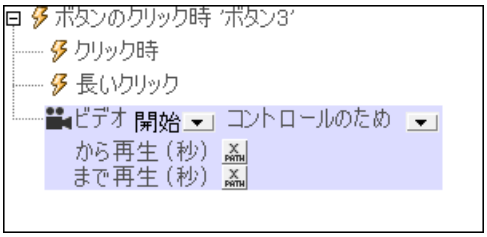
メモ Windows Phone では、マルチチャンネルのオーディオおよびビデオの再生はサポートされていません。1つのオーディオ**または**ビデオファイルのみ1度の再生することが可能です。これは最後に再生されたファイルを意味します。

メモ オーディオ/ビデオファイルは、MobileTogether Designer プロジェクトのサーバーメカニズムにデプロイを使用して MobileTogether Server にデプロイすることはで**きません**。しかしながら、URL を使用してストリームすることはできませんが、サーバーに手動でオーディオ/ビデオファイルをコピーすることができます。MobileTogether Server に存在する、オーディオ/ビデオファイルをストリームするには、以下を行います: (i) バイナリのロード アクションを使用して、バイナリのオーディオ/ビデオデータをデータソースノードにロードします。(ii) バイナリの保存 アクションを使用して、このノード内のデータをクライアント上のファイルに保存します。(iii) オーディオ/ビデオの再生アクションを使用して、クライアントデバイス上に保存されたファイルを再生します。または、オーディオ/ビデオファイルを MobileTogether Server に保存する代わりにWeb サーバーに保存し、URL を使用して、Web サーバーからオーディオ/ビデオファイルをストリームします。

ビデオの再生アクション

各 ビデオアクション(*下のスクリーンショット*): (i) ビデオコントロール名を使用して適用されるビデオコントロールを識別します、(ii) コントロールに関連したビデオファイルで実行されるアクションを指定します。これらは次のアクションです: *ビデオの開*

始、一時停止、再開、停止、シーク(ジャンプ)。ビデオアクションはファイル全体の代わりに再生される特定のファイルのセグメントを指定することを許可します。詳細に関しては、ビデオアクションの詳細を参照してください。



メモ: ソリューションが中断された場合 にビデオストリームが再生されている場合、再生は一時停止されます。ソリューションが再開すると再生が継続されます。

ビデオの再生イベント

ビデオの再生イベントは、各ビデオコントロールで定義され、ビデオコントロールに適用されます。これらのイベントは(右クリックして開くことのできる) コントロールのコンテキストメニュー、または、ビデオコントロールのコントロールアクションプロパティを使用してアクセスすることができます。各イベントのために、実行されるアクションを左側のアクションペインからイベントタブにドラッグアンドドロップすることにより定義することができます。

- **ビデオの開始時**: イベントが発生する前に(すなわち、ビデオの再生が開始する前に)、ビデオファイルの詳細を使用することはできない場合は、ビデオの高さ 幅、時間、および現在の位置を取得する関数を呼び出してはなりません(下を参照) 。この時点では、**`mt-video-is-playing`** 関数が有効な情報を返します。このイベントは、例えば、(ノードの更新アクション などを仕使用して)ビデオの再生の再生をXML ツリーノードにログするために使用できます。
- **ビデオのエラー時**: 可能性のあるエラー: ファイルが見つからないエラー、ファイルフォーマットのエラー、またはダウンロード/再生の中断エラーなど。MobileTogether XPath 実行関数 **`mt-external-error`** を使用してエラーに関する情報を取得することができます。アクションがイベントのために定義されている場合、これらのアクションは実行されます。それ以外の場合は、エラーはメッセージボックスに表示されます。
- **ビデオの完了時**: ビデオの再生は(停止アクションが実行されることなく)ファイルまたは指定されてセグメントの再生が最後に到達すると完了したものと考えられます。これらのアクションはビデオが中断(プロジェクトプロパティ*他のソリューションに切り替える*)または一時停止されると**実行されません**。

ビデオに関連した MobileTogether XPath 拡張関数

次のビデオに関連したMobileTogether XPath 拡張関数 を使用することができます:

- **`mt-video-get-current-position(`** ***`VideoControlName`*** を`xs:string` として) を `xs:decimal` として
- **`mt-video-get-duration(`** ***`VideoControlName`*** を`xs:string` として) を`xs:decimal` として
- **`mt-video-height(`** ***`VideoControlName`*** を`xs:string`として **)** を`xs:integer` として
- **`mt-video-width(`** ***`VideoControlName`*** を`xs:string` として) を`xs:integer` として
- **`mt-video-is-playing(`** ***`VideoControlName`*** を`xs:string` として) を`xs:boolean` として

XPath 式内のこれらの関数を使用することができます。例えば、ビデオの高さまたは幅の条件を処理する場合など。の再生が開始する前は、ビデオファイルの詳細を確認することができません。この結果、高さ 幅、長さ および現在の再生箇所などに関する情報を確認することができません。関連する機能は再生が開始してからのみ使用してください。

オーディオ /ビデオフォーマット

MobileTogether ソリューションを実行する多種のクライアントデバイスによりサポートされるオーディオとビデオのフォーマットについての情報を含んでいるインターネットページへのリンクです。有効と思われるWeb ページを選択しました。しかしながら、下にリストされるページに必要な情報が見つからない場合、代替のソースを検索してください。

メモ MIDI ファイルの再生はWeb ブラウザー以外のクライアントデバイスでサポートされています。iOS デバイスでの、MIDI ファイルの再生はサウンドバンクファイルが必要となります。

Android

Android 開発者 Web ページ: サポートされるメディアのフォーマット

オーディオ録音フォーマット
一般的に、オーディオの録音のためにサポートされるフォーマット /コーデックは、デバイスとOS のバージョンにより異なります。デバイスのOS の組み合わせの数を考慮すると、特定のフォーマットまたはエンコーダー指定しないことが奨励され、また、デフォルトのフォーマット /コーデックをそれぞれのデバイスに対して使用することが奨励されます。以下の制約に注意してください:

- `AMR-NB` と`AMR-WB` オーディオフォーマットの録音に関しては、AMR-NB とAMR-WB コードのみをそれぞれに対して使用してください。
- `AAC` オーディオフォーマットはAAC-Low Complexity、High Efficiency-AAC、およびEnhanced Low Delay-AAC codecs を使用して録音することができますが、与えられたデバイスでの作業にすべてのコーデックを使用できない場合があります。

---

iOS

iOS は以下を含む多種の産業標準のビデオフォーマットと圧縮基準をサポートしています:

- H.264 ビデオ、1.5 Mbps まで、640 x 480 ピクセル、1秒当たり30 フレーム、低い複雑度バージョンを持ち、160 キロビット毎秒までのAAC-LC オーディオを持つ、48 kHz までのH.264 ベースラインプロファイル、m4v、.mp4、および.mov ファイルフォーマットのステレオオーディオ
- H.264 ビデオ、上限 768 キロビット毎秒 まで、320 x 240 ピクセル、1秒当たり30 フレーム、160 キロビット毎秒までのAAC-LC を持つレベル1.3 までのベースラインプロファイル、 48 kHz、 m4v、.mp4、および.mov ファイルフォーマットのステレオオーディオ
- MPEG-4 ビデオ、上限 2.5 Mbps まで、640 x 480 ピクセル、一秒当たり30 フレーム、160 キロビット毎秒までのAAC-LC オーディオを持つ単純型プロファイル、48 kHz、m4v、.mp4、および.mov ファイルフォーマットのステレオオーディオ
- オーディオ技術 にリストされるフォーマットを含む多種のオーディオフォーマット。

Apple 開発者 Web ページ: オーディオ情報
Apple 開発者 Web ページ: ビデオ情報

Windows

Windows のためのオーディオ/ビデオ コーデックとフォーマット

オーディオの録音のフォーマット
WMA とMP3 ファイルはWindows Phone での録音フォーマットとしてはサポートされていません。

Web ブラウザー
HTML で再生するオーディオの基本的概要
HTML で再生するビデオの基本的概要
異なるブラウザーでサポートされる オーディオ/ビデオフォーマット

現在サポートされる最も高い数値は以下の通りです:

- *オーディオ*: **mp3**, **aac**
- *ビデオ*: **H.264 (mp4)**

11.4 NFC

近距離通信 (NFC) は短距離でNFC が有効化されている2台のデバイス間でデータを送信することのできるワイアレスの技術です (通常 4cm (1.5 インチ) または以下) 。この技術はNFC が有効化されている クレジット/デビット カードとNFC が有効化されている支払いターミナル間で支払いを行う際に最も頻繁に使用されています。NFC により、通常はテキストまたは数字である、少量のデータのペイロードを送信することができます。しかしながら、NFC が有効化されているデバイス間で(イメージ とファイルなど)のデータを送信する際に使用することもできます。

NFC に関する詳細は、nearfieldcommunication.org、Wikipedia、とNFC フォーラムを参照してください。

NFC が有効化されているデバイス

NFC が有効化されているデバイスは、アクティブ またはパッシブであることができます。例えばNFC タグなどのパッシブ デバイスは、スマートフォンなどの他の(アクティブな) NFC デバイス、が読み取るデータを含んでいます。アクティブなデバイスは、情報を読み取り/送信することができます。スマートフォンはアクティブな NFC デバイスです。アクティブな NFC デバイスは、パッシブな NFC デバイスから情報を読み取り、他のNFC が有効化されているデバイスと情報を交換することができます。

安全な通信が重要な場合、NFC は、安全な接続を構築し、暗号化された情報を使用することができます。

Android 上での Android ビーム? とNFC 使用

Android ビーム? (今後 Android ビーム) はAndroid 4.0. からAndroid デバイス上で使用することのできるアプリです。2台のビームが有効化されているデバイス間でデータを共有するために使用することができます。通常、デバイス上で現在開かれている(Web ページ、写真、連絡先情報などの) コンテンツをデバイスをタッチすることにより他のデバイスにビームすることができます。

メモ Android ビームをOS で使用できる場合でも、デバイスがNFC をサポートしない場合はこの機能を使用することはできません(例えば、デバイスがNFC チップを含んでいない場合など)。

デバイス上でAndroid ビームを使用するには、(i) デバイスがNFC をサポートしている(ii) NFC とAndroid ビームがデバイス上で有効化されていることを確認してください。NFC とAndroid ビームが使用可能かを確認するには、Android デバイスの **「設定 | ネットワーク」** タブへ移動してください。*NFC* と*Android ビーム* オプションを使用することができる場合、有効化してください。OS がAndroid 4.0 以降の場合は、両方のオプションを使用することができなく、NFC をデバイス上で使用することはできません。

メモ Android デバイスとWindows デバイスでNFC が有効化されている場合、NFC を使用して通信することができます。

Windows 上での NFC の使用とiOS

- *Windows*: Windows デバイス上でNFC が使用可能かをチェックするには、**「設定 | タップして送信」**に移動します。
- *iOS*: Apple Pay でNFC は使用されています。他の種類の情報を共有する場合にこの機能を使用することはできません。

メモ Android デバイスとWindows デバイスでNFC が有効化されている場合、NFC を使用して通信することができます。

NDEF 技術

NFC データ交換 フォーマット(NDEF) メッセージは、NFC データ書式内で送信され受信されます。NDEF フォーマットでは、各通信は**NFC タグ**です。各 NFC タグには、**NDEF メッセージ**が含まれており、各 NDEF メッセージには1つ以上の**NDEF レコード**が含まれています。アクティブな NFC デバイスのロックが解除されると、近辺で NFC タグを自動的に検索します。NFC タグの用途により、デバイスは NFC タグの最良の処理方法を決定します。デバイスがユーザーに実行するアクションを**問わない**ことは重要です。これはユーザー入力によりデバイスが NFC タグから離され、接続を切断するからです。詳細に関しては、次を参照してください：Android 開発者のためのガイド.

MobileTogether 内で NFC をデザインする際に重要な点は、NFC メッセージの**ペイロード**(すなわち、メッセージのコンテンツ) が hexBinary フォーマットで保管され、転送されるということです。hexBinary フォーマットの構文スペースは 16 進数値としての(データポイントの) シンプルなコードです。例: 文字列 `hi` が hexBinary フォーマットに変換されると`6869` になります(`h` の16 進表現は、`68` で、`i` の16 進値は`69` だからです)。

NFC タグ

用語 NFC タグは2つの概念を参照するために使用されています:

- NDEF 技術を使用して転送される**データ**(次を参照: 上記の*NDEF 技術*)
- NFC データを含むパッシブ NFC **デバイス**

上記の後者の種類の NFC タグは、マイクロチップを含むハードウェアオブジェクトです。最も簡単なフォームでは、この種類の NFC タグには以下の特徴があります：(i) 読み取ることのできるデータを含んでおり、(ii) NFC タグがロックされるまで、含まれるデータを複数回上書きすることができ、(iii) ロックされると NFC タグを上書きすることができません。

詳細に関しては、次を参照してください：NFC タグの説明。

MobileTogether と NFC

MobileTogether ソリューションは NFC を次の方法でサポートします:

- (Android と Windows デバイス上で) NFC タグを読み取ることができ、読み取られたデータは更に処理することができます
- (Android と Windows デバイスから) メッセージを一台のデバイスから他のデバイスにプッシュすることができます
- Android ビームが有効化されている Android デバイスから他の Android ビームが有効化されている Android デバイスにファイルをビームすることができます。

メモ iOS デバイス上では NFC はサポートされていません。

このセクション

- NFC タグの検出と読み取り
- 他のデバイスにデータをプッシュする
- NFC に関連したイベント
- NFC のためのデザインコンポーネント

NFC タグの検出と読み取り

NFC 開始アクションを使用して、NFC が開始されると、NFC タグの検出は自動的に開始されます。NFC タグが検出されると、タグ内のNFC メッセージが自動的に受信され、メッセージ内の情報が**$MT_NFC** ツリー内に保管されます。NFC アクションの開始/停止 がデザイン内に追加されると、このツリーがデザイン内に作成されます。ツリーの構造全体が下に表示されています。NDEF メッセージは、複数のNDEF レコードを含むことができ、NDEF レコードは再帰的であることができます。受信されたNFC タグ情報が**NDEFMessage** または**NDEFRecord** 要素の属性の情報を含まない場合、**$MT_NFC** ツリー内でこれらの属性は作成されません。

$MT_NFC ツリーの全体の構造

```
<Root>
   <Tag Id=""/>
   <NdefMessage
      CanMakeReadOnly=""
      IsWriteable=""
      MaxSize=""
      Type="">
      <NdefRecord
         Id=""
         TypeNameField=""
         RecordTypeDefinition=""
         Type=""
         Text=""
         Language=""
         URI=""
         Payload=""
         MimeType=""
         ExternalDomain=""
         ExternalPackageName="">
         <NdefRecord />
      </NdefRecord>
      <NdefRecord />
          ...
      <NdefRecord />
   </NdefMessage>
</Root>
```

$MT_NFC ツリー内の情報を更に処理し、他のページソースデータと同じ方法で表示することができます。例えば、**$MT_NFC/Root/Tag/NdefMessage/NdefRecord/@Text** ノードはラベル内のメッセージテキストを表示するためにラベルにリンクすることができます。

メモ NFC タグが検出されると、実行される追加アクションを**NFC タグの検出時** イベントのタブを使用して指定することができます。

メモ NFC からの情報は、**$MT_NFC** ツリー内の既存の情報を上書きします。タグが検出されると、**$MT_NFC** ツリー内の前のタグの情報が置き換えられます。

他のデバイスにデータをプッシュする

MobileTogether ソリューションは、ソリューションを実行しているデバイスからデータをNFC が有効化されているデバイスに送信するために使用することができます。エンドユーザーの取るステップは以下のとおりです:

1. 送信するデバイスで、ソリューションを開始します。
2. (例えば、ボタンをタップするなどして) NFC の開始 アクションをトリガーします。
3. データの送信元のデバイスをNFC が有効化されているデータを受信するデバイスの裏に設置します。
4. NFC プッシュ アクションをトリガーして送信を開始します (例えば、ボタンをタップするなど)。NFC プッシュ アクションが開始されると、NFC メッセージまたはファイルを(複数のデバイスに)複数回送信することができます。エンドユーザーは送信デバイスのNFC 範囲内に受信するデバイスを設定するのみです。NFC が停止されると、または、キャンセルオプションを選択して、新規 NFC プッシュ アクションを追加することにより、NFC プッシュがキャンセルされると継続的な送信は停止されます (*下のスクリーンショットを参照*)。

しくみ

NFC データ送信はNFC プッシュ アクションで次のように定義することができます (*スクリーンショット参照*)。また、以下の方法で定義することができます。

- *NFC プッシュ オプション*を使用して特別に定義された型のメッセージを送信することができます。Android と Windows デバイスで使用することができます。メッセージに関する詳細は、NFC プッシュ アクション を参照してください。
- Android ビームが有効化されているデバイスからファイルを他のAndroid ビームが有効化されているデバイスにビームすることができます。この機能はAndroid デバイスでのみ使用することができます。

アクションの設定を定義する方法に関する詳細はNFC プッシュ アクションを参照してください。

メモ NFC データ送信はiOS デバイス上ではサポートされていません。

NFC に関連したイベント

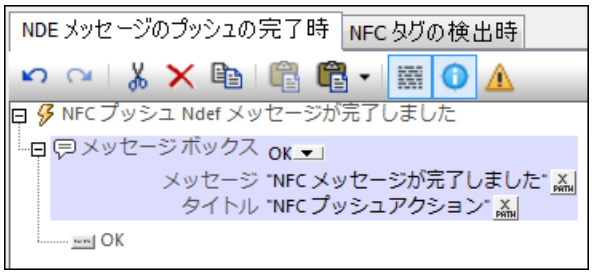
NFC に関連したイベントには以下の重要な機能が搭載されています (*下のスクリーンショット*):

- **NDE メッセージのプッシュの完了時** は、データの送信を行う時に実行されるアクションを(NFC プッシュを使用して)指定します。
- **NFC タグの検出時** は、NFC タグが検出されると実行される追加アクションを指定します。

これらのアクションにアクセスするには、 スタイル & プロパティ ペイン | プロジェクトプロパティ | NFC アクションプロパティに移動し、プロパティの **追加ダイアログ** ボタンをクリックします。

NDE メッセージのプッシュの完了時

メッセージまたはファイルの送信に成功するとイベントがトリガーされます。以下の状況で使用することができます (i) 次のアクションを定義する場合 (ii) 送信の完了をユーザーに通知する場合。下のスクリーンショットは、イベントの使用方法を表示しています。

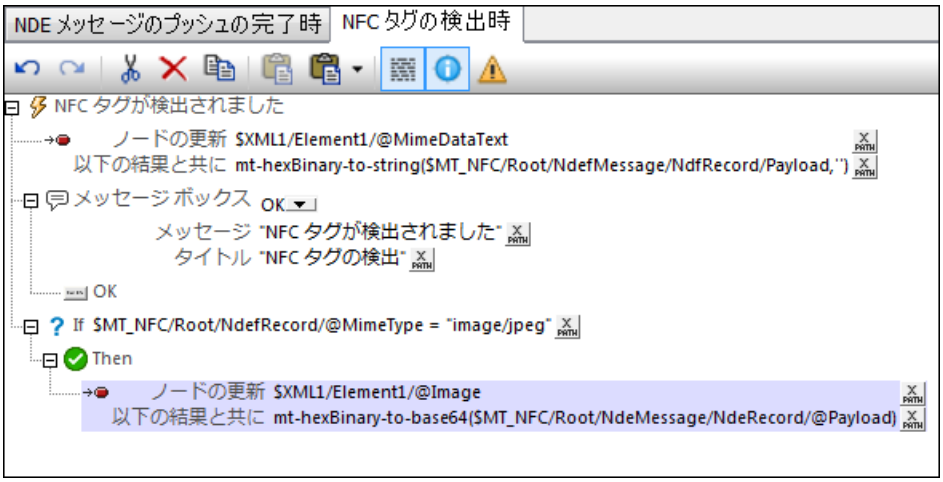


次も参照してください：他のデバイスにデータをプッシュする

NFC タグの検出時

NFC タグが検出されるとイベントはトリガーされます。タグが検出されると、情報は自動的に読み取られ `$MT_NFC` ツリー内に保管されます。 イベントにより実行される追加のアクションを指定することができます。例えば、下のスクリーンショット内で示されるように、`$MT_NFC` ツリーの `NdefRecord` 要素の `Payload` 属性からのデータを使用してソースツリー ノードを更新することができます。

`Payload` 属性は、hexBinary フォーマット内にコンテンツが保存されます。ペーロードにテキスト文字列が存在する場合、を使用してソースツリーノード内に文字列を配置する前にテキスト文字列を取得するために拡張関数 `mt-hexBinary-to-string` を使用することができます (*下のスクリーンショット内の最初の ノードの更新 アクションを参照してください*)。同様に、ペイロードにイメージが存在する場合、ペイロードのコンテンツを `mt-hexBinary-to-base64` 拡張関数を使用してイメージの Base64 エンコードに変換することができます。



次も参照してください：NFC タグの検出と読み取り

NFC のためのデザインコンポーネント

NFC 固有のデザインコンポーネントを使用してNFC 機能は実装されます:

NFC の開始 /停止 アクション

このアクションはメッセージのプッシュおよび/または受信の開始または停止のために使用されます。

NFC の開始 アクションがトリガーされるとプッシュおよび/または受信が開始されます。アクションを実行するためのステップのシーケンスは、以下のとおりです:

1. NFC がデバイス上で有効化されている必要があります。NFC が有効化されていない場合、開始アクションをトリガーすると、ユーザーにNFC の有効化を問うプロンプトが表示されます。
2. NFC が有効化された後、MobileTogether Client アプリがNFC のために登録されます。
3. この後、NFC タグの検索が自動的に開始され、NFC タグ内のNFC メッセージは自動的に受信されます。NFC プッシュアクションを使用して、プッシュを開始することができます。プッシュは自動的には開始されません。

NFC の停止 アクション全てのメッセージのプッシュと受信を停止します。メッセージのプッシュと受信を再開するには、開始アクションを再度トリガーしてください。

次も参照してください: NFC タグの検出と読み取り。

NFC プッシュアクション

NFC プッシュアクションソリューションを実行しているデバイスから データを送信し、他のNFC が有効化されているデバイスへの送信を有効化します。更に、Android ビームが2台のAndroid デバイスで有効化されている場合、ファイルは一台のデバイスからもう一台のデバイスにビームされます。 NFC プッシュアクションは、プッシュするメッセージ、または、ビームするファイルを定義します。アクションがトリガーされると、指定されたメッセージまたはファイルがNFC を介して送信されます。

メモ NFC データ送信はiOS デバイス上ではサポートされていません。

詳細に関しては、次を参照してください: 他のデバイスにデータをプッシュするとNFC プッシュアクション。

$MT_NFC ページソース ツリー

NFC の開始/停止 アクションが定義される際にデザイン内のページソースとして `$MT_NFC` ツリーが作成されます。NFC タグが検出されるとツリーは自動的に表示されます。NFC タグからのデータは、`$MT_NFC` ツリーのノード内に保管されています。シミュレーションに関しては、NFC サンプルファイルを使用して、NFC タグからのデータが `$MT_NFC` ツリー内に保管されているかを確認することができます(NFC シミュレーションに関する詳細は、NFC サンプルファイルを参照してください)。

NFC に関連したイベント

2つのNFC に関連したイベントは以下の重要な機能を搭載しています:

- **`NDE メッセージのプッシュの完了時`** は、データの送信を行う時に実行されるアクションを(NFC プッシュを使用

して)指定します。

- **NFC タグの検出時** は、NFC タグが検出されると実行される追加アクションを指定します。例えば、イベントがトリガーされるとノードの更新 アクションを使用して、検出されたNFC タグからのデータをデータソースツリーを更新するために使用することができます。

NFC に関連した XPath 拡張関数

NFC に関連した次の関数を使用することができます:

- **mt-nfc-started**: ブール値は、ソリューションに開始済みのNFC が存在するかをチェックします。
- **mt-hexBinary-to-string**: は hexBinary をテキスト文字列に変換します。
- **mt-hexBinary-to-base64**: は hexBinary をBase64 エンコードイメージに変換します。
- **mt-string-to-hexBinary**: は、テキスト文字列をhexBinary 文字列に変換します。
- **mt-base64-to-hexBinary**: は、Base64 エンコードイメージをhexBinary 文字列に変換します。

メッセージのペイロードは、hexBinary フォーマットにより送信されるため、変換関数はデータを送信するために(hexBinary に)準備し、(テキストとイメージなどの)使用できるフォーマットに変換します。これら関数の詳細に関しては、次を参照してください: MobileTogether 拡張関数。

シミュレーションのための NFC サンプルファイル

シミュレーションのために、you can create an NFC サンプルファイルを作成し、このファイルをNFC タグからデータが **$MT_NFC** ツリーに正確にインポートされているかをテストするために使用することができます。 NFC シミュレーションに関する詳細は、NFC サンプルファイルを参照してください。

11.5 プッシュ通知

プッシュ通知 (PN) はデバイス上にインストールされているアプリに関連しており、モバイルデバイス上に表示されるテキストメッセージです。デバイスでプッシュ通知が受信されると、ユーザーはアプリ、または、デバイスを使用する必要がありません。デバイスがオンに切り替えられていることを確認してください。プッシュ通知はアプリの開始を待たず、また、受信されるメッセージを待つことなく、アプリの発行元にユーザーと直接通信する方法を提供します。プッシュ通知は通常アプリに関連する更新ニュースなどの情報を提供するだけではなく、招待を受ける、サイトにリンクする、データベースを変更するなどのアクションを実行することもできます。

MobileTogether 内では、MobileTogether ソリューションからプッシュ通知は送信され、同じソリューション (または他のソリューション) により受け取られます。このため、通知は複数のデバイス上の1つのソリューションから「プッシュ」されます。

MobileTogether AppStore App アプリストアから他のにダウンロードすることのできるアプリとしてコンパイルされた MobileTogether ソリューションである MobileTogether AppStore App— によりプッシュ通知を送信することもできます。

MobileTogether プッシュ通知の作動のしくみ

MobileTogether プッシュ通知は、短いメッセージ、長いメッセージ、キー値ペアとして構成されているデータにより構成されています。 通常、プッシュ通知が受信されるとデバイスに短いメッセージが表示されます。プッシュ通知をタップすると長いメッセージが表示されます。ユーザーがメッセージが受信されると取られるアクションを決定することが出来るボタンをメッセージは持つことが出来ます。プッシュ通知のペイロードは、デバイス上のデータツリーに転送され、他のアクションで使用することができます。これにより、プッシュ通知イベントに関連するデータである新しいデータが自由に MobileTogether アクションの全ての範囲によりにより処理されることが出来、MobileTogether コントロール と使用できるようになります。

プッシュ通知のしくみは以下のとおりです:

- ソリューションの送信の プッシュ通知の送信 アクションは、プッシュ通知の主要のパラメーター (メッセージの詳細と受け取り左記のリスト) を定義します。メカニズムのこの部分は、このセクションの ソリューションの送信 サブセクション内で大まかに説明されています。
- ログイン資格情報、または、登録情報をベースに受け取り先デバイスは識別されます。**特定の名前**を用いて特定の MobileTogether Server にデバイスはログインされます。更に、特定の**プッシュ通知トピック**を受け取るために、デバイスを**外部プッシュ通知キ**を使用して登録することができます。プッシュ通知の送信 アクション内では、ユーザー名、外部プッシュ通知キー、またはプッシュ通知のトピックをベースにして受け取り先デバイスを定義することができます。このため、プッシュ通知は柔軟的に構成することのできる受け取り先のリストに送信することができます。
- デバイスでプッシュ通知が受信されると、短い/長いメッセージが表示され、ペイロードは自動的にソリューションの受信先の `$MT_PUSHNOTIFICATION` ページデータソースに転送されます。ソリューションの受信先はプッシュ通知が受信、または、プッシュ通知のボタンがユーザーによりタップされると取られる追加アクションを定義します。 サブセクション ソリューションの受信 は、プッシュ通知メカニズムのこの部分の概要について説明しています。

このセクション

ソリューションの送信 と ソリューションの受信 はサブセクションでプッシュ通知の主要なメカニズムについて説明していますが、その後の2つのサブセクションは、追加機能について説明されています。AppStore App 内のプッシュ通知の実装に成功するためには (MobileTogether ソリューションからコンパイルされるアプリ)、追加ステップが必要になります。AppStore App 内のプッシュ通知 サブセクション内でこれらのステップについて説明されています。プッシュ通知のシミュレーション は、ソリューションの送信と受信が同じソリューションからではない場合のシナリオについて説明しています。

ソリューションの送信

ソリューションの送信内では、プッシュ通知 (PN) の送信をトリガーするイベントを選択し、このイベントのためにプッシュ通知の送信アクションを定義します (*下のスクリーンショット*)。プッシュ通知の多種のパラメーターはプッシュ通知の送信アクション内で定義されています。主要のパラメーターは下で大まかに説明されています。

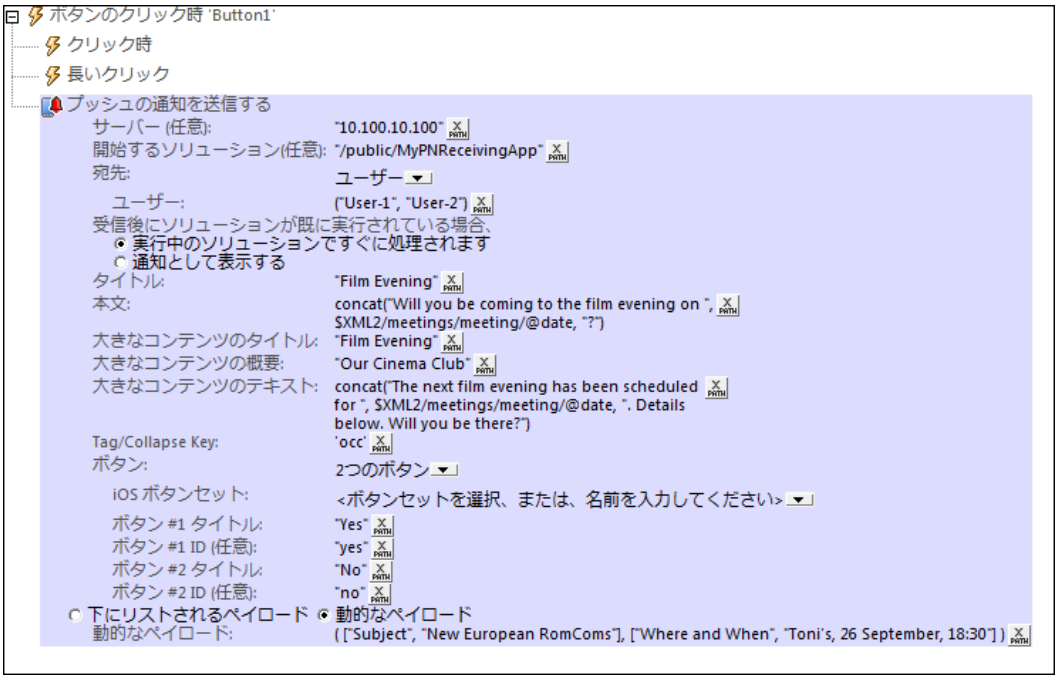
プッシュ通知の送信アクションのパラメーターの定義に加えプッシュ通知内で送信されるデータに関連する、ソリューションの送信内で必要とされる他の準備項目です。ペイロードおよび/または短いおよび長いメッセージのためのペイロードのためのデータが、ソリューションの送信のページデータソースから動的に追加される場合、これらのページソースが作成され、プッシュ通知のペイロードのために正確なデータが使用できる必要があります。

下にリストされるメカニズムは、ソリューションの送信の一部ですが、ソリューションの送信はソリューションの受信と同じソリューションであることが出来ます。このシナリオの場合、ソリューションの送信と受信のソリューションのメカニズムは1つのソリューションにまとめることが出来ます。

ソリューションの送信の準備: プッシュ通知の送信アクション

ソリューションの送信に関する準備はプッシュ通知の送信アクション内のプッシュ通知のパラメーターの受信に関連しています。主要のパラメーターは以下のとおりです:

- 短いメッセージ、長いメッセージ、ペイロードが定義されます。これらの項目の定義は複雑ではありません。
- ユーザー名 (またはユーザーロール)、外部プッシュ通知キー、プッシュ通知トピックをベースにして受け取り先のリストはここで定義されます。特定のユーザー名を使用して、デバイスは特定の MobileTogether Server にログインします。 ユーザー名/ロールは MobileTogether Server 上で構成されたユーザーを参照します。更に、デバイスをお以下の項目に登録することができます (i) 外部プッシュ通知キー、および/または (ii) 特定のプッシュ通知トピックを受信 (*これらの登録の詳細に関しては、ソリューションの受信を参照してください*)。このため、これらの選択条件を使用して受信者のリストを柔軟的に構成することができます。



- プッシュ通知はプッシュ通知ボタンを含むことが出来ます。最大3つまでのプッシュ通知ボタンのテキストと ID を定義することができます。ユーザーがボタンをタップすると、ボタンの ID がソリューションのページソースノードにパスされます。これにより、ユーザーがプッシュ通知に対応するかを決定することができ (**プッシュ通知の受信時** イベントのタブ内で)相対するアクションを決定することができます。iOS デバイス上で表示されるアクションは、AppStore App のた

めに有効化されており、標準の MobileTogether ソリューションのためには有効化されていないことに注意してください。

AppStore App としてソリューションを送信する

AppStore App としてソリューションの送信が作成されている場合、追加ステップに関しては、AppStore App 内のプッシュ通知 を参照してください。

ソリューションの受信

このセクションはをソリューションの受信内に存在するプッシュ通知 (PN) メカニズムの異なる部分により構成されています。下にリストされるメカニズムはソリューションの受信の一部です。ソリューションの受信はソリューションの送信と同じソリューションであることが出来ます。この場合、ソリューションの送信と受信のメカニズムは1つのソリューションにまとめることが出来ます。

メモ 標準のMobileTogether ソリューションの大きなコンテンツはAndroid とWindows デバイス上でのみ表示されます。iOS デバイス上で大きなコンテンツを表示する場合、AppStore App を受け取る側のソリューションでコンパイルする必要があります。

プッシュの通知の受信時 イベント

- デザイン時では、**プッシュの通知の受信時** イベントはソリューションの受信のプロジェクトプロパティによりアクセスすることができます。イベントのタブ内で、プッシュ通知が受信されると取られるアクションを定義します。イベント デザインに追加されると、`$MT_PUSHNOTIFICATION` ページソースが自動的にされます。
- ランタイムでは、**プッシュの通知の受信時** イベントは、ユーザータップ、または、プッシュ通知内のボタンによりトリガーされます。イベントがトリガーされると、以下が発生します: (i) ソリューションの受信が既に作動している場合ソリューションの受信が開始されます。(ii) プッシュ通知のペイロードは`$MT_PUSHNOTIFICATION` ページソースに自動的に参照されます。プッシュ通知ボタンがパスされると、ボタンのID (文字列) がページソースにパスされます。(iii) イベントの アクションが実行されます。If-Then、または、If-Then-Else アクションを使用して、押されるプッシュ通知ボタンにより取られるアクションを作成することができます (*次のセクションを参照*)。

$MT_PUSHNOTIFICATION ページソース

`$MT_PUSHNOTIFICATION` ページソースには次の固定された構造が存在します:

```
$MT_PUSHNOTIFICATION
Root
|   @button
|
|-- Entry
|      @key
|      @value
```

ランタイム:

- プッシュ通知がタップされると、`$MT_PUSHNOTIFICATION/Root/@button` ノードにボタンのID がパスされます。`@button` 属性の値はIf-Then、または、If-Then-Else アクションを使用して条件付きの処理のために使用されます。例えば、`@button` ノードが *受け取り*ボタンのID を含む場合、受け取りSMS 自動的に送信され、または、データベースが変更されます。他のボタンID のために代替アクションを定義することもできます。
- `Entry` 要素の数値がランタイムで決定され、プッシュ通知のペイロード内で含まれるキー値ペアの数値と同じになります。各キー値ペアのデータは対応する`Entry` 要素にパスされます。`$MT_PUSHNOTIFICATION` ページソース内のデータはデザイン内でデータを表示するなと希望する方法で処理されます。

外部プッシュ通知キー

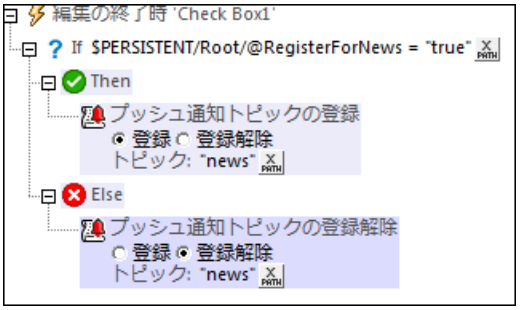
プッシュ通知を受信するモバイルデバイスを外部プッシュ通知キーに登録することができます。このキーは外部プッシュ通知キーの登録 アクションにより生成されたテキスト文字列です (*下のスクリーンショットを参照*)。モバイルデバイス上の同じソリューションにより同じキーを生成することができるため、外部プッシュ通知キーは、モバイルデバイスの特定のセットを識別することができます。

外部プッシュ通知キーに送信されると、そのキーに登録されている全てのデバイスはプッシュ通知を受け取ります。

メモ AppStore Apps 内でも外部プッシュ通知キーを使用することができます。

プッシュ通知 トピック

プッシュ通知 を受信するモバイル デバイスは1つまたは複数の特定のトピックに関するプッシュ通知を受信するために登録することができます。プッシュ通知 トピックの登録 アクションを使用して1つまたは複数のトピックにデバイスを登録することができます (*下のスクリーンショットを参照*)。ランタイムでは、プッシュ通知 トピックにプッシュ通知が送信されると、そのトピックに登録されている全てのデバイスがプッシュ通知を受け取り、ターゲット トピックのどれかに登録されているデバイスはプッシュ通知を受け取ります。



理論的には、特定のデバイス上のソリューションはトピックにこのデバイスを登録するために使用することができます。作業上では、プッシュ通知を受け取るソリューションから登録をおこなうことが奨励されています。

---

iOS プッシュ通知ボタンセットの定義

プッシュ通知ボタンを含むプッシュ通知がデバイスに届くと、ボタンがタップされ、ソリューションの受信が開始され、プッシュ通知ボタンの ID がソリューションの `$MT_PUSHNOTIFICATION` ページソースの `$MT_PUSHNOTIFICATION/Root/@button` ノードにパスされます。

ソリューションの送信のプッシュ通知の送信 アクション内で、非-iOS デバイスのためにプッシュ通知ボタンは定義されますが、iOS プッシュ通知ボタンセットは、**「プロジェクト | iOS プッシュ通知ボタンセット」** コマンドを使用して、ソリューションの受信内で定義されます。

AppStore App としてのソリューションの受信

AppStore App としてソリューションの受信が作成される場合、追加ステップに関しては、AppStore App 内でのプッシュ通知 を参照してください。

AppStore App 内のプッシュ通知

AppStore Apps はエンドユーザーがダウンロードすることのできる異なるオペレーションベースのアプリにコンパイルされる MobileTogether アプリです。(詳細に関しては次を参照してください: AppStore Apps)。異なるオペレーティングシステムのためにプログラムコードをコンパイルするには、それぞれのオペレーティングシステムのためにプログラムコードを生成する必要があります。プログラムコードの生成ウィザードを使用しておこなうことができます。**受信するソリューション**の MobileTogether アプリがプッシュ通知 (PN) を受信するように設定されている場合、ウィザードの スクリーン6 内の関連する設定に情報を入力する必要があります(ソリューションの **プッシュ通知の受信時** イベントにアクションが定義されている場合のみスクリーンが表示されます)。

AppStore Apps 内でのプッシュ通知の使用は、下記の追加ステップを必要とします。プログラムコードを生成する前にこれらのステップを実行する必要があります。これは、プログラムコードの生成ウィザードのスクリーン6 の設定を完了するために登録情報が必要になるからです。

メモ プッシュ通知を受信するためのアプリを準備するステップは **ソリューション受信のアプリ**にのみ適用されます。ソリューションの送信と受信が異なるアプリの場合、送信するソリューションには適用されません。

Android とiOS

Android とiOS アプリがプッシュ通信を受け取るには、Firebase アカウントを所有し、Firebase Server キーを作成する必要があります。

プログラムコードの生成ウィザードのスクリーン6 で、Firebase Server キーを使用することに注意してください。

Android

Firebase Console 内でAndroid アプリを作成します。アプリページに移動し`google-services.json` ファイル。をダウンロードします。

`google-services.json` ファイルの場所を記録してください。ファイルの名前を変更しないでください。プログラムコードの生成ウィザードのスクリーン6 内のこのファイルをポイントする必要があります。

---

iOS

以下が必要になります: (i) *Apple プッシュ通知 サービス(APN)* キーを生成し(ii) Firebase 内で構成を行い、`GoogleService-Info.plist` ファイルをダウンロードします。これを行うには下のステップに従ってください。

APN キーを生成します *(Apple デベロッパーアカウント1 つに対して1 つに必要です。アカウントは多種のプロジェクトに使用することができます)*:

1. `developer.apple.com` でAPN キーを作成します。これを以下のように行います: *アカウント> 証明書、ID & プロファイル> キー、* **追加** ボタン(`+`) をクリックし、キーを作成します。
2. 1度のみ生成される`.p8` 拡張子を持つAPN キーファイルを確認しダウンロードします。キーファイルを安全な場所に保管します。10 桁の認証キーIDも保管してください。キーファイルと認証 ID はApple デベロッパーアカウントをセットアップするために1度必要になります。

Firebase を構成し、`GoogleService-Info.plist` ファイルをダウンロードして生成します:

1. Firebase コンソール内でiOS アプリを作成します。
2. 生成された`.p8` ファイルをアップロードします。

3. バンドルID プレフィックスではなくアプリID プレフィックスと(APN キー生成中に取得された10桁の認証キーID と共に指定します。
4. **`GoogleService-Info.plist`** ファイルをダウンロードします。

ダウンロードされたファイルの場所に注意してください。ファイルの名前を変更しないでください。プログラムコードの生成ウィザードのスクリーン6 内のこのファイルをポイントする必要があります。

トラブルシューティング

Q: *プッシュ通知が受信されない場合の発生する問題は何ですか?*
A: *以下に挙げられる問題が原因の可能性があります:*

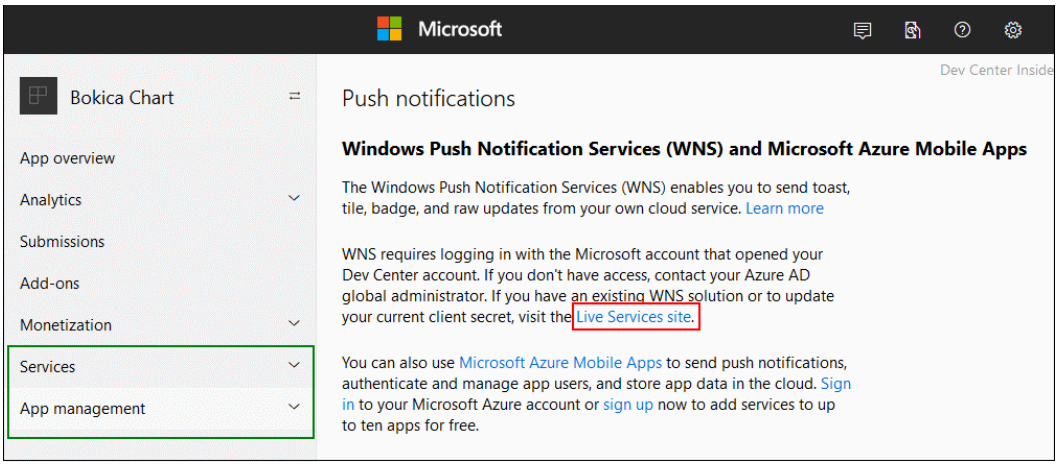
- AppStore App がインストールされると、アプリを1度開始する必要があります。プッシュ通知の受信の許可を問われると、**「はい」**をクリックします。アプリが開始し、サーバーに接続されると、クライアントのプッシュ通知アドレスがサーバーに送信されます。
- コンパイルされたアプリのためではなく、特定のトピックのためにメッセージを受け取る場合、プッシュ通知が正確に送信されたかを確認するために、ソリューション名を確認します。
- **`p8`** ファイルと2つのキー(APN と認証キー) がアップロードされていることをチェックします。
- Firebase 設定のバンドルID をチェックし、and in 生成されたAppStore App 内のID が同一かをチェックします。
- Apple デベロッパーコンソール内のアプリID のためのプッシュ通知をチェックします。デベロッパーコンソールが有効化されます。
- 上記が適用されない場合、**`xcode/your project/capabilities`** に移動し、プッシュ通知スイッチをオフに切り替え、オンに切り替えます。

Q: *iOS 9 では「通知として表示」が作動しません。*
A: エラーではありません。サポートはiOS 10、または以降のみに与えられています。

Windows

プッシュ通知 受け取りソリューションのWindows アプリのためにプログラムコードを生成する場合、プログラムコードの生成ウィザードのスクリーン6 のために2 つの情報が必要になります: (i) パッケージSID、と(ii) アプリケーションの秘密。取得方法は、以下のとおりです:

1. https://developer.microsoft.com/en-us/dashboard にユーザー名とパスワードを使用してログインします。
2. アプリがストア内に存在しない場合、アプリのための名前の受信の後のステップに従います。
3. アプリのページに移動し、**サービス** > **プッシュ通知** > **`WNS/MPNS`** を選択します。一部の場合、 **`WNS/MPNS`** アイテムは**`App management`** の下にリストされています (*下のスクリーンショットを参照してください*)。



4. 表示されるページ内で、(*上のスクリーンショット内で赤枠で囲まれている*) ライブサービスサイトをクリックします。
5. アプリの登録ページが開かれます。プログラムコードの生成ウィザードを使用してプログラムコードを生成するために必要なこのページの次の詳細を記録してください。プログラムコード: (i) パッケージSID、(ii) アプリケーションの秘密。 **「保存」**を押して、手続きを完了します。
6. プログラムコードの生成ウィザードを開始してプログラムコードを作成します。

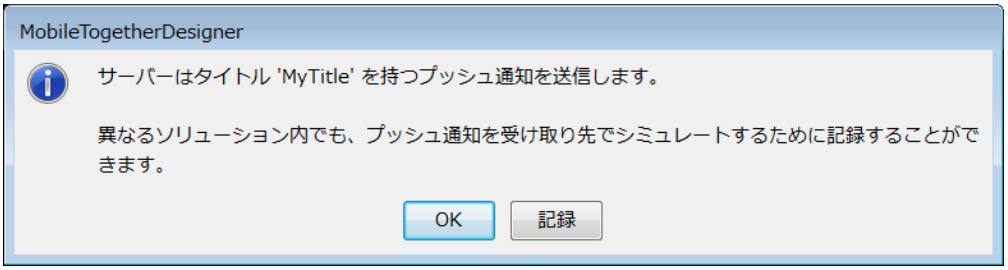
プッシュ通知のシミュレーション

プッシュ通知 (PN) には以下に関連下データが含まれています: (i) プッシュ通知の短いメッセージ。(ii) プッシュ通知の長いメッセージ。 (iii) プッシュ通知のペイロード。ソリューションの送信と受信は1つで同じソリューションです。シミュレーション内での送信と受信間のデータの転送は1つのシミュレーション内で行われます。この場合のシミュレーションはわかりやすいです。

しかしながら、ソリューションの送信と受信が異なる場合、シミュレーションのメカニズムは以下のとおりです: ソリューションの送信のシミュレーション中に送信されるプッシュ通知データはMT プッシュ通知シミュレーションファイル (`.mtpnsim` 拡張子を持つ) に記録されます。ソリューションの受信がシミュレートされる場合、`.mtpnsim` ファイルをロードすることができます。シミュレーターは `.mtpnsim` ファイル内にプッシュ通知データのセットを表示します。また、シミュレートすることを希望するプッシュ通知を選択することができます。

プッシュ通知シミュレーションデータの記録

ソリューションの送信をシミュレート中にプッシュ通知を送信するイベントをトリガーする場合、下に表示されるダイアログが表示されます。

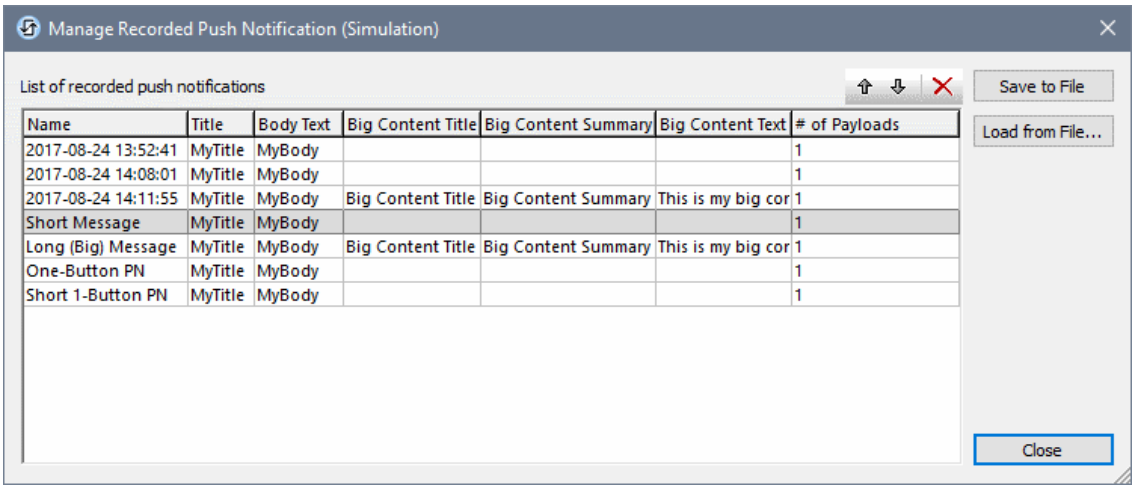


メモリにプッシュ通知データを記録するには、 **「記録」**をクリックします。メモリに複数のプッシュ通知をこの方法で記録することができます。シミュレーションを閉じる場合、記録されたプッシュ通知はメモリ内に存在しますが、ファイルに保存されていないことに注意してください。

ソリューションファイルを保存、または、閉じる場合、メモリ内に保存されていない記録されたプッシュ通知データが存在することを通知され、このデータをファイルに保存するかどうかが問われます。 **「はい」**を選択すると、メモリ内の記録されたプッシュ通知データの異なるセットがソリューションとして同じフォルダー内のMT プッシュ通知シミュレーションファイルに保存されます。このファイルは次のパターンで名前が付けられます: `YourSolutionName.mtpnsim`。追加のプッシュ通知は現在の、または、シミュレーションの次のシミュレーション中に同じファイル内に保存されます。ファイル内のプッシュ通知データのセットはプッシュ通知の記録日時である名前と共に識別されます。

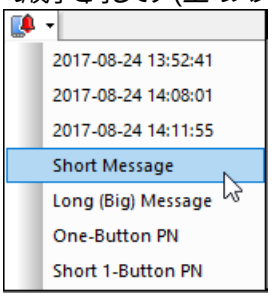
シミュレーターに記録されたシミュレーションデータをロードする

記録されたプッシュ通知シミュレーションデータをMT プッシュ通知シミュレーションファイル (`.mtpnsim` ファイル) からロードするには、ソリューションの受信 (MTD ファイル) を開始し、シミュレーターのツールバー内の **プッシュ通知** アイコン を クリックします。記録されたプッシュ通知 (シミュレーション) の管理ダイアログが表示されます(*下のスクリーンショットを参照*)。 **「ファイルからロードする」**をクリックしロードする `.mtpnsim` ファイルを参照し、 **開く**をクリックします。このファイル内の記録されたプッシュ通知データはダイアログとメモリ内にロードされます (*スクリーンショットを参照*) 。 MTD ファイルに自動的に保存されないことに注意してください。MTD ファイルを保存せずに再ロードした場合、記録されたプッシュ通知データを `.mtpnsim` ファイルから再ロードする必要があります。 **「ファイルに保存する」**をクリックして、MTD ファイルに記録されたプッシュ通知データを保存します。これにより `.mtpnsim` ファイルに再ロードすることを回避できます。プッシュ通知データを他の `.mtpnsim` ファイルからロードした場合、メモリ内のプッシュ通知データを新しいデータが上書きします。MTD ファイル内の記録されたプッシュ通知データを上書きするには、このダイアログ内の **「ファイルに保存する」**をクリックします。



Name	Title	Body Text	Big Content Title	Big Content Summary	Big Content Text	# of Payloads
2017-08-24 13:52:41	MyTitle	MyBody				1
2017-08-24 14:08:01	MyTitle	MyBody				1
2017-08-24 14:11:55	MyTitle	MyBody	Big Content Title	Big Content Summary	This is my big cor	1
Short Message	MyTitle	MyBody				1
Long (Big) Message	MyTitle	MyBody	Big Content Title	Big Content Summary	This is my big cor	1
One-Button PN	MyTitle	MyBody				1
Short 1-Button PN	MyTitle	MyBody				1

記録されたプッシュ通知 (シミュレーション) の管理ダイアログ内で (*上のスクリーンショット*)、各記録されたプッシュ通知は個別のラインに名前、短いメッセージデータ 長いメッセージデータ ペイロードの情報別に表示されています。プッシュ通知の順序を(右上の) **「上に移動」**と**「下に移動」**ツールバーアイコンをクリックすることにより変更することができます。 **「削除」**ツールバーアイコンをクリックしてプッシュ通知を削除することもできます。簡単に識別するためにプッシュ通知の名前を編集することもできます。これを行うには、名前をダブルクリックして編集します。ダイアログ内での変更はメモリ内で行われます。MTD ファイルに変更を保存するには、**ファイル「ファイルに保存する」**をクリックします。
シミュレーション内で使用されているプッシュ通知を選択するには、シミュレーターツールバー内の **プッシュ通知** アイコン のドロップダウン矢印をクリックします。これにより現在メモリ内に存在する全てのプッシュ通知のリストが表示されます (*下のスクリーンショットを参照*)。プッシュ通知の表示の順番は記録されたプッシュ通知 (シミュレーション) ダイアログの現在の順序と同じです (*上のスクリーンショットと下のスクリーンショットを比較してください*)。



このリストからプッシュ通知を選択すると、選択されたプッシュ通知が受信された状態をソリューションはシミュレートします。他のプッシュ通知の受信をシミュレートするには、ドロップダウンリストから新規のプッシュ通知を選択してください。

11.6 チャート

チャートは、ソースドキュメントのデータをグラフィカルに表示します。XPath 式を定義してチャートの各軸のアイテムのシーケンスを指定し、チャートを設定します。MobileTogether Designer は、自動的にチャートを生成します。下のテーブルは、作成することのできるチャートの種類とチャートの軸に必要なデータアイテムを表示しています。

チャートの種類	X-軸 (カテゴリー)	Y-軸 (値)	シリーズの数 (Z軸)
円グラフ(2D, 3D)	テキスト	数値	1
グループ化されていない棒グラフ (2D, 3D)	テキスト	数値	1
グループ化されている棒グラフ (2D, 3D)	テキスト	数値	> 1
カテゴリー線グラフ	テキスト	数値	1 線 = 1 系列
値線グラフ	数値	数値	1 線 = 1 系列
面グラフ スタックされた面グラフ	テキスト	数値	1 面 = 1 系列
ローソク足チャート	テキスト	数値	3 または4
ゲージ	—	数値	1
オーバーレイチャート	テキスト	数値	= 1 または>1チャート

このセクションは次の通り整理されています:

- 「チャートデータセクション」は異なる軸のためのデータ選択の方法を説明しています。
- 「チャート設定と概観」はチャートのプロパティの定義方法を説明しています。

チャートの作成と構成

このセクション:

- チャートの作成
- コンテキストノード
- チャート構成ダイアログ
- チャート設定の編集とデータの選択

チャートの作成

デザインにチャートを挿入するには、[チャート] コントロールを[コントロール] ペインからドラッグして、チャートを挿入するデザイン内の場所にドロップします。

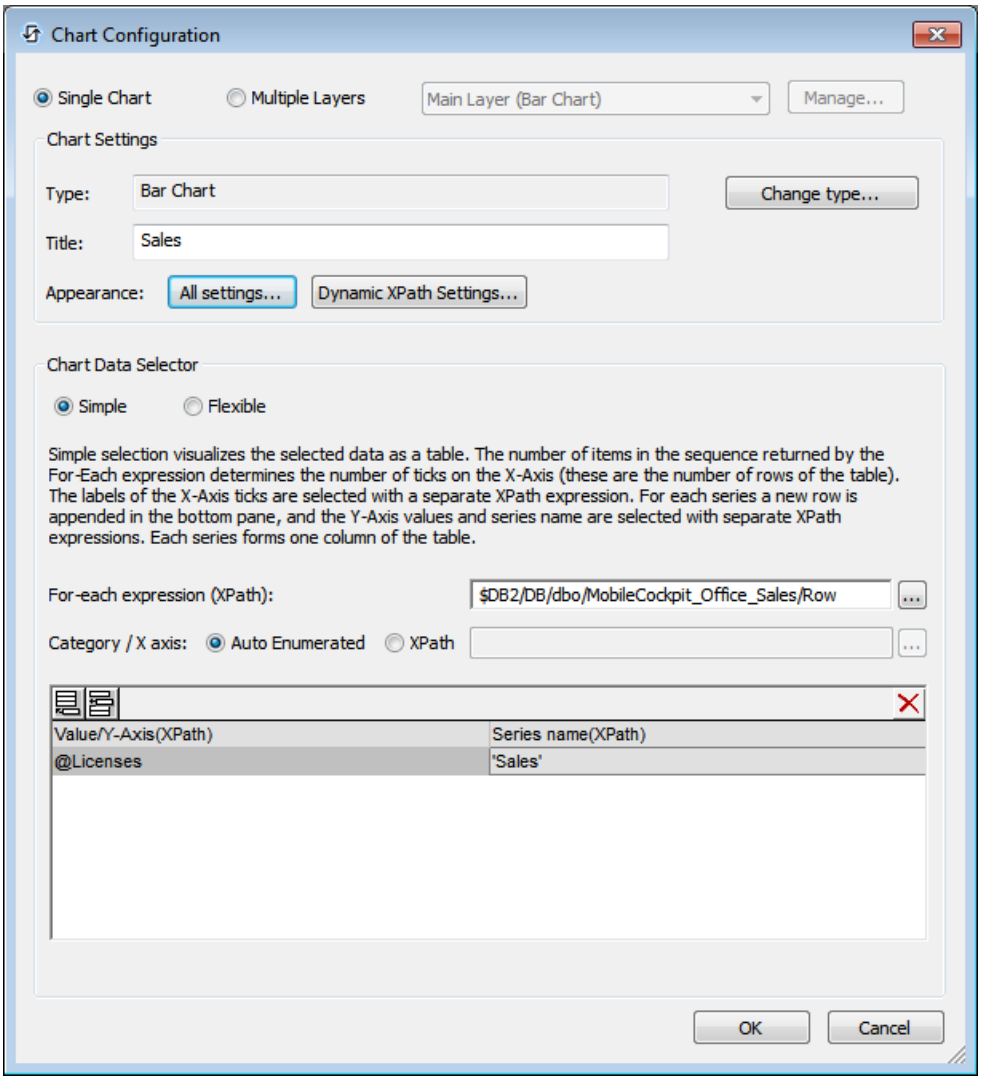
コンテキスト ノード

XML ノードを[ページ ソース] ペインからデザイン内のチャートにドラッグして、このXML ノードをチャートのXPath 式のコンテキスト ノードにします。チャートのコンテキスト ノードは、新しいXML ノードをチャートにドラッグすることによりいつでも変更することができます。コンテキスト ノードは式内のパスロケーターの開始点であるため、チャートのコンテキスト ノードに注意することは重要です。

チャート構成ダイアログ

チャートは[チャート構成] ダイアログ (*下のスクリーンショット*) で、チャート作成時に構成することができます。チャートの構成は次を含みます: (i) チャートの軸のデータの選択の指定 (ii) チャートのプロパティの定義。これらの設定は後で編集することができます。チャートにXML コンテキスト ノードを割り当てた後、以下を行い[チャート構成] ダイアログを表示します:

- チャートをダブルクリックする
- デザイン内のチャートを選択して、[スタイル & プロパティ] ペインのチャート設定プロパティのXPath の編集ボタンをクリックする。



[チャート構成] ダイアログは3 つのパートから構成されています:

- *単一または複数レイヤー*: 複数のレイヤーを選択して、オーバーレイチャートを作成することができます。チャートを重ねることにより、コンポジットを作成することができます。
- *チャート設定*: チャートの種類を選択し、チャートの外観を定義します。
- *チャートデータの選択*: *シンプル オプション* または *フレキシブル オプション* を使用して、チャートの異なる軸のデータを選択します。

チャート設定の編集とデータの選択

チャートが作成された後、チャートの設定またはデータの選択を変更するには、デザインのチャートを右クリックして、**[チャートの作成設定]** を選択します。これにより[チャートの構成] ダイアログ (*上のスクリーンショット*) がポップアップします。ダイアログのチャートの設定またはデータの選択を編集することができ、**[OK]** をクリックして終了します。

チャートデータの選択

このトピックは、チャートデータの選択がどのように作動するか説明する簡単な例を含みます。

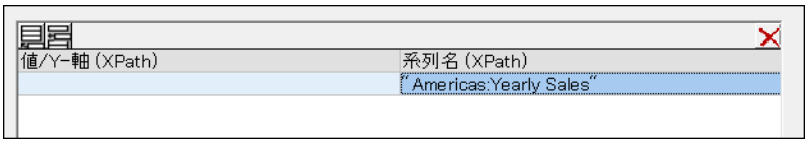
- このサンプルで使用されるXML ファイル: YearlySales.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Data xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="YearlySales.xsd">
        <ChartType>Pie Chart 2D</ChartType>
        <Region id="Americas">
                <Year id="2005">30000</Year>
                <Year id="2006">90000</Year>
                <Year id="2007">120000</Year>
                <Year id="2008">180000</Year>
                <Year id="2009">140000</Year>
                <Year id="2010">100000</Year>
        </Region>
        <Region id="Europe">
                <Year id="2005">50000</Year>
                <Year id="2006">60000</Year>
                <Year id="2007">80000</Year>
                <Year id="2008">100000</Year>
                <Year id="2009">95000</Year>
                <Year id="2010">80000</Year>
        </Region>
        <Region id="Asia">
                <Year id="2005">10000</Year>
                <Year id="2006">25000</Year>
                <Year id="2007">70000</Year>
                <Year id="2008">110000</Year>
                <Year id="2009">125000</Year>
                <Year id="2010">150000</Year>
        </Region>
</Data>
```

---

4 つの XPath 式を使用したチャートデータの選択

下のスクリーンショットは [チャート構成] ダイアログを表示しています。ペインの下に、データ選択のための4 つのXPath 式を入力するための [チャートデータセレクター] ペインがあります。

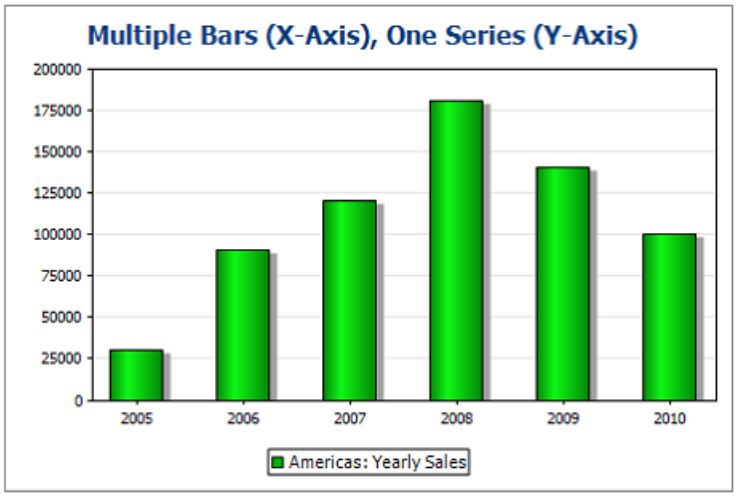


[チャートデータセレクター] ペイン内の4 つのXPath 式は共に作動し、次を行います:

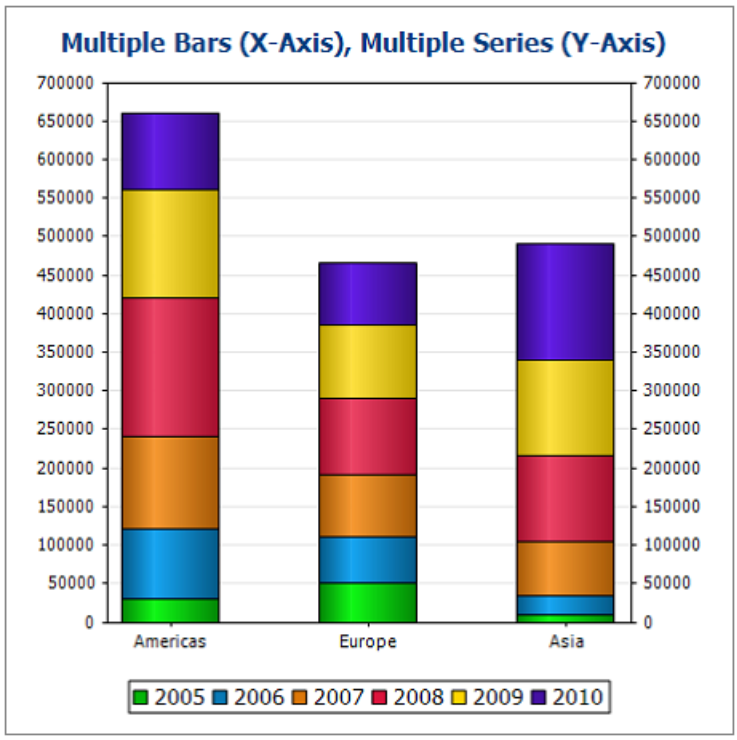
XPath	説明
For-Each	• 他の3 つのXPath 式のコンテキストを設定する。 • X-軸の目盛の数として返されたシーケンスのアイテムの数を設定します。 • 上のスクリーンショットの場合、`Region[1]/Year` 式は6 つのノードアイテムを返します。ですから、X-軸には6 つの目盛があります(*下のスクリーンショッ*

	A)。
X-軸	• 返されたシーケンスのアイテムは、X-軸の目盛に対応したラベルテキストを与えます。 • 上の例では、`@id` 式は、`Year` 要素の各 `id` 属性の値を返します。これらの値は対応する目盛のラベルとなります (*下のスクリーンショット*)。 • 棒グラフを指定したので、棒が目盛に作成されます。
Y-軸	• Y-軸 は、Y-軸のテーブルの行で定義された複数の系列を表示することができます。 • 2 つのXPath 式(系列の値と系列の名前) により各系列は定義されます。 • 例では、`self::node()` XPath 式 (ピリオドで省略されたフォーム) は、コンテキストノードである`Year` 要素である現在のノードを選択します。(X-軸の棒で表示される) 各 `Year` 要素のために、`Year` 要素のコンテンツが、その年のY-軸の値として、として読み込まれ、棒の高さとして表示されます (*下のスクリーンショット参照*)。更に下のスクリーンショットは、Y-軸の複数の系列のチャートを表示しています。
系列名	• この式は、系列の凡例のテキストを提供します。例では、(チャートの下に表示される、*下のスクリーンショット*) 凡例のテキストはテキスト文字列であるXPath 式から取得されます (*上のスクリーンショット参照*)。

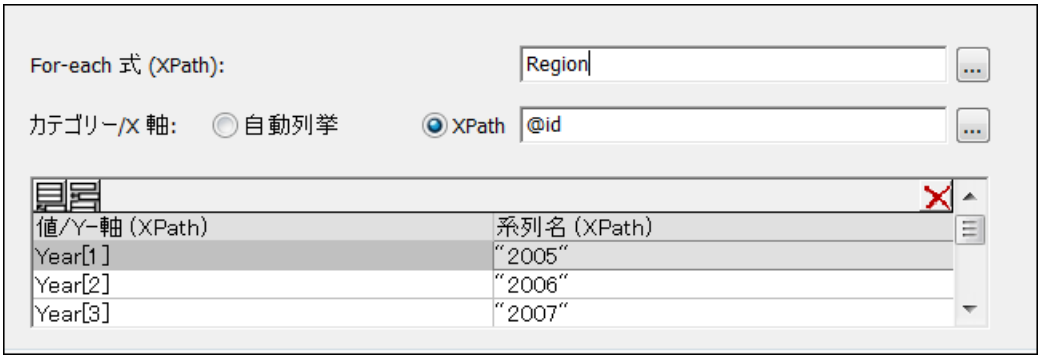
上のスクリーンショットで表示されるデータの選択と`YearlySales.xml` のXML データにより生成されたチャートは、次のスクリーンショットで表示されるチャートのようになります。



上のスクリーンショットは、単一系列の棒グラフを表示しています。下のスクリーンショットは、複数の系列の棒グラフを表示しています。後者の例の場合、各系列の値は棒にスタックされています。



このチャートのXPath 式は下のスクリーンショットに表示されています。



メモ 円グラフおよびゲージチャートには、名前を必要としない単一の系列があります。系列名がデータの選択に入力されると、系列名は無視されます。しかし、グループ化されていない棒グラフでは、単一系列の名前が存在する場合、凡例に使用されます。ゲージチャートに関しては、系列名エントリが無視されることに加え、X-軸データの選択も無視され、Y-軸の選択のみがゲージチャートに使用されます。

チャートデータの選択: シンプル

このセクション:

- はじめに
- コンテキスト ノード
- X とY 軸のためのデータの選択
- For-Each 式がノード以外のアイテムを返す場合

はじめに

[チャート構成] ダイアログ の [チャート データ セレクター] ペインのシンプルオプションにより、データの選択をテーブルとしてビジュアル化することができます。下にリストされる XML ドキュメントを使用して、ビジュアル化のプロセスを説明します。

このサンプルで使用されるXML ファイル: YearlySales.xml

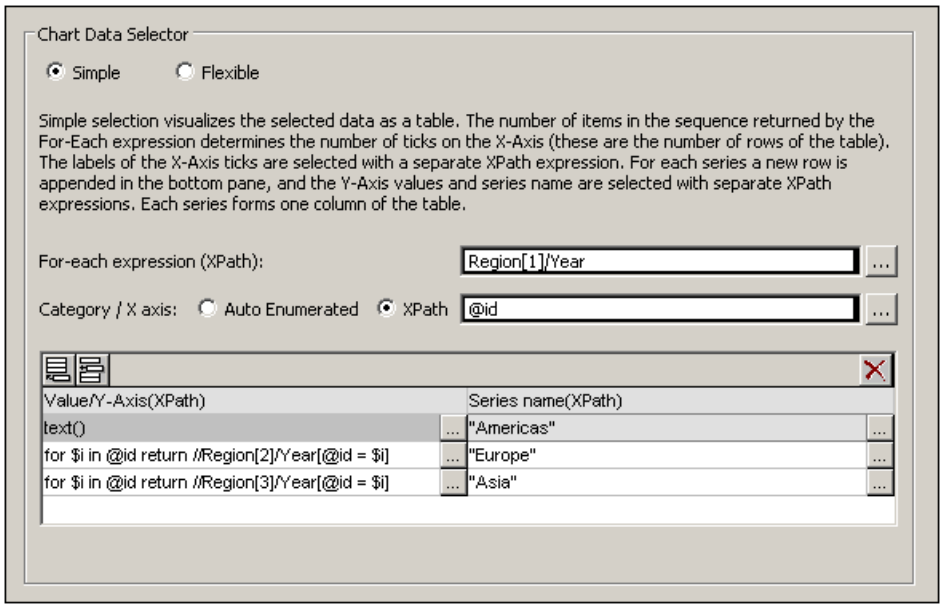
```
<?xml version="1.0" encoding="UTF-8"?>
<Data xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="YearlySales.xsd">
        <ChartType>Pie Chart 2D</ChartType>
        <Region id="Americas">
                <Year id="2005">30000</Year>
                <Year id="2006">90000</Year>
                <Year id="2007">120000</Year>
                <Year id="2008">180000</Year>
                <Year id="2009">140000</Year>
                <Year id="2010">100000</Year>
        </Region>
        <Region id="Europe">
                <Year id="2005">50000</Year>
                <Year id="2006">60000</Year>
                <Year id="2007">80000</Year>
                <Year id="2008">100000</Year>
                <Year id="2009">95000</Year>
                <Year id="2010">80000</Year>
        </Region>
        <Region id="Asia">
                <Year id="2005">10000</Year>
                <Year id="2006">25000</Year>
                <Year id="2007">70000</Year>
                <Year id="2008">110000</Year>
                <Year id="2009">125000</Year>
                <Year id="2010">150000</Year>
        </Region>
</Data>
```

コンテキスト ノード

XML ノードを[ページ ソース] ペインからデザイン内のチャートにドラッグして、このXML ノードをチャートのXPath 式のコンテキスト ノードにします。チャートのコンテキスト ノードは、新しいXML ノードをチャートにドラッグすることによりいつでも変更することができます。コンテキスト ノードは式内のパスロケーターの開始点であるため、チャートのコンテキスト ノードに注意することは重要です。

X とY 軸のためのデータの選択

[チャート データ セレクター] ペイン (*下のスクリーンショット*) で、スクリーンショットに表示されているようにデータの選択をします。チャートは、データ ノード内に挿入されているため、For-Each 式のコンテキスト ノードは データ ノードです。



チャートデータは以下のテーブルのようにビジュアル化することができます。各 `Region[1]/Year` 要素に、新しい行が作成され、X-軸 とY-軸のXPath 式が対応する`Region[1]/Year` 要素のコンテキストで評価されます。

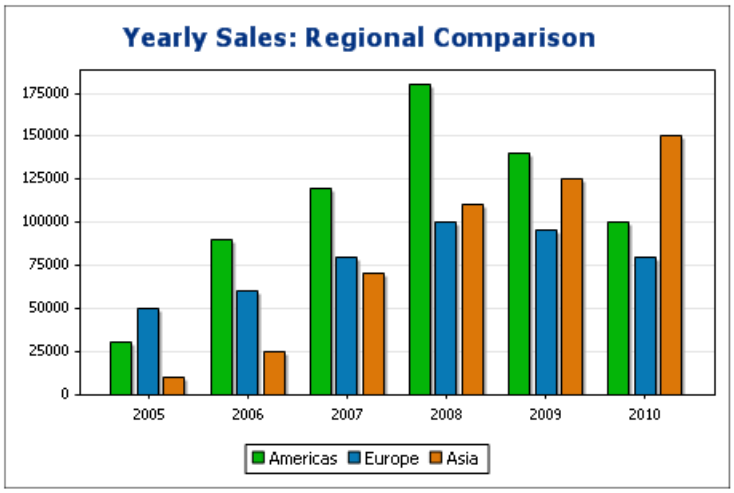
For-Each XPath	X-軸	系列のY-軸		
		Americas	Europe	Asia
`Region[1]/Year[1]`	`@id`	`text()`	`XPath-1`	`XPath-2`
`Region[1]/Year[2]`	`@id`	`text()`	`XPath-1`	`XPath-2`
`Region[1]/Year[3]`	`@id`	`text()`	`XPath-1`	`XPath-2`
`Region[1]/Year[4]`	`@id`	`text()`	`XPath-1`	`XPath-2`
`Region[1]/Year[5]`	`@id`	`text()`	`XPath-1`	`XPath-2`
`Region[1]/Year[6]`	`@id`	`text()`	`XPath-1`	`XPath-2`

- For-Each 式 `Region[1]/Year` は(テーブルの行となる) 6つのノードを返します。For-Each 式により返されたシーケンスのアイテムの数は、X-軸の**目盛の数**を決定します。
- X-軸 のXPath 式は、各 `Region[1]/Year` 要素の`@id` 属性の値を返します。これらの値は**X-軸の目盛のラベル** となります。目盛の数よりもラベルの数が多い場合は、メモリが生成され、すべてのラベルが表示されるようになります。目盛よりもラベルの数が少ない場合、(対応するラベルが存在しないため) 後方の目盛にラベルが付きません。自動的に列挙されたオプションは1 から始まる整数のシーケンスを生成し、各整数をX-軸の目盛に順番に割り当てます。
- `Americas` series (`text()`) のXPath 式は、`Region[1]/Year` 要素のコンテンツを返します。この式

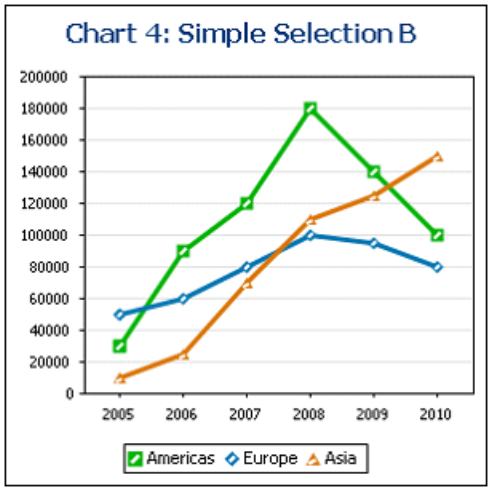
は、希望する値を効果的に返す限り、(下で説明される) Europe および Asia 系列 に類似しています。

- `Europe` 系列の XPath 式は `for $i in @id return //Region[2]/Year[@id=$i]` です。式は次を実行します: (i) 現在の `Region[1]/Year/@id` 属性の値を検索し、(ii) 現在の `Region[1]/Year` 要素の `@id` 値と同じ `@id` 値を持つ `Region[2]/Year` 要素のコンテンツを返します。
- `Asia` 系統の XPath 式は `Europe` 系統の XPath 式と類似しています。

このデータの選択で生成された棒グラフは以下のように表示されます:

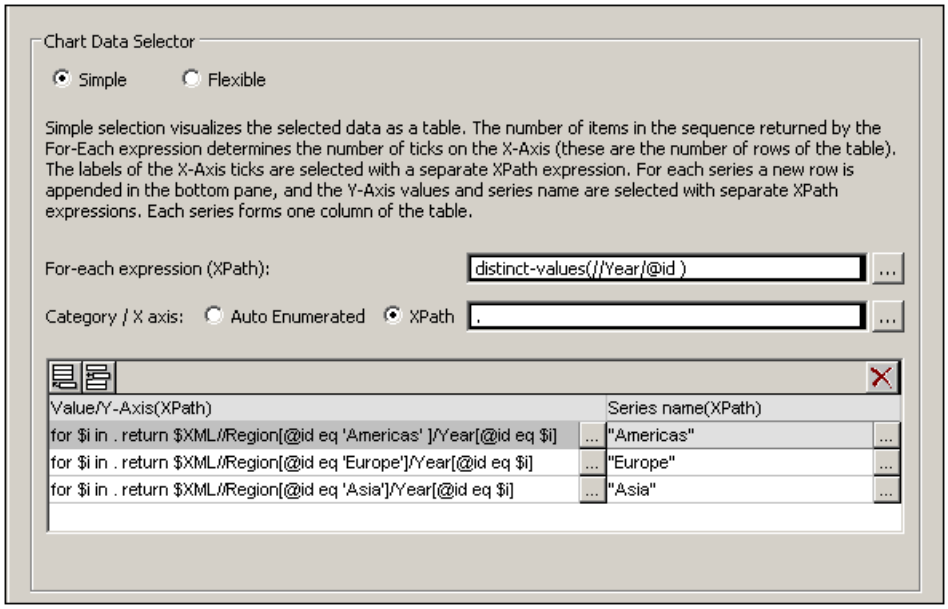


このデータの選択の折れ線グラフは次のように表示されます:



For-Each 式がノード以外のアイテムを返す場合

X-軸の目盛の数は、For-Each XPath 式により返されたアイテムの数によるため、6 つの一意の年度の値を返す、下のスクリーンショット(`distinct-values(//Year/@id)`) の XPath 式も 6 つの目盛を X-軸に生成します。シーケンスにより返されたアイテムは、しかしながら、ノードではなくアトミックな値です。この結果、コンテキスト アイテムとして使用することもできますが、ツリー内のロケート ノードのコンテキスト ノードとしては使用することができません。しかし、値をベースにして、ノードをロケートするために使用することができます。



上のスクリーンショットで表示されているデータの選択で、以下の点に注意してください:

- X-軸 とY-軸 のデータ選択は、直接の出力とテストの値のフィルターとしてFor-Each 式により返されたアトミックな値を使用します。
- XPath 式のロケーション ステップは、ドキュメント ノード(`$XML` in `$XML//Region...`) から開始します。アトミックな値がロケーションコンテキストを提供しないため、これは必要です。

チャート データテーブルは以下を評価します:

For-Each XPath	X-軸	系列のためのY-軸		
		Americas	Europe	Asia
2005	2005	XPath-1	XPath-2	XPath-3
2006	2006	XPath-1	XPath-2	XPath-3
2007	2007	XPath-1	XPath-2	XPath-3
2008	2008	XPath-1	XPath-2	XPath-3
2009	2009	XPath-1	XPath-2	XPath-3
2010	2010	XPath-1	XPath-2	XPath-3

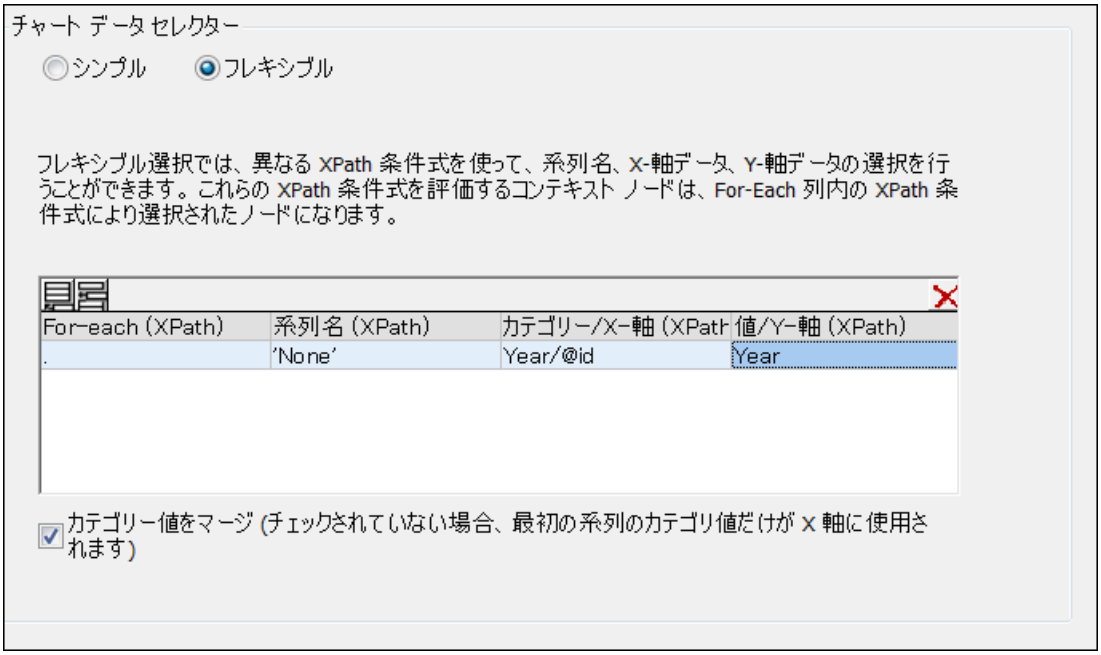
チャートデータの選択: フレキシブル

このセクション:

- フレキシブルなチャートデータの選択について
- 1 つの行、1 つの系列
- 3 つの行、3 つの系列、カテゴリーの値はマージされていません
- 3 つの行、3 つの系列、カテゴリーの値はマージされています
- 1 つの行、3 つの系列
- チャートデータの選択のルール

フレキシブルなチャートデータの選択について

[チャート構成] ダイアログの [チャートデータセレクター] ペイン (*下のスクリーンショット*) で、フレキシブルオプションにより、XPath 式を使用して、系列軸 (Z-軸)、X-軸、および Y-軸のデータを自由に選択することができます。軸の XPath 式は、その軸にプロットされたアイテムのシーケンスを返します。軸に対してのこれらの (アイテムの) シーケンスは、チャートを生成するために順番になれらべられます。



次の点に注意してください:

- 系列は X-軸 (カテゴリー軸) 目盛のセットに対してプロットされた値の系列を指します。2番目の系列は、同じ X-軸の目盛の2番目の値のセットに対してプロットされます。例えば、X-軸が 2008、2009、と 2010 年度を表し、Y-軸が売上高を表す場合、系列 1 は、 米国 (米国でのこれら3 年間の売り上げ)、系列 2 は、欧州 (欧州でのこれら3 年間の売り上げ) を表示します。このデータに対し棒グラフが選択されている場合、X-軸の各年度 (2008、2009、2010) には、2 本の棒 (米国と欧州) が表示されます。円グラフおよび単一棒グラフの場合、1 つの系列のみ使用することができます。各チャートの種類に関下は チャートの種類のテーブルを参照してください。
- [チャートデータセレクター] ペインの各行は、系列を表します。
- チャートの XPath コンテキスト ノードは、デザイン内のチャートコントロールの上にページソースペインからノードをドロップすることにより定義されます。
- For-Each 列の XPath 式は、他の3 つの XPath 式の評価のコンテキストを提供します。For-Each XPath 式は、デザインに挿入されたノードのコンテキストで評価されます。

次の例は、軸に対してデータを選択する際の重要な点を説明しています。下にリストされるXML ドキュメントを参照しています。

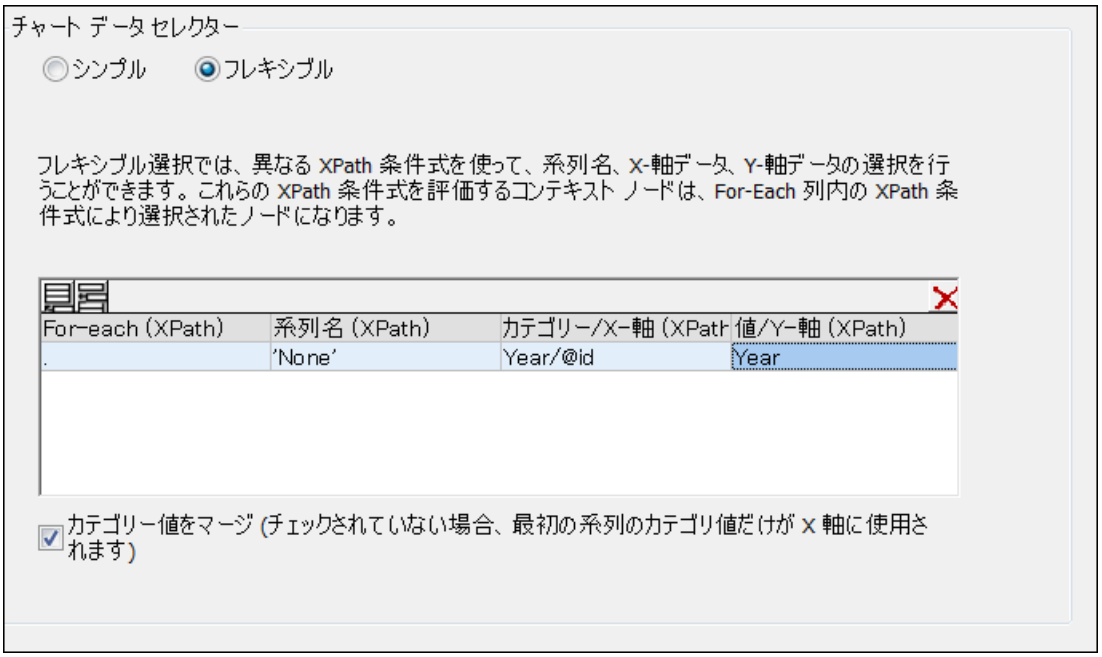
- このサンプルで使用されるXML ファイル: YearlySales.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Data xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="YearlySales.xsd">
        <ChartType>Pie Chart 2D</ChartType>
        <Region id="Americas">
                <Year id="2005">30000</Year>
                <Year id="2006">90000</Year>
                <Year id="2007">120000</Year>
                <Year id="2008">180000</Year>
                <Year id="2009">140000</Year>
                <Year id="2010">100000</Year>
        </Region>
        <Region id="Europe">
                <Year id="2005">50000</Year>
                <Year id="2006">60000</Year>
                <Year id="2007">80000</Year>
                <Year id="2008">100000</Year>
                <Year id="2009">95000</Year>
                <Year id="2010">80000</Year>
        </Region>
        <Region id="Asia">
                <Year id="2005">10000</Year>
                <Year id="2006">25000</Year>
                <Year id="2007">70000</Year>
                <Year id="2008">110000</Year>
                <Year id="2009">125000</Year>
                <Year id="2010">150000</Year>
        </Region>
</Data>
```

---

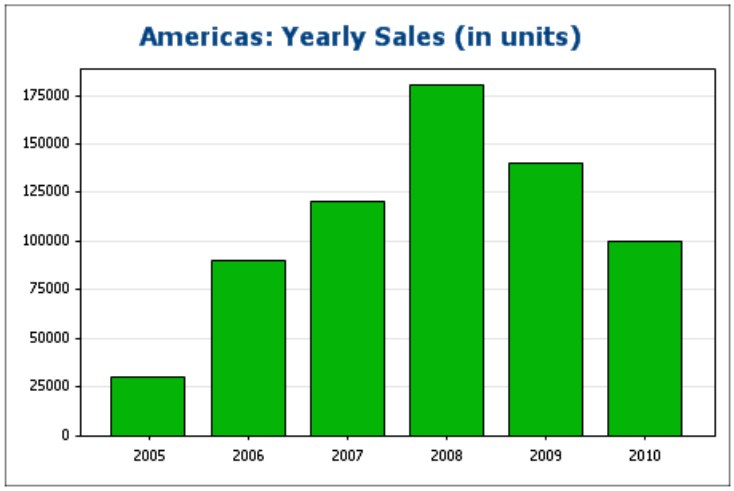
1 つの行、1 つの系列

2D 棒グラフを`Region` 要素(3つの要素があります:米国、欧州、アジア)に対して生成するとします。要素ノードの中のデザインにチャートを作成するとします。チャート データセレクターのFor-Each XPath 式のコンテキストノードは、`Region` 要素となります。



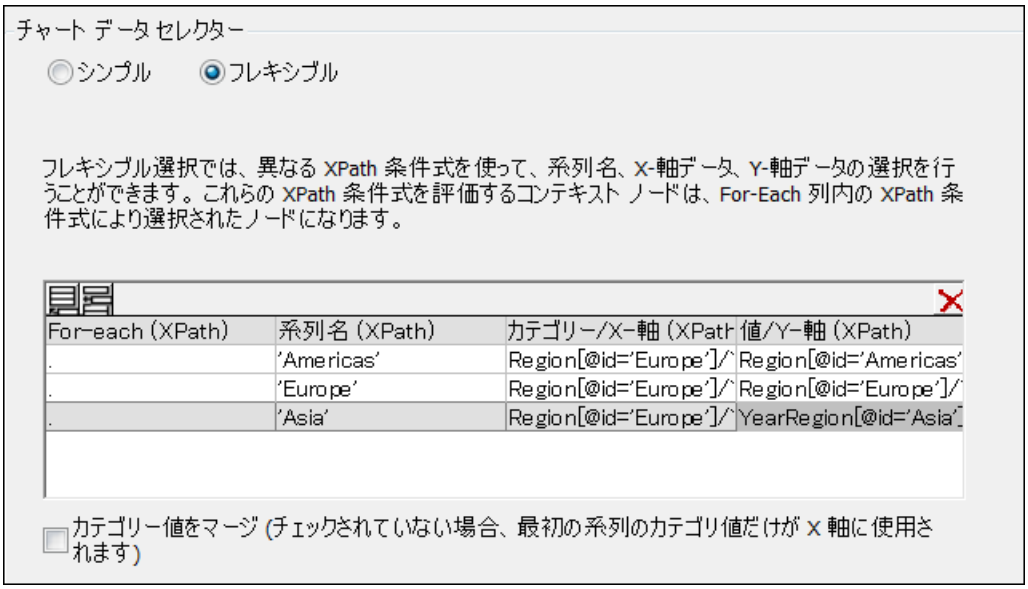
上のスクリーンショットで表示されているデータの選択では、For-Each 式は、(`Region` 要素である) 現在のノードを返します。ですから、`Region` 要素は、その他3つのXPath 式 (系列、X-軸、およびY-軸) のコンテキスト ノードとなります。このチャートに系列は1つしか存在しないため、系列名は必要なくこの列は空白になります。X-軸の選択は6 つの値を返します。6 はX-軸の目盛の数となり、シーケンスの6つのアイテムはX-軸の目盛の相対するラベルとなります。Y-軸の選択もまた6つのアイテムを返します。各アイテムは対応するX-軸の目盛に対してY-軸にプロットされます。チャートは、`Region` 要素内に作成されているため、チャートは3 つの`Region` 要素のためにそれぞれ作成されます。各チャートに対して、その特定の`Region` 要素の子孫ノードが使用されます。

米国 地域のチャートの出力は以下のように表示されます:



3 つの行、3 つの系列、カテゴリーの値はマージされていません

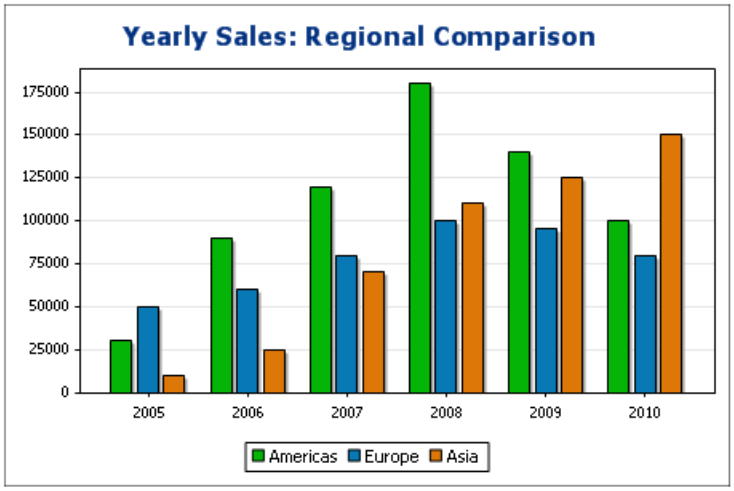
複数の系列を作成する場合、下のスクリーンショットで表示されているように、行を追加することができます。



上のデータの選択に関して次の点に注意してください:

- (チャートは`Data` ノードをXPath コンテキストノードとするため) 各行は、系列を定義し、すべての行は `Data` 要素をコンテキストノードとして持ちます。
- 最初の行は、米国の系列を定義し、文字列式を系列の名前として与えます。X-軸の値は欧州地域の`Year/@id` 値を使用して選択されます。(すべての地域はの`Year/@id` 値を持つため、どの地域を選択してもかまいません。)。最初(米国)の系列 Y-軸の値は、述語フィルターを使用して、米国地域のために選択されます。
- 2 番目と3 番目の系列は、1 番目の系列と同じパターンで行われます。各系列のX-軸の選択は同じですが、*カテゴリー値をマージ*のチェックボックスがチェックされていないため、2 番目と3 番目の式は無視されます。(値がマージされたとしても、各系列の値は同じなため、違いはありません; 新しい異なる値のみがカテゴリーの値に追加されます。)

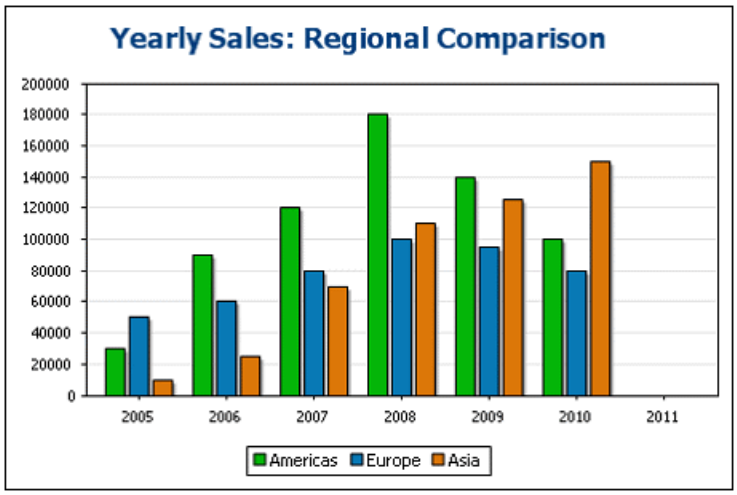
上のデータ選択を使用して生成されたチャートは以下のように表示されます:



3 つの行、3 つの系列、カテゴリーの値はマージされています

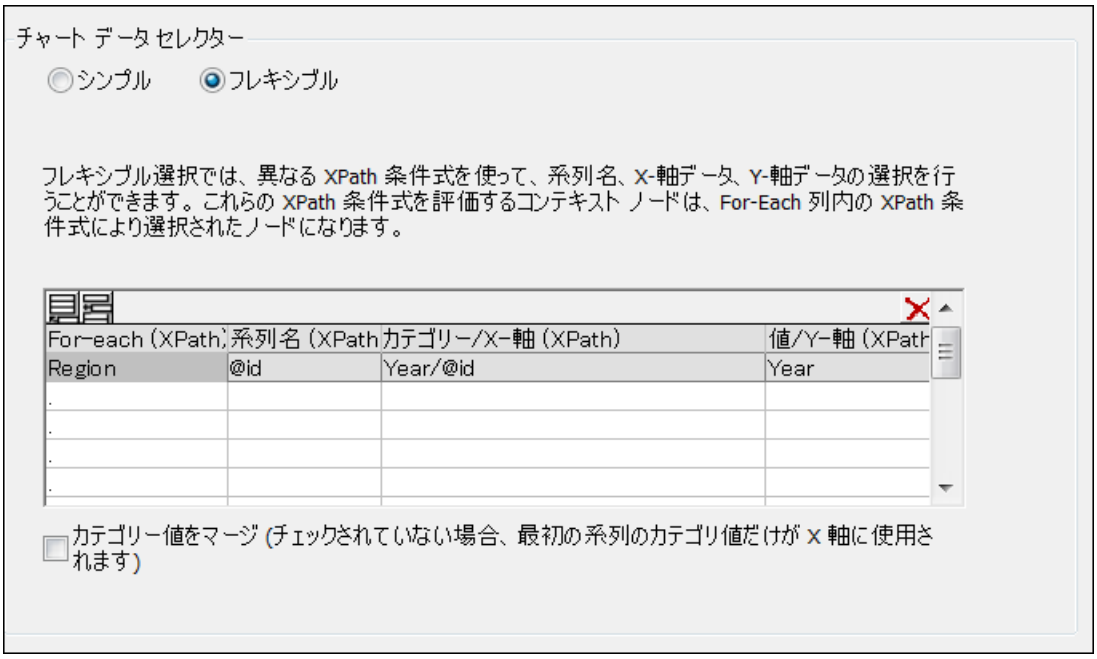
この例のデータの選択 (*下のスクリーンショット参照*) は、前述の2 つの例と3 つの面で異なります: (i) 3 番目の系列の X-軸の選択に余分なアイテム (2011) が追加されています (ii) *カテゴリーの値をマージ*チェックボックスがチェックされています (iii) Y-軸の目盛の間隔が手動で 20000 に設定されています。

この変更の効果は新しいアイテム (2011) をX-軸の結果シーケンスに追加することです。チャートは以下のように表示されます:



1 つの行、3 つの系列

この例で作成されるチャートにはXPath コンテキストノードとしてData ノードがあります (上のXML ドキュメントを参照してください)。1 つの行のみがデータの選択に使用されていますが、3 つの系列を生成します。これは、For-Each 列の XPath 式が3 つのアイテムのシーケンスを返し、明確的に3 つの系列を作成するからです。



各系列は異なるRegion 要素をコンテキストノードとして持つため、するためFor-Each 系列、系列名、X-A軸、Y-軸の選択は、異なる地域に対応します。このデータの選択のチャートは次のように表示されます:

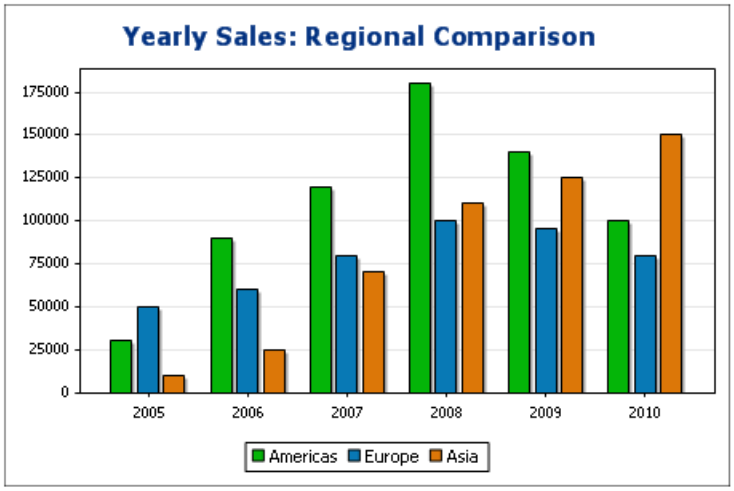


チャート データの選択のルール

異なるチャートの軸のためにデータを選択するチャートデータセレクターを使用する場合、次の点に注意してください:

1. 棒の数 (または円グラフの扇形の数など) は、X-軸 またはY-軸の単一のデータ行の選択のシーケンスの大きい方のアイテムの数と同じです。ですから、(ラベルを与える) X-軸 には5 つのアイテムがあり、(値を与える) Y-軸 に

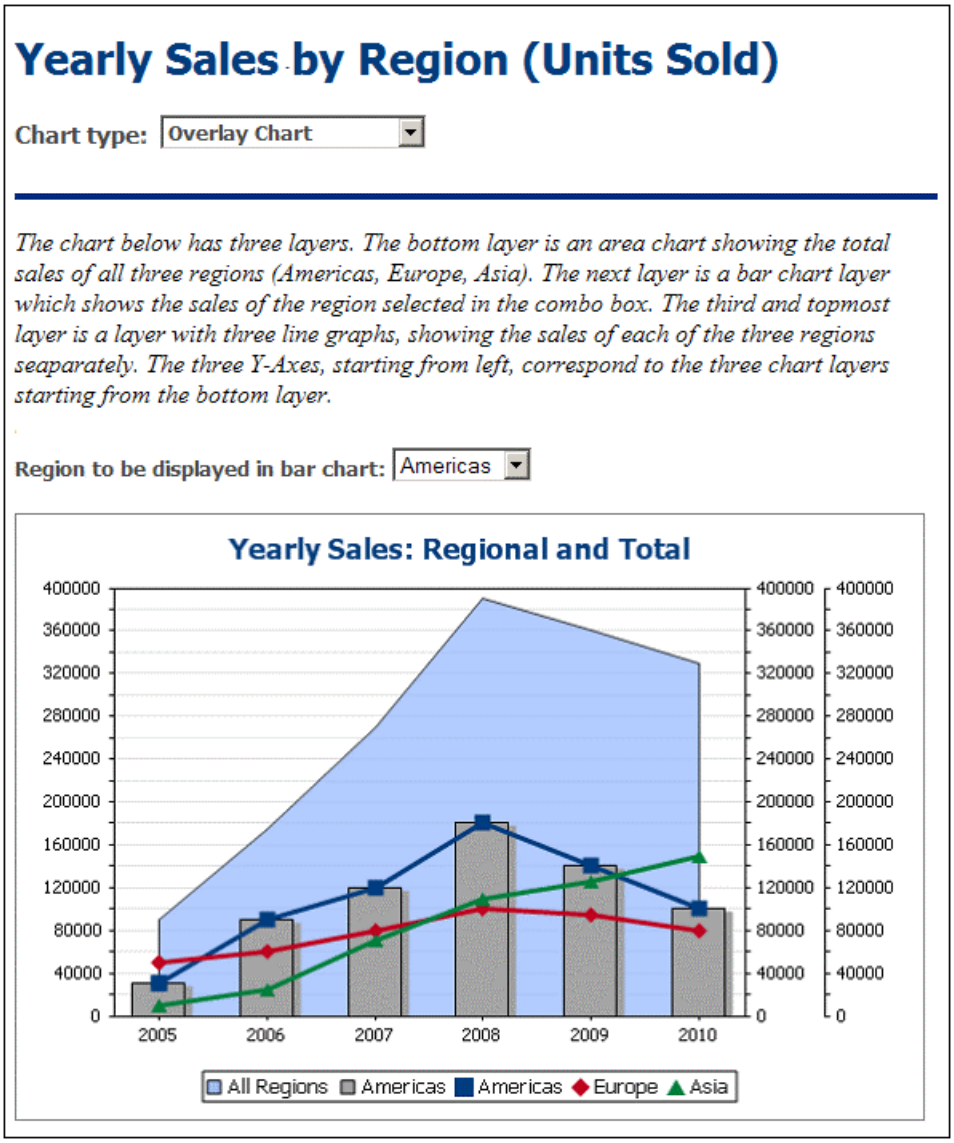
は6 つのアイテムがある場合、6 本の棒がプロットされますが、最後の棒にはラベルがありません。X-軸に6 つアイテムがあり、Y-軸に5 つアイテムがある場合、6 本の棒がプロットされ、最後の棒にはラベルがありますが値は0 となります。

2. 系列の数は、For-Each 列の式により返されたすべてのシーケンスのアイテムの累積された数となります。
3. 系列の名前はZ-軸 (または系列の名前 軸) のXPath 式と共に選択されます。データの選択行のXPath 式が空の場合、名前の系列は作成されません。また、XPath 式が系列の数より少ない数のアイテムのシーケンスを返す場合、系列には名前が与えられません。

オーバーレイチャート

オーバーレイチャート (*下のスクリーンショット*) を選択することにより、異なる種類の複数のチャートを結合することができます。下のスクリーンショット内のオーバーレイチャートはスクリーンショット内のテキストにより説明されています。以下の種類のチャートを結合することができます:

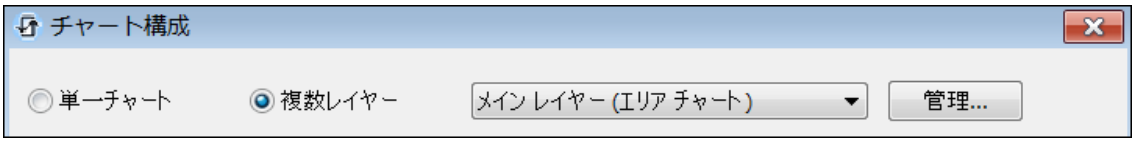
- 2D 棒グラフとスタックされた棒グラフ
- 折れ線グラフ
- 面グラフとスタックされたエリアチャート
- ローソク足グラフ



オーバーレイチャートの作成

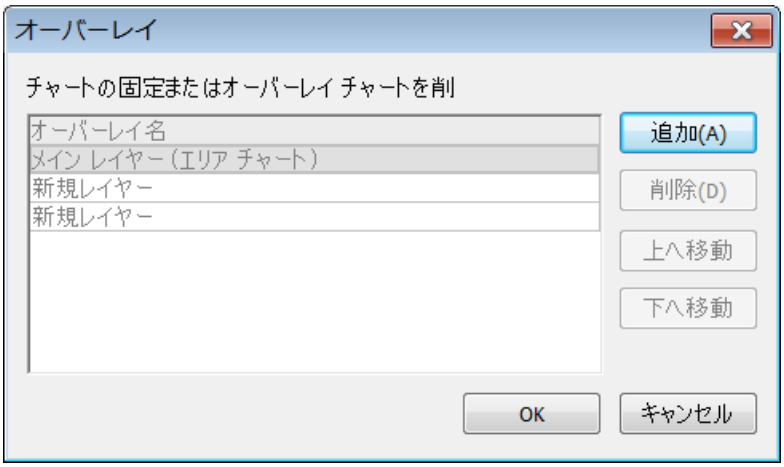
チャート構成ダイアログ内の複数のレイヤーボタンを選択することによりオーバーレイチャートを作成することができます (*下のス*

クリーンショット参照)。メインのれーやーは自動的に作成されます。メインのレイヤーの種類を選択下の地に、チャートの設定とデータの選択を通常通り行います。



オーバーレイチャートのレイヤーの管理

新しいレイヤーをオーバーレイチャートに追加し、削除することもできます。これを行うには、チャート構成オーバーレイ内の**管理**ボタンをクリックします(*上のスクリーンショット参照*)。これによりオーバーレイが表示されます（*下のスクリーンショット*)。新しいレイヤーを追加するには**追加**をクリックします。レイヤーを削除するには**削除**をクリックします。



レイヤーはチャートの上に重ねられますが、下のレイヤーの外観を損ねる可能性があることに注意してください。チャートを透明にすることも可能ですので、レイヤーのアレンジメントによっては最適の結果を得ることができない場合があります。例えば、棒グラフが折れ線グラフの上に配置されている場合、折れ線の一部の外観が損なわれる場合があります。これらの点を考慮して、レイヤーの順序を決定してください。

多種のチャートのためのデータ選択

各チャートのためのデータ選択はチャートの型によって異なり、各チャートの通常の方法で行われます。チャートのためのデータ選択の概要に関しては、チャートデータの選択: シンプルとチャートデータの選択: フレキシブルのセクションを参照してください。

オーバーレイチャートに関連する以下の一般的な説明は重要ですので、チャート作成の際に考慮してください：

- すべてのチャートレイヤーではX-軸 は共有され、メインレイヤーで定義されるX-軸 です。このX-軸 定義は、他のチャートで評価することはできず、チャートを描くことはできません。他のレイヤーで定義されるX-軸 の定義は無視されます。
- 各チャートのY-軸の系列は、個別に選択されます。各レイヤーのY- 軸は一番下のレイヤー(メインレイヤー)のために左から右に互いに平行に描かれます。チャート設定内で各 Y-軸 のためにラベルを指定することができます。

重要なチャート設定

以下のチャート設定に関連します:

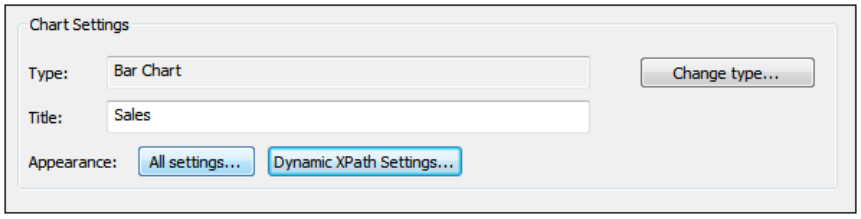
- チャートのタイトルは、メインレイヤーのチャートタイトルの設定で指定することができます。
- X-軸 設定は、メインレイヤーの設定で指定されます。
- Y-軸 ラベルは各チャートのために個別に設定されます。
- Y-軸 開始点 (最小値) と最終点 (最大値)、およびメモリの間隔は、各チャートのために指定することができます。これにより軸が互いに対して調整されます。
- 各チャート内の系列の色は個別に設定することができます。

(C) 2018 Altova GmbH

チャートの設定と外観

チャートの設定は次の通り構成されています:

- チャートの基本設定 により、チャートの種類とタイトルを選択することができます。基本チャート設定はチャート構成ダイアログのチャート設定ペインにより定義することができます。 (*下のスクリーンショット参照*)。
- チャートの詳細設定により、チャートの外観 (タイトル、凡例、色、フォントなど) を変更することができます。詳細設定は、外観の変更ダイアログで定義されます。このダイアログにアクセスするには、チャート構成ダイアログの **[すべての設定]** をクリックしてください (*下のスクリーンショット参照*)。
- 動的な XPath 設定 はチャート設定ペイン内の **[動的な設 XPath 設定]** をクリックすることによりアクセスすることができます(*下のスクリーンショット参照*)。



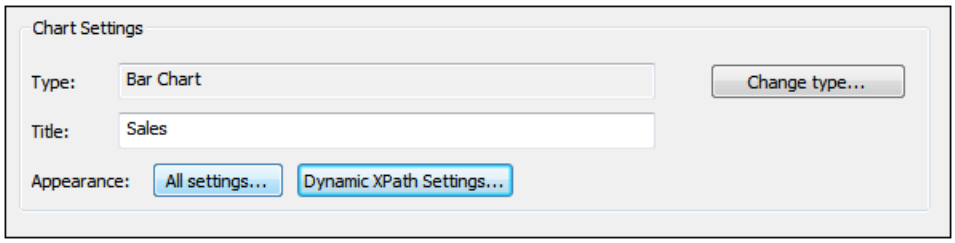
チャートの基本設定

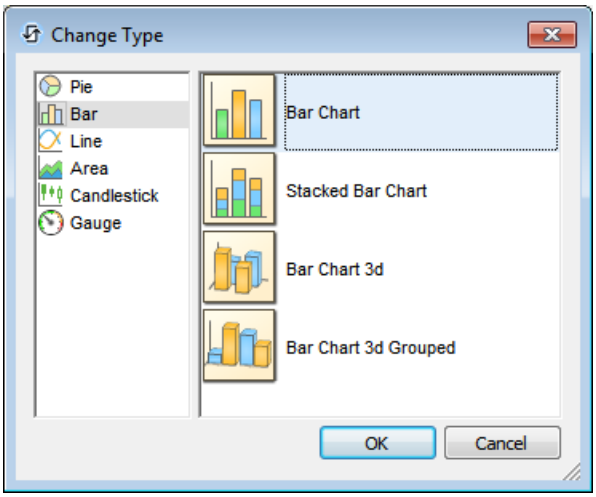
このセクション:

- チャートの種類の設定
- チャートの種類のリスト
- 他の基本設定

チャートの種類の設定

最も基本的な設定はチャートの種類です。チャートの種類を選択するには、チャート構成ダイアログのチャート設定ペイン内の **「種類の変更」** をクリックします。

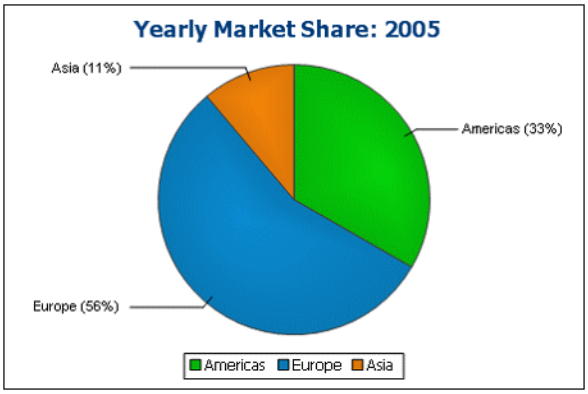




チャートの種類は、チャートウィンドウにある **「種類の変更」** ボタンをクリックすることで表示される、種類の変更ダイアログボックスにて定義されます。(上のスクリーンショット)、チャートの種類を選択して、**[OK]** をクリックします。

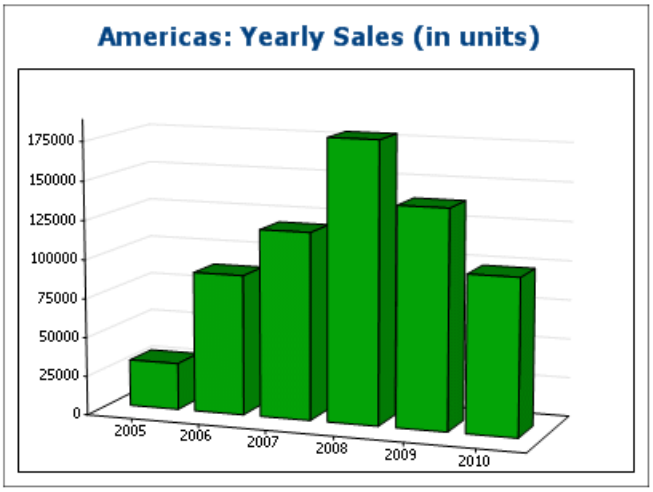
▾ *円チャート*

円グラフでは、単一軸の値を表示し、別の軸により値のラベル付けを行います。

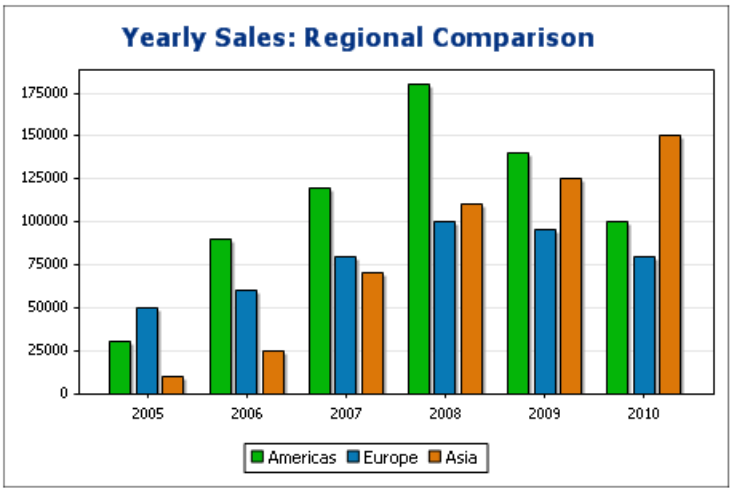


棒グラフ

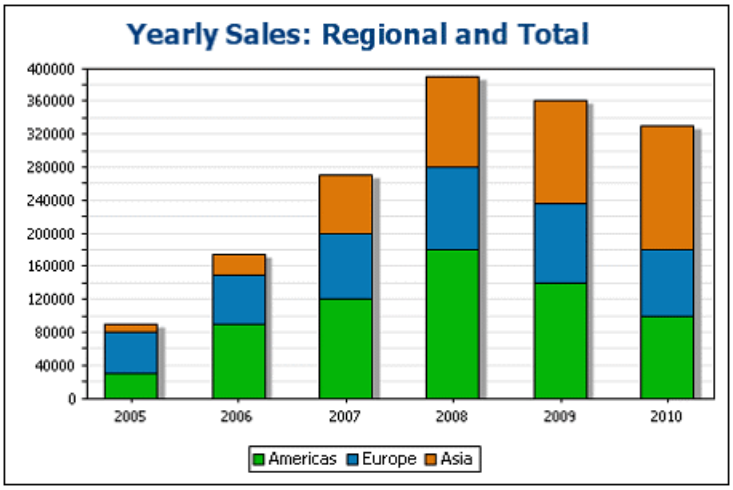
棒グラフでは、X-軸に数値以外の値を使用することができます。



上のサンプルで表示されるように次のセットの3つの値を使用することもできます: (i) 地域、(ii) 年度、(iii) 売り上げ。 棒グラフは2D ならびに3D にて表示することができます。

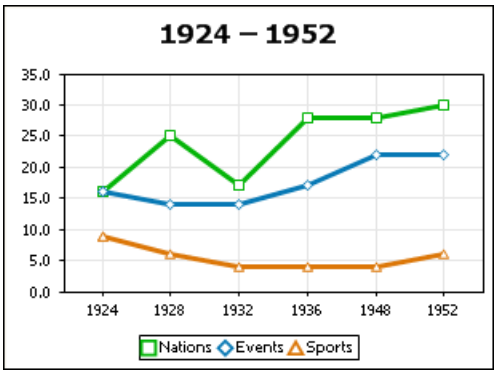


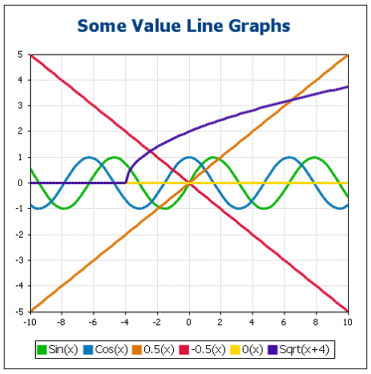
更に、スタックされた棒グラフを使用することで、チャートに含まれている系列の合計を表示することができます。



- *折れ線グラフ*

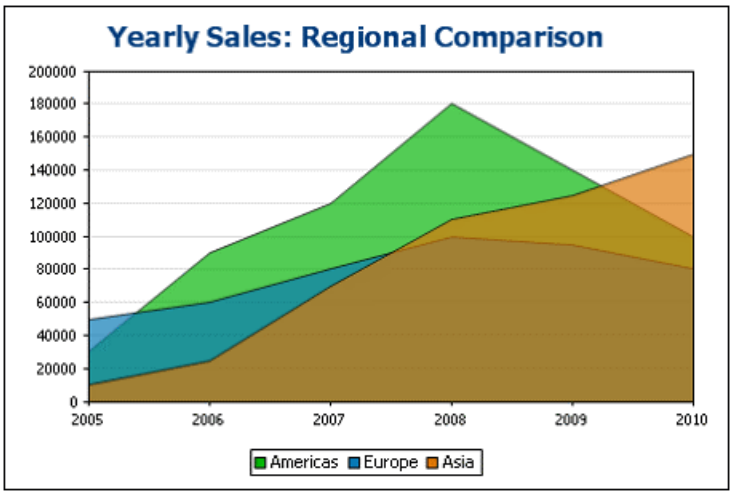
  通常の折れ線グラフ (下左のスクリーンショット) と値付き折れ線グラフ (下右のスクリーンショット) の違いは、値付き折れ線グラフでは、X-軸に数値カラムしか使用することができない点にあります。折れ線グラフのX-軸に文字列を使用したい場合には、**[種類の変更]** ダイアログボックスの右ペインから折れ線グラフを選択してください。





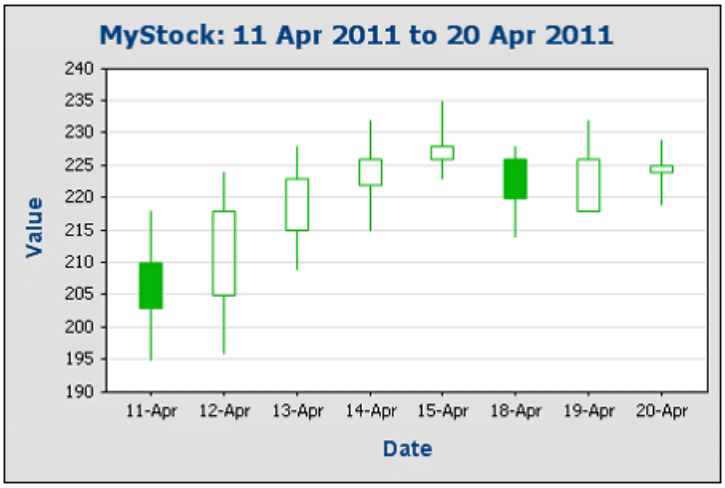
エリアチャート

エリアチャートとは、線の下の面が色付けされる折れ線グラフの1種類です。エリアチャートはスタックすることができます。



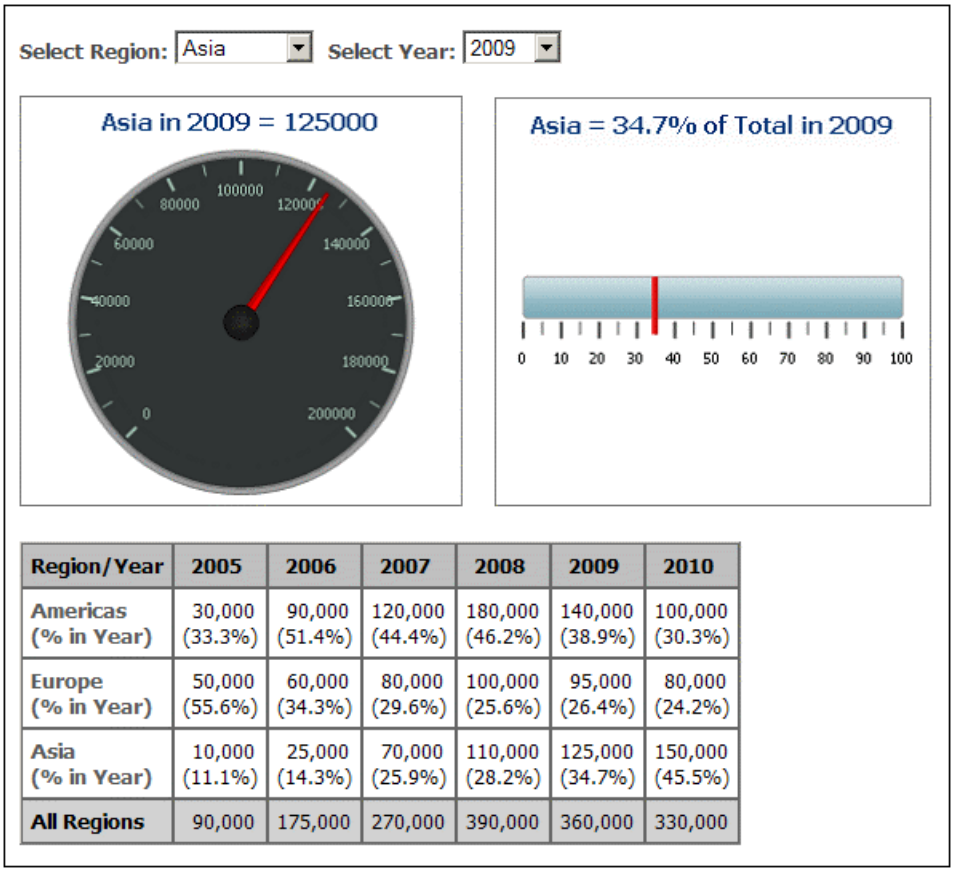
ローソク足チャート

ローソク足チャートは、ある期間内における証券やデリバティブ、または通貨といった値の動きを表示するために使用されます。チャートには資産の動きが表示され、毎日の終値だけでなく、高値、安値、そしてオプションとして始値を表示することができます。4つの系列（始値、高値、低値、終値）を定義すると、チャートは以下のように表示されます。



- ゲージチャート

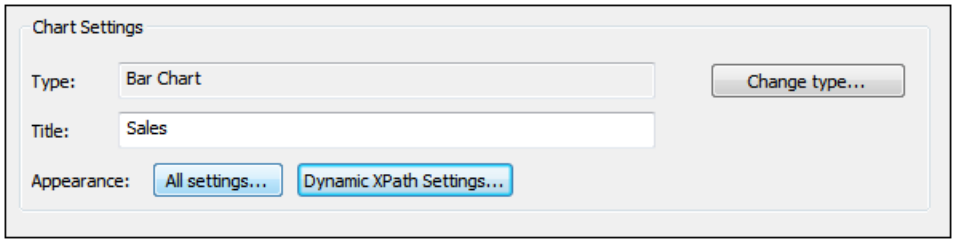
  ゲージチャートには1つの値か最低ならびに最高値とともに表示されます。



Region/Year	2005	2006	2007	2008	2009	2010
Americas (% in Year)	30,000 (33.3%)	90,000 (51.4%)	120,000 (44.4%)	180,000 (46.2%)	140,000 (38.9%)	100,000 (30.3%)
Europe (% in Year)	50,000 (55.6%)	60,000 (34.3%)	80,000 (29.6%)	100,000 (25.6%)	95,000 (26.4%)	80,000 (24.2%)
Asia (% in Year)	10,000 (11.1%)	25,000 (14.3%)	70,000 (25.9%)	110,000 (28.2%)	125,000 (34.7%)	150,000 (45.5%)
All Regions	90,000	175,000	270,000	390,000	360,000	330,000

他の基本設定

チャート設定ペインでは、チャートのタイトルを設定することができます (*下のスクリーンショット参照*)。



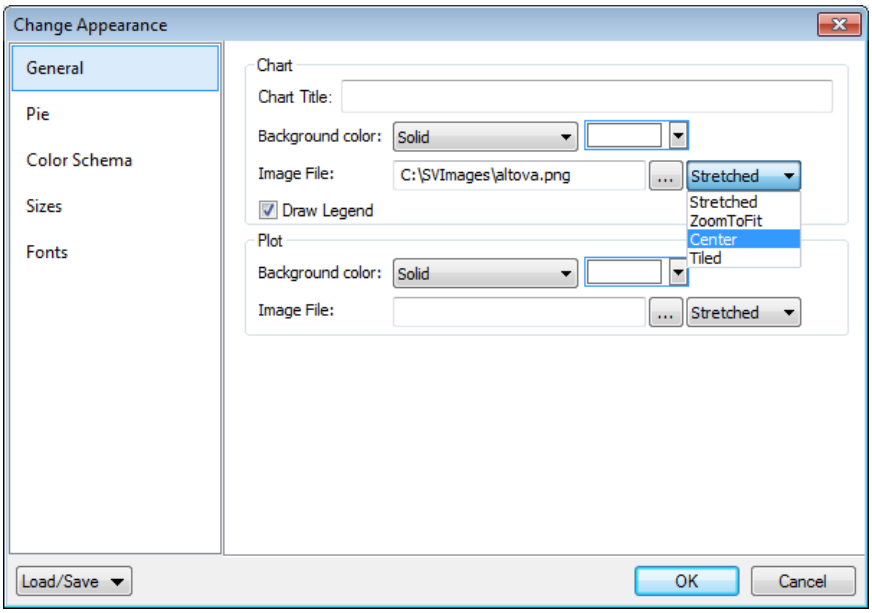
チャートの詳細設定

このセクション:

- 詳細設定へのアクセス
- 詳細設定の概要
- チャート設定のロード、保存、リセット

詳細設定へのアクセス

チャートの詳細設定にアクセスするには以下を行います: チャート構成ダイアログ内の**すべての設定** をクリックします。。これにより特定のチャート種類のための外観の変更ダイアログが表示されます。 (*下のスクリーンショットは円グラフの外観の変更ダイアログを表示しています*)。



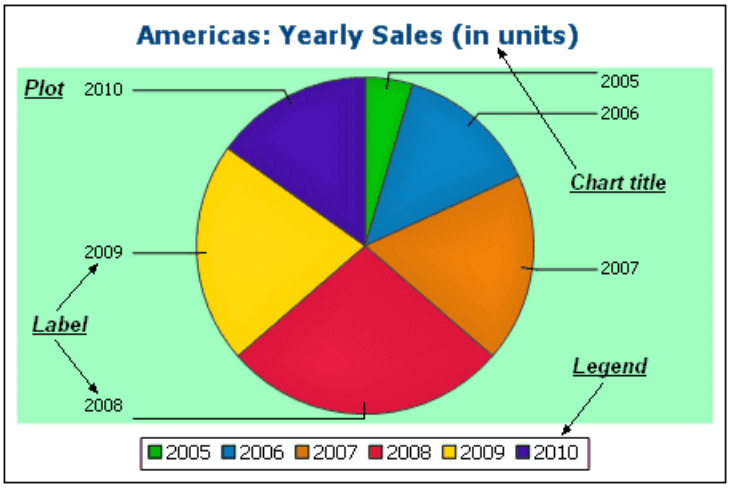
詳細設定の概要

詳細設定はすべてのチャートの種類に共通するタブとチャートの種類に特有のタブにより整理されています。

チャート設定全般

▾ *全般*

チャートのタイトル(*下のスクリーンショット参照*) は基本設定と同じですが(*上を参照*) 詳細設定でも編集することができます。このダイアログの他の設定は、チャートの背景色とプロットです。下のスクリーンショットでは、プロットには薄い緑色の背景色が与えられています。チャートおよび/またはプロットの背景のイメージとしてイメージファイルを設定することもできます。このイメージはチャートまたはプロットのエリア全体をカバーするように拡大、(チャートまたはプロットの) 2つのディメンションにズームがマッチするようにズーム、中央揃え、並べて表示することができます。凡例はチャートの配色コードのキーとなりオン/オフすることができます。



- *配色パターン*

  4つの設定済み配色パターンとユーザーにより定義される配色パターンを使用することができます。色を配色パレットに追加または削除することにより配色パターンを変更することができます。このタブにより選択される配色パターンはチャートで使用されます。

- *サイズ*

  ピクセルまたはパーセンテージ率としてチャートのさまざまな部分のサイズを設定することができます。

- *フォント*

  チャートのタイトル、凡例、ラベルのプロパティをこのタブで指定することができます。サイズはチャートのサイズのパーセンテージまたは絶対ポイントとして設定できます。

- *ロード/保存ボタン*

  設定はXML ファイルに保存され、正しい構成を持つXML ファイルからロードされることができます。構成を確認するには、チャートの設定を保存しXML 、ファイルを開きます。このボタンをクリックして、チャートの設定をディフォルトに再設定することもできます。

特定の種類のチャートの設定

- *円グラフ*

  設定: (i) 最初の扇形が描かれる角度; (ii) 扇形が描かれる方向; (iii) アウトラインの色; (iv) 色に対するハイライト有無(3D 円グラフ ドロップシャドウ 透明度の使用); (v) ラベルの有無; および(vi) 値とパーセンテージがラベルに追加されるか、およびパーセンテージへの小数点の数。

- *棒グラフ*

  設定: *(全般)* X とY 軸を交換して横向きの棒グラフを生成する *(棒)* 棒のアウトラインおよびドロップシャドウ(ドロップシャドウは2D 棒グラフのみ); *(X-軸)* X-軸の ラベル配色と垂直方向の枠線; *(Y-軸)* Y-軸のラベル、配色

と水平方向の枠線、値の範囲の表示、Y-軸の目盛; *(Z-軸、3D のみ)* Z-軸のラベルと配色; *(3D)* 垂直方向への傾斜、水平方向への回転、ビューの幅。

- *折れ線グラフ*

  設定: *(全般)* X とY 軸を交換してグラフの生成; *(ライン)* プロットポイントの有無; *(X-軸)* X-軸の ラベルと配色と垂直方向の枠線; *(Y-軸)* Y-軸のラベル、配色と水平方向の枠線、値の範囲の表示、Y-軸の目盛。

- *ゲージチャート*

  設定: (i) ゲージが開始する角度とスケールの掃引角度; (ii) 表示される値の範囲; (iii) メジャーおよびマイナーなメモリの間隔と配色; (iv) ダイアル、針、境界線の配色。

- *エリアチャート*

  エリアの透明度は0 (透明度無し) から255 (最大透明度) の値で設定することができます。この場合、スタックされていないエリアチャートでの透明度の使用は他のエリアの下にあるエリアをビューアが可視することができます。エリアのアウトラインを指定することができます。

- *ローソク足チャート*

  塗りつぶしの色は2つの場面で使用することができます: (i) 終値が始値よりも高い場合、(ii) 始値が終値よりも高い場合。後者の場合、系列の配色もオプションとして使用することができます。系列の色は、外観の変更ダイアログの配色パレットタブで指定することができます。

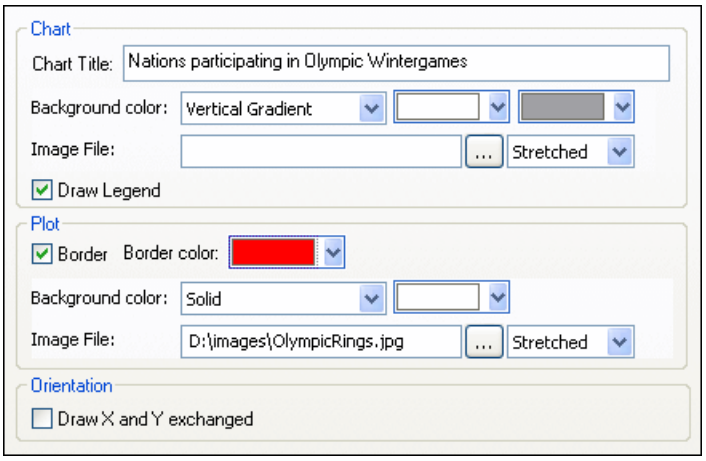
チャート設定のロード、保存、リセット

チャート設定は、XML ファイルに保存することのできるデフォルトの設定と異なります。これらの設定は、後にチャートの設定としてロードすることができ、時間と動力をセーブすることができます。**「ロード/保存」**ボタン(このセクションの最初のスクリーンショット参照) をクリックすると以下のオプションを使用することができます:

- **デフォルトとして設定**: 設定に加えられた変更を拒否し、**すべての**設定セクションのデフォルトの設定を復元します。
- **ファイルからロード**: XML ファイルに以前保存された設定をインポートすることができます (*次のコマンドを参照*)。コマンドは、必要なファイルの場所を入力することのできる **「参照」**ダイアログを表示します。
- **ファイルに保存**: **「名前を付けて保存」**ダイアログボックスを開きます。保存するXML ファイルを指定することができます。このファイルはデフォルト設定と異なる設定のファイルをリストします。

一般

「外観の変更」ダイアログボックスの一般セクションでは、チャートのタイトル、凡例の追加や削除、背景イメージ、更に棒グラフ 折れ線グラフ エリアチャート、ローソク足チャートに対してはチャートの方向を指定することができます。



チャート

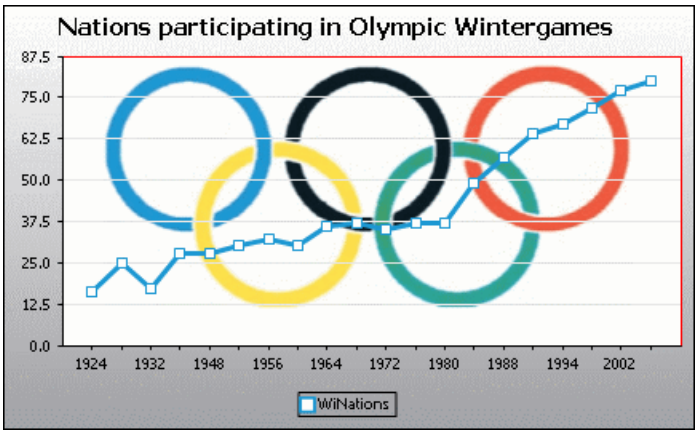
チャートタイトルフィールドにチャートを説明する様なタイトルを入力し、チャート全体に適用される背景色をドロップダウンリストから選択します。塗りつぶしや、垂直または水平グラデーション、そしてグラデーションでは2つの色を選択します。更に、背景色の代わりに背景イメージを指定し、ドロップダウンリストから表示オプションを選択することもできます：

- 伸縮：イメージがチャートの高さならびに幅に伸ばされて表示されます
- 合わせる：イメージが、アスペクト比を保ったままチャートのフレームに合うように表示されます
- 中央に表示：イメージがオリジナルのサイズで中央に表示されます
- 並べて表示：イメージがチャートより小さい場合、イメージの複製が背景エリアに表示されます

凡例を描画チェックボックスはデフォルトでオンになっており、チャート内に凡例を表示したくない場合はチェックを外してください。

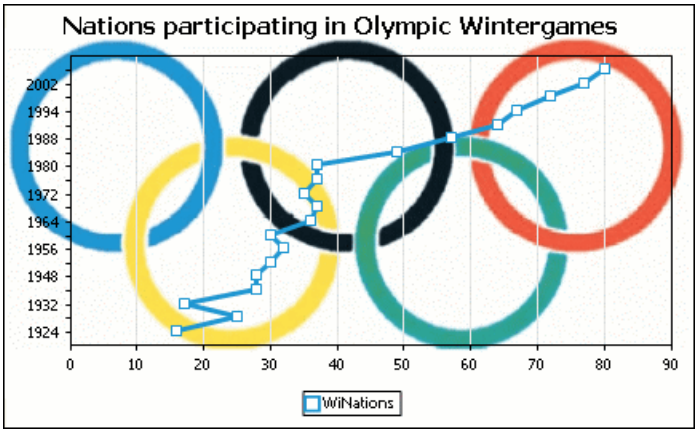
プロット

プロットとは実際のチャートが表示される領域です。プロットの周りに線をひいたり、プロット領域に対して背景色やイメージを指定することができます。以下のスクリーンショットでは、プロット領域の色は白のままでチャートの背景色がグレー（垂直グラデーション）に、境界線の色が赤に変更され、背景イメージが加えられています。



方向

少数の系列と、多くの値がデータに存在する場合、X-軸とY-軸を入れ替えた方が、より見やすい表示となることもあります。以下のスクリーンショットでは、プロットの背景色が「透明」にセットされ、イメージがチャートの背景に適応されていることにも注目してください。



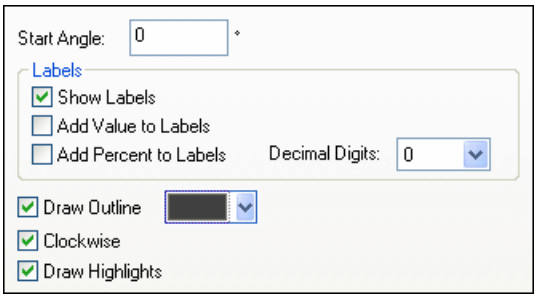
このオプションは円グラフならびにゲージチャートにお適用できないことに注意してください。

各チャートの機能

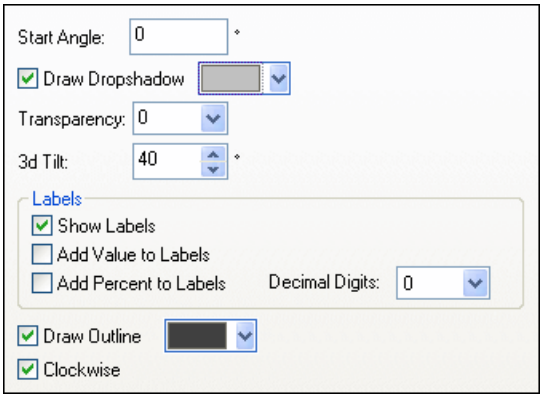
「外観の変更」ダイアログボックスでは、チャートの種類に関連した機能の定義を行うことができます。

円グラフ

2D ならびに3D バージョンで行うことのできる設定にそこまで違いはありません。2D の円グラフではハイライトを付与することができます。

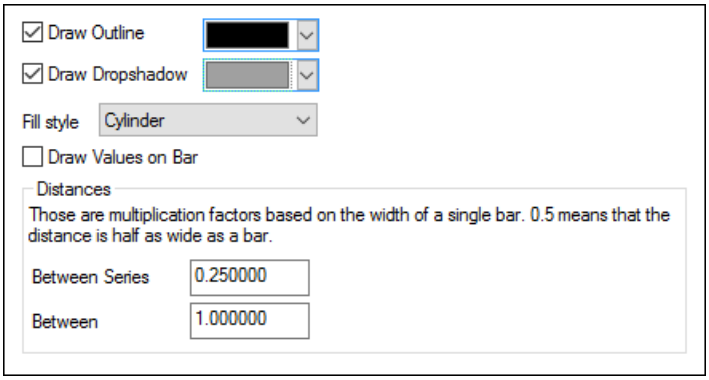


3D の円グラフでは、影の描画や透明度、3D の傾きを指定することができます。



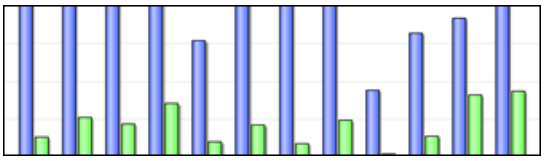
[開始角度] の値により、選択されたカラム内にある最初の行が表示されるチャートの位置を指定します。0° の角度が時計の12時方向に対応します。
凡例やラベルを表示したり、値やパーセントをラベルに付与したり、パーセントに使用される桁数を指定することができます。
[輪郭を描画] チェックボックスの隣から選択できる色により、チャートや個々の円グラフセグメントの周りに描かれる境界線の色を指定することができます。時計回りのチェックボックスにより、行の表示を時計回りの、または反時計回りに行うかを指定できます。
3D の円グラフでは、影の描画やその色、チャートの透明度、そして3D の傾きを指定することができます。2D のチャートでは、ハイライトを描画オプションにより、更なる側面をチャートに付与することができます。

棒グラフ

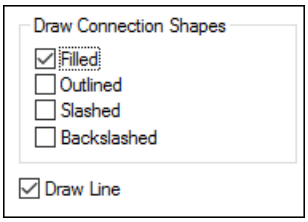


棒グラフのために以下を設定することができます:

- 棒に輪郭を追加し、色を定義することができます
- 2D の棒グラフでは、影を付けることができ、色を定義することもできます (3D 棒グラフではこのオプションを使用することはできません)。
- デフォルトで、棒の形式は、円柱ですが、「グラデーション(縦方向)」または「塗りつぶし」をドロップダウンリストから選択することができます。
- (Y-軸上のバーの高さに対応する) 棒の値を棒上に表示することができます。
- 棒グループの系列と棒グループの距離は、単一の棒の幅の小数として指定することができます。例えば、下のスクリーンショットでは、青い系列と緑の系列から構成される棒グループが表示されており、系列間の距離は、棒の幅の25% (`=0.25`) に設定されています。棒グループの距離は棒の幅の100% (`=1.0`) に設定されています。



折れ線グラフ

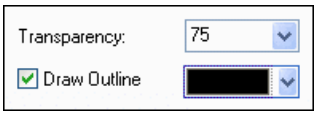


チャート内にある値を結びつける接続点を描画するには、接続点の描画グループボックスにあるチェックボックスのうち少なくとも1つにチェックを入れる必要があります。系列をマークするために、最大5つの形が使用されます :正方形、ひし形、三角形、逆三角形、円。6つ以上の系列がチャート内に存在する場合、接続点のオプションを2つ以上選択することで、接続点の形を組み合わせて使用することができます。以下のスクリーンショットでは、**塗りつぶし**と**スラッシュ**の2つが選択されており、6つ目以降の系列にはスラッシュが使用されます。

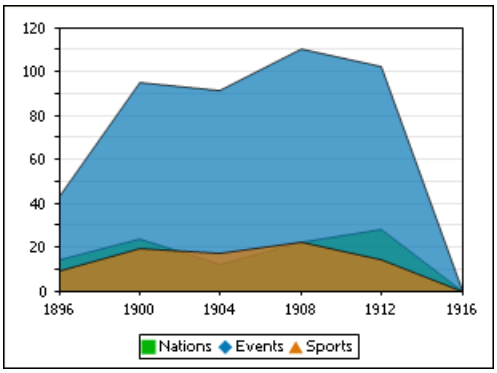
線を描くオプション (i) 接続点のみ (ii) 接続点と線により線をで使用して描くことができます。

接続点は棒グラフと値付き棒グラフの両方で使用することができます。

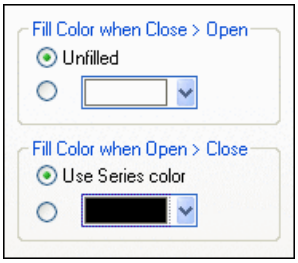
エリアチャート



エリアチャートでは透明度を指定することができ、1つの系列がチャートにある他の系列により隠れてしまうのを防ぐことができます。更に、データエリアに対して輪郭を描画、ならびに色を指定することができます。(*下のスクリーンショット参照*)。



ローソク足チャート



始値と終値の値が系列として既に定義されている場合、使用される色と、終値が始値よりも大きい場合、ローソクの塗りつぶしを行うかを指定することができます。

ゲージチャート

角グループボックスにある「開始」値により0 のある位置と、「可動角度」によりチャートの表示に使用される角度が指定されます。値の範囲グループボックスでは、表示する最小ならびに最大値を指定することができます。大きな目盛り間隔では目盛りが値とともに、小さな目盛り間隔では価なしで表示され、それぞれに異なる色を指定することができます。色のグループボックスでは、ダイアルや針、チャートを囲む境界線や針のベース（中央部にある部分の針）の色を指定することができます。現在の値と追加ラベルを希望する角度で表示することができます。

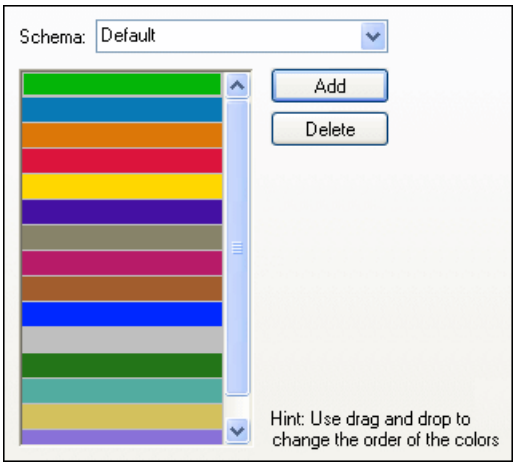
色

選択されたチャートの種類によりMobileTogether Designer ではチャート内で使用される色の指定を行う2つのセクションが用意されます：

- 円グラフ、折れ線グラフ、エリアチャート、ローソク足チャートに対して使用される塗りつぶし
- ゲージチャートに対して使用される色の範囲

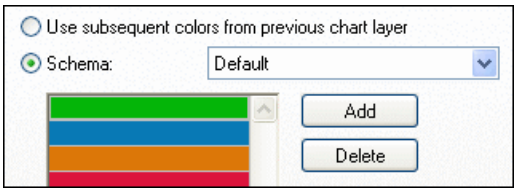
色スキーマ

「外観の変更」ダイアログボックスにある塗りつぶしセクションでは、（デフォルト、グレイスケール、カラー、パステルから成る）4つの定義済み塗りつぶしセットが用意されており、カスタマイズすることができます。更に、独自の色を最初から作成することもできます。



最初の色が最初の系列に対して使用され、2 つ目の色が2 つ目の形式に対して使用されます。色を選択してマウスのドラッグ操作により別の位置に移動することで色の順番を変更することができます。新たな色を追加したり既にある色を削除するには、それぞれ対応するボタンをクリックしてください。ローソク足チャートでは、最初の色だけが使用されます。

1 つ以上のオーバーレイチャートレイヤーをチャートウィンドウに追加した場合、外観の変更ダイアログボックスに、デフォルトで選択されている**[前のチャートレイヤーから継続して色を使用する]** ラジオボタンが表示されます。



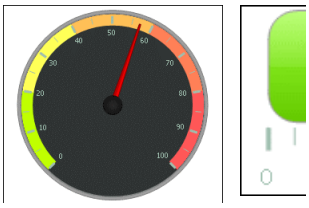
このラジオボタンが選択されていると、前のレイヤーから色が継承され、そのオーバーレイに対して独自の色を選択できなくなります。アクティブなレイヤーにおける系列は、その前のレイヤーで使用された色の集合が継続して使用されます。これにより、チャートウィンドウ内に表示される系列全てに異なる色を持たせ、より簡単に系列が識別できるようになります。

このオプションは任意のレイヤーで無効にすることができ、異なる色の集合を指定して、更に続くレイヤーにてそれらの色を継承することもできます。

色の範囲

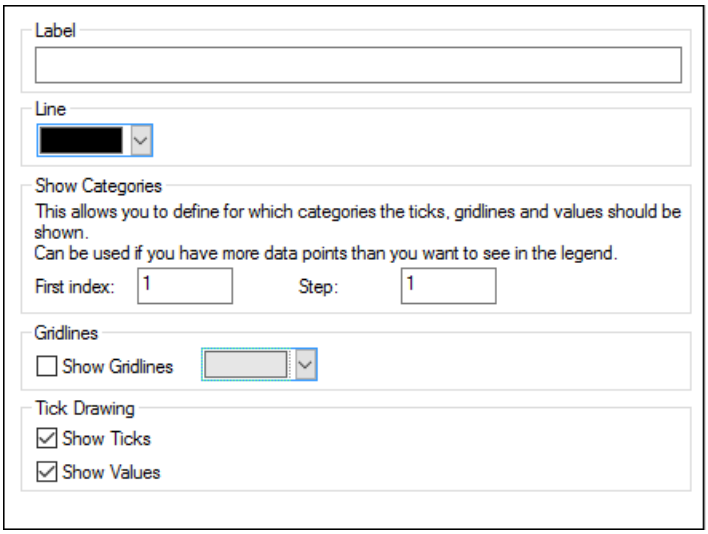
ゲージチャートでは、値の範囲に対して色を指定することで、ゲージの外観をカスタマイズすることができます。

上のスクリーンショットで定義された設定により、ゲージチャートは以下のような外観になります：

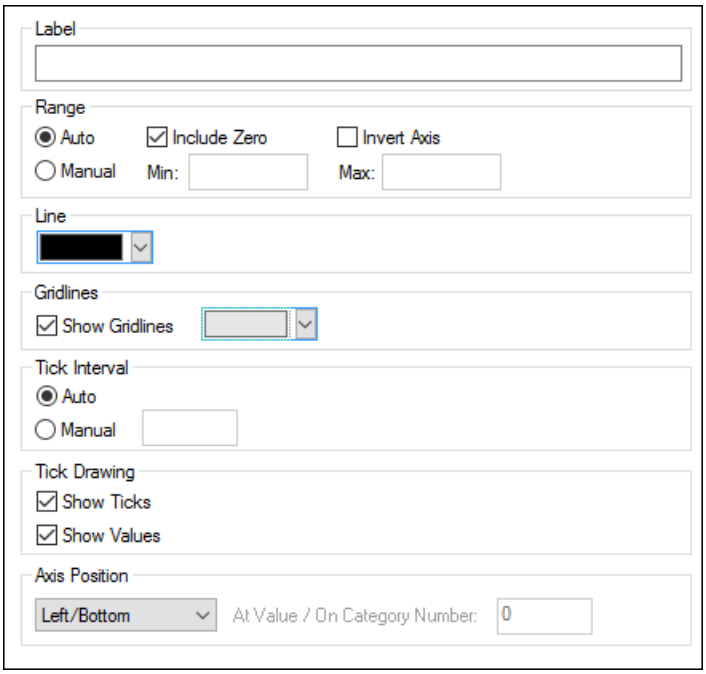


X- 軸

外観の変更」ダイアログボックスのX-軸セクションでは、軸のラベルの入力や、軸ならびにグリッド（表示されている場合）の色を定義、目盛りや軸の値を表示するか指定することができます。このセクションにおける記述は全ての棒グラフ、折れ線グラフ、エリアチャートローソク足チャートに対して共通するものです。カテゴリの表示オプションにより、全てのカテゴリ(X-Axis 値) のサブセットが表示されるように指定することができます。すなわち、メモリ、グリッド線、選択されたカテゴリの値が表示されます。表示されたカテゴリのサブセットを以下を入力して作成することができます (i) 表示する最初の値のインデックス、および、(ii) ステップの数量 例えば、`1900`、`1901`、`1902 ... 1999`、`2000`、から開始する、101 件のカテゴリが存在する場合、最初のインデックスを1 に、ステップを10 に設定することにより、1900 年から2000 年の間の10 年後との期間を表示することができます。



折れ線グラフの場合、値の範囲ならびに表示される目盛りの間隔を定義することもできます。



ラベル
ラベルフィールド内に入力されたテキストは、X-軸の説明文として軸の下に表示されます。

範囲
デフォルトでは範囲グループボックスの自動ラジオボタンが選択されています。チャートの一部分における詳細を表示したい場合、「手動」ラジオボタンを選択して、**最小**ならびに**最大値**を対応するフィールドに入力します。
X-軸のために使用される列に0 が含まれない場合、「ゼロを含む」チェックボックスを無効化することができます。X-軸は系列内で使用することのできる最小の値と共に開始されます。「軸の反転」オプションによりY-値の値を反転することができます。例えば、**0** から**360** の間の値が実行される場合、このオプションの選択は、Y-軸を生成します。これにより、**360** の値が元の値となり、Y-軸の値が上がるごとに、0に近づきます。

ライン
チャート内に表示される軸の色は、**「ライン」**ドロップダウンリストから指定することができます。既に表示されている色を選択するか、**「他の色」**ボタンをクリックして標準的な色を選択するか、カスタマイズされた色を作成することができます。カスタムタブにある**「選択」**ボタンをクリックすると、スポイトが表示され、画面上に表示されている色を選択することができます。

グリット線
グリット線を表示チェックボックスが有効になっている場合、隣にあるドロップダウンボックスから色を選択することができます。

目盛り間隔
デフォルトで表示される目盛り間隔で都合が悪い場合は、目盛り間隔グループボックスにある手動ラジオボタンを有効にして、目盛り間の値をフィールドに入力してください。

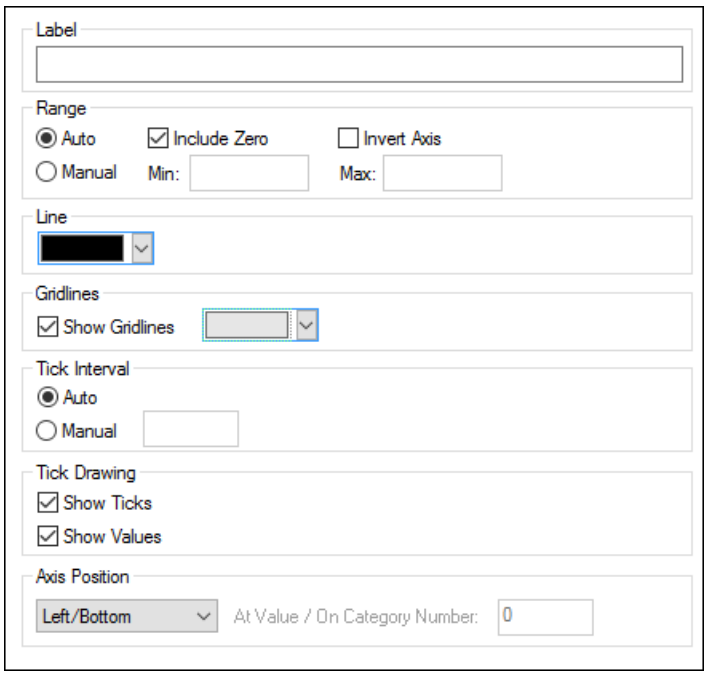
目盛り描画
軸にある目盛りや、軸上にある値の表示を切り替えることができます。

軸の位置
軸が表示される位置をドロップダウンリストから選択することができます。「値 /カテゴリー番号」を選択すると、プロット内の自

由な位置に軸を配置することができます。

Y- 軸

「外観の変更」ダイアログボックスのY-軸セクションでは、軸のラベルの入力、軸の色や（表示される場合は）グリッド線の定義、値の範囲指定、目盛りの表示やその間隔、そして軸の値を表示するかなどを設定することができます。このセクションにおける記述は全ての棒グラフならびに折れ線グラフに対して共通するものです。



ラベル
ラベルフィールドに入力された値は、Y-軸の説明として軸の左側に表示されます。

範囲
デフォルトでは「範囲」グループボックスの「自動」ラジオボタンが選択されています。チャートの一部分における詳細を表示したい場合、「手動」ラジオボタンを選択して、**最小**ならびに**最大値**を対応するフィールドに入力します。
X-軸のために使用される列に0 が含まれない場合、「ゼロを含む」チェックボックスを無効化することができます。X-軸は系列内で使用することのできる最小の値と共に開始されます。「軸の反転」オプションによりY-値の値を反転することができます。例えば、`0` から`360` の間の値から実行される場合、このオプションの選択は、Y-軸を生成します。これにより、`360` の値が元の値となり、Y-軸の値が上がるごとに、0に近づきます。

ライン
チャート内に表示される軸の色は、**「ライン」**ドロップダウンリストから指定することができます。既に表示されている色を選択するか、**「他の色」**ボタンをクリックして標準的な色を選択するか、カスタマイズされた色を作成することができます。カスタムタブにある**「選択」**ボタンをクリックすると、スポイトが表示され、画面上に表示されている色を選択することができます。

グリッド線
グリッド線を表示チェックボックスが有効になっている場合、隣にあるドロップダウンボックスから色を選択することができます。

目盛り間隔
デフォルトで表示される目盛り間隔で都合が悪い場合は、目盛り間隔グループボックスにある手動ラジオボタンを有効にし

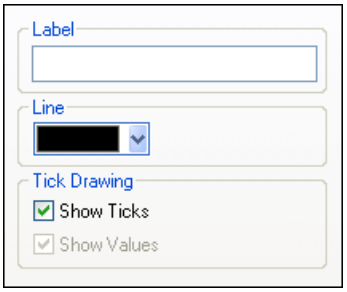
て、目盛り間の値をフィールドに入力してください。

目盛り描画
軸にある目盛りや、軸上にある値の表示を切り替えることができます。

軸の位置
軸が表示される位置をドロップダウンリストから選択することができます。「値 /カテゴリー番号」を選択すると、プロット内の自由な位置に軸を配置することができます。

Z- 軸

「外観の変更」ダイアログボックスのZ-軸セクションでは、軸のラベルの入力や、軸ならびにその色を定義、軸上の目盛りを表示するか指定することができます。このセクションにおける記述は全ての3D 棒グラフ（3D 棒グラフならびにグループ化された3D 棒グラフ）対して共通するものです。



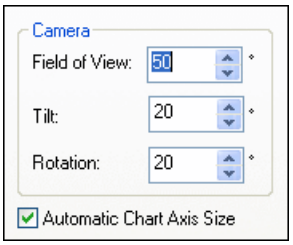
ラベル
ラベルフィールドに入力された値は、Z-軸の説明として軸の右側に表示されます。

ライン
チャート内に表示される軸の色は、**「ライン」**ドロップダウンリストから指定することができます。既に表示されている色を選択するか、**「他の色」**ボタンをクリックして標準的な色を選択するか、カスタマイズされた色を作成することができます。カスタムタブにある**「選択」**ボタンをクリックすると、スポイトが表示され、画面上に表示されている色を選択することができます。

メモリの描画
目盛りの表示を切り替えることができます。

3D アングル

「外観の変更」ダイアログボックスの3D アングルセクションでは、チャートの3D 的な外観を3D 棒グラフに対して設定することができます。



「傾き」の値によりX-軸に対する角度が決定され、「回転」の値によりY-軸に対する角度が決定されます。対応する

チェックボックスにチェックを入れることで、チャート軸の大きさをチャートのウィンドウに自動的に合わせることができます。

「自動チャート軸サイズ」チェックボックスが有効になると、現在表示されているチャートの大きさに対して最適なX-軸ならびにY-軸の大きさがMobileTogether Designer により計算されます。チャートの幅と高さは、チャートウィンドウの大きさを変更することで動的に変更されます。

サイズ

「外観の変更」ダイアログボックスの大きさセクションでは、様々な余白や軸、またはゲージ目盛りの間隔を定義することができます。以下に示す全てのプロパティが全てのチャートに対して使用できるわけでは無いことに注意してください。

一般

外部の余白	プロットとチャートウィンドウ端の空間
タイトルからプロット	チャートタイトルとプロット上部の端との空間
凡例からプロット	プロット下部の端と凡例との空間

円グラフ

プロットからラベル	円グラフにて、円グラフの左端ならびに右端とラベル間の空間
グラフの高さ	3D 円グラフの高さ
グラフの影	3D 円グラフにて（有効化されていた場合）表示される影の長さ

X-軸

X-軸から軸ラベル	棒グラフと折れ線グラフにて表示されるX-軸とラベル間の空間
X-軸からプロット	2D 棒グラフと折れ線グラフにて表示されるX-軸とプロット間の空間
X-軸目盛りサイズ	棒グラフと折れ線グラフにて表示されるX-軸上の目盛り長さ

Y-軸

Y-軸から軸ラベル	棒グラフと折れ線グラフにて表示されるY-軸とラベル間の空間
Y-軸からプロット	2D 棒グラフと折れ線グラフにて表示されるY-軸とプロット間の空間
Y-軸目盛りサイズ	棒グラフと折れ線グラフにて表示されるY-軸上の目盛り長さ

Z-軸

Z-軸から軸ラベル	3D 棒グラフにて表示されるZ-軸とラベル間の空間
Z-軸目盛りサイズ	3D 棒グラフにて表示されるZ-軸上の目盛り長さ

ライン描画

接続点のサイズ	折れ線グラフにてチャート内の値をマークする接続点の大きさ

3D 軸サイズ

ベースの手動X-軸サイズ	3D 棒グラフにおいてX-軸の長さとチャートウィンドウの大きさの関係を定義します。3D アングルセクションにある**自動チャート軸サイズ**のチェックを外さなければサイズが自動的に計算されることに注意してください
ベースの手動Y-軸サイズ	3D 棒グラフにおいてY-軸の長さとチャートウィンドウの大きさの関係を定義します。3D アングルセクションにある**自動チャート軸サイズ**のチェックを外さなければサイズが自動的に計算されることに注意してください
Z-軸系列余白	3D 棒グラフのZ-軸における各系列間の距離

ゲージ

境界線の幅	円ゲージチャートのゲージ周りに表示される境界線の幅

ゲージ間隔

領域線から目盛りまでの距離	円ゲージチャートにおいて、領域内側の端と値を表示する目盛りの空間
大きな目盛りの長さ	円ゲージチャートに表示される（値を持った）大きな目盛りの長さ
大きな目盛りの幅	円ゲージチャートに表示される（値を持った）大きな目盛りの幅
小さな目盛りの長さ	円ゲージチャートに表示される値を持たない目盛りの長さ
小さな目盛りの幅	円ゲージチャートに表示される値を持たない目盛りの幅

ゲージ針

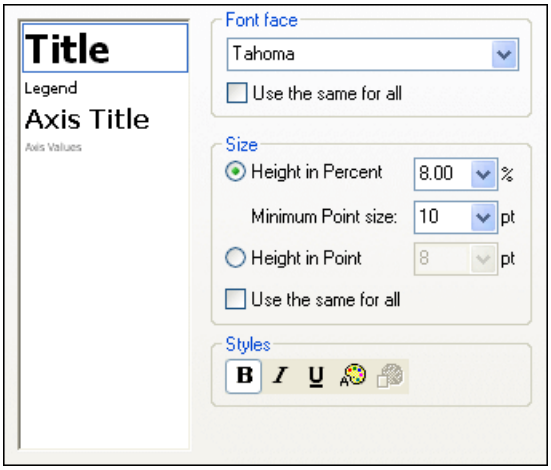
針の長さ	円ゲージチャートにおける針の長さ（パーセントはゲージの直径から計算されるため、50% を超えた値を選択すると、針の先端がゲージの外側に出てしまうことに注意してください！）
ベースにおける針の幅	円ゲージチャートの中心における針の幅（太さ）
針ベースの半径	円ゲージチャート中央部に表示されるベースの半径

ゲージ色の範囲

境界線から色の範囲までの距離	円ゲージチャートにおける境界線内側にある端と、色の範囲との間にある空間
色の範囲幅	円ゲージチャートにおけるカスタマイズ可能な色が表示される範囲の幅（パーセントはゲージの直径から計算されることに注意してください！）

フォント

「外観の変更」ダイアログボックスのフォントセクションではチャートウィンドウ内のオブジェクトに使用されるフォント設定を行うことができます。



フォント設定

フォントセクションではチャートウィンドウ内に表示される各要素のフォントフェイス、大きさ、そしてスタイルを変更することができます。大きさはチャートサイズに対するパーセントとして定義することができるとともに、最低サイズをポイントで指定、または絶対サイズをポイントで指定することができます。同一のフォントや大きさを全てのテキスト要素に適用したい場合、**[全てに適用]** チェックボックスにチェックを入れます。
リストボックス内にある要素は以下のように定義されます：

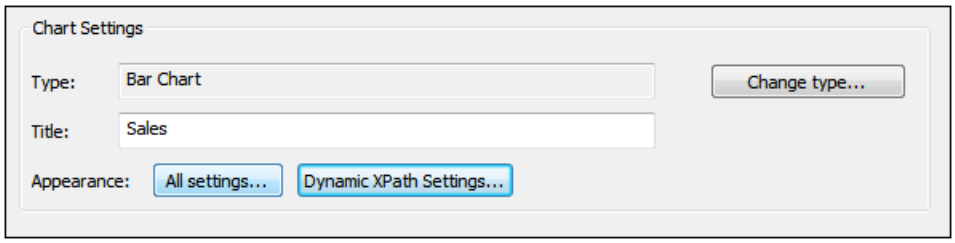
- **タイトル** :チャートの名前
- **凡例** :チャート内で使用されている色の説明
- **ラベル** :円グラフにおける各パイの説明
- **軸タイトル** :棒グラフおよび折れ線グラフにおけるX- 、Y- 、Z- 軸の名前
- **軸の値** :棒グラフおよび折れ線グラフの軸上に表示される単位
- **目盛りの値** :ゲージチャートにて表示される単位

(C) 2018 Altova GmbH

動的な XPath 設定

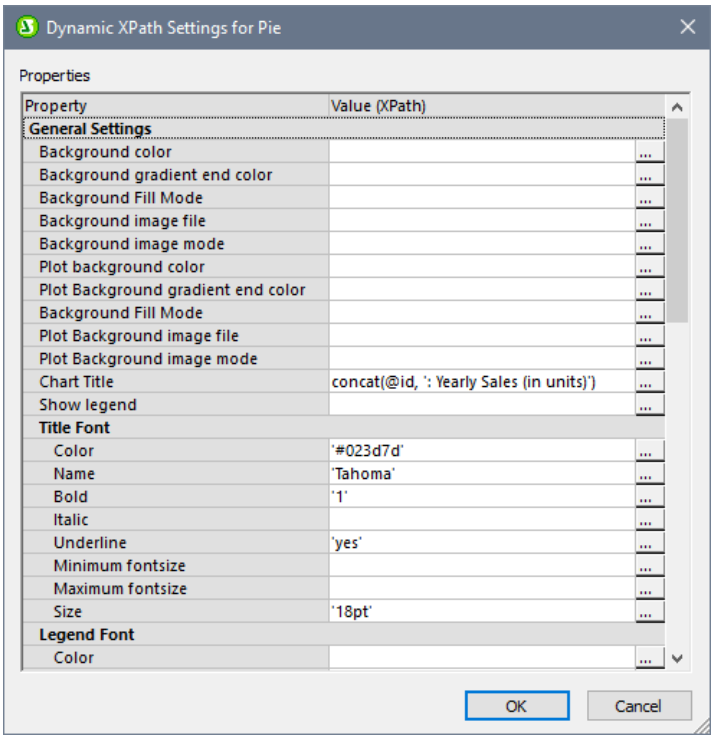
動的なXPath 設定はチャート設定でXML ドキュメントから動的データを使用する際にとても役に立ちます。例えば、Region 要素のチャートがRegion 要素のデータ(例えば名前) がチャートのタイトル内で必要とします。複数のRegion 要素がある場合、そのうちの1 つがチャートに使用されますが、チャートのタイトルのためのデータはXPath 式を使用してのみ取得することができます。動的なXPath 設定ダイアログによりXPath を使用してデータにアクセスすることができます。

動的なXPath 設定ダイアログにアクセスするには、チャート構成ダイアログのチャート設定ペイン内の **動的なXPath 設定**」をクリックします。(*下のスクリーンショット参照*)。



動的な XPath 設定

動的なXPath 設定ダイアログ(*下のスクリーンショット参照*)は、チャート構成ダイアログのチャート設定ペインの **動的なXPath 設定**」ボタンをクリックすることによりアクセスすることができます。このダイアログで複数のチャート設定を入力することができます。



以下の点に注意してください:

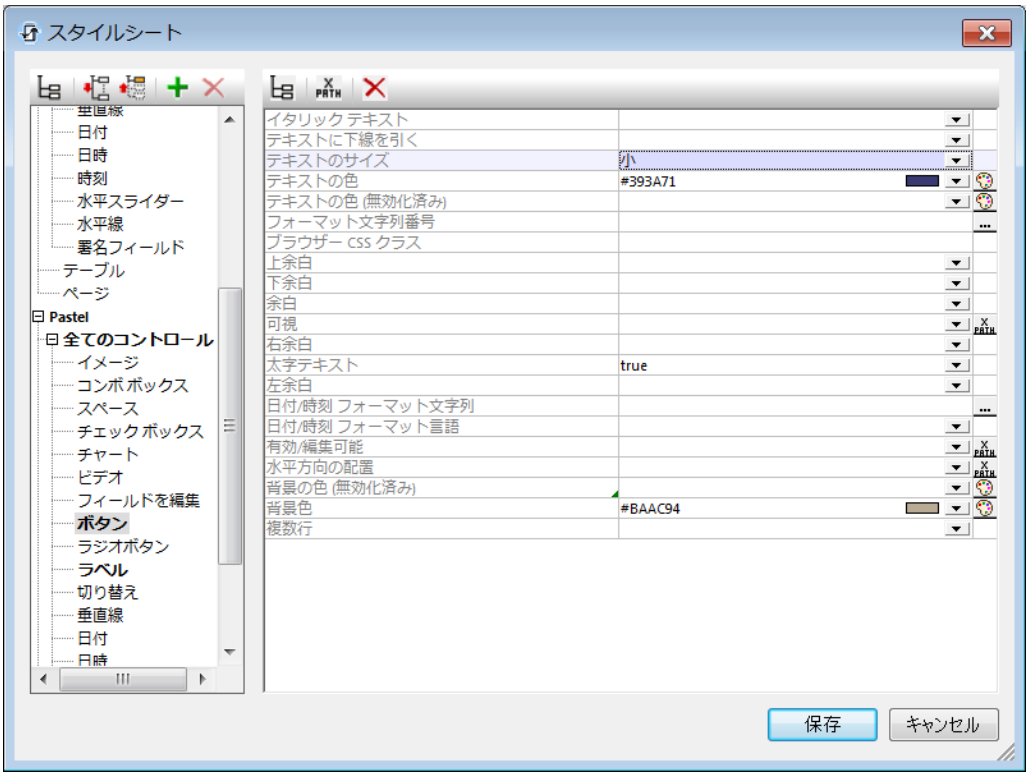
- このダイアログの全てのエントリはXPath 式として評価されます。ですから、文字列リテラルは引用符で囲む必要が

あります。例: 'Tahoma'、'1'、'18pt'、および'#FF3366'。

- 設定または値にポイントすると、列挙とフォーマットの情報を表示するヒントが表示されます。
- 動的なXPath 設定には、チャート構成ダイアログまたは外観の変更ダイアログで作成された設定に対する優先順位があります。例えば、動的なXPath 設定として作成されたチャートのタイトルはチャート構成ダイアログの設定に対する優先順位を有します。
- 使用する配色パターンとして、外観の変更ダイアログ内でユーザーにより定義された配色パターンが選択されると配色パターンの色が使用されます。配色はRGB 16 進数フォーマットとして設定されています: #RRGGBB。ですからXPath 式により指定される赤色は以下の通り指定されます: '#FF0000'。

11.7 スタイルシート

MobileTogether Designer のスタイルシート機能により、プロジェクト、ページ、テーブル、とコントロールレベルでグローバルなスタイルを定義することができます。スタイルシートは **「プロジェクト | スタイルシート」** コマンドにより開かれるスタイルシートダイアログで作成され、定義されます（下のスクリーンショット）。ユーザーにより定義される複数スタイルシートを作成することができます。これらのスタイルシートは、デザインの多種のコンポーネントに適用することができます。



ユーザーにより作成されたスタイルシートの追加と削除

2つの種類のスタイルシートの型は以下のとおりです: (i) プロジェクトレベルで**自動的**に適用され、削除することのできないプロジェクトのスタイルシート (ii) ユーザーにより作成されたスタイルシートそれぞれのページ、テーブル、コントロールに個別に適用することができます。

ユーザーにより作成されたスタイルシートを追加するには、 **「スタイルシートの追加」** をクリックします。ユーザーにより作成されたスタイルシートの名前を変更するには、名前をダブルクリックして編集します。

ユーザーにより作成されたスタイルシートを削除するには、 **「スタイルシートの削除」** をクリックします。

スタイルの定義

左側のペイン内のスタイルシートから、スタイル（ページ、テーブル、または、コントロール）を定義するレベルを選択し、右側のペイン内でそのスタイルプロパティに値を割り当てます。静的なプロパティの値を選択、または、入力することができます。また、プロパティの値を評価するXPath 式を入力することもできます。動的な割り当ての例はエンドユーザーのモバイルデバイスのスクリーンの幅など、プロパティの値を条件付きにするなどです。

スタイルの定義の優先順位

コンポーネントにスタイルの定義の場所が近いほど、定義の優先順位は、同じプロパティのための遠い場所にある定義よりも高くなります (次を参照: *スタイルシート全体の優先順位*)。例えば、ユーザーにより作成されたスタイルシートが ボタンインスタンス に割り当てられたとします。ユーザーが作成したスタイルシート内のスタイルは、プロジェクトのスタイルシート内のスタイルよりも、(ボタンのスタイル プロパティに関しては、)高い優先順位を与えられます。このようにして、カスケードスタイルを持つデザインコンポーネントにスタイルを与えることができます。更に、 スタイルシート内の優先順位のレベル 自身がカスケード効果を定義するための柔軟性を与えます。

ユーザーにより作成されたスタイルシート は、デザインコンポーネントにスタイルシート名をコンポーネントの スタイルシート プロパティの値として入力するだけで、デザインに適用することができます。スタイルシートの割り当ては、(名前を直接入力することにより) 静的におこなうことができ、(XPath 式を使用することにより)動的におこなうことができます。XPath 式を使用することにより、を動的なコンテキストに従いユーザーにより作成されたスタイルシートを選択することができます。例えば、エンドユーザーが現在使用しているモバイルデバイスの種類によりスタイルシートの選択をおこなうことも可能です。

プロジェクト CSS ファイルについて

Web クライアント、すなわちブラウザーのためのスタイルに関するMobileTogether Designer 機能がありますが、スタイルシート機能とは個別の問題です。これは、プロジェクトの ブラウザー設定を使用してプロジェクトにCSS ファイルを割り当てるプロジェクトCSS ファイル機能を指します。プロジェクトCSS ファイル内で、各コンポーネントの ブラウザー CSS クラス プロパティを使用して、デザインコンポーネントに割り当てられているクラスのためのスタイルを定義することができます。現在のセクション内では、プロジェクトCSS ファイルは**作業しません**。この機能に関する詳細は、プロジェクトの ブラウザー設定を参照してください。

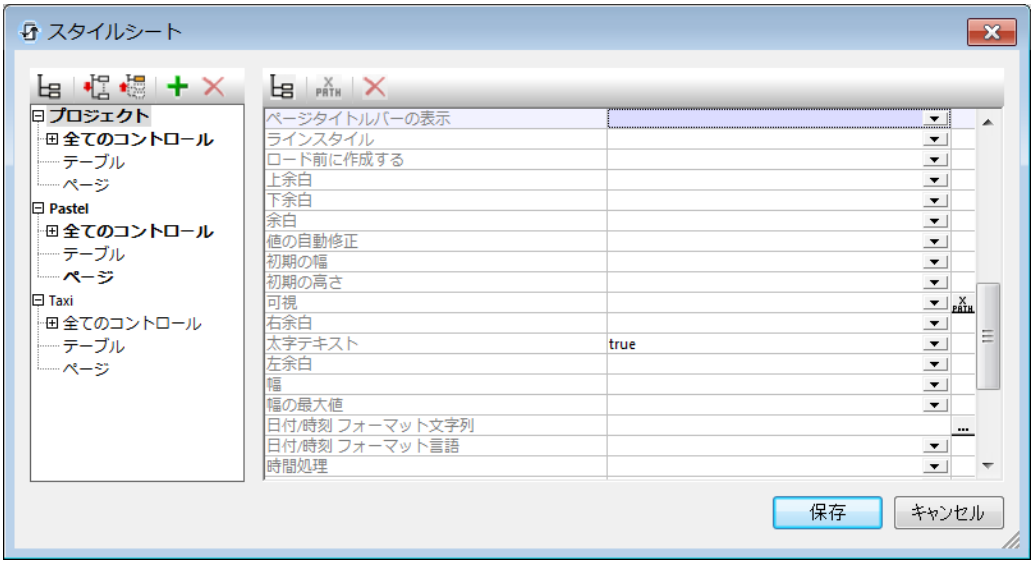
このセクション

このセクションは以下のように整理されています:

- スタイルシートの型とスコープ は、2つのスタイルシートの型とそれぞれのスコープについて説明しています。
- スタイルシート内の優先順位 は、単一のスタイルシート内で使用することのできる優先順位のレベルについて説明しています。
- スタイルシート全体の優先順位 は、複数のスタイルシートが使用されている場合、スタイルの優先順位の決定方法について説明しています。
- ユーザーにより作成されたスタイルシートの適用 は、ユーザーにより作成されたスタイルシートがどのようにデザインコンポーネントに適用されるかを説明しています。
- スタイルシート プロパティ は、スタイルシートダイアログ内でどのようにコンポーネントスタイルと作業するかの概要について説明されています。

スタイルシートの型とスコープ

スコープに従い、スタイルシートは次の2つの型にグループ化することができます:



- *プロジェクト*の名前を持つプロジェクトのスタイルシート。この名前は固定されていて変更することはできません。*プロジェクト*スタイルシート内で定義されているスタイルは、**プロジェクト全体**に**自動的に適用**されます(すなわち、プロジェクトの全てのページに対して適用されます)。*プロジェクト*スタイルは、対応する箇所で高い優先順位を定義されているスタイルによりオーバーライドされます。
- ユーザーにより与えられた名前を持つ、ユーザーが作成したスタイルシート。作成することのできるスタイルシートの数量に制限はありません。(上のスタイルシートでは、2つのユーザーが作成したスタイルシートが存在します: 「パステル」と「タクシー」)。ユーザーが作成したスタイルシート内で定義されたスタイルは、個々のコントロール、テーブル、ページに適用することができます。これは、ユーザーが作成したスタイルシートの名前を、対応するコントロール、テーブルまたはページのスタイルシートのプロパティの値として入力することにより行います。ユーザーが作成したスタイルシートがこのように適用されると、*プロジェクト*スタイルシートより高い優先順位を有するようになります。

  メモ ユーザーが作成したスタイルシートを追加するには、**「スタイルシートの追加」** をクリックします。ユーザーが作成したスタイルシートの名前を変更する場合は、名前をダブルクリックして編集します。

メモ スタイルグループ(コントロール、テーブル、またはページ)が、1つ以上のスタイルの定義を含む場合、そのスタイルのグループと、そのグループに含まれているスタイルシートは、太字で表示され、それ以外の場合は、通常のフォントで表示されます。例えば、上のスクリーンショットでは、「*taxi*」スタイルシート内では、スタイルが定義されていませんが、他の2つのスタイルシートでは少なくとも1つ以上のスタイルが定義されています。「*Pastel*」スタイルシートでは、1つのスタイルがスタイルのページグループ内で定義されています。

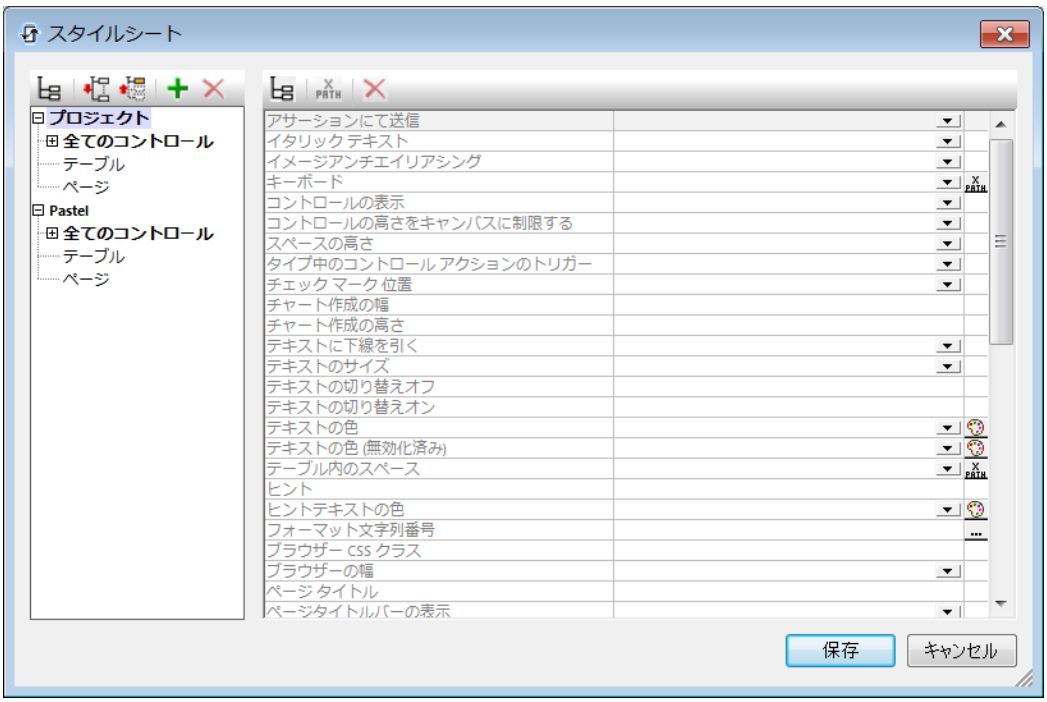
メモ ダイアログ(左から右)の各ペインには、空でない内アイテムの表示を有効化/無効化するアイコンが存在します。空ではないアイテムの表示は、定義済みのスタイルのリストを確認する場合荷役にたちます。例えば、現在定義されているスタイルの概要を確認する場合など。左側のペインには、次を行うためのツールバーアイコンが存在します:(i) 全てのアイテムの展開、および、(ii) 全てのアイテムの折りたたみ。

スタイルシート内の優先順位

*プロジェクト スタイルシート*とユーザーが作成した*スタイルシート*は3段階のレベルにより構成されています:

```
スタイルシート (レベル-1)
|
|-- 全てのコントロール(レベル-2)
|    |
|    |-- ControlType-1 (レベル-3)
|    |    ...
|    |-- ControlType-n (レベル-3)
|
|-- テーブル (レベル-2)
|
|-- ページ (レベル-2)
```

下の*スクリーンショット*内でこの階層構造を確認することができます。

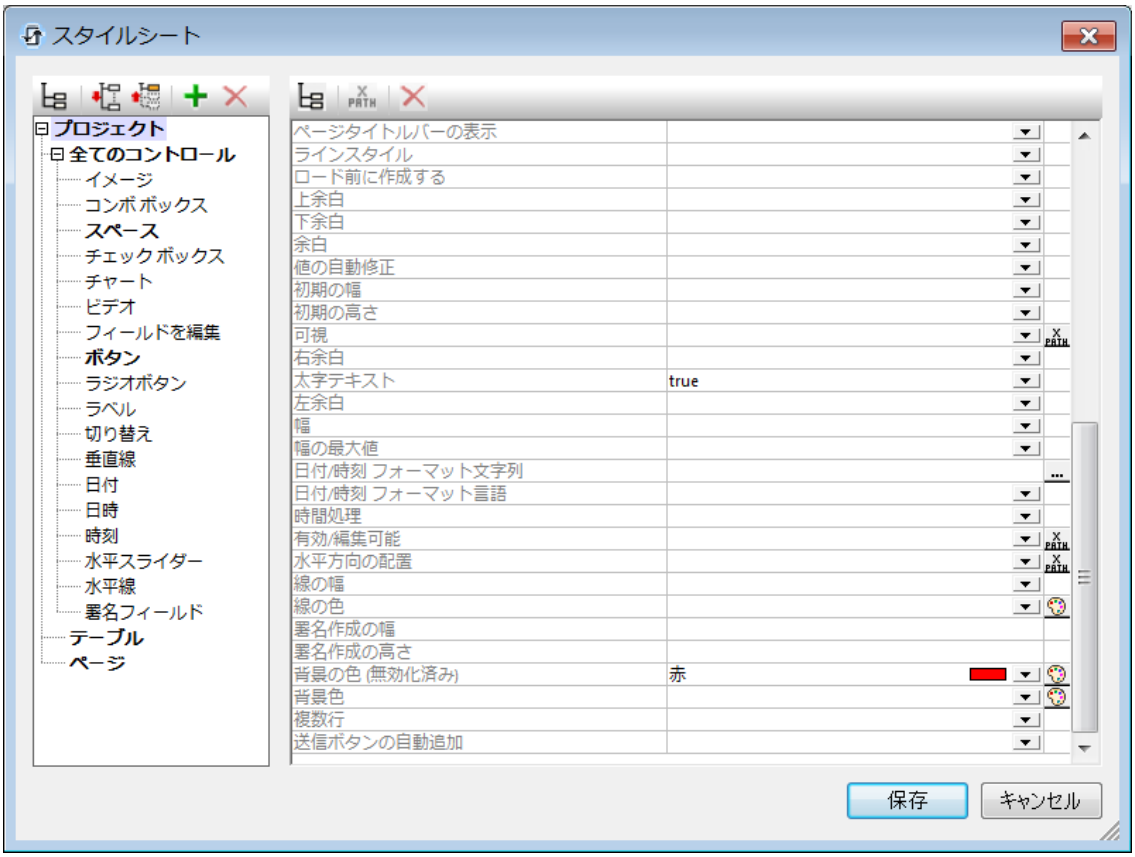


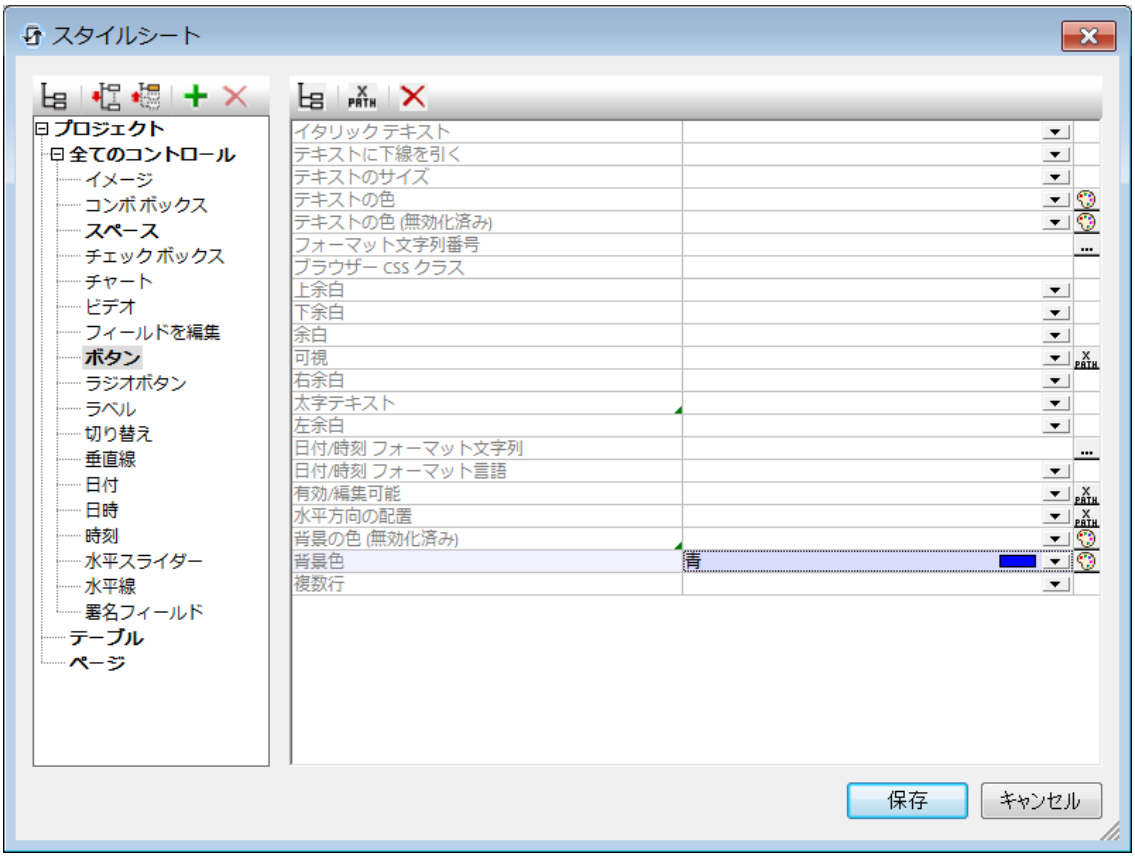
階層構造内の高いレベルは*スタイル*プロパティを下のレベルにパスします。ですから、全てのコントロールのレベルは、自身のスタイルの定義をレベル3のコントロールの型にパスします。レベル(低いレベル) にある全てのコントロールは、全てのコントロールレベルで設定されたプロパティの値を継承します。

*スタイルシート*レベルで使用することのできるプロパティは、全てのコントロールレベルの全てのプロパティ、および、(それぞれテーブルとページのためのプロパティを定義する)テーブルとページプロパティです。ですから、例えば、*スタイルシート*レベルで、**背景色** プロパティを設定すると(右下の*スクリーンショット*に表示されるように、赤の値を設定すると)、レベル3の全てのコントロールの型とデザイン内のテーブル、およびデザインページが、これらの子孫プロパティのために定義が存在しない場合、(この場合は赤)値を継承します。

高いレベルで割り当てられているプロパティの値をオーバーライドするには、オーバーライドする値を低いレベルに割り当てます。例えば、ボタン コントロール型が、青の**背景色** プロパティの値に割り当てられたとします。(左側のスクリーンショット内の高いレベルで割り当てられている)**背景色** プロパティ(とテーブルとページ)を持つ全てのコントロールの型は、赤い背景色を継承し、全てのボタンコントロールは青い背景色を継承します。特定のボタンインスタンスに、青以外の背景色を割り当てる場合、その特定のボタンインスタンスの**背景色** プロパティを指定します(デザイン内のボタンコントロールを選択し、

スタイルとプロパティペイン内で**背景色** プロパティの値を設定して行います)。





デザインコンポーネントに近く存在する定義のための優先順位

スタイルプロパティが複数のレベルで存在し、定義が比較的デザインコンポーネントに近いプロパティが比較的高い優先順位を有します。例えば、コントロールの型のスタイルの定義が、スタイルシートレベルの同じプロパティのための定義よりも、そのコントロールの型に対して高い優先順位を有する場合などが挙げられます。

下のテーブルは、それぞれの列に対して、異なるレベルの優先順位が降順に表示されています。列内で低いレベルを持つスタイルは比較的高い優先順位を有します。

スタイルシート	スタイルシート	スタイルシート
すべての コントロール	テーブル	ページ
Individual コントロールの型		

メモ 個々のコントロールの型、テーブル、またはページの単一インスタンスのためにプロパティを設定するには、デザイン内でそのインスタンスを選択し、自身のプロパティの値を割り当てます。デザインコンポーネントに直接割り当てられるため、デザインコンポーネントにより近く存在するため、この定義は、スタイルシート内の定義よりも高い優先順位を有します。

スタイルシート: スコープとアプリケーション

プロジェクト スタイルシートは、自動的にプロジェクト全体に適用されます。これは、例えば、*プロジェクト* スタイルシートのスタイルシートレベルで定義されている**背景色** プロパティの値が自動的に、プロパティ内の全ての**背景色** プロパティにより継

承されることを意味します。ユーザーが作成したスタイルシートは、その一方、ページのインスタンス、テーブル、テーブル個々のコントロールのみに適用され、プロジェクト全体にはお適用されません。下のテーブルは、ページ、テーブル コントロール インスタンスに適用された場合、どのデザインコンポーネントが特定のスタイルシートのレベルで定義されたスタイルを継承するかを示しています。

スタイルシート内の定義のレベル	適用されるスタイルシート ...		
	ページ インスタンス	テーブル インスタンス	コントロール インスタンス
スタイルシート	ページ インスタンス。ページ上の全てのテーブルとコントロール	テーブル インスタンス、テーブル内の全てのコントロール	コントロール インスタンス
全てのコントロール	ページ上の全てのコントロール	テーブル内の全てのコントロール	コントロール インスタンス
コントロールの型	ページ上のその型の全てのコントロール	テーブル内のその型の全てのコントロール	その型のコントロール インスタンス
テーブル	ページ上の全てのテーブル	テーブル インスタンス	- -
ページ	ページ インスタンス	- -	- -

スタイルシート全体の優先順位

(異なるレベルの異なるスタイルシート内で) 単一のスタイルプロパティに対して(異なるレベル、または、異なるスタイルシートで)複数の定義が存在する場合、これらの定義の1つ以上が単一のデザインコンポーネントのスタイルプロパティに定義される場合、優先順位に関する問題が発生します。この場合、MobileTogether は、このプロパティのための定義を、下にリストされる異なるスタイルシートのレベルで確認します。最初に一致するスタイルが使用されます。したのテーブルは、ボタンコントロールの **背景色** プロパティの例を使用しています。

デザイン内でボタンコントロールのために定義される背景色	**高い優先順位**
デザイン内のボタンコントロールが スタイルシート1 を参照する場合	
スタイルシート1 内のボタンコントロールのために定義された背景色	
スタイルシート1 内のすべてのコントロールのために定義された背景色	
スタイルシート1 のために定義された背景色	
デザイン内のボタンコントロールが スタイルシート2 を参照する場合	
スタイルシート2 内のボタンコントロールのために定義された背景色	
スタイルシート2 内のすべてのコントロールのために定義された背景色	
スタイルシート2 のために定義された背景色	
デザイン内のボタンコントロールが スタイルシート3 を参照する場合	
スタイルシート3 内のボタンコントロールのために定義された背景色	
スタイルシート3 内のすべてのコントロールのために定義された背景色	
スタイルシート3 のために定義された背景色	
プロジェクトスタイルシート内のボタンコントロールのために定義された背景色	
のために定義された背景色内のすべてのコントロールのために定義された背景色	
のために定義された背景色 スタイルシートのために定義された背景色	**低い優先順位**

XPath 式を使用してプロパティの値を定義する場合、次に注意してください:

- 式がからのシーケンスを評価する場合、リストは上から下に検索されます。
- (可視性、太字、 および斜線 などの) ブール値をとるプロパティのために式が定義されており、値が `true` ではない場合、XPath のルールに従い、値は `false` になります。この結果リストを更に検索することはできません。

プラットフォームのデフォルトの値

モバイルデバイス プラットフォーム(Android、iOS、Windows) には、特定のスタイルプロパティのためのデフォルトの値が存在します。例えば、iOS デバイスのデフォルトのページの背景色は、白ですが、Android デバイスのデフォルトの背景色は黒である場合があります。しかしながら、そのプラットフォームのデフォルトの設定は、全てのプロパティで使用できるとは限らない点に注意してください。 **「プラットフォームのデフォルトの値の設定」** コマンドを使用して、そのプロパティのデフォルトのプラットフォームにプロパティ値を設定することができます。プラットフォーム デフォルトの値は、次の定義レベルで設定することができます:

- デザインコンポーネントに対して直接設定する スタイル & プロパティペイン内のデザインコンポーネントのプロパティ定義を右クリックし、 **「プラットフォームのデフォルトの値の設定」** を選択します。
- スタイルシート内のプロパティに対して設定する (プロジェクト、全てのコントロール、特定のコントロールの型、テーブ

ル、および、ページなどの)全てのスタイルシートのレベルで定義されているプロパティを右クリックし、**「プラットフォームのデフォルトの値の設定」**を選択します。

その他のスタイル定義と同様、プラットフォームデフォルトの値は、(コンポーネントから)相対的に遠い場所で定義されている値は、コンポーネントに相対的に近く定義されているスタイル定義にオーバーライドされます。

ユーザーにより作成されたスタイルシートの適用

ユーザーが作成したスタイルシートをページインスタンス、テーブルインスタンスとコントロールインスタンスに適用することができます。ユーザーが作成したスタイルシート内のスタイルの定義は、即時に選択されたデザインコンポーネントに適用され、優先順位の低い既存のスタイルの定義をオーバーライドします。

ユーザーが作成したスタイルシートを(ページ、テーブル、またはコントロールなどの) デザインコンポーネントを以下のように適用することができます:

1. デザイン内で、ユーザーが作成したスタイルシートを適用するデザイン コンポーネント(ページ、テーブル、またはコントロール) を選択します。
2. スタイル & プロパティペインで、スタイルシートを適用するページ、テーブル、またはコントロールのスタイルプロパティを選択します。下のスクリーンショットでは、コントロールのスタイルシートプロパティが選択されています。

3. スタイルシート プロパティのコンボボックスのドロップダウンリスト(*上のスクリーンショット参照*) から、デザインコンポーネントを適用するユーザーが作成したスタイルシートを選択します(ドロップダウンリストは、現在のプロジェクトのすべてのユーザーが作成したスタイルシート含みます) または、ペインのツールバー内のXPath アイコンをクリックし、適用するスタイルシートの名前を評価するXPath 式を入力してください。

メモ デザインコンポーネントがスタイルシートによりスタイルを与えられている場合、プロパティの名前を含むセルの右下に緑色のマーカーで表示されています(上のスクリーンショット内の**スペースの高さ**プロパティを参照してください)。マウスのカーソルをマーカーニポイントすると、スタイルシートの情報がポップアップで表示されます。マーカーをクリックすると、スタイルシート ダイアログ内の対応する定義に移動することができます。

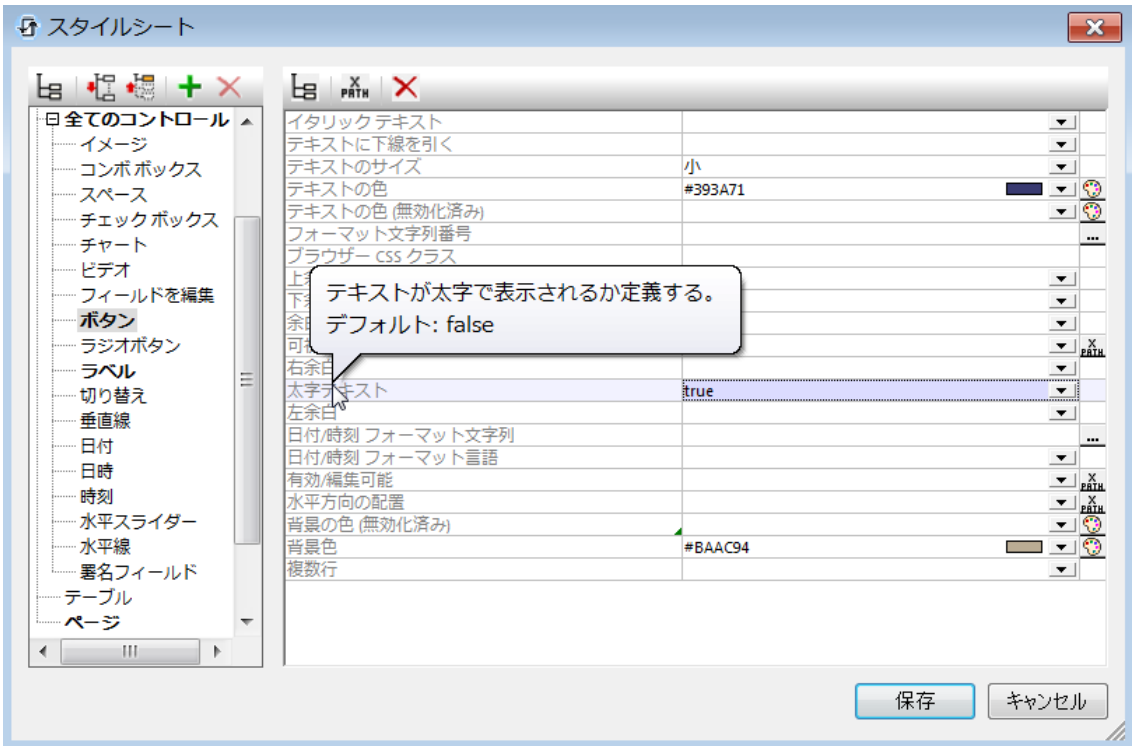
XPath を使用してスタイルシートを選択する利点

ユーザーが作成したスタイルシートを選択するためにXPath 式を使用する大きな利点は、選択が動的な環境条件に対して条件付きで行えることです。例えば、 iOS デバイスのために一枚のスタイルシートを指定し、その他のデバイスのために他のスタイルシートを指定する場合、次のXPath 式を使用することができます: `if ($MT_iOS=true()) then 'iOSStyleSheet' else 'GeneralStyleSheet'`.

メモ スタイルシートを頻繁にランタイムに切り替えることは、ソリューションの実行のスピードを下げる可能性があります。

スタイルシートプロパティ

スタイルシート内では、(ボタンコントロールとラベルコントロールなどの)個々のコントロールの型、テーブル、および、ページのためにスタイルを定義することができます。左側のペイン内でスタイルを定義するデザインコンポーネントを選択してください。選択されたデザインコンポーネントのプロパティが右側のペインに表示されます。各プロパティのために値を選択または入力することができます。完了すると **保存** をクリックします。



以下の点に留意してください:

- プロパティ名にマウスをポイントすると、プロパティのデフォルトの値を含むプロパティについての情報が表示されます (*スクリーンショット参照*).
- 色の値を取るプロパティに関しては、プロパティのカラーピッカーをクリックして適切な色を素早く選択します。
- XPath 式をプロパティの値として入力することができます。(存在する場合) プロパティフィールドの右側の XPath アイコンをクリックまたは、ダイアログのツールバー内の **XPath** アイコンをクリックします。
- ツールバーアイコンをクリックして、空ではないプロパティを表示します。これにより、低優先のプロパティを省き、選択されているデザインコンポーネントの現在定義されているスタイルのみを確認することができます。
- 設定したプロパティの値を削除するには、ダイアログのツールバー内の **「リセット」** をクリックします。
- e プロパティの プラットフォームのデフォルトの値 にプロパティの値を設定することもできます。これをおこなうには、プロパティをダブルクリックして、**「プラットフォームのデフォルトの値の設定」** を選択します。

11.8 リッチテキスト

リッチテキストの機能により、リッチテキスト書式を持つXML ページソースをこの書式を持つソリューションを表示することができます。異なる要素のために自身のスタイルを定義することができます。Web クライアントとWindows クライアント上では、このようなコンテンツを編集し、新規のコンテンツがリッチテキストの書式をもつことができるように、また、適切なXML マークアップと共に保存できるようにXML ページソースに戻して保存することができます。

機能は次のメカニズムを使用して実装することができます:

- リッチテキスト コントロールを、リッチテキストで表示を希望するデザイン内に配置します。コントロールには、2つの主要な関連性が存在します: (i) XML ページソース(ii) リッチテキスト スタイルシート。
- リッチテキスト スタイルシート ダイアログ内では、既存のスタイルルールのための異なる要素とスタイルマッピングのためのスタイルルールを定義することができます。

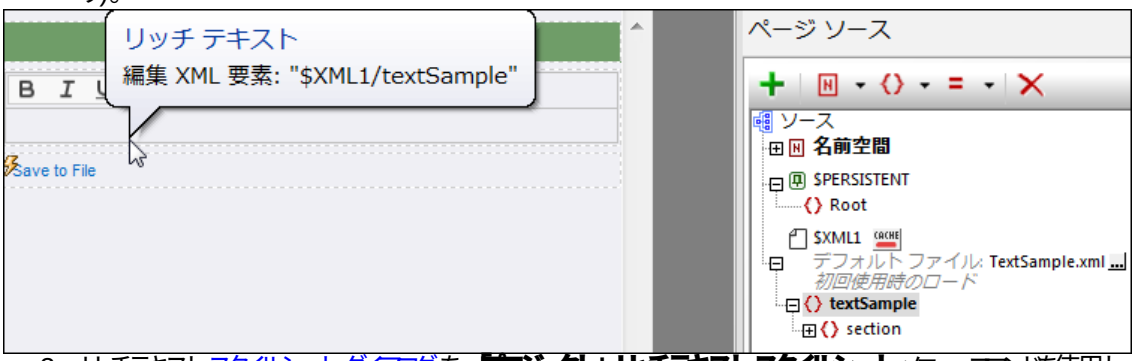
このセクション
このセクションは次のサブセクションに整理されています:

- リッチテキスト コントロール
- リッチテキスト スタイルシート: セットアップ
- リッチテキスト スタイルシート: スタイル
- リッチテキスト コンテンツの編集

リッチテキストコントロール

デザイン内でリッチテキストを有効化する最初のステップは、ページ内のテキストを表示する箇所にリッチテキストコントロールを追加することです。コントロールを以下のようにセットアップします。

1. リッチテキストコントロールをページにドラッグアンドドロップします（*下のスクリーンショット参照*）。
2. リッチテキストを含むページソースの **「ルート要素」** をコントロールにドラッグアンドドロップします（コントロールのページソースリンクは、ページソースのルート要素である必要があり、コントロールは属性、または、ルート要素以外の要素にリンクすることができません）。例えば、下のスクリーンショットでは、コントロールは、要素 **`textSample`** である **`$XML1`** のルート要素にリンクされています(表示するリッチテキストはこのXML ページソース内に保管されています)。



3. リッチテキスト スタイルシート ダイアログを **「プロジェクト | リッチテキスト スタイルシート」** メニューコマンドを使用して開き、新規のスタイルシートを作成します。プロジェクトのために複数のスタイルシートを定義することができます。これにより、異なるスタイルシートを異なるリッチテキスト コントロールに割り当てることができます。スタイルシートのセットアップの方法は、リッチテキスト スタイルシートのセクションで説明されています。
4. スタイルシートを作成すると、リッチテキストスタイルシートプロパティを使用して、デザイン内でリッチテキスト コントロールを選択し、コントロールにスタイルシートを割り当てます。各 リッチテキスト コントロールには1度に1つのスタイルシートのみしか割り当てることはできません。

メモ コントロールの **リッチテキストの高さ** プロパティを使用してコントロールの高さを設定することができます。

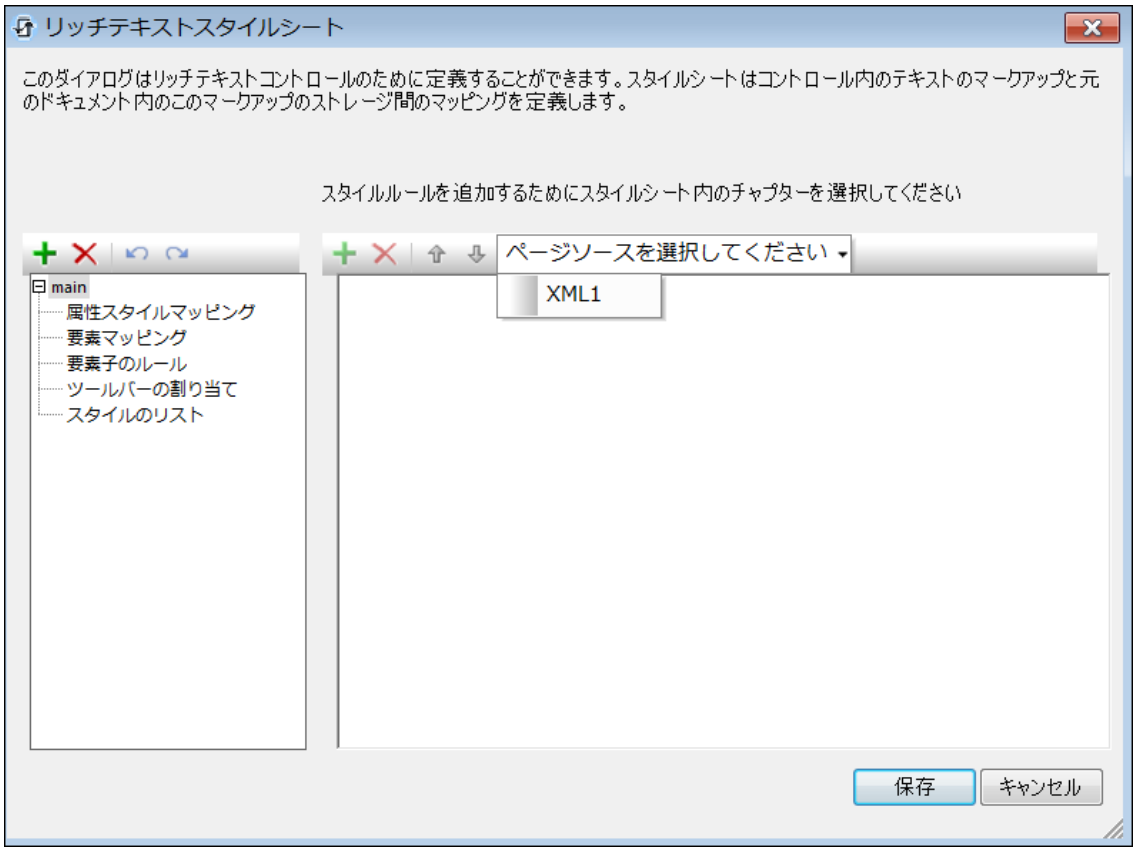
リッチテキスト スタイルシート: セットアップ

リッチテキスト スタイルシートはXML (ページソース内のデータ) と[リッチテキスト コントロール]内で表示する) HTML 間の変換を説明する複数のルールにより構成されています。プロジェクトのために複数のスタイルシートを定義することができます。これらのスタイルシートを[リッチテキスト コントロール]に割り当てることができます。スタイルシート内のスタイルは、[リッチテキスト コントロール]内で表示されるテキストのスタイルのために使用されます。

スタイルシートの作成方法

リッチテキスト スタイルシートを作成するには、以下を行います:

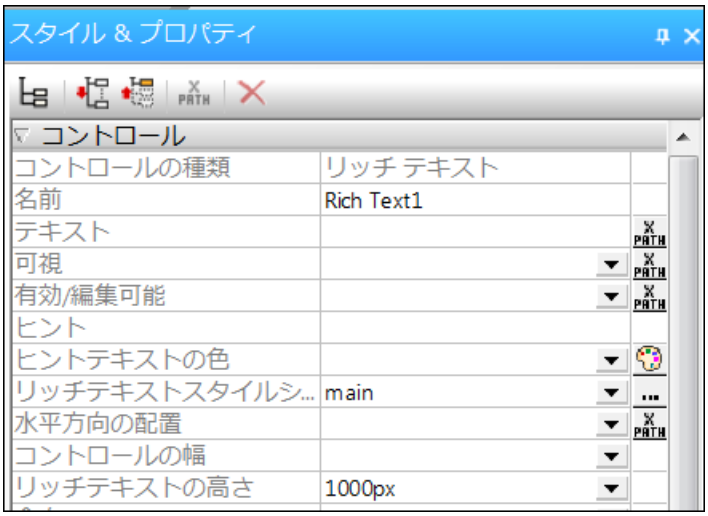
1. メニューコマンド **「プロジェクト」** **リッチテキスト スタイルシート**を選択します。リッチテキスト スタイルシート ダイアログが表示されます (*下のスクリーンショット参照*) 。
2. 左側のペインのツールバー内の **「スタイルシートの追加」**をクリックします。
3. プロジェクトに適したスタイルシートの名前を編集します。下のスクリーンショットでは、例えば、スタイルシートは `main` 名前が変更されています。
4. スタイルシートが適用されるページソースを選択してください。
5. プロジェクトと共にスタイルシートを保存するために **「保存」**をクリックします。



スタイル マッピングの作成方法に関しては、セクション[リッチテキスト スタイルシート: スタイル]内でルールが詳細に説明されています。

リッチテキスト コントロールにスタイルシートを割り当てる方法

スタイルシートの作成、保存後、コントロールの リッチテキスト スタイルシート プロパティを使用してリッチテキスト コントロールに割り当てることができます。下のスクリーンショットでは、例えば、リッチテキスト コントロールは **main** という名前のスタイルシートに割り当てられています。



リッチテキスト スタイルシート: スタイル

リッチテキスト スタイルシートが作成された後のスタイルのルールとマッピングはリッチテキスト スタイルシート ダイアログのセクション内で定義されています。これらのルールとマッピングはページソースのXML 内のスタイルのクライアントデバイス上で表示することのできるHTML へのスタイルの変換に関連しています。Web クライアントとWindows クライアントの場合、リッチテキスト コンテンツはエンドユーザーにより編集し書式変更されることが可能です。ルールとマッピングはこのため、変更されたHTML スタイルをXML ページソースへ戻すために使用されます。

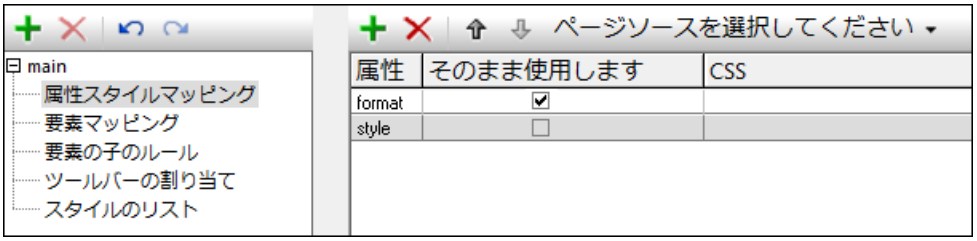
リッチテキスト スタイルシート ダイアログ内では、スタイルルールとマッピングはこれらのセクションは以下で説明されています:

- 属性スタイルマッピング
- 要素マッピング
- 要素の子のルール
- ツールバーの割り当て
- スタイルのリスト

属性スタイルマッピング

このセクションで定義されている属性はリッチテキスト コントロール内のコンテンツにページソース内の同じ名前の属性にマップされます (*下のスクリーンショット参照*) 。ページソース内の**すべて**の要素の属性にマッピングは適用されます。ここで定義されている属性は、親要素がどのようにスタイルされるかを説明しています。他方、ここで**定義されていない**属性は親要素の**スタイルをパスしません**。

右側のペインのツールバー内の **「スタイルの追加」**をクリックして、属性の名前を入力し、スタイルされている属性のリストに属性を追加することができます。



以下の点について注意してください:

- *そのまま使用する*は、ページソース内の属性のコンテンツが有効なCSS であることを期待します。
- *CSS:* 1つ、または、複数のCSS プロパティ値ペアをとります。XML ページソース内の属性からプロパティ値を使用する場合、`$value` を使用して、その属性の値を取得します。例えば、ページソース内の属性が`format` という名前の属性を持つ場合 : (i) `<myelement format="red">...</myelement>` と(ii) `format` 属性はスタイルシート内で上のスクリーンショットで示されるように定義されており、`red` である`format` 属性の全体の値はスタイルの定義内の`$value` と置き換えられます。ですから、`myelement` の場合、(`format` 属性のスタイルの定義から取得される) スタイルの定義は以下に対して解決されます: `font-weight:bold; color:red`。ページソース内で、他の要素が`@format="blue"` を持つ場合、要素のスタイルの定義は次に対して解決されます: `font-weight:bold; color:blue`.
- If 要素 in XML ページソース内の要素がスタイルシート内で存在する*そのまま使用する*スタイルの定義の両方の属性を持つ場合、その要素をスタイルするために2つのスタイルの定義は結合されます。

サンプル

- 上のスクリーンショット内の属性の定義がページソースのXML 内の次の要素に適用されている場合:
  `<heading style="font-style:italic;" global="font-weight:bold;">Text Formatting</heading>`
  `@style` 属性はそのまま使用されるため、結果するスタイルの定義は`font-style:italic` となります。

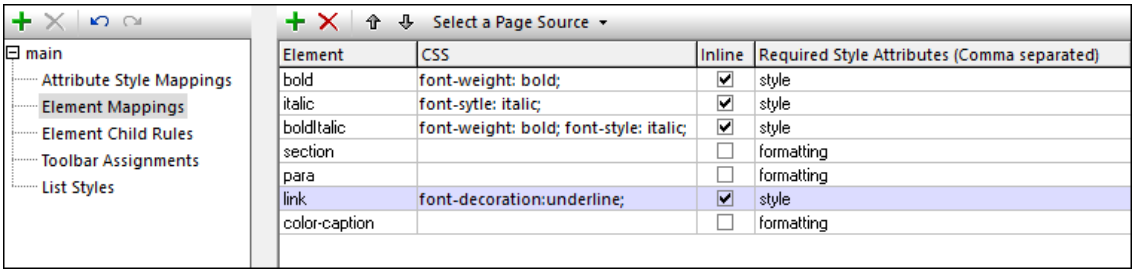
@global 属性はスタイルシート内で定義されていないため無視されます。

- 上のスクリーンショット内の属性の定義が次の要素に適用されている場合:
  <heading format="red">Text Formatting</heading>
  スタイルシート内の format 属性の定義であるため、適用されるスタイルは、 font-weight:bold; color:red; になります。

要素マッピング

各要素マッピング (*下のスクリーンショット参照*) はリストされる要素の特定のプロパティを定義し、以下を行うことができます:

- 要素のCSS スタイルプロパティを定義します。定義されている場合、これらのスタイルプロパティは*属性スタイルマッピング*セクション内で属性のために定義されるスタイルより高い優先順位を持ちます。
- 要素が(HTML div に対応する) ブロック要素 、または、(HTML span に対応する) インライン要素であるかを指定します。デフォルトの設定はブロックです。このため、デフォルトでは、インラインと指定されない限り、すべての要素は新規のラインに表示されます。一方、インライン要素はライン内で発生します。共通のインライン要素はライン内のテキストを太字、または、斜体でマークアップします。下のスクリーンショットでは、どの要素がインラインとして指定されているかに注意してください。
- どの要素の属性が要素のスタイルプロパティを持つか指定してください。これらの属性は、(クライアントから) スタイルプロパティが書き込まれるXML ドキュメント内の属性になります。



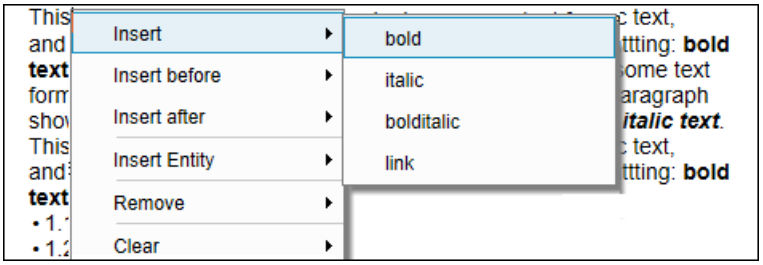
右側のペインのツールバー内の **「スタイルの追加」** をクリックして、定義する要素の名前を入力してリストに要素を追加することができます。

要素の子のルール

このセクションでのルールは要素が持つことのできる子要素を定義することです (*下のスクリーンショット参照*)。入力する子要素のリストは、スキーマで許可されているすべての子要素のリストに完全に一致する必要はありません。 リッチテキスト コントロール内のコンテンツを編集する際にエンドユーザーが追加することを許可する子要素のリストです。要素がここにリストされていない場合、その要素の子としてすべての要素を使用することができます。

右側のペインのツールバー内の **「スタイルの追加」** をクリックし、要素の名前を入力して、リストに要素を追加することができます。

上のスクリーンショットでは、例えば、**para** 要素は次の子を持つように定義されています: **bold**、**italic**、**bolditalic** と**link**。エンドユーザーがこのスタイルシートを使用するリッチテキストコントロール内のコンテンツを編集中に**para** 要素内を右クリックすると、下のスクリーンショット内に表示されるコンテキストメニューが表示されます。挿入することのできる要素は**para** 要素の子として定義されている要素です。

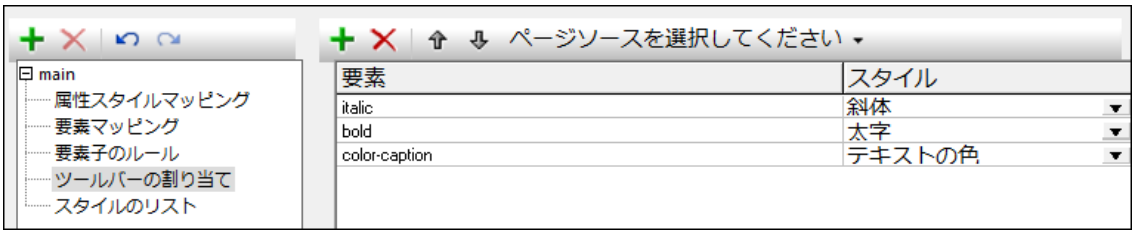


ツールバーの割り当て

リッチテキスト コントロールがWeb クライアント、または、Windows クライアントのソリューション内で表示される場合、コントロールのコンテンツを編集することが可能です。これらのデバイス上では、ツールバーの編集は、コントロールの上の部分に表示されます (*下のスクリーンショット参照*) 。エンドユーザーはツールバー内のアイコンをクリックして、選択されたテキストコンテンツにアイコンのスタイルを割り当てることができます。

リッチテキスト スタイルシート ダイアログの*ツールバーの割り当て* セクションで XML 要素をツールバーアイテムにマップします (*下のスクリーンショット*)。下のスクリーンショットでは、例えば、(i) (*スタイル*列内のコンボボックスで選択済みの)**斜体** アイコンは**italic** という名前の要素にマップされます。(ii) **太字** アイコンは**bold** という名前の要素にマップされます。(iii) **テキストの色** の選択は、**color-caption** という名前の要素にマップされます。エンドユーザーがツールバーアイコンをクリックすると、または、ツールバーのコンボボックスの1つを選択すると、ツールバーアイテムに関連したスタイルが選択されたテキストに適用されます。

XML レベルでは、ツールバーアイテムにマップされている要素は選択されたテキストの周りで作成されます。この要素に関連したスタイルが存在しない場合、関連するスタイルを含む属性が要素に追加されます。例えば、下のスクリーンショットでは、**italic** 要素がダイアログの要素マッピング セクション内の斜線のスタイルに関連付けられており、(**斜体** ツールバーアイコンをクリックし) テキストの選択の周りで **italic** 要素を作成することは、テイストの選択が斜体で表示されることを意味します。下のスクリーンショット内に表示されている**テキストの色** ツールバーアイテムの場合、**color-caption** 要素のためにスタイルプロパティは定義されていません(*要素マッピング セクション内のスクリーンショットを参照してください*) 。この結果、ツールバー内でエンドユーザーが選択するテキストの色は、**color-caption** 要素の**formatting** 属性内に保管されます。編集されたソリューションからどのXML 要素 スタイルの属性がマップされるかは、要素のマッピング セクション内の要素の定義内で指定されています (スクリーンショット内の**color-caption** の定義 を参照してください)。



右側のペインのツールバー内の **「スタイルの追加」** をクリックし、要素の名前を入力して *ツールバーの割り当て* リストに要素を追加することができます。次の列内では、コンボボックスから対応するツールバーアイテムを選択します (使用することのできる ツールバーアイテムすべては、コンボボックスのドロップダウンリスト内でリストされています)。

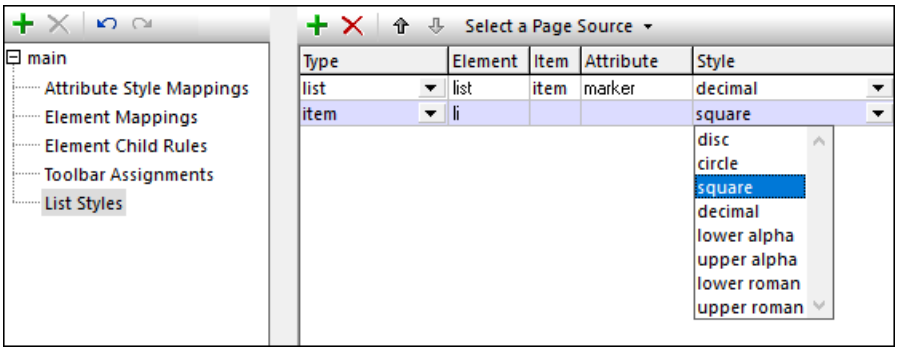
スタイルのリスト

スタイルのリストは、リスト構造の2つの型に対して使用することができます:

- *コンテナーのリスト(編集可能な マーカー はリスト全体に適用されます)*: リストアイテム子要素を含むリスト要素としてリストは構成されています。例: `<list> <item/>...<item/> </list>`。下のスクリーンショットでは、最初の定義はこの型のリストです。エンドユーザーはリストマーカーのリスト全体を変更することができます(例えば、マーカーが数字、または、スクエアであるかなど)。デザイナーとしてリストの最初のマーカーを定義します。
- *アイテムシーケンスリスト(編集可能な マーカー は個別のリストアイテムに適用されます)*: リストアイテム要素のシーケンスにより構成されるリスト要素を含まないリストです。例: `<li>aaa</li>...<li>nnn</li>`. 下のスクリーンショットでは、2番目の定義がこの型のリストです。**個別** リストアイテムのマーカー文字は、エンドユーザーにより編集されることができますが、マーカーを1度に編集することは **できません**。

エンドユーザーは、リストの両方の種類のを編集することができ、リストアイテムを追加、または、削除することもできます (*リッチテキスト コンテンツの編集を参照してください*).

メモ 選択されたマーカーは、リストが番号別、または、アイテム別に整理されるか決定します。



スタイルのリストを定義するには(*上のスクリーンショット参照*)、以下を行います:

1. 右側のペインのツールバー内の **「スタイルの追加」** をクリックしてリストのためにエントリを追加します。
2. *要素* 列内: コンテナーのリストのために、リストに対応する要素の名前を入力します。アイテムシーケンスリストのためにリストアイテム要素の名前を入力します。
3. *アイテム*列内: コンテナーのリストのために、リストに対応する要素の名前を入力します。アイテムシーケンスリストのためには、入力をしません。
4. *属性* 列内: コンテナーのリストのために、リストマーカーの名前を持つ属性の名前を入力します。アイテムシーケンスリストのためには何も入力しません。エンドユーザーが他のマーカーを最初のマーカー以外でコンテナーのリストのために選択する場合、新規のマーカー名がこの属性のために入力され、リストはこの属性の値を基にスタイルされます。

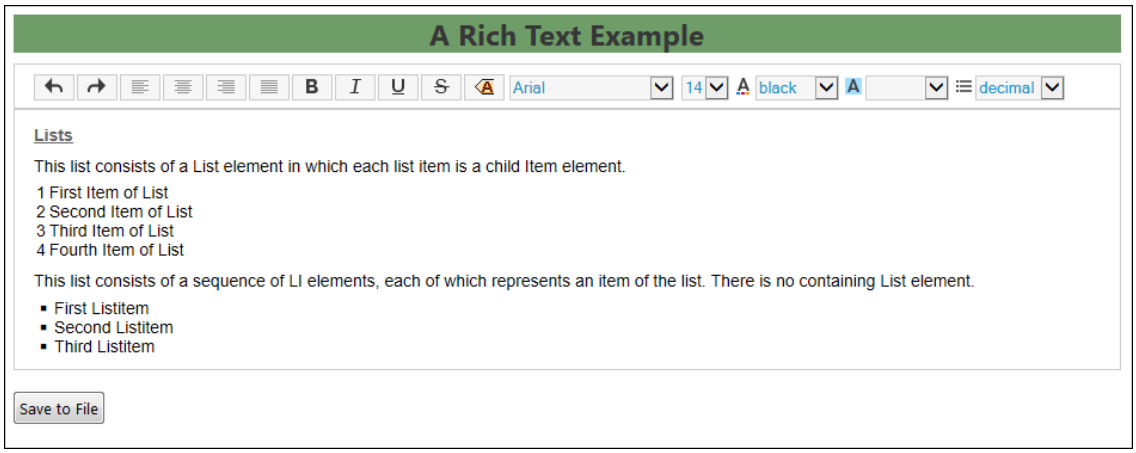
リッチテキストコンテンツの編集

リッチテキスト コントロール内のテキストコンテンツはすべてのクライアントデバイス上で**表示**されます。コントロールに割り当てられているスタイルシートは内でスタイルは定義されます。Web クライアントとWindows クライアント上では、エンドユーザーはテキストコンテンツを更に編集することができます。このセクションは、Web 、または、Windows クライアント上のリッチテキストの編集方法について説明されています。

リッチテキスト コントロール

リッチテキスト コントロールがWeb 、または、Windows クライアント上で表示される場合、このコントロールは2つの部分により構成されています (*下のスクリーンショット*)。:

- コントロールの上部で柔軟に使用することのできる編集のためのツールバー。
- テキストを編集することのできるコンテンツの部分 (変更済み、追加済み、または、削除済み)。コンテンツがデバイス上のコントロールに与えられているスペースよりも広いスペースを必要とする場合、コントロールのコンテンツはスクロールバーを必とします。コントロールの高さは**リッチテキストの高さ** プロパティを使用して設定することができます。



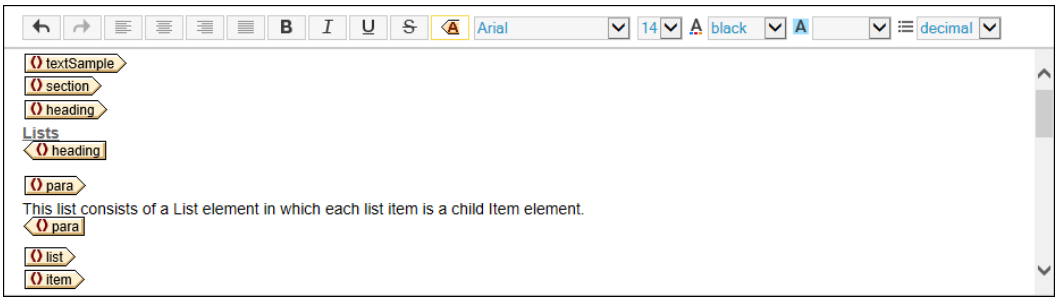
コントロールのツールバー

リッチテキスト コントロールのツールバーはWeb とWindows クライアントのみで表示されます(*下のスクリーンショット*)。

(左から)次のアイコンによりこのルールバーは構成されています。:

- *元にもどす* (**Ctrl+Z**) アイコンおよび *やり直し* (**Ctrl+Y**) アイコン.
- 選択されたテキストをフォーマットするため使用される太字 (**Ctrl+B**)、斜体 (**Ctrl+I**)、下線 (**Ctrl+U**) および取り消し線アイコン。マークアップのレベルでは、それぞれのスタイルが、そのアイコンを割り当てられている要素と共に囲まれた選択済みテキストに適用されます。 (要素 、または、割り当てられた属性がそれぞれのスタイルを指定します。次を参照してください: ツールバーの割り当て)
- コンテンツを切り取り(**Ctrl+X**)、コピー(**Ctrl+C**)、および、貼り付け(**Ctrl+V**) することができます。リッチテキスト コントロールからコンテンツがコピーされた場合、貼り付ける際にコンテンツをXML 、または、テキストとして貼り付けるかを問うポップアップが表示されます。コピーされたXML ツリーはXML 構造 がコピーされた要素のコピーを許可するものとして貼り付けられます。それ以外の場合は、テキストコンテンツのみが貼り付けられます。
- 要素 マークアップタグをオンとオフに切り替えるアイコン(*下のスクリーンショットはオンに切り替えられている マーク*

アップタグを表示しています)。

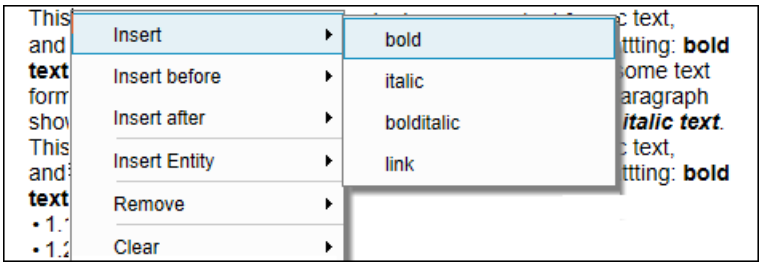


- フォントのファミリー、フォントのサイズ、フォントの色、および、フォントの背景をそれぞれ選択することのできる4つのコンボボックス。
- リストアイテムのマーカーを選択することのできるコンボボックス(i) リスト全体 (コンテナーのリストの場合)、または、(ii) 選択されたリストアイテム(アイテムシーケンスリストの場合)。

メモ ツールバーをベースにしたスタイルを適用することのできない場所にカーソルがポイントされている場合、ツールバーアイテムは無効化されています。

要素の挿入と削除の方法

テキスト コンテンツ内に要素を挿入するには、スタイルシート内で編集することのできる要素の子要素を定義することが最善策です。これが行われていると、エンドユーザーが要素内のテキストの選択を右クリックしてコンテキストメニューから挿入する要素を選択することができます (*下のスクリーンショット参照*)。

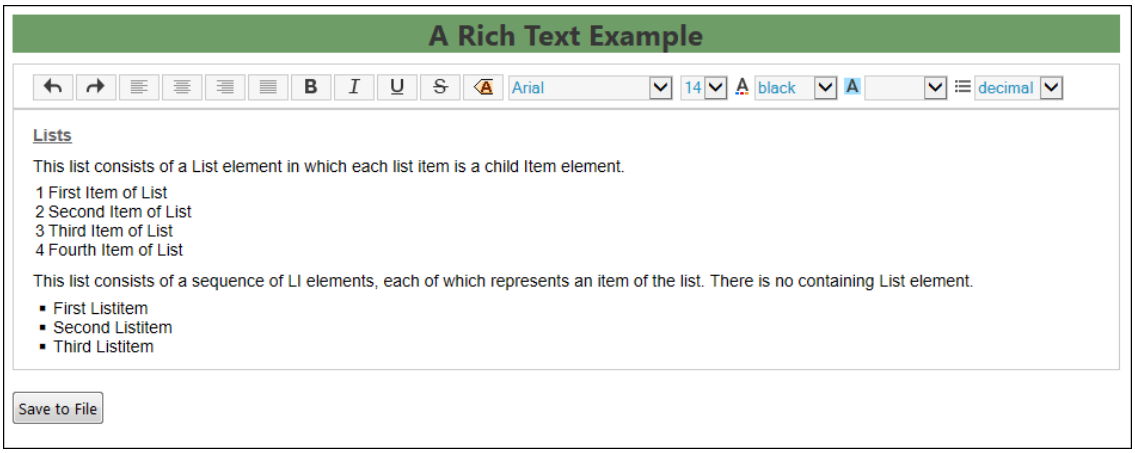


コンテキストメニュー内で表示される要素は、カーソルが置かれている要素により異なります:

- *挿入:* カーソルの位置、または、テキストの選択箇所で子要素を挿入します。
- *前 後に挿入:* カーソルがポイントされている要素の前、または、後に兄弟要素を挿入します。コンテキストメニューは、スタイルシート内で洗濯されているページソースの構造に従い兄弟要素を表示します。
- *エンティティの挿入:* XML (特殊文字アンパサンド、アポストロフィ、より小さい、より大きい、文字、引用符) をカーソルの選択の位置で文字エンティティ、または、テキストの選択の置換えとして挿入します。
- *削除:* カーソルがポイントされている箇所、または、祖先要素内で要素を削除します。コンテキストメニューは、スタイルシート内の洗濯されたページソースの構造を基にして現在の要素とすべての祖先を表示します。
- *クリア:* 囲まれたテキストの選択に追加されているマークアップをクリアします。例えば、テキストの選択が太字の場合、太字のスタイルを作成するマークアップはクリアされます。

リストの編集方法

次のスクリーンショットは2つのリストを表示しています。最初のリストはコンテナーのリストで2番目のリストはアイテムシーケンスリストです。



リスト固有の編集機能について以下の点に注意してください：

- ツールバー内のリストメーカーコンボボックスを使用してリストアイテムのマーカーを変更してください。コンテナーのリスト内のアイテムが選択されている場合、リスト内のすべてのアイテムのマーカーは変更されます。アイテムがアイテムシーケンスリストが選択されたとおりのままの場合、マーカーおよびそのリストアイテムのみが変更されます。
- リストアイテムを挿入するには、新規のリストアイテムを挿入するすぐ後のカーソルをリストアイテムの終わりに配置し、「**Enter**」を押します。
- リストアイテムを削除するには、マークアップタグを切り替え、削除するアイテムリストを選択し、コンテキストメニュー内の「**削除**」を選択します。

11.9 ソリューションへのハイパーリンク

ソリューションへのハイパーリンクを以下の方法で作成することができます:

- ページまたはコントロールイベントのURL を開くアクションを使用します。
- エンドユーザーが送信する電子メールを使用します。

ハイパーリンクのURL がクエリ文字列を含まない場合、ソリューションは最初のページから開かれます。URL がクエリ文字列を含まない場合は、ソリューションはソリューションとクエリ文字列のロジックに従い開かれます。(クエリ文字列無しと有り) の 2 つのURL 種類の例としては、 Google などの検索エンジンのURL があげられます。

- クエリ文字列無しのURL はGoogle スタートページを開きます: https://www.google.com/
- 検索エンジンで "Altova MobileTogether" を検索するクエリ文字列 (疑問符の後はすべてクエリ文字列としてみなされます) を含むURL の場合、(Google のスタートページではなく) URL は直接検索の結果を含むページを開きます : https://www.google.com/search?q=Altova+MobileTogether&ie=utf-8&oe=utf-8&gws_rd=cr&ei=3YAaVdDDA4SYsgGOm4A4

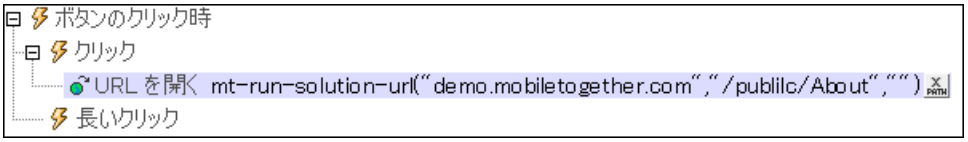
メモ サーバーの設定を更新するリンクは、Gmail 内および他の電子メールアプリケーション内では、作動しませんが、AquaMail、K9、およびMailWise などのよく使用されるクライアントでは作動します。AquaMail およびK9 内でテスト済みでこれらのアプリケーションでは正常に作動します。

デザインコンポーネントからソリューションをリンクする

URL を開くアクションのコンポーネントを使用して、デザインコンポーネントをソリューションにリンクすることができます。例えば、ボタンがクリックされると、ボタンのURL を開くアクションはソリューションが開かれることを指定することができます。

ソリューションのリンクの作成方法は以下の通りです:

1. ソリューションリンクとして指定するイベントのためにURL を開くアクションを作成します。(下のスクリーンショット参照)。
2. ソリューションのURL を生成するために、**mt-run-solution-url** 関数を使用するXPath 式を作成します。



- mt-run-solution-url

  mt-run-solution-url(ServerAddress? を*xs:string* として, **SolutionName?** を *xs:string* として, **InputParameters?** を*xs:string* として) を**xs:string?**として 3 つの提供された引数からソリューションのURL を生成します:

  - `ServerAddress`: 実行するソリューションがデプロイされるMobileTogether Server のIP アドレスまたは名前を使用します。
  - `SolutionName`: サーバー上のソリューションのデプロイされたパスを使用します。例えば、: `/public/MySolution` は`/Public` コンテナー内の`MySolution.mtd` ファイルをポイントします。
  - `InputParameters`: `mt-run-solution-url-parameters` を入力として使用します。関数の引数は、クエリのパラメーターの値を与える文字列の値のシーケンスです。 `mt-run-solution-url-parameters` 関数は、正しくエンコードされた、パーセントを除いたクエリ文字列のエンコードのルールに従ったURL のクエリ文字列 のパラメーター (名前と値) を含む文字列を返します。 結果文字列内のパラメーター名は、自動的に関数に(`inputparameter1, inputparameter2 ... inputparameterN`) より生成され、それぞれ関数の引数の文字列アイテムから、インデックス順に組み合わされた名前と値により値を

割り当てられます。(更に、 InputParameters 引数は、既にURL の一部のクエリ文字列としてエンコードされている文字列として提供されます(下の4番目のサンプルを参照)。)

mt-run-solution-url 関数はですから、MobileTogether Server 上でソリューションにアクセスするクエリパラメーターを持つまたは持たないURL を作成します。クエリパラメーターは、ソリューションがURL を使用して開かれるとソリューションにパスされます。これらのパラメーターの値は、**$MT_InputParameters** グローバル変数を使用して、他のデザインコンポーネントでアクセスすることができます。

サンプル

- **mt-run-solution-url**('100.00.000.1', '/public/MyDesign', '') は、IP アドレス100.00.000.1. を持つMyDesign ソリューションをポイントする URL を返します。URL には、クエリパラメータはありません。
- **mt-run-solution-url**('', '/public/MyDesign', '') は、現在のサーバー上のMyDesign ソリューションをポイントするURL を返します。URL には、クエリパラメータはありません。
- **mt-run-solution-url**('', '', mt-run-solution-url-parameters(('2015', 'USA', 'true'))) は、現在のサーバー上の現在のソリューションをポイントするURL を返します。**mt-run-solution-url-parameters** 関数の引数は、クエリのパラメーターの値である文字列の値のシーケンスです。最初の文字列は、最初のパラメータの値です。第 2 の文字列は、第 2 のパラメーターの値です。**mt-run-solution-url-parameters** 関数は関数は、正しくエンコードされ、パーセントを除いたURL クエリ文字列のエンコードのルールに従った文字列を返します。
- **mt-run-solution-url**('', '', 'inputparameter1=value1&inputparameter2=value2%3FAndMoreValue2') は、現在のサーバー上の現在のソリューションをポイントするURL を返します。InputParameters 引数は、URL クエリ文字列として既にエンコードされた文字列として提出されます。

以下の点に注意してください:

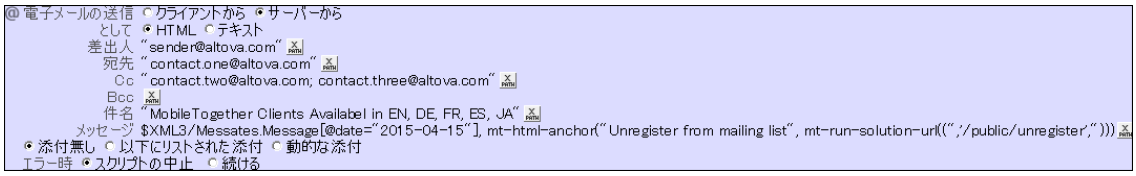
- 最初の引数 ServerAddress が空の文字列の場合、現在のサーバーが使用されています。
- 最初の引数 ServerAddress がクライアントに保管されているサーバー情報を検索するために使用されています。サーバー名と関連付けられているポート番号、ユーザー名、ユーザーパスワードはサーバーに接続するために使用されています。クライアントが認識しないサーバー名を持つURL が生成された場合、URL は作動しません。
- 第 2 の引数 SolutionName が空の文字列の場合、現在のサーバーが使用されています。
- 第 2 の引数 SolutionName は (i) ソリューションがサーバー上で実行される場合、(サーバー上に) デプロイされたパスを生成します。(ii) シミュレーション用のファイルパスを生成します。
- 第 3 の引数 InputParameters は、**mt-run-solution-url-parameters** という名称の<%MT%>-特有のXPath 拡張関数を使用しクエリのパラメータの値のペアをエンコードします。(クエリのパラメーターをエンコードする) **mt-run-solution-url-parameters** 関数と(URL 全体を生成する) **mt-run-solution-url** 関数を混同しないよう注意してください。

他のデザインコンポーネント内のハイパーリンク クエリパラメータの値を使用する

コントロールイベントまたはページイベントに関連付けられたハイパーリンクによりトリガーしてソリューションが開かれる場合、ハイパーリンクのURL 内のパラメーターの値はソリューションにパスされ、ターゲットソリューション内の他のデザインコンポーネントで使用することができます。ターゲットソリューションの **$MT_InputParameters** グローバル変数内で文字列の値のシーケンスとして値は保管されます。**$MT_InputParameters** シーケンス内の文字列の値の順序は、URL のクエリパラメーターを生成する**mt-run-solution-url-parameters** 関数のために送信されたシーケンスと同じです。**$MT_InputParameters** 内の文字列の順序は、既知のため、位置述語を使用して、各文字列はXPath 式内でアクセスすることができます。例えば **$MT_InputParameters[1]** は、シーケンス内の最初の文字列の値を返し、**$MT_InputParameters[2]** は、第 2 文字列の値を返します。

エンドユーザーが送信する電子メールからソリューションにリンクする

電子メールの送信 アクションにより、クライアントおよびサーバーから電子メールを送信することができます。電子メールが HTML として送信される場合、メールのメッセージにハイパーリンクを追加することができます。リンクは MobileTogether ソリューションを開くことができます。電子メールのメッセージにリンクを追加する場合、メッセージオプションの XPath 式内の **mt-html-anchor** 関数を使用します (*下のスクリーンショットを参照*)。



mt-html-anchor 関数は以下の 2 つの引数を必要とします: **LinkText** と **TargetURL**。関数はこれらの 2 つの引数を使用して HTML ハイパーリンク要素を作成します: `<a href="TargetURL">LinkText</a>`

サンプル:

```
mt-html-anchor('Unregister from mailing list', mt-run-solution-url('', '/
public/unregister', ''))
```

は、次のパターンの HTML コードフラグメントを作成します:

```
<a href="LinkTo unregister.mtd">Unregister from mailing list</a>
```

mt-run-solution-url 関数は、ソリューションへリンクする URL を生成します。この URL はハイパーリンクの `href` 属性の値として保管されます。(`mobiletogether:// スキーム`の使用) この URL は、ハイパーリンクの `href` 属性の値として保管されます。

メモ `mt-run-solution-url` 関数を使用してリンクが作成される場合、モバイルデバイスの電子メールアプリからソリューションを開くことを有効化する (`http:// スキーム`ではなく) `mobiletogether:// スキーム` を使用して作成されます。 しかし、電子メールが Web クライアントで開かれた場合、ソリューションを開くリンクは `http:// スキーム`を使用する必要があります。この場合、ですから、`http://` リンクは手動で作成される必要があります。`mt-run-solution-url` 関数の使用は奨励されません。

メモ Web クライアントの場合、サーバー上のクライアントに直接移動するリンクを作成することができます。例えば、`http://localhost:8085/run?d=/public/BizBudget`。サーバー上のソリューションのコンテナーが匿名のアクセスを許可する場合、エンドユーザーはサーバーにログインすることなく、直接ソリューションを使用することができます。サーバーのアクセスレベルの設定に関する上は MobileTogether Server ユーザーマニュアルを参照してください。

チャプター 12

XPath/ XQuery: 式、関数、変数

12 XPath/ XQuery: 式、関数、変数

このセクションは以下のセクションで構成されています:

- XPath/XQuery 式と関数
- グローバル変数

メモ Altova の一般的な XPath 拡張関数ライブラリの機能に関しての詳細は、Altova 拡張関数のセクションを参照してください。(これらの拡張関数は、MobileTogether を含むすべての Altova 製品で使用することができます。)

(C) 2018 Altova GmbH

12.1 XPath/ XQuery 式と関数

このセクションはMobileTogether Designer のXPath/XQuery 機能と1 つにまとめられます: MobileTogether Designer が各プロジェクトで使用することのできる、[XPath/XQuery 条件式の編集] ダイアログ、拡張関数およびユーザー定義関数。

- [XPath/XQuery 条件式の編集] ダイアログ
- MobileTogether 拡張関数
- ユーザー定義 XPath/XQuery 関数
- XPath/XQuery についてよくある質問

XPath/ XQuery 条件式の編集ダイアログ

XPath 式の編集ダイアログ(*下のスクリーンショット*) は、MobileTogether 機能のためのXPath 式を作成および編集するために使用されます。例えば、多くのダイアログ内のファイルパスはXPath 式により構成されることができ、これによりファイルパスの動的な構成が許可され、すべてのページソース内でパスがソードコンテンツをベースにすることができます。

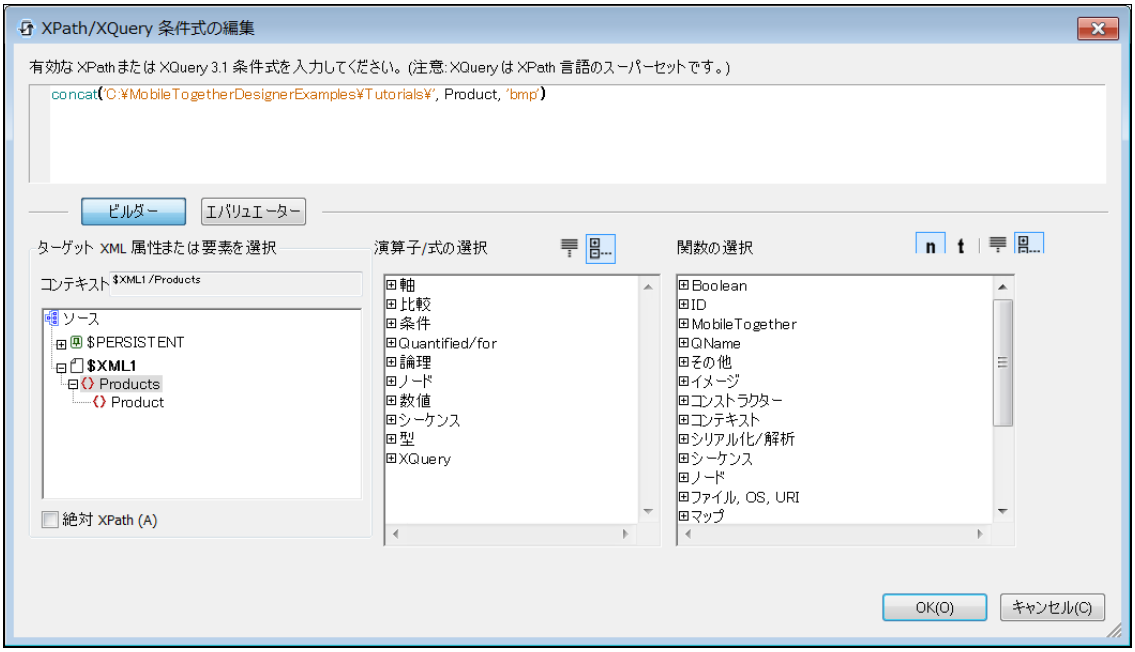
XPath 式の編集ダイアログは、データソースのツリーとXPath/XQuery 3.1 の演算子と関数のライブラリを提供することにより、有効なXPath/XQuery 3.1 式の作成をサポートします (*下のスクリーンショット参照*)。ダイアログには次の 2 つのモードがあります: (i) XPath 式を作成するためのビルダーモードと(ii) 現在編集されているXPath 式の結果をチェックするためのエバリュエーターモード。2つのモードを各ボタン(**ビルダー**と**エバリュエーター**) をクリックすることにより切り替えることができます。

XPath 式の編集が完了すると**[OK]** をクリックします。

メモ XPath 式の編集ダイアログからコピーされたテキストは、スタイル & プロパティペイン内にXPath として貼り付けられます。

ビルダー モード

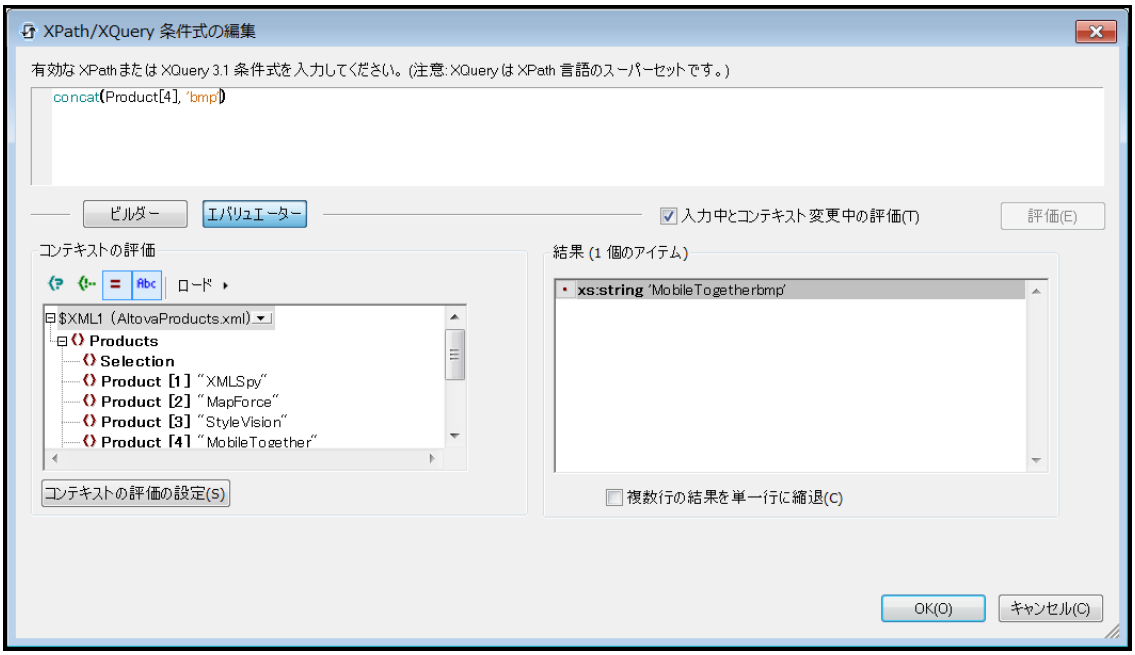
ビルダー モードで、(i) キーボードで式テキスト ボックスにXPath 式を入力します、または(ii) ビルダー モードのヘルパーを使用して、ノード、演算子、関数を各リストからダブルクリックして入力することにより(*下のスクリーンショット参照*) XPath 式を素早く正確に作成することができます。式テキスト ボックスに入力された式がエラーを含む場合、その式に赤の下線が表示され、問題を通知します。ビルダー モードについては、XPath 式ビルダーのセクションで詳しく説明されています。



エバリュエーター モード

エバリュエーター モードでは、ダイアログ右の結果ペイン(*下のスクリーンショット参照*) で現在入力されたXPath 式の結果

を評価することができます。評価コンテキストペインは、現在割り当てられている作業 XML ドキュメントの構造とコンテンツを表示します。XPath 式の編集ダイアログのエバリュエーターモードの詳細は XPath 式エバリュエーターのセクションで説明されています。



MobileTogether 拡張関数

MobileTogether デザイン内での使用のために特別に作成された、以下のXPath 拡張関数を、デザイン内の任意の場所で使用することができます。XPath のデフォルトの名前空間はこれらの関数を呼び出すために使用されます。

XPath 関数の戻り値の更新
XPath 関数は、XPath 式が評価された場合のみ評価されます。これは通常 XPath 式を含むアクションが実行された場合、または、データが変更されXPath 式が評価される場合に発生します。

例えば、**mt-audio-is-playing** 関数を含むXPath 式をサンプルとします。この関数は**true** または**false**. を返します。式がある特定の時間に評価されると、例えば戻り値が(オーディオが再生されているため) **true** の場合。この値がソリューション内で表示されると、オーディオの再生が停止されても、この値は自動的に変更されません。設定をこのようにするには、関数がもう一度呼び出され、新しい値が表示内の値を更新する必要があります。

このような値が更新される理由は使用されている特定のデザインメカニズムにより異なります。このような値を更新する方法のひとつとしては、表示の更新 アクションを使用したページの更新時 イベントのタイマーなどが挙げられます。

メモ Altova の一般的なXPath 拡張関数ライブラリの機能に関しての詳細は、Altova 拡張関数のセクションを参照してください。(通常の拡張関数は、MobileTogether を含むすべてのAltova 製品で使用することができます。)

mt-audio-get-current-position

mt-audio-get-current-position(ChannelNumber *as xs:integer***)** as **xs:decimal**
ターゲットオーディオファイルが再生されているチャンネル番号を引数として取ります。現在のオーディオ再生のポジションを小数で表した数値(秒)で返します。現在のオーディオ再生のポジションを引数として秒で返します。現在の再生箇所に関する情報は再生が開始してのみ、取得することができます。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
```
mt-audio-get-current-position()
```

mt-audio-get-duration

mt-audio-get-duration(ChannelNumber *as xs:integer***)** as **xs:decimal**
ターゲットオーディオファイルが再生されているチャンネル番号を引数として取ります。オーディオ再生の長さを小数で表した数値(秒)で返します。再生の長さに関する情報は再生が開始してのみ、取得することができます。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
```
mt-audio-get-duration()
```

mt-audio-is-playing

mt-audio-is-playing(ChannelNumber *as xs:integer***)** as **xs:boolean**
テストするチャンネルの番号を引数として取ります。そのチャンネルでオーディオファイルが再生されている場合は`true()` を返します、それ以外の場合は`false()` を返します。

使用方法
```
mt-audio-is-playing()
```

mt-audio-is-recording

説明
デバイスがオーディオを録音している場合は`true()`を返します、それ以外の場合は`false()` を返します。

使用方法
`mt-audio-is-recording()`

mt-available-languages

説明
ローカライズ ダイアログで定義されている言語を返します。

ローカライズダイアログ内では、ISO 言語コードにより各言語 (例: **en-US**) 及び名前 (例: **English**)が指定されています。デフォルトの言語の言語コードは常に空の文字列ですが、名前には文字列を与えることが出来ます。

関数は各言語を2つの文字列の配列として返します。例: **["en-US", "English"]**。複数の言語は配列アイテムのシーケンスとして返されます。例: **(["en-US", "English"], ["de-DE", "German (DE)"])**。シーケンス内の最初の配列は常にデザインのデフォルトの言語を返します。ですから、言語名は(ローカライズダイアログ内で)デフォルトの言語のために指定されており、最初の配列アイテムの両方の文字列は空になります。それ以外の場合、最初の配列アイテムは空の文字列とデフォルトの言語に与えられた名前になります (例: **["", "MyDefaultLanguage"]**)。

返されるシーケンスはスペースで区切られた文字列として表示されます。

使用方法
mt-available-languages() は次の値を表示する可能性があります: **en-US English**
mt-available-languages() は次の値を表示する可能性があります: **en-US English de-DE German (DE)**
mt-available-languages() は次の値を表示する可能性があります: **MyDefaultLanguage en-US English**

mt-base64-to-hexBinary

mt-base64-to-hexBinary(Base64Image *を xs:base64Binary として* **)** を**xs:string** として
この関数は、Base64 エンコードイメージをhexBinary 文字列に変換します。 `Base64Image` 引数は `base64Binary` 内でエンコードされるテキストである必要があります。このようなテキストを与えるデータソースノードを送信することができます。

使用法
mt-base64-to-hexBinary`($XML1/Element1/@image)` Base64 イメージを、hexBinay に変換します。

mt-cache-update-dateTime

説明
ページソースのキャッシュが更新されると時刻を返します。ページソースがキャッシュされていない場合、空のシーケンスが返されます。

使用法
`mt-cache-update-dateTime($XML1)`

mt-change-image-colors

mt-change-image-colors(Base64Image を *xs:base64Binary* **として、SourceColors を** *xs:string+* **として、TargetColors を** *xs:string+* **として、Quality を** *xs:integer* **として)** を**xs:base64Binary** として
関数は、Base64-エンコードイメージを最初の引数として取り、ソースの色 引数として与えられたこれらのイメージの色を関連するターゲットの色 として変換し、変換されたイメージをBase64-エンコードイメージとして返します。

- `Base64Image` は、`base64Binary` でエンコードされる必要があります。そのようなテキストを返すノードを与えることができます。
- *ソースの色* と*ターゲットの色* は、1つまたは複数の文字列アイテムのシーケンスである必要があります。両方のシーケンスのアイテムの数は同等である必要があります。
- *クオリティ*は、`1` から`100` の整数です。クオリティのレベルを指定し、`100` が最も高い数値です。

サンプル

- **`mt-change-image-colors`**`(Base64ImageNode, ('#000000'), ('#666666'), 90 )` は、黒 (`#000000`) が灰色 (`#666666`) に変えられたBase64 イメージを返します。
- **`mt-change-image-colors`**`(xs:base64Binary(Base64ImageNode), ('#000000', '#FF0000'), ('#666666', 'blue'), 90 )` は、黒 (`#000000`) が灰色 (`#666666`) に、赤 (`#FF0000`) が青に変えられたBase64 イメージを返します。

mt-client-ip-address

`mt-client-ip-address()` を**`xs:string`** として

サーバーからみられるクライアントのIP アドレスを返します。シミュレーションに関しては、MobileTogether Designer のオプションダイアログの内のシミュレーションタブのIP アドレスを設定します(**「ツール | オプション」**)。

説明

- **`mt-client-ip-address`**`()` は、クライアントIP アドレスを返します。シミュレーション内では、MobileTogether Designerのオプションタブ内で設定された値を返します。

mt-connected-via-lan

説明

モバイルデバイスがLAN を使用して接続されている場合、`true()`を返します、それ以外の場合は`false()` を返します。

使用方法

`mt-connected-via-lan()`

mt-connected-via-wifi

説明

WiFi によりモバイルデバイスが接続されている場合 `true()` を、それ以外の場合は、`false()` を返します。

使用法

`mt-connected-via-wifi()`

mt-control-width

`mt-control-width(Text` *`as xs:string*`*`, Parameters`** *`as map(*)`*`)`**を**`xs:integer?`** として

テキスト 文字列がコントロールの表示テキストの場合、最小の幅をピクセルで返します。**テキスト** 引数は、コントロールに表示されるテキストです。**パラメーター** 引数は、コントロールのプロパティを定義する**キーと値** マップです。下のリストを参照してください。返された整数は、送信された**テキスト** 文字列が**パラメーター** 引数内で指定されたプロパティを使用して表示される、ピクセルで示された最小の幅です。この値は、コントロールが表示される列内のテーブルの幅など、コントロールに関連した他のプロパティを計算と指定するために使用されます。

メモ この機能はWeb クライアントでの表示には使用することができません。

メモ この関数は、以下の場合でのみ使用することができます(i) デザインアクション、と(ii) ページソースツリーノード

のロード時に存在することを確認 (*XPath 値*) オプション。これらはスタイルプロパティの XPath 式では使用することは許可されていません。

パラメーター 引数として与えられた ***キーと値*** マップは以下の通りです。マップ内のキーと値のペアの順序は固定されていません。キーと値ペアにより指定されているとおりコントロールプロパティが提出されない場合、プロパティのデフォルトの値が (それぞれ ラベル または ボタン コントロールのために) 使用されます。この結果、コントロールの種類 パラメーターのみが必須となります。

- "コントロールの種類" : "ラベル"|"ボタン"
- "テキストのサイズ" : "小"|"中"|"大"
- "ユニット" : "px"|"" デフォルトは "px" です。
- "太字" : "true"|"false"
- "斜体" : "true"|"false"
- "下線" : "true"|"false"
- "ボタンの外観" : "+"|"-"|">"|"共有"

サンプル

- `mt-control-width("Send", map{"Control Kind" : "Button", "Text Size" : "medium", "Unit" : "", "Bold Text" : true(), "Italic Text" : false(), "Underline Text" : false(), "Button Look" : "+"})`

mt-db-any-changed-fields

説明
行要素が新しい、変更された、または削除された列を含む場合 `true` を返します。フィールドが変更されていない場合は `false` を返します。関数は、指定された DB 行に変更が加えられたかテストします。

使用法
`mt-db-any-changed-fields($DB1/DB/RowSet/Row[3])`

mt-db-any-changed-rows

説明
(DB を表示する) `$DB` 変数が新しい、変更されたまたは削除された行を含む場合を `true` 返します。DB が変更されていない場合は `false` を返します。関数は DB に変更が加えられたかをテストします。

使用法
`mt-db-any-changed-rows($DB1)`

mt-db-deleted-original-fields

説明
元の行要素からフィールド属性を返します:

- *新しい行:* フィールド属性ではありません。新しい行のために関数が呼び出された場合、空のリストを返します。
- *変更された行:* 削除されたフィールド属性。変更された行のために関数が呼び出された場合、行要素の下にリストされていない対応する `OriginalRow` 要素からこれらのフィールドを返します。
- *削除された元の行:* すべてのフィールド属性。関数が (対応する行要素が削除された) `OriginalRow` のために呼び出された場合、すべてのフィールドを返します。

使用法
`mt-db-deleted-original-fields($DB1/DB/RowSet/Row[1])`

mt-db-deleted-original-rows

説明
行要素の無いすべての`OriginalRow` 要素を返します。関数はデータベースから読み込まれたデータへの変更に使用することができます。ページソースで`OriginalRowSet` が有効化された場合のみ機能します！

使用法
`mt-db-deleted-original-rows($DB1)`

mt-db-modified-fields

説明
指定された行要素の属性を返します:

- *新しい行:* すべてのフィールド属性。関数が新しい行のために呼び出された場合すべてのフィールドを返します。
- *削除された元の行:* すべてのフィールド属性。関数が(対応する行要素が削除された) `OriginalRow` のために呼び出された場合、すべてのフィールドを返します。
- *変更された行:* 変更されたフィールド属性。関数が変更された行のために呼び出されると、対応する `OriginalRow` 要素に保管された値と異なる値を持つフィールドを返します。

使用法
`mt-db-modified-fields($DB1/DB/RowSet/Row[3])`

mt-db-modified-rows

説明
変更されたすべての行要素の属性のリストを返します。関数はデータベースから読み込まれたデータへの変更に使用することができます。ページソースで`OriginalRowSet` が有効化された場合のみ機能します！

使用法
`mt-db-modified-rows($DB1)`

mt-db-new-fields

説明
指定された行要素の新しいフィールド属性を返します:

- *新しい行:* すべてのフィールド属性。関数が新しい行のために呼び出された場合、すべてのフィールドを返します。
- *変更された行:* 新しいフィールド属性。関数が変更された行のために呼び出されると、対応する `OriginalRow` 要素のためにリストされていないフィールドを返します。
- *元の行:* 空のリスト。関数が(対応する行要素が削除された) `OriginalRow` 要素のために呼び出された場合、空のリストを返します。

使用法
`mt-db-new-fields($DB1/DB/RowSet/Row[1])`

mt-db-new-rows

説明
`OriginalRowSet` ではなく`RowSet` 要素の下にリストされている、行要素のリストを返します。関数はDB から読み込まれるから読み込まれたデータへ加えられた変更を判別します。ページソースで`OriginalRowSet` が有効化された場合のみ機能します！

使用法
`mt-db-new-rows($DB1)`

mt-email-attachment

mt-email-attachment(Filename *xs:string* を*xs:string* として, **Content** を*item()* として **, ContentType** を*xs:string* として **) array(*)** として
Content 引数により与えられたXML、Base64 または、テキストをコンテンツを電子メールの添付として準備します。コンテンツが XML、Base64 または、テキスト として解析されるかは、値として、XML、Base64、または テキスト を受け入れる**ContentType** 引数により決定されます。添付に関連したファイル名が**Filename** 引数により与えられます。

メモ mt-email-attachment は、電子メールの送信アクションの動的な添付のオプションを使用する際の必要条件です。

メモ HTML として送信される電子メールに関しては、電子メールの内容が正確な HTML 、つまり、**html** 要素から始まる、HTML である必要があります。有効な内容は次のXPath/XQuery 構成により開始することができます:
element html { element body { "Test" } }

メモ 添付はクAndroid およびiOS ライアントのみで使用することができます。

サンプル

- **mt-email-attachment**('MTNewFeatures.txt', $XML2/Releases/Release[@date='2015-04-15']/Features, 'XML')は 機能ノード を返します。
- **mt-email-attachment**('MTLogo.jpg', $XML4/Images/Image[@name='MTLogo'], 'Base64')は イメージファイル を返します。

mt-external-error-code

説明
最後のDB 、ロード、保存アクションのエラーテキストを返します。 「404 Web ページが見つかりません」などのOS または データベースのネイティブなエラーコードを返します。

使用法
mt-external-error-code()

mt-external-error-text

説明
最後のDB 、ロード、保存アクションのエラーテキストを返します。エラーテキストは返されたエラーコードと共に提供されるテキストです。

使用法
mt-external-error-text()

mt-extract-file-extension

mt-extract-file-extension(FilePath *を xs:string として*) を**xs:string?** として
FilePath 引数として与えられたファイルパス内のファイルの(xml などの)拡張子を返します。 **ファイルパス** 引数内に与えられた文字列は、絶対または相対パスの構文パターンを有する必要があります。**ファイルパス** 引数として**mt-last-file-path** 関数を使用することができます。

サンプル

- **mt-extract-file-extension**(/storage/emulated/0/Download/Myfile.xml) は、**'xml'** を返します。
- **mt-extract-file-extension**(mt-last-file-path()) は、mt-last-file-path() により返されたファイルパス内のファイル名のファイル名の拡張子を返します。

mt-extract-file-name

mt-extract-file-name(FilePath を*xs:string* として**)** を**xs:string?** として**?**

FilePath 引数として与えられたファイルパス内のファイルの名前(ファイル型拡張子の前の部分)を返します。**ファイルパス**引数内に与えられた文字列は絶対または相対的な構文パターンを有する必要があります。 **mt-last-file-path** 関数を**ファイルパス**引数として使用できることに注意してください。

サンプル

- **mt-extract-file-name**(/storage/emulated/0/Download/MyFile.xml) は、**'MyFile'** を返します。
- **mt**(mt-last-file-path()) は、mt-last-file-path() により返されたファイルパス内のファイル名の一部である、mt-last-file-path() により返されたファイルパス内のファイル名のみのファイル名の拡張子を返します。前のセクションを参照してください。

mt-font-height

mt-font-height(TextSize を*xs:string** として、**Unit** を*xs:string* として **)** を **xs:string?** として

テキストサイズ引数として提出された文字のサイズの高さをピクセルで返します。任意の**ユニット**引数は、必要とされる返される数値の高さでユニットを指定します。現時点では、ピクセルの高さのみが返されます。返された値は同じコントロールまたは他の関連するコントロールの値を計算するために使用することができます。**テキストサイズ**引数に許可される値は、以下の通りです: 一番小さい|小さい|中|大きい|一番大きい。**ユニット**引数に許可される値は以下の通りです: "px" と空の文字列。

メモ この関数は、以下の場合でのみ使用することができます(i) デザインアクション、と(ii) ページソースツリーノードのロード時に存在することを確認 (*XPath 値*) オプション。これらはスタイルプロパティの XPath 式では使用することは許可されていません。

サンプル

- **mt-font-height**("small", "px") は、33 を返します(値はクライアントにより異なります)
- **mt-font-height**("smallest", "") は、27 を返します(値はクライアントにより異なります)

mt-format-number

mt-format-number(Number を*xs:numeric* として、**PictureString** を*xs:string* として **)** を **xs:string** として

数値を第 1 の引数とし、第 2 (PictureString) の引数に従いフォーマットし、フォーマットされた数値を文字列として返します。これは読みにくい数値を読みやすい書式にフォーマットする際に役に立ちます。ピクチャ文字列は、通貨記号などの文字も含むことができ、フォーマットされた出力内に使用することができます。その位置の入力数値に数値がない場合にゼロを数値の位置に入力する場合はゼロをピクチャ文字列の数値の位置に使用します(*下のサンプルを参照*)。ゼロ(または他の文字) の強制を希望しない場合は、番号記号 (#) を使用します。

小数識別子の前の数値は決して短縮化されません。数値の小数部分 (小数識別子の右側) および単位数値(小数識別子の左側の最初の数値)は、ピクチャ文字列の数値の部分が入力数値の小数の位置より先短い場合、四捨五入されます。

メモ モバイルデバイス上のフォーマットされた出力内のグループ化識別子および小数識別子はモバイルデバイスで使用される言語のものです。

サンプル

- **mt-format-number**(12.3, '$#0.00') は$12.30 を返します。
- **mt-format-number**(12.3, '$00.00') は$12.30 を返します。
- **mt-format-number**(12.3, '$0,000.00') は$0,012.30 を返します。
- **mt-format-number**(12.3, '$#,000.00') は$012.30 を返します。

- **mt-format-number**(1234.5, '$#,##0.00') は$1,234.50 を返します。
- **mt-format-number**(1234.5, '$#0.00') は$1234.50 を返します。
- **mt-format-number**(123.4, '$0') は$123 を返します。
- **mt-format-number**(1234.5, '$0') は$1235 を返します。
- **mt-format-number**(1234.54, '$0.0') は$1234.5 を返します。
- **mt-format-number**(1234.55, '$0.0') は$1234.6 を返します。

mt-geolocation-started

説明
ソリューションが位置情報のトラッキングを開始すると、true() を返します。それ以外の場合、false() を返します。

使用法
mt-geolocation-started()

mt-has-serveraccess

説明
サーバーへのアクセスが可能な場合はtrue を返します。その他の場合はfalse を返します。この関数は、MobileTogether Server への接続が関数のタイムアウトの秒数 引数により定義されている秒数内に確立することができるかチェックします。

使用法
mt-has-serveraccess(TimeoutSeconds を整数として)

mt-hexBinary-to-base64

mt-hexBinary-to-base64(HexBinary *を xs:string として* **)** を**xs:base64Binary** として
この関数は、hexBinary 文字列を(通常はイメージである) Base64 エンコード文字列に変換します。必要とされるhexBinary 文字列を与えるノードは、HexBinary 引数として送信することができます。

使用法
mt-hexBinary-to-base64('48656C6C6F20576F726C64') は、Base64 文字列 'SGVsbG8gV29ybGQ=' を返します。

mt-hexBinary-to-string

mt-hexBinary-to-string(HexBinary *as xs:string,* **Encoding** *as xs:string***)** as **xs:string**
この関数は、hexBinary 文字列を**Encoding** 引数内のエンコード名を使用してエンコードされるテキスト文字列に変換します。hexBinary 文字列を与えるノードは、**HexBinary** 引数。として送信されることができます。空の文字列がEncoding 引数として与えられると、デフォルトの**'UTF-8'** エンコード内で結果テキスト文字列が生成されます。Encoding 引数が指定されていない場合、エラーが発生します。

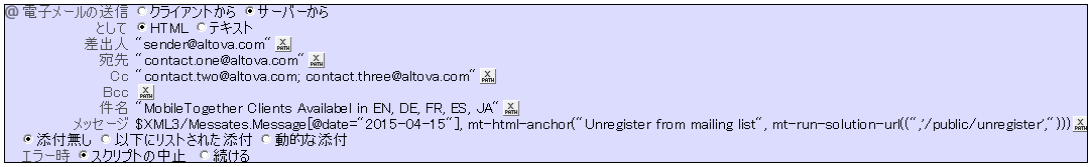
サンプル

- **mt-hexBinary-to-string**('48656C6C6F20576F726C64', 'ASCII') は、'Hello World' を返します。
- **mt-hexBinary-to-string**('48656C6C6F20576F726C64', '') は、'Hello World' を返します。
- **mt-hexBinary-to-string**('48656C6C6F20576F726C64') は、エラーを返します

mt-html-anchor

説明

mt-html-anchor 関数は、以下の2 つの引数を必要とします: **LinkText** と**TargetURL**。この関数はこれら2 つの引数を使用して、HTML ハイパーリンク要素を作成します: **LinkText**。リンクは、電子メールの送信アクションを使用してHTML として送信される電子メールに挿入することができます。リンクはインターネットのページ、または、MobileTogether ソリューションを開くことができます。電子メールの本文にリンクを追加する場合は、本文オプションのXPath 式内の**mt-html-anchor** 関数を使用します (*下のスクリーンショット参照*)。



サンプル

- **mt-html-anchor('Unregister from mailing list', 'http://www.altova.com/unregister.html'))**は **Unregister from mailing list** を返します。
- **mt-html-anchor('Unregister from mailing list', mt-run-solution-url('', '/public/unregister', ''))** は**Unregister from mailing list** を返します。

mt-image-width-and-height

mt-image-width-and-height(Image *を base64encoded-image として* **)** を **xs:integer+** として

引数 `Image` は、ディメンションを検索するイメージのBase64 エンコードです。引数は`xs:base64Binary` の型である必要があります。通常、引数は、Base64 エンコードされたデータを含むノードを検索します。関数は2 つの整数のシーケンスを返します: (i) 幅、(ii) 高さ.

説明

- **mt-image-width-and-height**`($XML1/images/png)` は、`364 76` を返します。

mt-invert-color

mt-invert-color(Color を*xs:string* として **)**を**xs:string** として

色 引数 は、RGB (16 進数フォーマット) カラーコードです。 例えば、**"#00FFFF"** 。コード内の各色のコンポーネント(R、G、およびB) が反転され、新しいカラーコードが返されます。

サンプル

- **mt-invert-color**`('#000000')` は、を返します `'#FFFFFF'` を返します。
- **mt-invert-color**`('#00FFFF')` は、`'#FF0000'` を返します。
- **mt-invert-color**`('#AA0000')` は、`'#55FFFF'` を返します。
- **mt-invert-color**`('#AA33BB')` は、`'#55CC44'` を返します。
- **mt-invert-color**`('#34A6D2')` は、`'#CB592D'` を返します。

mt-last-file-path

mt-last-file-path() を**xs:string?** として

以下のアクション内で最後に使用された(ロードされたまたは保存された) クライアントファイルのフルパスを返します。(例えば、Android では : `/storage/emulated/0/Download/MyFile.xml`) : オーディオの録音、ファイ

ルのロード保存、バイナリのロード保存、イメージのロード。Android 以外のオペレーティングシステムはこの関数をサポートしない場合があります。

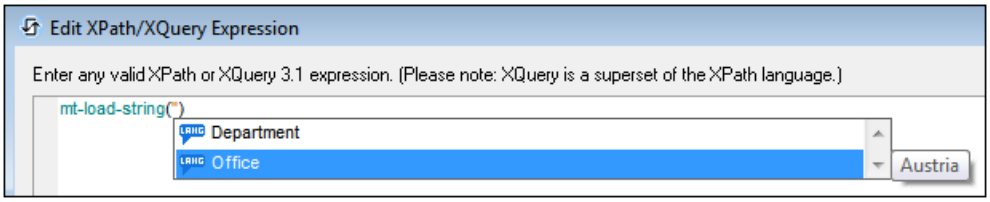
使用法
`mt-last-file-path()`

mt-load-string

説明
`StringName` 引数により、識別されたカスタム文字列を返します。各カスタム文字列 はローカライズ ダイアログ内で定義された文字列のプールの一部です。各 `StringName` は、文字列プール内で複数のローカライズされた文字列と関連しています。選択されたローカライズされた文字列の言語は、モバイルデバイスの言語、またはシミュレーションの言語です。

使用法
`mt-load-string('StringName')`

`mt-load-string` 関数が[XPath/XQuery 式の編集] ダイアログ内に入力されると、すべての使用可能なカスタム文字列がポップアップで表示されます(*下のスクリーンショット参照*) 。区切りアポストロフィ内、または `'StringName'` の引用にカーソルをポイントして、**[Ctrl+Space キー]** をクリックします。



[上へ] および **[下へ]** キーボードボタンを使用して、リストを循環します。します。選択されたカスタム文字列の値は、ポップアップの横に表示されます (*下のスクリーンショット参照*)。表示された値のローカライズ言語は MobileTogether Designer で現在選択されているシミュレーションする言語 です。XPath 式内のカスタム文字列の名前を入力するには、文字列を選択、またはカスタム文字列 リストを循環し文字列を検索し**[Enter キー]** を押します。

mt-localized-string-name

`mt-localized-string-name(Text` を*`xs:string`* として **`)`** を**`xs:string*`** として
`mt-localized-string-name(Text` を*`xs:string`* として、 **`Lang`** を*`xs:string`* として **`)`** を **`xs:string*`** として
関数は、デフォルトの言語内またはローカライズされた言語の(最初の) 引数をテキスト文字列の値として、テキストの値として与えられたテキスト文字列の値を持つコントロールまたは文字列の名前を返します。詳細に関してはローカライズおよび **[プロジェクト] ローカライズ]** を参照してください。関数には 2 つの署名があります。第 2 の署名には、テキスト文字列の言語は第 2 の引数です (`Lang`)。`Lang` 引数はローカライズされた言語の名前に一致する必要があります。`Lang` が指定されている場合、ローカライズされた言語の文字列のみを対象に、テキスト引数内で提出されたテキスト文字列に一致する文字列が検索されます。

サンプル

- **`mt-localized-string-name`**`('City')` は `'CityButton'` を返します。
- **`mt-localized-string-name`**`('Stadt', 'DE')` は `'CityButton'` を返します。
- **`mt-localized-string-name`**`('Stadt')` は `'CityButton'` を返します。
- **`mt-localized-string-name`**`('Stadt', 'ES')` は `''` を返します。
- **`mt-localized-string-name`**`('Stadt', 'German')` は `''` を返します。
- **`mt-localized-string-name`**`('Ciudad', 'ES')` は `'CityButton'` を返します。

上のサンプルは、**CityButton** という名前のボタンコントロールの文字列のためのサンプルです。文字列のデフォルトの言語は、英語で、**DE** および**ES** という名前の言語にローカライズされています。

- mt-nfc-started

説明
ソリューションがNFC を開始すると、 `true()`を返します。それ以外の場合、`false()` を返します。

使用法
`mt-nfc-started()`

- mt-refresh-userroles

説明
現在使用可能なユーザーロールをサーバーからロードします。この関数は、グローバル変数 `MT_UserRoles` からクエリすることのできるサーバーからユーザーロールを更新します。

使用法
`mt-refresh-userroles()`

- mt-reload-dateTime

説明
ページソースが再ロードされた時刻を返します。ロードされていない場合は、空のシーケンスを返します。

使用法
`mt-reload-dateTime($XML1)`

- mt-run-appstoreapp-url

`mt-run-appstoreapp-url(Scheme?` を*`xs:string`* として、**`Host?`** を*`xs:string`* として、**`InputParameters?`** を*`xs:string`* として) を**`xs:string?`** として
3つの与えられた引数からMobileTogether AppStore App のURL を生成します。URL は、(通常電子メールで送信される) アプリをハイパーリンクから開きます ターゲットURL のハイパーリンクには、以下のフォーマットが必要です : **`<url-scheme>://<url-host>`** 。スキームの情報はアプリのマニフェストファイルに保管されます。デバイスには、このアプリはこのスキームを使用してURL を開くように表示されます。このスキームを持つURL のリンクがタップされると、デバイスはこのAppStore App を開きます。詳細に関しては、AppStore App を参照してください。

- `Scheme`: アプリと関連する一意のスキーム名。スキームは、生成されるアプリのプログラムコードに割り当てられます ([プログラムコード生成ウィザードのスクリーン1] 内)。
- `Host`: アプリと関連する一意のホスト名。ホストは、生成されるアプリのプログラムコードに割り当てられます ([プログラムコード生成ウィザードのスクリーン1] 内)。
- `InputParameters`: は、関数 `mt-run-solution-url-parameters` を入力として取ります。関数の引数は、クエリのパラメーターの値を与える文字列のシーケンスです。 `mt-run-solution-url-parameters` 関数は、正確にエンコードされクエリ文字列のエンコードのルールに従いパーセントエスケープされたURL クエリ文字列のパラメーター(名前と値) を含む文字列を返します。結果文字列内のパラメーター名は、関数 (例: `in1, in2 ... inN`) により自動的に生成され、関数の引数の文字列アイテムの値に、名前と値がインデックスの順序に従い組み合わされ、それぞれ割り当てられます。(また、`InputParameters` 引数は既にURL の一部のクエリ文字列のために文字列として与えられることができます(下の2番目のサンプルを参照してください)。)

`mt-run-appstoreapp-url` 関数は、ですから、MobileTogether AppStore App を開き、アプリがURL を使用して開かれた場合、アプリにパスされるクエリパラメータのあるまたはクエリパラメータ無しのURL を作成します。これらのパラメータの値は、**`$MT_InputParameters`** グローバル変数を使用して、他のデザインコンポーネント内でアクセスすることができます。

サンプル

- **mt-run-appstoreapp-url**('myappscheme', 'myfirstapp', '') は、URL **myappscheme://myfirstapp** を返します。モバイルデバイス上では、URL は、スキームとホストで識別されるAppStore App を開きます。URL には、クエリパラメーターはありません。
- **mt-run-solution-url**('myappscheme', 'myfirstapp', 'in1=value1&in2=value2%3FAndMoreValue2') は、URL **myappscheme://myfirstapp**を返します。モバイルデバイス上では、URL は、スキームとホストで識別されるAppStore App を開きます。InputParameters 引数は既にURL クエリ文字列としてエンコードされた文字列として提出されます。

mt-run-solution-url

mt-run-solution-url(ServerAddress? を*xs:string* として, **SolutionName?** を *xs:string* として, **InputParameters?** を*xs:string* として) を**xs:string?**として 3 つの提供された引数からソリューションのURL を生成します:

- ServerAddress: 実行するソリューションがデプロイされるMobileTogether Server のIP アドレスまたは名前を使用します。
- SolutionName: サーバー上のソリューションのデプロイされたパスを使用します。例えば、: /public/MySolution は/Public コンテナー内のMySolution.mtd ファイルをポイントします。
- InputParameters: mt-run-solution-url-parameters を入力として使用します。関数の引数は、クエリのパラメーターの値を与える文字列の値のシーケンスです。 mt-run-solution-url-parameters 関数は、正しくエンコードされた、パーセントを除いたクエリ文字列のエンコードのルールに従ったURL のクエリ文字列 のパラメーター(名前と値) を含む文字列を返します。 結果文字列内のパラメーター名は、自動的に関数に(inputparameter1, inputparameter2 ... inputparameterN) より生成され、それぞれ関数の引数の文字列アイテムから、インデックス順に組み合わされた名前と値により値を割り当てられます。(更に、InputParameters 引数は、既にURL の一部のクエリ文字列としてエンコードされている文字列として提供されます(下の4番目のサンプルを参照)。)

mt-run-solution-url 関数はですから、MobileTogether Server 上でソリューションにアクセスするクエリパラメーターを持つまたは持たないURL を作成します。クエリパラメーターは、ソリューションがURL を使用して開かれるとソリューションにパスされます。これらのパラメーターの値は、**$MT_InputParameters**グローバル変数を使用して、他のデザインコンポーネントでアクセスすることができます。

サンプル

- **mt-run-solution-url**('100.00.000.1', '/public/MyDesign', '') は、IP アドレス100.00.000.1. を持つMyDesign ソリューションをポイントする URL を返します。URL には、クエリパラメータはありません。
- **mt-run-solution-url**('', '/public/MyDesign', '') は、現在のサーバー上のMyDesign ソリューションをポイントするURL を返します。URL には、クエリパラメータはありません。
- **mt-run-solution-url**('', '', mt-run-solution-url-parameters(('2015', 'USA', 'true'))) は、現在のサーバー上の現在のソリューションをポイントするURL を返します。**mt-run-solution-url-parameters** 関数の引数は、クエリのパラメーターの値である文字列の値のシーケンスです。最初の文字列は、最初のパラメータの値です。第 2 の文字列は、第 2 のパラメーターの値です。 **mt-run-solution-url-parameters** 関数は関数は、正しくエンコードされ、パーセントを除いたURL クエリ文字列のエンコードのルールに従った文字列を返します。
- **mt-run-solution-url**('', '', 'inputparameter1=value1&inputparameter2=value2%3FAndMoreValue2') は、現在のサーバー上の現在のソリューションをポイントするURL を返します。InputParameters 引数は、URL クエリ文字列として既にエンコードされた文字列として提出されます。

以下の点に注意してください:

- 最初の引数 ServerAddress が空の文字列の場合、現在のサーバーが使用されています。
- 最初の引数 ServerAddress がクライアントに保管されているサーバー情報を検索するために使用されています。サーバー名と関連付けられているポート番号、ユーザー名、ユーザーパスワードはサーバーに接続するために使用されています。クライアントが認識しないサーバー名を持つURL が生成された場合、URL は作動しません。
- 第 2 の引数 SolutionName が空の文字列の場合、現在のサーバーが使用されています。
- 第 2 の引数 SolutionName は (i) ソリューションがサーバー上で実行される場合、(サーバー上に) デプロイされたパスを生成します。(ii) シミュレーション用のファイルパスを生成します。
- 第 3 の引数 InputParameters は、**mt-run-solution-url-parameters** という名称の<%MT%>-特有のXPath 拡張関数を使用しクエリのパラメータの値のペアをエンコードします。(クエリのパラメーターをエンコードする) **mt-run-solution-url-parameters** 関数と(URL 全体を生成する) **mt-run-solution-url** 関数を混同しないよう注意してください。

mt-run-solution-url-parameters

mt-run-solution-url-parameters((Parameters* を*xs:string* として) as **xs:string?**

mt-run-solution-url-parameters 関数は**mt-run-solution-url** 関数の3 番目の関数として使用されることを目的としています。**mt-run-solution-url-parameters** 関数の引数は、文字列の値のシーケンスです。これらはクエリ文字列のパラメーターの値で**mt-run-solution-url** 関数により生成されます。mt-run-solution-url-parameters 関数は、正しくエンコードされた、パーセントを除いたクエリ文字列のエンコードのルールに従ったURL のクエリ文字列 のパラメーター(名前と値) を含む文字列を返します。結果文字列内のパラメーター名は、自動的に関数に(inputparameter1, inputparameter2 ... inputparameterN) より生成され、それぞれ関数の引数の文字列アイテムから、インデックス順に組み合わされた名前と値により値を割り当てられます。

メモ Parameters 文字列が2 重引用符を含む場合、これらを一重引用符に変更してください。この手順は、MobileTogether がパラメーター文字列を作成するために2 重引用符を使用するからです。XPath replace 関数を使用して、2 重引用符を一重引用符に変換することができます: **replace(<string>, '"', "'")**.

これらのパラメーターの値は**$MT_InputParameters** グローバル変数を使用してデザインコンポーネントの順序にアクセスすることができます。

サンプル

- **mt-run-solution-url-parameters**(('2015', 'USA', 'true')) は '&inputparameter1=2015&inputparameter2=USA&inputparameter3=true' を返します。

mt-server-config-url

mt-server-config-url(ServerSettings *as map*) as **xs:string?**

mt-server-config-url 関数は、マップを引数として取り、URL である文字列を返します。クライアントデバイスへのリンクとしてURL が送信された場合、そして、リンクがタップされた場合、クライアント上のサーバー設定 は、自動的に更新されます。URL は以下に類似したURL になります: **mobiletogether://mt/change-settings?settings=<json encoded settings>**

URL に含まれているJSON-エンコードサーバー設定は、**mt-server-config-url** 関数 の ServerSettings 引数 により提供されます。ServerSettings マップは以下に表示されています。関数の使用方法の例に関しては、MobileTogetherExamples/SimpleApps フォルダー内のサンプルファイル **ClientConfiguration.mtd** をテストしてください。

```
mt-server-config-url(
  map{
    "DelOthSrv": false(),    (: whether existing server list should be
deleted before import :)
    "DetView": true(),       (: whether the details view should be used or
```

```
the grid :)
    "Refresh": true(),      (: refresh solutions on start :)
    "RetToSln": true(),     (: Windows clients only :)
    "ActSrvURL": "",        (: the first server with this URL gets the
active one :)
    "Servers": array{
       map{
         "Name": "",
         "URL": "",         (: if DelOthSrv is false then this property is
used as key to merge the new settings with the existing ones :)
         "LoginProvider": map{
             "NameSuffix": "",
             "NamePrefix": "",
                              },
         "Port": "",
         "User": "",
         "StorePW": true(),
         "Password": "",
         "SSL": false()
          }                  (: , map {...} to add another server :)
                    }
     }
)
```

mt-string-to-hexBinary

mt-string-to-hexBinary(Text *を xs:string,* **Encoding** *as xs:string として*) を **xs:string** として

この関数は、テキスト文字列をhexBinary 文字列に変換します。テキスト文字列を返すノードは、**Text** 引数として送信することができます。関数は**Text** 文字列を、**Encoding** 引数。内で指定されるエンコードを持つものとして読み取ります。空の文字列が**Encoding** 引数として与えられると、デフォルトのエンコード **'UTF-8'** が使用されます。**Encoding** が指定されていないと、エラーが発生します。

サンプル

- **mt-string-to-hexBinary**('Hello World', 'ASCII') は、'48656C6C6F20576F726C64' を返します。
- **mt-string-to-hexBinary**('Hello World', '') は、'48656C6C6F20576F726C64' を返します。
- **mt-string-to-hexBinary**('Hello World') は、エラーを返します

mt-test-case-run

mt-test-case-run()を map(*) として

現在実行されているテストランの情報が含まれてるマップを返します。マップには**key:value** ペアが含まれます: **"name":<the name of the test case>**、**"step":<the current step>**、**"count":<the total number of steps>**。再生が現在実行されていない場合、マップのキーには空の値が含まれます。

サンプル

- **mt-test-case-run**() は以下のような単純型マップを返します {"name":"MyTestCase", "step":"2", "count":"10"}
- **mt-test-case-run**() は以下のような単純型マップを返します {"name":"", "step":"", "count":""}

mt-text-to-speech-is-language-available

mt-text-to-speech-is-language-available(Language *as xs:string*) as **xs:boolean**

言語引数は、フォーム**en** (言語コード) または**en-US** (言語国名コード)の文字列の値をとることができます。言語の引数内で指定された言語がモバイルデバイス上で使用できる場合、関数が`true()`を返します。それ以外の場合は、`false()` を返します。

使用法
mt-text-to-speech-is-language-avaialable("en") は、**en** または**en-<country>** 言語のバリアントがモバイルデバイス上で使用することができる場合、`true()` を返します。それ以外の場合は、`false()` を返します。
mt-text-to-speech-is-language-available("en-US") は、モバイルデバイス上で**en-US** を使用することができる場合、`true()` を返します。それ以外の場合は、`false()` を返します。

- mt-text-to-speech-is-speaking

mt-text-to-speech-is-speaking() を**xs:boolean** として
読み上げアクションの再生が実行中の場合は、`true()` を返し、それ以外の場合は`false()` を返します。

- mt- transform- image

Altova 拡張関数の イメージに関連した関数 セクション内のこの関数に関する説明を参照してください。

- mt-user-tried-to-cancel-actions

説明
ユーザーが**[戻る]** ボタンを押した場合、またはソリューションを終了しようとした場合、関数は、`true()`を返します。それ以外の場合は、`false()` (デフォルトの値)を返します。

使用方法
`mt-user-tried-to-cancel-actions()`

- mt-video-get-current-position

mt-video-get-current-position(VideoControlName を*xs:string* として**)** を**xs:string?** として
引数としてビデオコントロールの名前を取り、このビデオコントロール内で再生されている現在のポジションを(秒) で返します。コントロール内でビデオが再生されない場合は、エラーが返されます。現在の再生箇所に関する情報は再生が開始してのみ、取得することができます。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
mt-video-get-current-position("Video-01") は、`Video-01` という名前のビデオコントロールで再生されているビデオのの現在のポジションを返します。

- mt-video-get-duration

mt-video-get-duration(VideoControlName を *xs:string として***)** を**xs:integer** として
ビデオの名前を引数として取り、ビデオコントロール内で再生されているビデオの長さ(秒) を返します。コントロールでビデオが再生されていない場合はエラーが返されます。長さに関する情報は再生が開始してのみ、取得することができます。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
mt-video-get-duration("Video-01") は、`Video-01` という名前のビデオコントロールで再生されているビデオのの長さを返します。

- mt-video-height

mt-video-height(VideoControlName を *xs:string として***)** を**xs:integer** として
ビデオの名前を引数として取り、ビデオコントロール内で再生されているビデオの高さを(ピクセル) 返します。ビデオが再生されていない場合はエラーが返されます。ビデオの高さに関する情報は再生が開始してのみ、取得することができま

す。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
mt-video-height("Video-01") は、Video-01 という名前のビデオコントロールで再生されているビデオのの高さを返します。

- mt-video-is-playing

mt-video-is-playing(VideoControlName を *xs:string として*) を**xs:boolean** として
ビデオの名前を引数として取り、ビデオコントロール内でビデオが再生されている場合は、true() を返し、それ以外の場合は false() を返します。

使用方法
mt-video-is-playing("Video-01") は、Video-01 という名前のビデオコントロール内でビデオが再生されている場合は、true() を返し、それ以外の場合は、false() を返します。

- mt-video-width

mt-video-width(VideoControlName を *xs:string として*) を**xs:integer** として
ビデオコントロール内で再生されているビデオの幅 (ピクセル) を返します。ビデオが再生されていない場合はエラーが返されます。ビデオの幅に関する情報は再生が開始してのみ、取得することができます。ですから、関数は、再生の開始の後に使用されることが奨励されます。

使用方法
mt-video-width("Video-01") は、Video-01 という名前のビデオコントロールで再生されているビデオのの幅を返します。

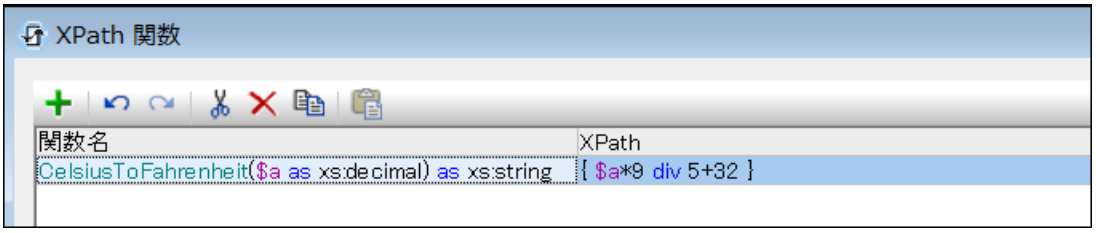
- mt-wait-cursor-shown

説明
クライアントデバイスが待機カーソルを表示する場合は、true() を返し、それ以外の場合は、false() を返します。

使用方法
mt-wait-cursor-shown()

ユーザー定義 XPath/ XQuery 関数

プロジェクトのすべてのXPath 式で使用することのできる、個別プロジェクトのための専用 XPath/XQuery 関数を作成することができます。これらのユーザーにより定義された関数を作成し、管理するアクセスポイントは、**[プロジェクト| XPath/ XQuery 関数]** コマンドによりアクセスできるXPath 関数ダイアログです。XPath 関数ダイアログ (*下のスクリーンショット*) は、プロジェクト内のすべてのユーザー定義 XPath 関数をリストしています。ツールバーまたは、ダイアログの対応するアイコンを使用して、関数を追加または削除することができます。関数の定義を編集する場合、は関数の**[XPath 式の編集]** ボタンをクリックします。

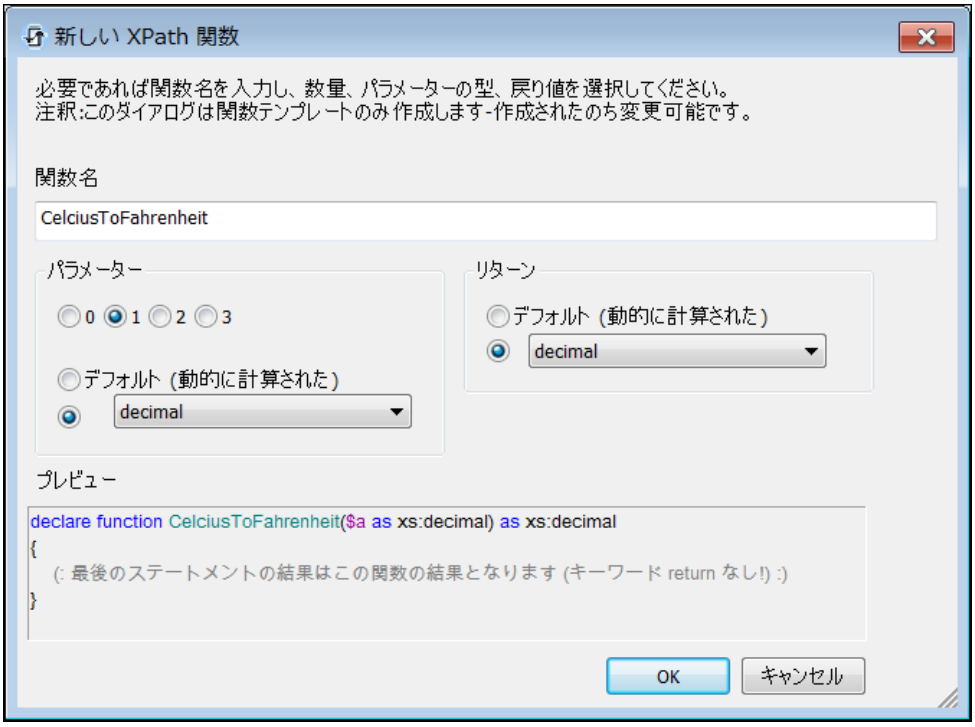


関数リストは関数名に順序付けることができます。これは、関数名列をクリックすることで実行できます。各クリックのサイクルは次のシーケンスで関数を順序づけます: (i) 昇順、(ii) 降順、(iii) ダイアログの順序。ダイアログ内の順序は関数を、リスト内でドラッグアンドドロップすることで変更することができます。リストを昇順 降順 の順序に置換えた後、関数をリスト内で他の場所に移動させると、新しく作成された順序がダイアログの順序となる点に注意してください。

ユーザにより定義された新しい XPath 関数の追加

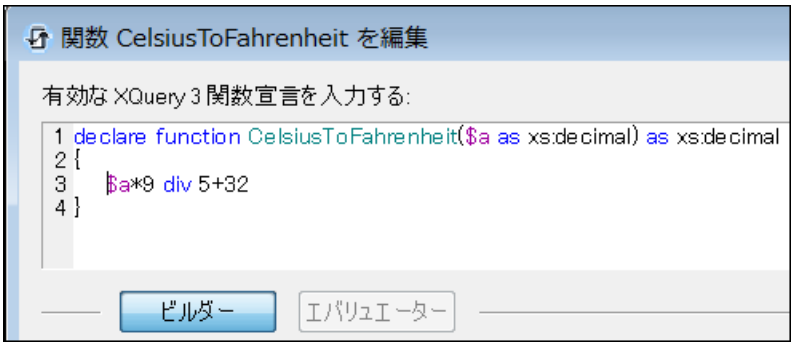
ユーザーにより定義された新しい関数を追加するには次の2 つのステップを踏みます: (i) 関数の宣言と(ii) 関数の定義。

新しい関数を追加するには以下を行います、XPath 関数ダイアログ(*上のスクリーンショット参照*) のツールバーから**[追加]** をクリックします。これにより新しいXPath 関数ダイアログ(*下のスクリーンショット*) が表示されます。



このダイアログで、関数の名前の宣言、関数パラメーター(引数) の数と種類の指定、関数の戻り値の種類を指定することができます。上のスクリーンショットで摂氏から華氏へ変換する関数の小数点の宣言をおこなっています。関数は摂氏の値を

小数とする1 つのパラメーターをもちます。これは、小数、華氏の気温、を出力します。関数の働きに関しては、次のステップで説明されています。関数 (*上のスクリーンショット*) を宣言した後、**[OK]** をクリックします。関数を定義することのできる、新しく宣言された関数のテンプレートを含む関数の編集ダイアログ (*下のスクリーンショット*) が表示されます。



波かっこに関数の定義を入力します。上のスクリーンショットに表示された定義では、**$a** が入力パラメーターです。 **[OK]** をクリックして終了します。関数はXPath 関数ダイアログのユーザー定義関数のリストに追加され、プロジェクトのすべての XPath 式で使用することができます。

メモ ユーザー定義 XPath 関数は個別の名前空間にある必要はありません。結果、ユーザー定義関数を定義または呼び出す際に名前空間プレフィックスは必要ありません。 XPath デフォルト名前空間 は、拡張関数およびユーザー定義関数を含む、すべてのXPath/XQuery 関数に使用されています。内蔵された関数を含むあいまいさを回避するために、ユーザー定義関数を大文字にすることが奨励されます。

XPath/ XQuery についてよくある質問

テーブルの繰り返し行の中に XPath 式があります。子属性をターゲットにするには、絶対または相対式のどちらを使用したらよいのでしょうか？

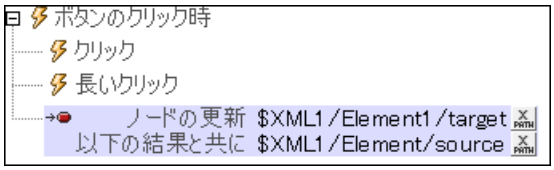
XPath 式がテーブルの繰り返し行にある場合、テーブルの行に対応する要素はコンテキストノードです。例えば、`Row` の場合。

- 絶対パスを使用する場合、例えば、`$XML/Row/@id`、XPath 式は、**すべての** `Row` 要素の `@id` 値を返します。アトミック値を期待する操作を使用する場合、操作はエラーを生成します。
- 相対パスを使用する場合、例えば、`@id`、すべての繰り返し行が、コンテキスト `$XML/` 行を持つため、XPath 式は、現在の `Row` 要素の1つの `@id` 属性の単一のアトミック値を返します。

混在したコンテンツ要素(テキストと要素)をロケートする XPath 式があります。（XPath により管理され、)ロケートされた要素とその子孫のテキストの値を取得する代わりに、ロケートされた要素のテキストコンテンツを取得しました。なぜでしょう?

混在したコンテンツ(テキストと要素)付きの要素がXPath ロケーター式によりロケートされる場合、混在したコンテンツ要素のテキストコンテンツのみが返されます。子孫要素のテキストコンテンツは無視されます。

ノードの更新アクションの例により最もよく説明されています。下のスクリーンショットで定義されるノードの更新アクションを熟考してください。



XML ツリーが次の構造とコンテンツを持つ場合:

```
<Element1>
   <source>AAA
      <subsource>BBB</subsource>
  </source>
  <target></target>
</Element1>
```

子要素 `subsource` のコンテンツを無視する一方、混在コンテンツ要素 `source` のテキストコンテンツと共に、`target` 要素が更新されます。`target` という名のノードは、`<target>AAA</target>` に更新されます。

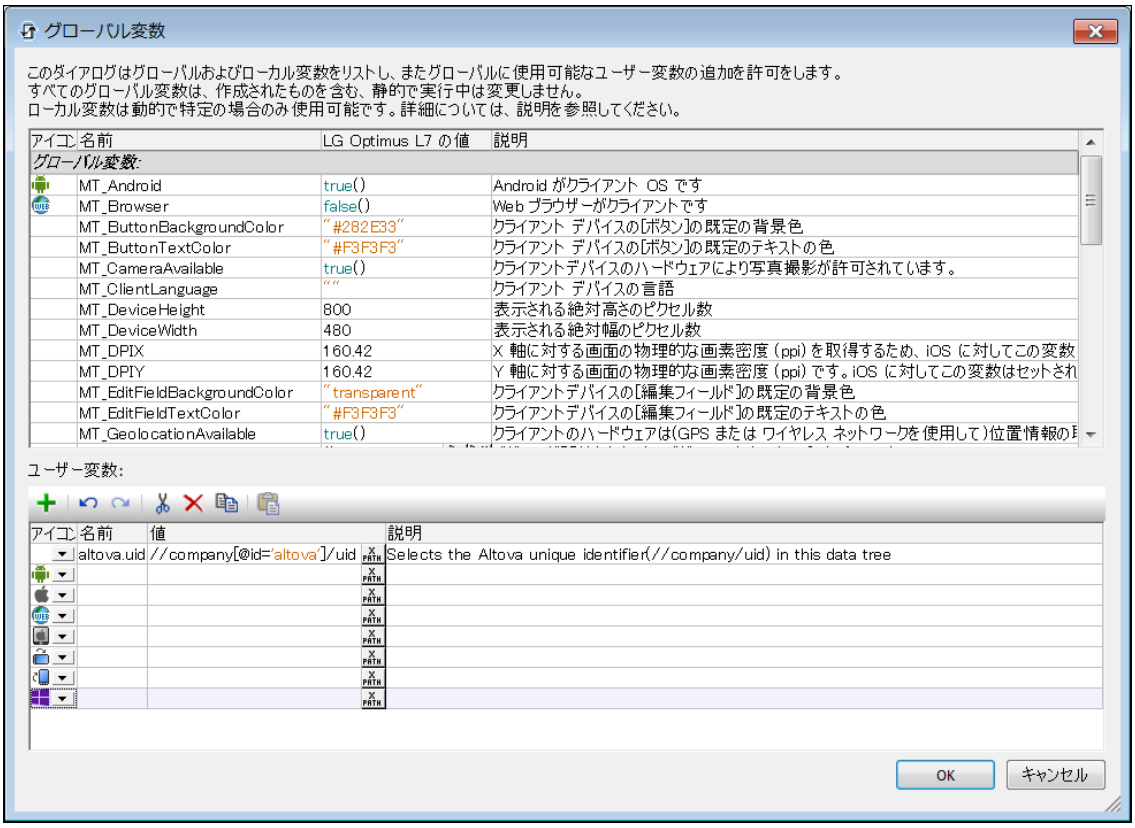
メモ 子孫ノードのテキスト コンテンツを含みたい場合、空の文字列付きの **`concat`** 関数を第 2 番目の引数として使用します。上の XML 例を使用することにより、たとえば、式 **`concat($XML1/Element1/source, '')`** は **`"AAABBB"`** を返します。

メモ XPath 対応メソッドのシリアル化を使用するチャート: 混在コンテンツ要素がXPath ロケーター式を使用してロケートされる場合、子孫要素のテキスト コンテンツもシリアル化されます。

グローバル変数

グローバル変数はクライアント モバイル デバイスの情報を含みます。例えば、ある変数はデバイスの種類を示し、もう1 つの変数はデバイスのディメンションを、もう1 つの変数はデバイスの現在の向き(横長または縦長) を示すなどです。これらすべての変数は、通常のモバイル通信の手順の一部として、クライアント デバイスからの実行時間内に取得されます。変数は XPath/XQuery 式に使用することができます。この結果、(サイズなどの) デバイス特有の静的プロパティに条件づけて、また、(向きなどの)変更可能な動的プロパティに応じて、処理を指定することができます。

MobileTogether Designer には、グローバル変数ダイアログ ([**プロジェクト | グローバル変数**]、*下のスクリーンショット*) で表示される、グローバル変数の標準ライブラリがあります。このダイアログでは、カスタム化した変数をプロジェクト内で使用するために定義することができます。カスタム化されたユーザー変数は XPath 式でで設定されます。



グローバル変数ダイアログ (*上のスクリーンショット*) は3つの種類の変数を表示しています:

- 静的値の変数: プロジェクトの実行中変更されない値を持つ変数です。ダイアログで、*値*の列のヘッダーは、デバイス選択コンボ ボックスで選択されたモバイル デバイスを表示します。例えば、使用されているモバイル デバイスが Android デバイスの場合、変数 **`$MT_Android`** は **`true()`** の値を持ちます。
- 動的値の変数: 動的値の変数はデバイスに関連した、およびプロジェクトに関連した情報を含み、プロジェクト実行中に変更できます。例えば、**`$MT_ControlNode`** 変数はプロジェクト実行中にどのノードが現在のノードであるかにより異なる値を持ちます。
- ユーザー変数: グローバル変数の標準ライブラリに加え、(ダイアログでは *ユーザー変数* と呼ばれる) 自身のグローバル変数を、ダイアログの下のペインで追加することができます。XPath 式を使用してユーザー変数に値を与えます。

メモ　ユーザー変数を定義する場合、変数名に**$** シンボルを使用しないでください。グローバル変数を XPath 式内で使用する場合は、通常通り**$** シンボルを使用しなくてはなりません。例:
`concat('http://www.', $company, '.com')`

静的なグローバル変数

グローバル変数ダイアログで静的値の変数は、*グローバル変数*と呼ばれています。グローバル変数は、デバイスの種類とディメンションなどのモバイルデバイスの静的な情報を含んでいます。プロジェクトの実行中、静的な値変数は変更されません。静的値の変数はグローバル変数ダイアログ **[プロジェクト] グローバル変数]** に表示されます。ダイアログで、*値*の列のヘッダーは、デバイス選択コンボボックスで選択されたモバイルデバイスを表示します。例えば、使用されているモバイルデバイスが Android デバイスの場合、変数 **$MT_Android** は **true()** の値を持ちます。(デバイスの情報は、標準のモバイル通信の手順として、デバイスから送信されます)。

メモ 変数の完全なリストと詳細に関しては、グローバル変数 ダイアログを参照してください。

▾ モバイルデバイスの種類を示す変数

説明

デバイスの種類を示す変数 (*下のテーブル参照*) のセットです。これらの変数は、条件付きデバイスの種類のアクションを指定するために使用されます。例えば **if ($MT_iOS=true()) then 'http://www.apple.com/' else 'http://www.altova.com'**。クライアントデバイスに関する情報がデバイスから送信されたとします。特定のデバイスでソリューションが実行された場合、対応するグローバル変数が(*下のテーブル参照*) **true()**に設定されている場合、グループ内の他の変数は**false()**に設定されます。これらの変数はXPath/XQuery 式 で使用することができます。

MT_Android	true() | false()
MT_Browser	true() | false()
MT_iOS	true() | false()
MT_iPad	true() | false()
MT_Windows	true() | false()
MT_WindowsPhone	true() | false()

▾ デバイスの通信機能を示す変数

説明

これらの変数は、デバイスに携帯ショートメール(SMS) および電話通話のサービス機能の有無を示し、SMS または電話通話を開始する前に確認します。通信機能に関しての情報は、クライアントデバイスから受信されます。値は **true()** または**false()**です。

MT_SMSAvailable	true() | false() | "" (empty string)
MT_TelephonyAvailable	true() | false() | "" (empty string)

▾ デバイスの機能の可用性を示す変数

説明

これらの変数は、カメラのアプリケーションと位置情報トラッキングのモバイルデバイスでの可用性について示しています。イメージ作成、位置情報に関連した、またはNFC に関連した アクションを開始する前にこれらを使用することができます。機能の可用性についての情報はクライアントデバイスから受信したものです。値は**true()** または**false()** です。(例えば、クライアントがWeb ブラウザーである場合など) これらの関数が使用できない場合、変数は定義されません。

MT_CameraAvailable	true() | false() | undefined

MT_GeolocationAvailable	`true() | false() | undefined`
MT_NFCAvailable	`true() | false() | undefined`

- デバイスのディメンション(寸法)と解像度を含む変数

説明
デバイスの高さ幅の値がピクセル値としてこれらの変数に表示されます。解像度はppi (ピクセルズ パー インチ) の単位で示され、X とY のディメンションで表示されます。iOS デバイスでは解像度は設定されていませんので、iOS デバイスの`$MT_DPIX` と`$MT_DPIY` は空白です。

MT_DeviceHeight	*ピクセルで表示される長さ*
MT_DeviceWidth	*ピクセルで表示される長さ*
MT_DPIX	*ピクセルズ パー インチで示される横長の画素密度*
MT_DPIY	*ピクセルズ パー インチで示される縦長の画素密度*

- デバイスの要素のデフォルトの色を含む変数

説明
ページおよび[ページ] コントロールの一部では、デバイスにより異なるデフォルトの色が設定されています。デフォルトの色を確認することは、ページのデザインに役に立ちます。例えば、ラベルの背景色は、デバイスのデフォルトのテキストの色に対応して設定することができます: **`if ($MT_LabelTextColor = '#000000') then '#FFFFFF' else '#000000'`**。クライアントデバイスから受け取ったデフォルトの色は16 進数の値です。例: `#336699` と `#ffaaff` 。

MT_ButtonBackgroundColor	*ボタンの背景色*; 16 進数の値、例: `#ffaaff`
MT_ButtonTextColor	*ボタンのテキストの色*; 16 進数の値、例: `#336699`
MT_EditFieldBackgroundColor	*編集フィールドの背景色*; 16 進数の値、例: `#ffaaff`
MT_EditFieldTextColor	*編集フィールドの背景色*; 16 進数の値、例: `#ffaaff`
MT_LabelBackgroundColor	*ラベルの背景色*; 16 進数の値、例: `#ffaaff`
MT_LabelTextColor	*ラベルの色*; 16 進数の値、例: `#336699`
MT_PageBackgroundColor	*ページの背景色*; 16 進数の値、例: `#ffaaff`

- その他

MT_ClientLanguage

クライアントデバイスの言語

MT_InputParameters

パラメーター値は、ソリューションが開始されるとソリューションにパスされます。これらの値は **`MT_InputParameters`** 変数に保管されます。ソリューションのハイパーリンクがクリックされるとパラメーター値はソリューションにパスされます。ハイパーリンクのURL にパラメーター値が含まれるクエリ文字列が存在する場合、これらは、リンクがクリックされソリューションが開始されると、ソリューションにパスされます。

MT_InputParameters 変数はパラメーター値を文字列値アイテムのシーケンスとして保ちます。単一のパラメーターの値を取得するには、シーケンス内のパラメーターのインデックス位置を知る必要があります。この後、XPath ロケーター式内のこの位置を使用することができます。例: **$MT-InputParameters[1]** ハイパーリンクと MT_InputParameters 変数の詳細に関しては、次を参照してください: ソリューションへのハイパーリンク。

- MT_IsAppStoreApp

  現在のソリューションが AppStore App として実行されているかを示します。許可されている値は true() または false() です。デフォルトの値は false() です。

- MT_IsEmbedded

  現在のソリューションが Web ページ内の埋め込み を実行しているかを示します。許可されている値は true() または false() です。デフォルトの値は false() です。

- MT_SimulationMode

  下のテーブルにリストされる値を使用して、どのようなシミュレーションが現在実行されているのか表示します。空のオプションは、シミュレーションではなく、ソシューションが実際のエンドユーザーのシナリオを実行していることを示します。$MT_SimulationMode はとても便利です。例えば、現在実行されているシミュレーション(または実際の用途)別に条件付きの処理を行いたい場合などです。詳細に関してはシミュレーションのセクションを参照してください。

"designer"	*デザイナーで直接シミュレーションを実行する*
"designer-server"	*シミュレーションをスタンドアロンサーバーで実行する*
"designer-client"	*シミュレーションをクライアントでトライアルランする*
""	*サーバーからクライアント/ブラウザー、エンドユーザーによる実行*

- MT_UserName

  MobileTogether Server にログインする名前。

動的なローカル変数

グローバル変数ダイアログで動的値の変数はローカル変数と呼ばれています。ローカル変数はデバイスに関連した、およびプロジェクトに関連した情報を含み、プロジェクト実行中に変更できます。例えば、デバイスの向き変数は、ユーザーがデバイスを持つ現在の向きにより変化します (*下のデバイスの向き変数の説明を参照*)。

現在のコントロール (*下部参照*) の情報を含む変数は、現在使用されているコントロールおよびノードの異なるアスペクトを参照するために使用できるため特に役に立ちます。現在のコントロールとノードを識別できることにより、条件付きの処理を行うことができます。例えば、**$MT_ControlNode** 変数はどのノードがプロジェクト実行時に現在のノードかテストする際に使用でき、また、これを基に他のノードを探すこともできます。**$MT_ControlValue** 変数は現在のコントロールに関連したノードの内容を含んでいます。

メモ 変数の完全なリストと詳細に関しては、グローバル変数 ダイアログを参照してください。

▾ デバイスの向きを示す変数

説明

MT_Portrait とMT_Landscape の値はtrue() またはfalse() と設定でき、プロジェクトの実行中に変更することができます。デバイスの向きに合わせ、ページまたはコントロールのプロパティを変更する際に使用することができます。

MT_Portrait	true() | false()
MT_Landscape	true() | false()

▾ デバイスのビューポートとディメンションを示す変数

説明

つの変数はそれぞれデバイスのビューポートの幅 (X ディメンション) と高さ(Y ディメンション) を示します。ビューポートはデザインコンポーネントが描かれるスクリーンのエリアです。ボタンのある上または下のバーを除いたスクリーンのエリアです。Web ブラウザークライアントでは、$MT_CanvasX と$MT_CanvasY 変数はMobileTogether Client アプリが表示されるカンバスのディメンションを与えます。(これは、ブラウザーウィンドウのディメンションからタイトルバー、リボン、ステータスバーおよび他のサイドバーを引いた数値です)。これらの変数の値は、ピクセルの値で、(**MT_DeviceHeight** と**MT_DeviceWidth** によりそれぞれ返される) デバイスの高さ幅ディメンションより小さくなくてはなりません。*iOS デバイスについてのポイントとピクセルを参照してください。*

MT_CanvasX	*ピクセルで表示される長さ*
MT_CanvasY	*ピクセルで表示される長さ*

⊟ *iOS デバイス上でのポイントとピクセル*

長さを定義するプロパティのためのピクセルの値を入力すると、iOS デバイスで表示する際には、これらの値は、ビューポート座標スペース内で**ポイント**として読み込まれます。ビューポート座標スペースは、デザインコンポーネントが描かれ、ポイントがこのスペース内で長さの単位として使用されているカンバスです。ここで使用されるポイントとは、一インチの1/72 に等しい単位ではありません。iOS デバイスは自動的にビューポート座標スペース内の**ポイント**をデバイス座標スペースで**ピクセル**にマップします。このようにして、iOS デバイスのスクリーンの精度に関係することなく、デザインコンポーネントは同じサイズの関係をカンバスに対して保つことができます。

MobileTogether Designer では、**$MT_CanvasX** と**$MT_CanvasY** 動的変数を使用して、現在のビューポート、カンバス、の次元を確認することができます。iOS デバイスでは、以下のように変数により返された値が計算されます: 現在のデバイス座標スペースの **ピクセル** ディメンションは、ビューポート座標スペース内で**ポイント**に変換されます。対応する数値は2 つの変数によりデザイン内で使用されるピクセルとして返されます。**$MT_CanvasX** と**$MT_CanvasY** 変数により返された値をベースにして、対応するサイズを計算することができます。例えば、ビューポートの幅の半分にイメージを作成する場合、に**$MT_CanvasX * 0.5** 等しいピク

セルの幅を与えます。イメージの幅の XPath 式は `concat($MT_CanvasX * 0.5, 'px')` になります。

- (Windows アプリとブラウザー内の)サイズ変更することのできるウィンドウのサイズを含む変数

説明
これらの変数はWeb ブラウザーとWindows アプリデバイスのみに適用することができます。ユーザーは(デスクトップアプリ内でウィンドウのサイズが変更できることと同様) ブラウザーウィンドウとWindows RT デバイスのアプリウィンドウのサイズを変更することができます。`$MT_WindowHeight` と`$MT_WindowWidth` 変数は、MobileTogether Client アプリが実行されるウィンドウの高さ幅を有します。ブラウザー内では、これらの変数はブラウザーウィンドウの高さと幅を与えます。(ブラウザー内では、`$MT_CanvasX` と`$MT_CanvasY` 変数により与えられる(ブラウザーウィンドウのディメンションからタイトルバー、リボン、ステータスバーとサイドバーを引いた値である) MobileTogether Client アプリのカンバスが表示されます)

MT_WindowHeight	*ピクセルで表示される長さの値*
MT_WindowWidth	*ピクセルで表示される長さの値*

- 現在のコントロールについての情報を含む変数

説明
説明
これらの変数は現在のコントロールと関連したデータソース ノード(コントロールのソースノード)に関連した情報を含みます。これらの変数の値は、現在処理されているコントロールにより変化します。例えば、**`$MT_ControlNode`** 変数は、関連したノードが変更すると現在のコントロールが変更されため異なる値を持ちます。(スペースおよび水平線コントロールなどの、コントロールの一部は、ソースノードを持ちませんが、チャート コントロールなどの他のコントロールは 関連したノードのコンテンツとしてXML 値を持つことに注意してください。)

`$MT_ControlNode` 変数は、ソースツリーノードへのポインターです。以下のようなテストのために使用することができます: **`$MT_ControlNode/localname()="first"`**.

これらの変数は、コントロールの値に基づいてコントロールのプロパティを変更する際に役に立ちます。例えば、エラー発生時、**`$MT_ControlValue`** 変数はラベルの背景色を赤に変更する際に使用できます: **`if ($MT_ControlValue = 'NaN') then '#FF0000' else '#FFFFFF'`**。

MT_ControlKind	*文字列の値*
MT_ControlName	*文字列の値*
MT_ControlNode	*コントロールのソースノードであるXML ノード*
MT_ControlValue	*コントロールのページソースリンクノードの値*
MT_ControlValueBeforeChange	*コントロールのページソースリンクノードの前の値、ノードまたはノードが編集される前の値*

メモ **`$MT_ControlValue`** 変数は、コントロールの`Visible`可視、 `XPath` から値を取得 および テキストプロパティの値の生成には使用**することができません。** これらのプロパティの値のために使用される場合、検証エラーが生じます。

その他

- MT_AudioChannel

  この変数はオーディオの再生イベントのために定義されたアクション内でのみ使用することができます。イベントをトリガー下オーディオのチャンネルの番号 (1 から 5) である整数を有します。

- MT_DBExecute_Result

  DB 実行アクションの最後の実行のXML 結果。DB 実行アクションにはすべてのSQL ステートメントが使用できます。ですから、アクションの実行は、DB空のデータ、ブール値、計算の結果などを含む多種のXML データを取得することができます。

- MT_FirstPageLoad

  現在のワークフローの実行中にpage がロードされる初回の場合、`true()` に設定する。

- MT_HTTPExecute_Result

  最後に実行されたSOAP リクエストの実行 またはREST リクエストの実行のXML 結果。

- MT_PageName

  ページの名前。

- MT_ServerConnectionErrorLocation

  この変数はサーバーの接続エラー時 イベントをトリガーしたアクションスタックを含む文字列のシーケンスです。アクション名を変更することができるため、この編集はデバッグのみに使用されることが奨励されます。

- MT_TableColumnContext

  動的な列を含むテーブルが生成されると、この変数には、現在の列のコンテキストノードが含まれます。これは動的な行および動的な列を含むテーブルと作業する場合必要不可欠です。このようなテーブルでは、セルのコンテンツが動的な行に関連した要素のコンテキスト内で定義されます。行コンテキスト内では、**`MT_TableColumnContext`** 変数が使用され、現在の列と関連する要素が検索されます。この変数の使用例に関しては、次のセクションを参照してください： 「テーブル| 動的な列」。

- MT_TargetNode

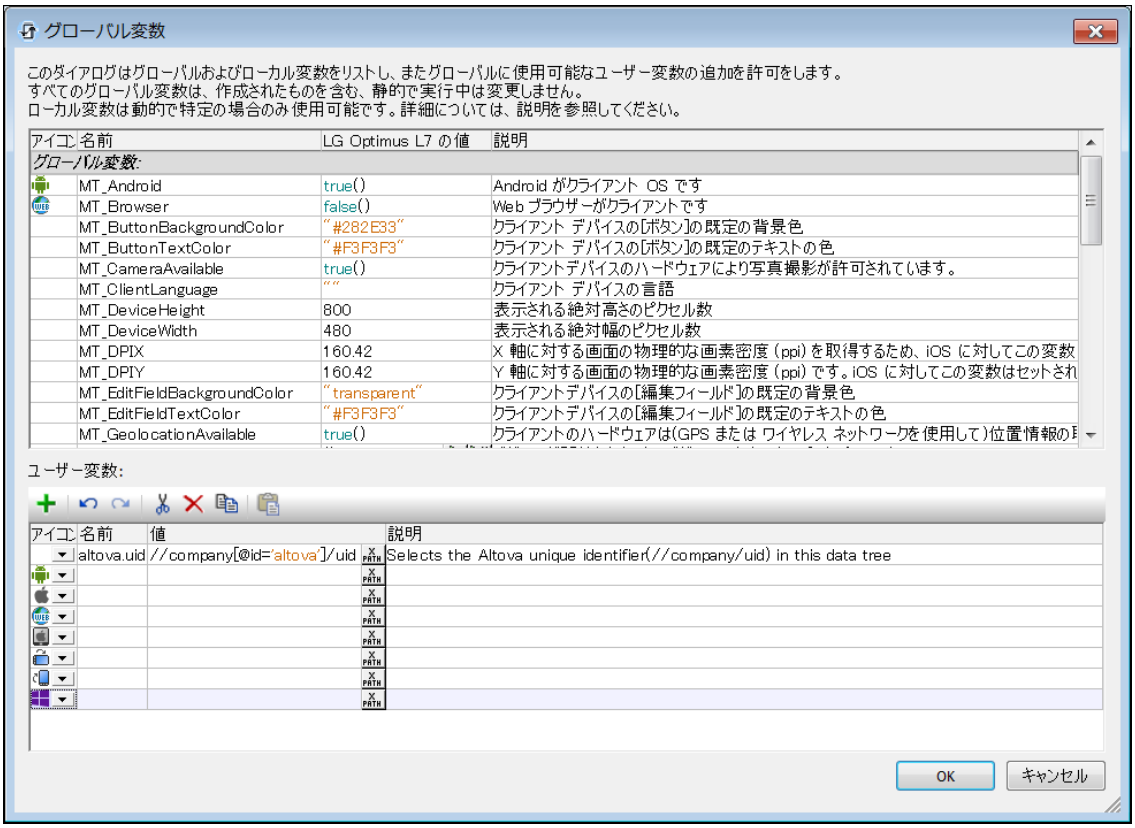
  この変数は ノードの更新、ノードの挿入、または ノードの追加 アクションのターゲット ノードを識別します。この変数は、更新された値、および、新しいノード プロパティをターゲット ノードに応じて生成するために使用することができます。どの様に変数を使用することができるのかそれぞれのアクションの説明を参照してください。`$MT_TargetNode` はDB 実行 アクションおよび **ロード時に存在することを確認 (XPath)** コマンドにも使用することができます。

- MT_UserRoles

  現在ログインしているユーザーのロールです。ロールはMobileTogether Server 管理者によりユーザーに割り当てられ、MobileTogether Server から取得することができます。

ユーザー変数

ユーザー変数は、グローバル変数ダイアログの下のペインで定義される変数です([**プロジェクト**] [**グローバル変数**]、下のスクリーンショット)。



ユーザー変数を追加するには、ダイアログ下のペインで以下を行います:

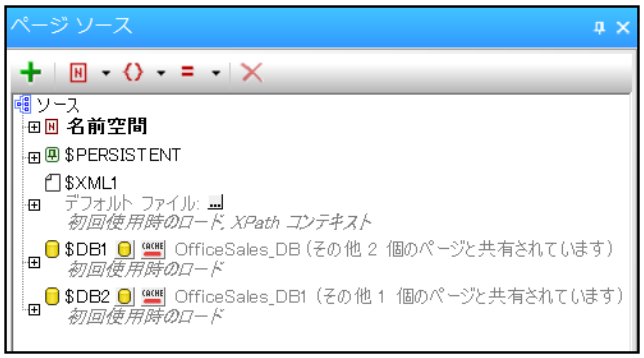
1. リストに新しいエントリを追加するには、(ペインのツールバーにある) **[追加]** または **[挿入]** アイコンをクリックします。
2. 新しい変数の名前を(*名前*列に、**$** シンボル無しで)入力し、変数に説明を与えます(*説明*列)上のスクリーンショット参照。
3. *値*フィールドをクリックして、[XPath/XQuery 式の編集] ダイアログを表示します、変数の値を決めるXPath 式を入力します。
4. 特定のグループに属する新しい変数を識別するためのアイコンを選択します。
5. **[OK]** をクリックして完了します。変数はグローバル変数として追加され、プログラミングのコンテキストとして使用することができます。

チャプター 13

データベース

13 データベース

データベース(DB) はMobileTogether デザインのデータソースを使用します。これにより、DB からのデータを MobileTogether ソリューションで表示することができます。また、エンドユーザーがモバイルデバイスからDB データの編集を行うことができます。複数の編集可能なDB データソースを使用することができます。これらのDB ソースのデータは、XQuery 式を含む多種のメカニズムを使用して、取得、編集、および保存 されることができます。



このセクション

このセクションは以下のように整理されています:

- データソースとしてのDB
- DB への接続
- DB オブジェクトをデータソースとして選択
- DB データの編集
- DB にデータを保存
- DB 実行アクション
- DB データの表示
- データベース クエリ

MobileTogether Client ソリューション内で編集することのできる、複数のデータソースを使用するMobileTogether デザインについての詳細に関してはデータベースチュートリアル、データベースとチャート を参照してください。また、データベースの使用のデモビデオも参照してください。

データベース サポート

次のデータベースはサポートされています。各データベースの使用できるルート オブジェクトもリストされています。Altova は他のODBC/ADO データベースへのサポートの向上のための努力をしていますが、以下にリストされたデータベースのみで接続とデータ処理のテストに成功しています。使用中のAltova アプリケーションが、64 ビット バージョンの場合、接続されているデータベース特定の64 ビットデータベース ドライバーにアクセスしてください。Altova アプリケーションはIBM iSeries データベースのロジカル ファイルをサポートし、ロジカル ファイルを表示します。

データベース	ルートオブジェクト	メモ

Firebird 2.5.4	データベース	
IBM DB2 8.x, 9.1, 9.5, 9.7, 10.1, 10.5	スキーマ	
IBM DB2 for i 6.1, 7.1	スキーマ	ロジカルファイルはサポートされ、ビューとして表示されています。
IBM Informix 11.70	データベース	
MariaDB 10	データベース	
Microsoft Access 2003, 2007, 2010, 2013	データベース	
Microsoft Azure SQL Database	データベース	SQL Server 2016 codebase
Microsoft SQL Server 2005, 2008, 2012, 2014, 2016	データベース	
MySQL 5.0, 5.1, 5.5, 5.6, 5.7	データベース	
Oracle 9i, 10g, 11g, 12c	スキーマ	
PostgreSQL 8.0, 8.1, 8.2, 8.3, 9.0.10, 9.1.6, 9.2.1, 9.4, 9.5, 9.6	データベース	PostgreSQL 接続は、ODBC または JDBC などのインターフェイス（ドライバ）によりネイティブとドライバーベースの接続でサポートされています。ネイティブな接続はドライバを必要としません。
Progress OpenEdge 11.6	データベース	
SQLite 3.x	データベース	SQLite 接続は、ネイティブとしてサポートされます。SQLite データベースファイルへの直接の接続です。別途のドライバーは必要ありません。
Sybase ASE 15, 16	データベース	
Teradata 16	データベース	

SQLite DB に接続するには、MobileTogether Designer 接続ウィザードを使用してください。

メモ
Linux とmacOS サーバーでサポートされるデータベース接続は JDBC のみです。

13.1 データ ソースとしての DB

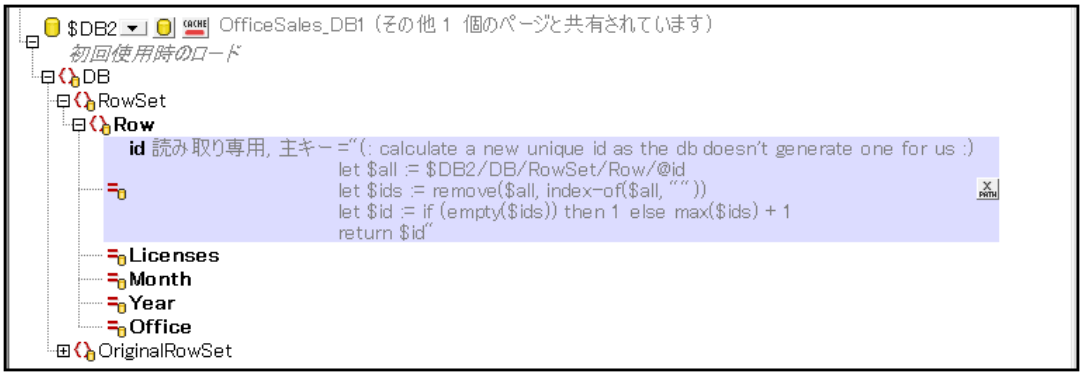
このセクション:

- DB データソースについて
- DB データソースのツリー構造について
- データソースの切り替え
- OriginalRowSet について
- MobileTogether Designer の主キー

DB データ ソースについて

ページのデザインにいくつものDB データソースを追加し、使用することができます。DB データソースが編集可能である、または編集不可能であるかはデータソースが追加される際 に定義されます。データがプレゼンテーションのみに必要な場合は、DB データソースを編集不可能に設定します。クライアントにデータの変更を許可する場合は、データソースを編集可能に設定します。

DB ソースが追加されると、データツリーが生成されます (*のスクリーンショットと下のツリー構造のセクションを参照してください*)。各 DB テーブル行は、行要素に対応します。テーブルの列は行要素の属性として追加されます。データソースが複数のページに使用される場合、単一のツリー構造がデータソースの全てのインスタンスで共有されます。他のページで使用されるデータソースが追加される都度、ツリー構造を共有するオプションを使用することができます。共有される構造が変更される場合、ページの複数インスタンスの共有されるデータソースを変更するオプションが与えられます。または、共有されるデータソースは、変更され他インスタンス内でのみ変更されます。



メモ　SQL ステートメントがページソース内に保管されている場合、クライアントデバイス上でデザインの実行中、ファイアーウォールルールがトリガーされる可能性があります。これを回避するために、以下を行うことが奨励されています: (i) ページソースプロパティを「サーバーのみにページソースプロパティのデータを保管する」に設定します。(ii) クライアント接続のためにSSL を使用します。(iii) SQL ステートメントをサーバー上で必要に応じてアセンブルします。

データソースの切り替え

データソースとしてDB を使用するデザインを作成し、他の同じ構造を持つDB に切り替え、元のデザインの使用を継続することができます。データソースのDB を切り替えるには、ツリーの$DB ルートノードを右クリックし、**[DB データソースの選択]** を選択して、[DB 接続プロセス] を続行します。

同じテーブル名、同じ列名、同じ列定義を持つ場合、2 つのDB は同じ構造を持つと考えられます。新しい構造が異なる場合は、DB への接続が設立されても、新しいDB からのデータを使用してデータソースは更新されません。DB の切り替えが中断されると、データソースは元 DB のを使用して接続を継続します。

メモ　DB が、大文字と小文字を区別するスイッチに関連する場合、SQL ステートメントXPath 式、および一致しない名前を使用する他の構成を変更する必要があります。

DB データ ソースのツリー構造について

すべてのDB データソースには次の構造があります:

```
$DBX (ルートノード)
|
|--DB (ルート要素)
|   |
|   |--RowSet (DB テーブルのコンテナー要素)
|   |   |
|   |   |--Row (DB テーブル)
|   |   |   |
|   |   |   |--<attributes> (DB テーブルのカラム)
```

XPath 式を使用して、ツリー内のノードをアドレスすることができます。(ノードのコンテキスト メニューを介して) ノードがページの*XPath コンテキスト ノード*として設定されている場合、XPath 式をコンテキスト ノードに対応して作成することができます。その他の場合は、ノードを次にリストされるルートノードから開始する絶対パスを使用してアドレスすることができます: `$DBX/DB/RowSet/Row/MyAttribute`。

XQuery 式を使用して、DB ツリー内のデータを取得または操作することができます。下の主キーセクションを参照してください。

OriginalRowSet について

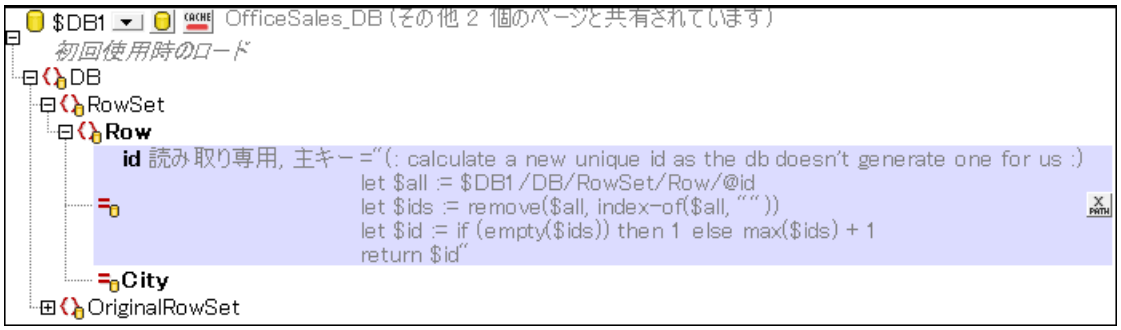
データを編集し、保存するには、ツリーはページソースに、RowSet 要素のコピーであるOriginalRowSet 要素を含む必要があります。オリジナルのデータはOriginalRowSet 要素に保存され、編集されたデータはRowSet 要素に保存されます。データソースが保存されると、OriginalRowSet とRowSet の 2 つのツリーの差異が計算され、ページソースは差異を基に更新されます。修正に成功すると、修正されたデータは、OriginalRowSet にコピーされ、OriginalRowSetが新しく保存されたDB データを含み、修正プロセスが繰り返すことができます。

ページソースのOriginalRowSet を作成するには、ページソースのルートノードを右クリックして、コマンド **[OriginalRowSet の作成]** を切り替えます。

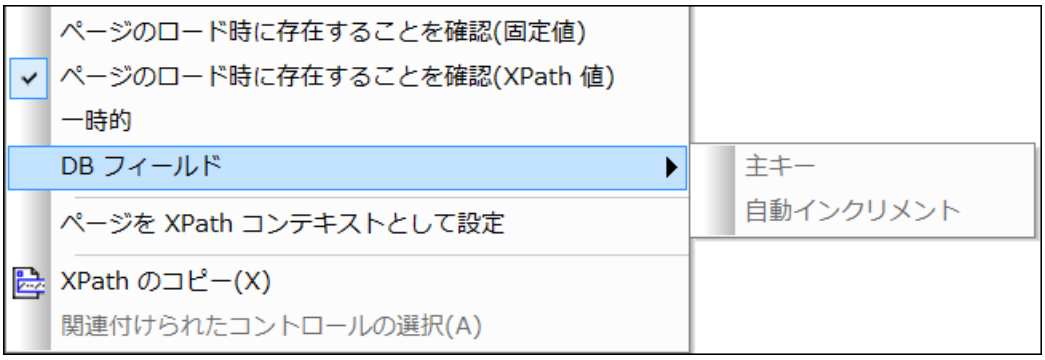
[OriginalRowSet の作成] コマンドは、データベース型 (`$DB`) ルートノードのために有効化されています。ページソースの元のデータを含むデータ構造を作成/削除する切り替えコマンドです。ユーザーにより修正されたデータはデータソースにより作成されたメインの構造に保存されています。修正されたデータがDB に保存されると、OriginalRowSet 構造は新たにDB に保存されたデータを含むために修正されます。これにより元のDB データはツリーで使用可能となります。

MobileTogether Designer の主キー

DB の主キーは通常、自動インクリメントです。この場合、新しい行がテーブルに追加され、追加された行の主キー列は自動的にインクリメントされます。MobileTogether Designer では、主キーからテーブルが取得されると、自動インクリメント情報が自動的に取得され[ページソース] ペインに表示されます(*下のスクリーンショット参照*)。

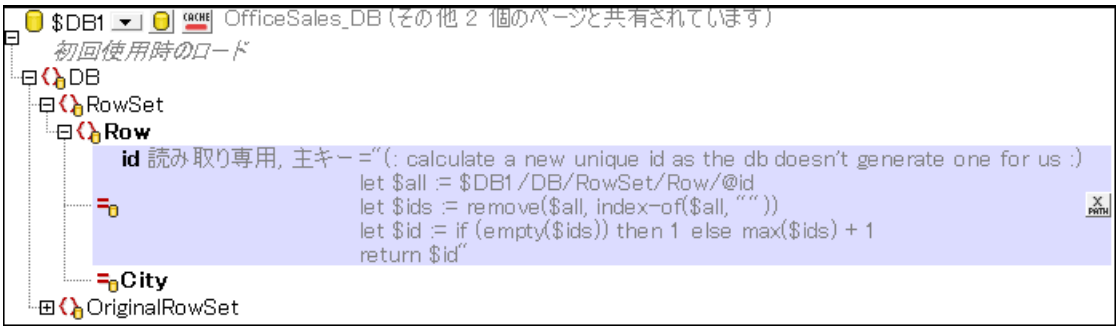


この情報の自動取得が成功しなかった場合、切り替えコマンドを含むツリーノードのコンテキストメニューによりノードに正確な注釈をつけることができます (*下のスクリーンショット参照*)。



主キー列が自動インクリメントでない場合、追加された行の新しい主キーの値はXQuery 式を使用して自動的に生成されなければなりません。これは、主キー列を編集することができないからです。XQuery 式は、主キーノードのコンテキストメニューコマンド**[ページのロード時に存在することを確認 (XPath 値)]** を使用して挿入されます。下の例では、次の XQuery 式を使用して新しい値が主キー `@id` のために生成されます:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```

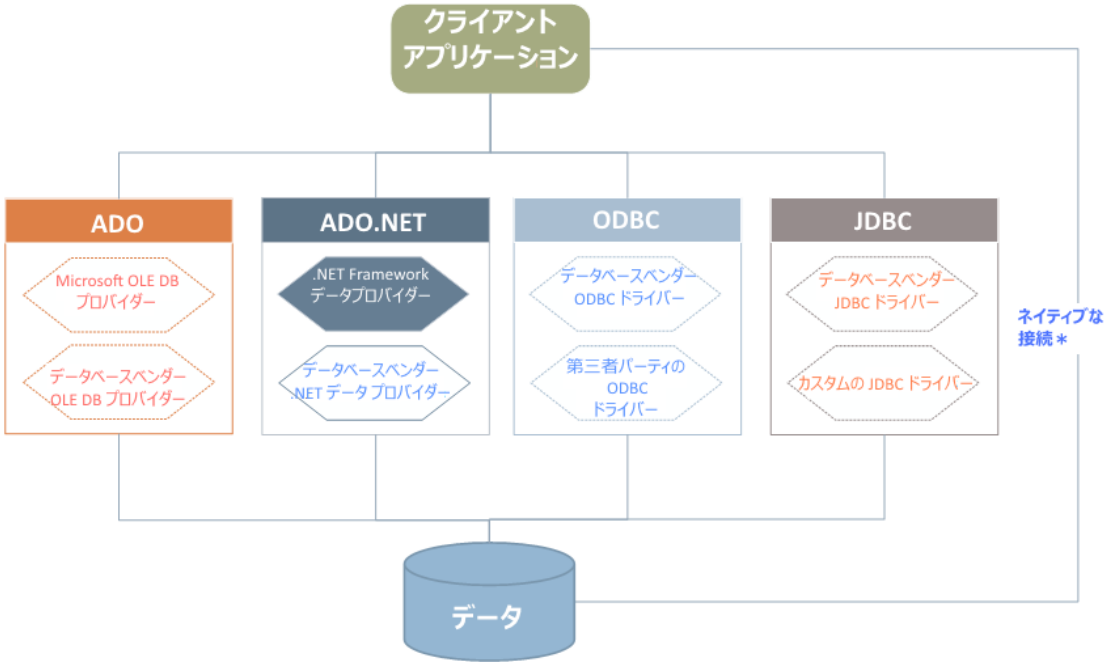


13.2 データベースへの接続

最も簡単なケースとして、データベースは、Microsoft Access、または、SQLite データベースファイルなどのローカルファイルが挙げられます。高度なケースの例としては、データベースは、アプリケーションが接続し、データを消費するオペレーティングシステムと同じオペレーティングシステムを使用するとは限らない、リモート、または、ネットワークデータベースサーバーに存在します。例えば、MobileTogether Designer が、Windows で作動し、データ(例えばMySQL) へアクセスするデータベースがLinux マシーンで作動することは可能です。

多種のデータベース型とリモートおよびローカルで作業するには、MobileTogether Designer は、データ接続インターフェイスとオペレーティングシステム内で使用することのできる、または、定期的にメジャーなデータベースベンダーによりリリースされるデータベースドライバーに従います。常に進化するデータベース技術の景観の中、このアプローチは、クロスプラットフォームに対する柔軟性および相互運用性に対応します。すなわち、MobileTogether Designer により、以下のデータアクセス技術により主要なデータベースにアクセスすることができます

次の図は、最も簡単な方法で、(ジェネリックなクライアントアプリケーションとして表示されている) MobileTogether Designer と(データベースサーバー、または、データベースファイルであることができる) データ保管の間で使用することのできるデータ接続のオプションを示しています。



** 直接なネイティブな接続 は、SQLite とPostgreSQL データベースのためにサポートされています。使用中のシステムにインストールするために追加ドライバーは必要ではありません。*

上の図で示されるように、MobileTogether Designer は、主なデータベース型に次のアクセス技術を使用してアクセスすることができます:

- 基となるOLE DB (Object Linking and Embedding, Database) プロバイダーを使用するADO (Microsoft® ActiveX® Data Objects)
- ADO.NET (データのインタアクションを有効化するMicrosoft .NET Framework 内で使用することのできるライブラリのセット)
- JDBC (Java データベース接続)
- ODBC (Open データベース接続)

ADO.NET の一部はサポートされていない、または、サポートが制限されている場合があります。次を参照してください：ADO.NET サポートメモ。

選択するデータベースへの接続のインターフェイスは、既存のソフトウェアのインフラストラクチャによります。データのアクセス技術と、接続するデータベースと密に統合することのできる、データベースドライバーを通常選択します。例えば、Microsoft Access 2013 データベースに接続するには、**Microsoft Office Access データベースEngine OLE DB プロバイダー** などのネイティブなプロバイダを使用するADO 接続文字列を作成します。に接続するには、他方、最新のJDBC、ODBC、またはADO.NET インターフェイスをOracle Web サイトからダウンロードします。

Windows 製品 (Microsoft Access またはSQL Serverなど) のためのドライバーは、Windows オペレーティングシステムで使用可能ですが、他のデータベースの種類では使用することができない可能性があります。メジャーなデータベースベンダーは、対応するデータベースにADO、ADO.NET、ODBC、またはJDBCの組み合わせからクロスプラットフォームアクセスをすることができる、一般使用可能なデータベースクライアントソフトウェアとドライバーを定期的にリリースします。更に、上記の技術のためのサードパーティーのドライバーを使用することができます。多くの場合は、1つ以上の方法で必要なデータベースへ使用中のオペレーティングシステムから接続することができます、この結果、MobileTogether Designer からも接続することができます。使用可能な機能、パフォーマンスパラメーター、と既知の問題はデータアクセス技術および使用されるドライバーにより異なります。

データベース接続ウィザードの開始

データベース接続を必要とするアクションには、接続をセットアップするために必要なステップを案内するウィザードが表示されます。

ウィザードのステップを開始する前に、データベースの一部の種類は、データベースドライバーまたはデータベースクライアントソフトウェアなどの複数のデータベースの必須条件を個別にインストールまたは構成する必要があることに注意してください。これらは、通常各データベースのベンダーにより提供され、特定のWindows バージョン用に調整されたドキュメントを含みます。データベースの種類別にグループ化されたデータベースドライバーのリストに関しては、をデータベースドライバーの概要参照してください。

データベース接続ウィザードを開始するには次を行ってください：

- (ページデザインビューの)ページソースペインから、**[ソースの追加]** ボタンをクリックして、新しい構造を選択します。
- DB クエリビューから、ビューの左上の**[クイック接続]** ボタンをクリックします。

データベースの種類を選択した後、**「次へ」**をクリックします。スクリーンの指示はデータベースの種類 (ADO、ADO.NET、ODBC、JDBC) およびドライバーにより異なります。

例えば、各データベースの種類に適用できるサンプルはデータベース接続のサンプルのセクションを参照してください。各データベースへのアクセス技術に関しては以下のトピックを参照してください：

- ADO 接続の設定
- ADO.NET 接続の設定
- ODBC 接続の設定
- JDBC 接続の設定

データベースドライバー概要

以下のテーブル、特定のデータベースに特定のデータアクセス技術を使用して接続するためのはよく使用されるデータベースのドライバーのリストです。このリストは、完全または規範的なものではありません。以下に表示されているドライバに加え他のネイティブまたはサードパーティー代替を使用することができます。

Windows オペレーティングシステムの複数のデータベースドライバーが既に使用されている場合がありますが、代替ドライバーをダウンロードする必要があります。データベースの一部に対して、オペレーティングシステムに付随するドライバーよりもデータベースベンダーより提供される最新のドライバーのパフォーマンスの方が良い場合があります。

データベースのベンダーは、個別にダウンロードすることのできるパッケージ、または、バンドルされたデータベースクライアントソフトウェアを提供します。後者の場合、データベースクライアントソフトウェアは、通常、データベースの管理と接続、およびデータベースクライアントとそのコンポーネントのインストールと構成を簡素化するために管理と構成ユーティリティにより構成されています。

データベースクライアントを正確に構成することは、データベースへの接続を成功させるために必須です。データベースをインストールして、使用する前に、各データベースバージョンおよび各 Windows バージョンにより通常異なるデータベースのインストールおよび構成の手順を確認することが奨励されます。

データベースの種類に関して各データアクセステクノロジーの機能と制限を理解するには、特定のデータベース製品のドキュメントを参照し、特定の環境に対しての接続をテストします。よく発生する接続問題を回避するために、以下に注意してください：

- ADO.NET の一部のプロバイダーはサポートされない、または、サポートが制限されています。次を参照してください：ADO.NET サポートメモ。
- データベースドライバーをインストールする際、Altova アプリケーション (32-ビット または 64-ビット) などの同じプラットフォームが使用されることが奨励されます。例えば、32ビットのAltova アプリケーションが64-ビットオペレーティングシステムで使用される場合、が使用される場合、32-ビットのドライバーをインストールし、32-ビットのドライバーを使用して、データベースの接続をセットアップします。詳細に関しては、使用可能な ODBC ドライバーの表示 を参照してください。
- ODBC データソースをセットアップする際、データソース名 (DSN) をユーザー *DSN* の代わりに *システム DSN* として作成することが奨励されます。詳細に関しては、使用可能な ODBC ドライバーの表示 を参照してください。
- JDBC データソースをセットアップする際、JRE (Java Runtime Environment) 、Java Development Kit (JDK) がインストールされている、またオペレーティングシステムの CLASSPATH 環境変数が構成されていることを確認してください。詳細に関しては、JDBC 接続のセットアップを参照してください。 .
- データベースベンダーからインストールするドライバーおよびデータベースクライアントソフトウェアのインストールの手順とサポートの詳細は、インストールパッケージのドキュメントを参照してください。公式なまたはサードパーティデータベースドライバーの場合、特定のオペレーティングシステムでの特定のドライバーに適用することのできる最も包括的な情報と構成手順は、通常ドライバーインストールパッケージの一部です。

データベース	インターフェイス	ドライバー
Firebird	ADO.NET	Firebird ADO.NET データプロバイダー (http://www.firebirdsql.org/en/additional-downloads/)
	JDBC	Firebird JDBC ドライバー (http://www.firebirdsql.org/en/jdbc-driver/)
	ODBC	Firebird ODBC ドライバー (http://www.firebirdsql.org/en/odbc-driver/)
IBM DB2	ADO	DB2 のための IBM OLE DB プロバイダー
	ADO.NET	.NET のための IBM Data Server プロバイダー
	JDBC	JDBC と SQLJ のための IBM Data Server ドライバー

データベース	インターフェイス	ドライバー
	ODBC	IBM DB2 ODBC ドライバー
IBM DB2 for i	ADO	• i5/OS IBMDA400 OLE DB プロバイダーのためのIBM DB2 • i5/OS IBMDARLA OLE DB プロバイダーのためのIBM DB2 • i5/OS IBMDASQL OLE DB プロバイダーのためのIBM DB2
	ADO.NET	IBM i のための.NET Framework Data プロバイダー
	JDBC	Java JDBC Driver のためのIBM ツールボックス
	ODBC	iSeries Access ODBC ドライバー
IBM Informix	ADO	IBM Informix OLE DB プロバイダー
	JDBC	IBM Informix JDBC ドライバー
	ODBC	IBM Informix ODBC ドライバー
Microsoft Access	ADO	• Microsoft Jet OLE DB プロバイダー • Microsoft Access Database Engine OLE DB プロバイダー
	ADO.NET	OLE DB のためのNET Framework データプロバイダー
	ODBC	• Microsoft Access ドライバー
MariaDB	ADO.NET	MariaDB のための専用の NET コネクタが不足している場合、MySQL のための **コネクタ/NET** を使用してください (http://dev.mysql.com/downloads/connector/net/)。
	JDBC	MariaDB コネクタ/J (https://downloads.mariadb.org/)
	ODBC	MariaDB コネクタ/ODBC (https://downloads.mariadb.org/)
Microsoft SQL Server	ADO	• SQL Server のためのMicrosoft OLE DB プロバイダー • SQL Server ネイティブなクライアント
	ADO.NET	• SQL Server のための.NET Framework データプロバイダー • OLE DB のための.NET Framework データプロバイダー
	JDBC	• SQL Server のためのMicrosoft JDBC ドライバー(http://msdn.microsoft.com/en-us/data/aa937724.aspx)
	ODBC	• SQL Server ネイティブなクライアント
MySQL	ADO.NET	コネクタ/NET (http://dev.mysql.com/downloads/connector/net/)
	JDBC	コネクタ/J (http://dev.mysql.com/downloads/connector/j/)
	ODBC	コネクタ/ODBC (http://dev.mysql.com/downloads/connector/odbc/)
Oracle	ADO	• OLE DB のためのOracle プロバイダー • Oracle のためのMicrosoft OLE DB プロバイダー
	ADO.NET	.NET のための Oracle データプロバイダー(http://www.oracle.com/technetwork/topics/dotnet/index-085163.html)
	JDBC	• JDBC Thin ドライバー • JDBC Oracle Call Interface (OCI) ドライバー これらのドライバーは、通常、データベースクライアントのインストール中にインストールさ

データベース	インターフェイス	ドライバー
		れます。(Thin ドライバーではない) OCI ドライバーを使用する接続は、Oracle XML DB のコンポーネントを使用しています。
	ODBC	• Oracle のための Microsoft ODBC • Oracle ODBC ドライバー(通常、Oracle データベースクライアントのインストール中にインストールされます)。
PostgreSQL	JDBC	PostgreSQL JDBC ドライバー(https://jdbc.postgresql.org/download.html)
	ODBC	psqlODBC (https://odbc.postgresql.org/)
	ネイティブな接続	使用可能。ネイティブな接続を使用する場合、ドライバーをインストールする必要はありません。
Progress OpenEdge	JDBC	JDBC コネクタ(https://www.progress.com/jdbc/openedge)
	ODBC	ODBC コネクタ(https://www.progress.com/odbc/openedge)
SQLite	ネイティブな接続	使用可能。ネイティブな接続を使用する場合、ドライバーをインストールする必要はありません。
Sybase	ADO	Sybase ASE OLE DB プロバイダー
	JDBC	JDBC のための jConnect™
	ODBC	Sybase ASE ODBC ドライバー
Teradata	ADO.NET	Teradata のための .NET データプロバイダー(http://downloads.teradata.com/download/connectivity/net-data-provider-for-teradata)
	JDBC	Teradata JDBC ドライバー(http://downloads.teradata.com/download/connectivity/jdbc-driver)
	ODBC	Windows のための Teradata ODBC Driver (http://downloads.teradata.com/download/connectivity/odbc-driver/windows)

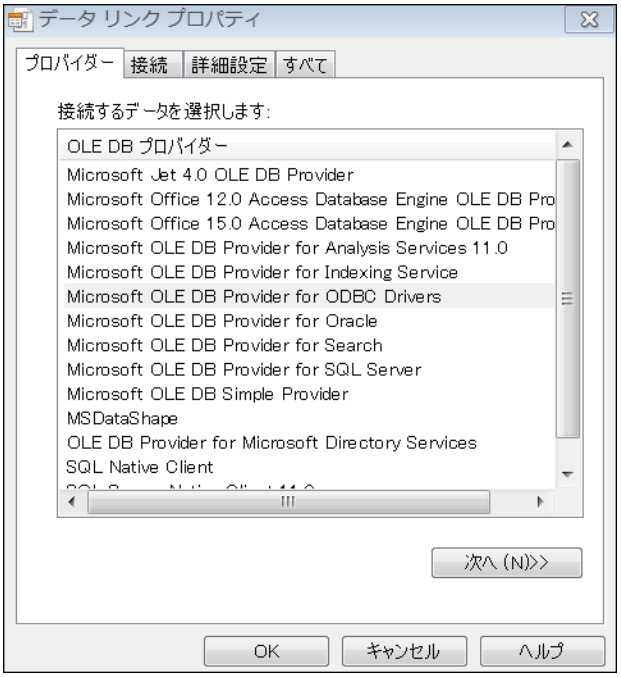
ADO 接続の設定

Microsoft ActiveX Data Objects (ADO) は、OLE DB を介して多種のデータソースに接続することのできるデータアクセス技術です。OLE DB は、ODBC または JDBC の代替インターフェイスので、COM (Component Object Model コンポーネント・オブジェクト・モデル) 環境内で、データへの一定のアクセスを提供します。ADO は、Microsoft Access および SQL Server 等の Microsoft ネイティブなデータベースへ接続する典型的な選択ですが、他のデータソースにも使用することができます。

ADO 接続のセットアップ:

1. データベース接続ウィザードの開始。
2. **[ADO 接続]** をクリックします。

3. **[構築]** をクリックします。



4. 接続するデータプロバイダーを選択します。下のテーブルは、よくある例をリストしています。

接続するデータベース...	使用するプロバイダー ...
Microsoft Access	• **Microsoft Office Access Database Engine OLE DB Provider** Microsoft Access 2003 に接続する際には、**Microsoft Jet OLE DB Provider** も使用することができます。
SQL Server	• **SQL Server Native Client** • **Microsoft OLE DB Provider for SQL Server**
他のデータベース	データベースに適用することのできるプロバイダーを選択します。 データベースへのOLE DB プロバイダーが使用できない場合は、データベースベンダー(データベースドライバー概要を参照)から必要なドライバーをインストールします。別の方法として、ODBC またはJDBC 接続をセットアップします。 オペレーティングシステムに必要なデータベースに対するODBC ドライバーがある場合、**Microsoft OLE DB Provider for ODBC Drivers** を使用することができます。

5. **「次へ」**をクリックして、ウィザードを終了します。

次のウィザードのステップは選択されたプロバイダーに固有の手順です。SQL Server には、データベースユーザー名とパスワードに加え、データベースサーバーのホスト名を提供またはお選択する必要があります。Microsoft Access には、データベースファイルへのパスを参照または提供するように求められます。

初期化プロパティ(接続パラメーター) の完全なリストは、接続ダイアログボックスの**すべて** タブで使用することができます？これらのプロパティは、選択されたプロバイダーにより異なります。次のセクションはMicrosoft Access and SQL Server データベースのための基本的な初期化プロパティの手引きを説明しています:

- SQL サーバー データリンクプロパティのセットアップ
- Microsoft Access データリンクプロパティのセットアップ

既存の Microsoft Access データベースへの接続

このアプローチは、パスワードで保護されていないMicrosoft Access データベースに接続する場合に適しています。データベースのパスワードの設定は、Microsoft Access (ADO) への接続で説明されています。

既存のMicrosoft Access データベースへの接続方法:

1. データベース接続ウィザードを実行します (データベース接続ウィザードの開始を参照)。
2. **Microsoft Access (ADO)** を選択して、**「次へ」**をクリックします。

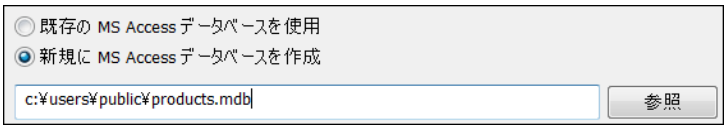
3. **[既存のMS Access データベースを使用]** を選択します。
4. データベースファイルを参照、または、(相対または絶対) パスを入力します。
5. **「接続」**をクリックします。

新規の Microsoft Access データベースの作成

既存のデータベースファイルに接続する代替として、Microsoft Access が使用中のマシンにインストールされていない場合でも、新規のMicrosoft Access データベースファイル(.accdb, .mdb) を作成して、接続することができます。MobileTogether Designer により作成されるデータベースは、空です。Microsoft Access または、DatabaseSpy (http://www.altova.com/ja/databasespy.html) などのツールを使用して、必要とされるデータベース構造を作成し、データを追加します。

新規の Microsoft Access データベースの作成方法:

1. データベース接続ウィザードを実行します (データベース接続ウィザードの開始を参照)。
2. **Microsoft Access (ADO)** を選択して、**「次へ」** をクリックします。



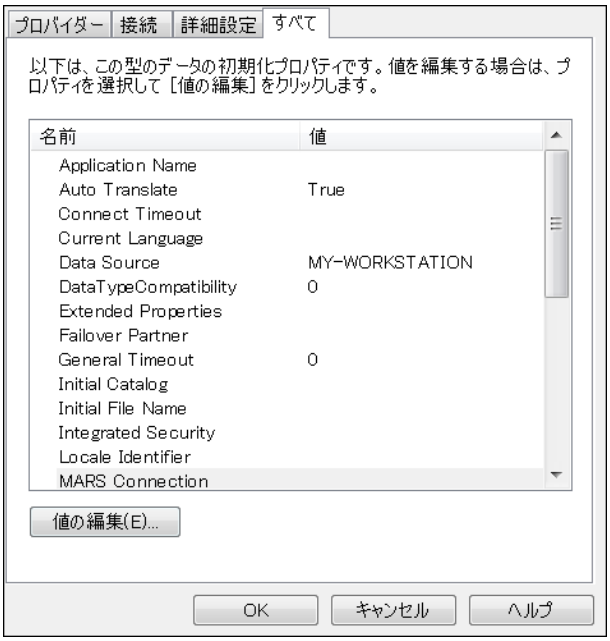
3. **[新規 MS Access データベースの作成]** を選択して、作成されるデータベースファイルの(相対または絶対)パスを入力します (例えば、**c:\users\public\products.mdb**)。または、**[参照]** をクリックして、フォルダーを選択して、ファイル名テキストボックスにデータベースファイルの名前(例えば、**products.mdb**)を入力し、**[保存]** をクリックします。

> **メモ**
> - データベースファイルを作成する場所に対して、書き込みアクセスの許可があることを確認してください。
> - **.accdb** フォーマットでファイルを作成する場合、Microsoft Access 2007 または以降が使用するコンピューターにインストールされている必要があります。**.mdb** フォーマットでファイルを作成する場合、上記の条件は必要ありません。

4. **「接続」** をクリックします。

SQL Server データ リンク プロパティの設定

Microsoft SQL Server データベースにADO (ADO 接続のセットアップを参照) を介して接続する場合、次のデータリンク プロパティが**全ての** タブのデータリンク プロパティダイアログボックスで正しく構成されていることを確認してください。

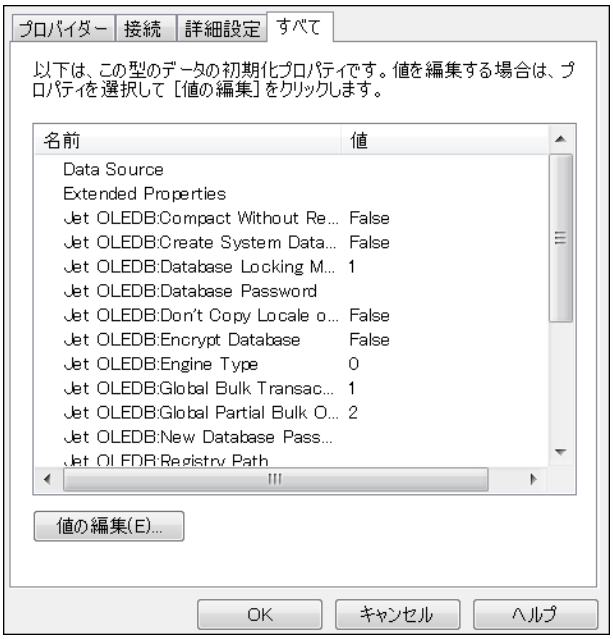


データ リンク プロパティダイアログ ボックス

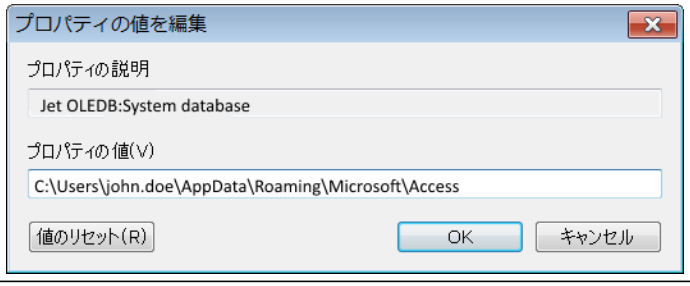
プロパティ	メモ
統合セキュリティ	**プロバイダー**タブで **SQL Server Native Client** がデータプロバイダーに選択された場合は、このプロパティを、空白文字に設定します。
セキュリティ情報を保持	このプロパティを**True** に設定する。

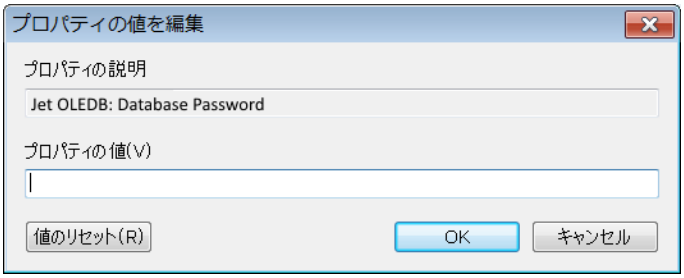
Microsoft Access データ リンク プロパティの設定

Microsoft Access データベースにADO (ADO 接続のセットアップを参照) を介して接続する場合、次のデータリンクプロパティが**全ての**タブのデータリンクプロパティダイアログボックスで正しく構成されていることを確認してください。



データリンクプロパティダイアログボックス

プロパティ	メモ
データソース	このプロパティは、Microsoft Access データベースファイルへのパスを保管します。データベースへの接続問題を回避するために、UNC (Universal Naming Convention 汎用名前付け規則) パスフォーマットの使用が奨励されます。例: `\\anyserver\share$\filepath`
Jet OLEDB: システムデータベース	このプロパティはワークグループ情報ファイルへのパスを保管します。データベースに接続する前に、このプロパティを明示的に設定する必要がある可能性があります。 "ワークグループ情報ファイル"エラーが理由で接続できない場合、ワークグループ情報ファイル (**System.MDW**) をユーザープロファイルに適用できるように設定します (手順は http://support.microsoft.com/kb/305542 を参照してください)、プロパティの値を **System.MDW** ファイルのパスに設定します。 

プロパティ	メモ
Jet OLEDB: データベースパスワード	データベースがパスワードで保護されている場合は、このプロパティの値をデータベースのパスワードに設定します。 

ADO.NET 接続のセットアップ

ADO.NET は、データベースからのデータを含むデータと作業するためにデザインされたMicrosoft .NET フレームワークライブラリのセットです。MobileTogether Designer から、データベースに接続するには、ADO.NET、Microsoft .NET フレームワーク4、または、以降を使用する必要があります。下に表示されているとおり、 NET プロバイダーを選択して、接続文字列を提供し、データベースにADO.NET を介して接続します。

.NET データプロバイダーは、コマンドを実行する特定の型のデータソースへの接続を可能にするクラスのコネクションです (例えば、SQL Server またはOracle データベースなど)。すなわち、ADO.NET とは、MobileTogether Designer などのアプリケーションがデータベースプロバイダーを介してデータベースと作動することを意味します。各 データプロバイダー は特定の型のデータソースと作動するように最適化されています。NET プロバイダーには以下の2つの種類があります:

1. デフォルトとしてMicrosoft .NET フレームワークを使用して提供されます。
2. 主要なデータベースベンダーを使用して NET フレームワークの拡張として提供されます。このようなADO.NET プロバイダーは、個別人ストールされている必要があり、通常それぞれのデータベースベンダーのウェブサイトからダウンロードすることができます。

メモ　特定のADO.NET プロバイダーは、サポートされていない、または、サポートが制約されています。ADO.NET サポートに関するメモを参照してください。

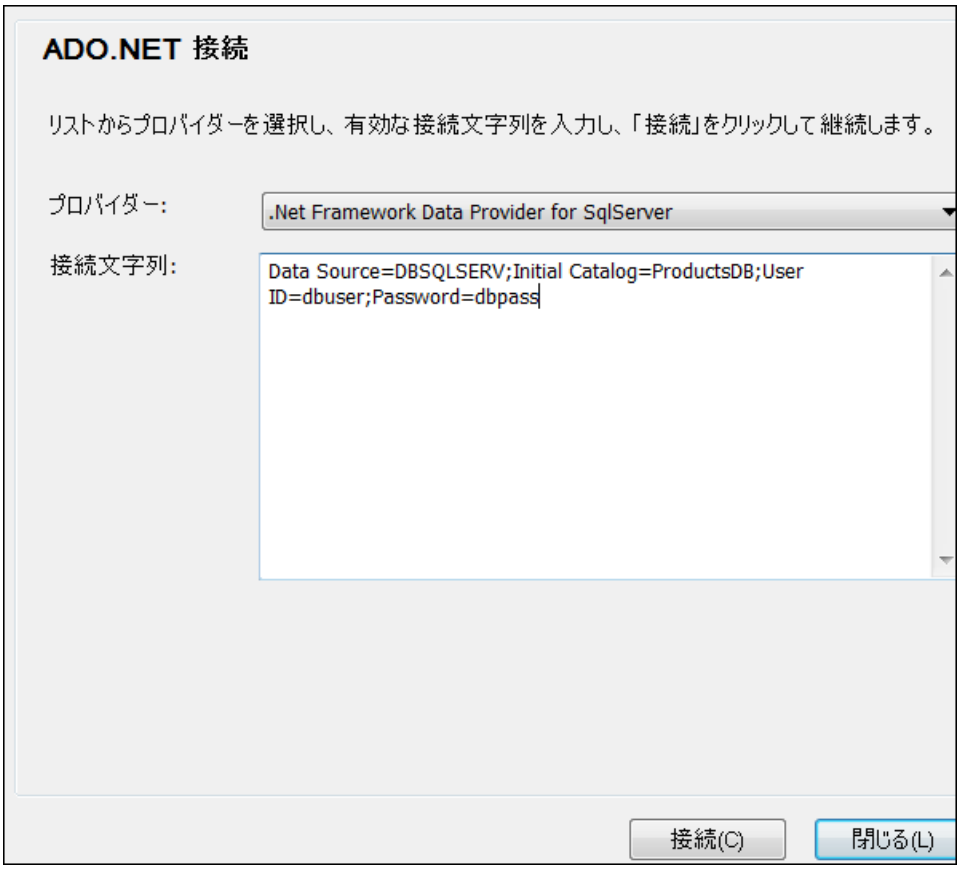
ADO.NET 接続の作成:

1. データベース接続ウィザードを開始します。
2. 「**ADO.NET 接続**」をクリックします。
3. .NET データ プロバイダーをリストから選択します。

> .NET フレームワークと共に使用することのできるプロバイダーのリストはデフォルトでプロバイダーのリストに表示されます。ベンダーに特有の NET データプロバイダーは、使用中のシステムに既にインストールされている場合のみ使用することができます。 NET データ プロバイダーは、データベースベンダーにより提供される、msi、または、exe ファイルを実行して、GAC (Global Assembly Cache) 内にインストールされる必要があります。

4. データベース接続文字列を入力します。接続文字列は、データベース接続情報をセミコロンで区切られた接続パラメーターのキー値ペアとして定義します。例えば、**`Data Source=DBSQLSERV;Initial Catalog=ProductsDB;User ID=dbuser;password=dbpass`** などの文字列は、サーバー `DBSQLSERV` 上で、ユーザー名 `dbuser` とパスワード `dbpass` を使用して、SQL Server データベース `ProductsDB` に接続します。接続文字列を、キー値ペアを直接「接続文字列」ダイアログボックスに入力することにより作成することができます。他の接続オプションはVisual Studio で作成することができます (次を参照: Visual Studio 内で接続文字列を作成する)。

> 接続文字列の構文は、プロバイダーリストより選択されたプロバイダーにより異なります。例えば、次を参照してください：サンプルADO.NET 接続文字列。



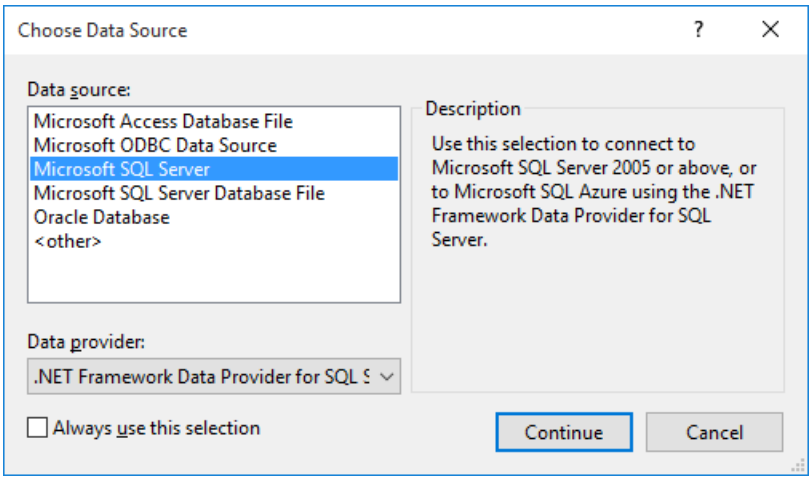
5. **接続**」をクリックします。

Visual Studio 内で接続文字列を作成する

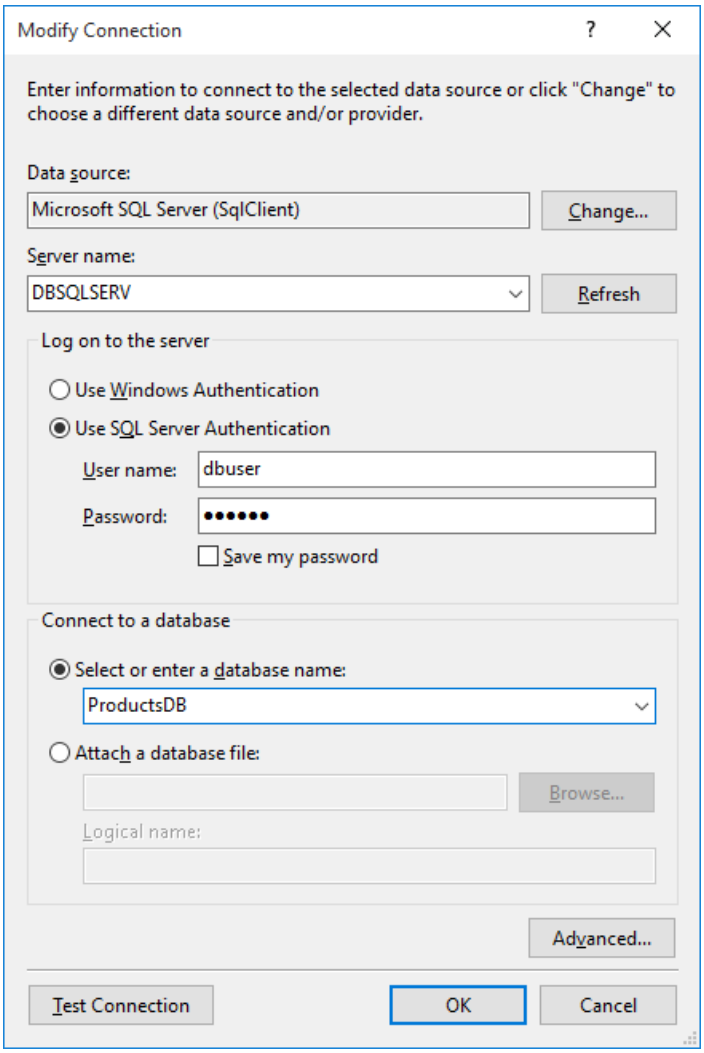
ADO.NET を使用してデータソースに接続するには、有効な接続文字列が必要です。次の手順は、Visual Studio から接続文字列を作成する方法を説明しています。

Visual Studio 内で接続文字列を作成する

1. **「ツール」**メニューから、**「データベースへの接続」**をクリックします。
2. リストからデータソースを選択します(この例の場合、Microsoft SQL Server)。データプロバイダーは、選択に従い自動的に記入されます。



3. **「接続」**をクリックします。



4. データベースへのサーバーホスト名、ユーザー名、およびパスワードを入力します。この例の場合、サーバー `DBSQLSERV` 上のデータベース `ProductsDB` にSQL Server 認証を使用して接続します。
5. **「OK」**をクリックします。

データベースへの接続に成功した場合、サーバーエクスプローラー ウィンドウに表示されます。サーバーエクスプローラー ウィンドウを、メニューコマンド **「表示 | サーバーエクスプローラ」**を使用して表示することができます。データベース接続文字列を取得するには、サーバーエクスプローラー ウィンドウ内の接続を右クリックし、**「プロパティ」**を選択します。接続文字列が Visual Studio プロパティウィンドウに表示されます。MobileTogether Designer の「接続文字列」ボックスに文字列を貼り付ける前に、アスタリスク(*) 文字を実際のパスワードと置き換える必要があることに注意してください。

サンプル ADO.NET 接続文字列

ADO.NET 接続をセットアップするには、データベース接続ダイアログボックスからADO.NET プロバイダーを選択し、接続文字列を入力します。(次も参照してください ADO.NET 接続のセットアップ)。異なるデータベースのサンプル ADO.NET 接続文字列は .NET プロバイダーの下にリストされています。

Teradata のための NET Data

このプロバイダーは、次の Teradata Web サイトからダウンロードすることができます (http://downloads.teradata.com/download/connectivity/net-data-provider-for-teradata)。サンプル接続は以下のとおりです:

```
Data Source=ServerAddress;User Id=user;Password=password;
```

IBM i のための NET Framework データプロバイダー

このプロバイダーは、*IBM i Access Client Solutions - Windows Application Package* の一部としてインストールされます。サンプル接続文字列は以下のとおりです:

```
DataSource=ServerAddress;UserID=user;Password=password;DataCompression=True;
```

詳細に関しては、インストールパッケージ内に含まれる「NET プロバイダー技術レファレンスヘルプファイル」を参照してください。

MySQL のための NET Framework データプロバイダー

このプロバイダーを MySQL ウェブサイトからダウンロードすることができます (https://dev.mysql.com/downloads/connector/net/)。サンプル接続文字列は以下のとおりです:

```
Server=127.0.0.1;Uid=root;Pwd=12345;Database=test;
```

次も参照してください: https://dev.mysql.com/doc/connector-net/en/connector-net-programming-connecting-connection-string.html

SQL Server のための NET Framework データプロバイダー

サンプル接続文字列は以下のとおりです:

```
Data Source=DBSQLSERV;Initial Catalog=ProductsDB;User
ID=dbuser;Password=dbpass
```

次も参照してください: https://msdn.microsoft.com/en-us/library/ms254500(v=vs.110).aspx

.NET Framework 4.0 のための IBM DB2 データプロバイダー 10.1.2

```
Database=PRODUCTS;UID=user;Password=password;Server=localhost:50000;
```

メモ このプロバイダーは、通常 IBM DB2 Data Server Client パッケージと共にインストールされます。IBM DB2 Data Server Client パッケージをインストール後 ADO.NET プロバイダーのリスト内で プロバイダーが不足している場合、次の技術メモを参照してください: http://www-01.ibm.com/support/docview.wss?uid=swg21429586。

次も参照してください: http://www.ibm.com/support/knowledgecenter/en/SSEPGG_10.1.0/com.ibm.swg.im.dbclient.adonet.ref.doc/doc/DB2ConnectionClassConnectionStringProperty.html

.NET (ODP.NET) のための Oracle データプロバイダー

ODP.NET プロバイダーを含むインストールパッケージをOracle ウェブサイトからダウンロードすることができます (次を参照してください: http://www.oracle.com/technetwork/topics/dotnet/downloads/index.html). サンプル接続文字列は以下のとおりです:

```
Data Source=DSORCL;User Id=user;Password=password;
```

Oracle への接続(ODBC)内で説明されているように**tnsnames.ora** ファイル内で定義されたOracle サービス名をさす `DSORCL` がデータソース名の場合。

tnsnames.ora ファイル内のサービス名を構成せず接続する場合は、以下のような文字列を使用してください:

```
Data Source=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=host)
(PORT=port)))(CONNECT_DATA=(SERVER=DEDICATED)
(SERVICE_NAME=MyOracleSID)));User Id=user;Password=password;
```

次も参照してください: https://docs.oracle.com/cd/B28359_01/win.111/b28375/featConnecting.htm

ADO.NET サポートに関するメモ

下のテーブルは、MobileTogether Designer 内でサポートされていない、または、制限の有るサポートを持つ、既知の ADO.NET データベースドライバをリストしたテーブルです。

データベース	ドライバ	サポートに関するメモ
すべてのデータベース	**.Net Framework Data Provider for ODBC**	制限つきサポート。Microsoft Access 接続の際に既知の問題があります。ODBC への直接の接続を代わりに使用することが奨励されます。次を参照してください: ODBC 接続のセットアップ。
	.Net Framework Data Provider for OleDb	制限つきサポート。Microsoft Access 接続の際に既知の問題があります。ADO への直接の接続を代わりに使用することが奨励されます。次を参照してください: ADO 接続のセットアップ。
Firebird	**Firebird ADO.NET Data Provider**	制限つきサポート。ODBC または JDBC を代わりに使用することが奨励されます。次を参照してください: Firebird (ODBC) への接続 と Firebird (JDBC) への接続.
Informix	**IBM Informix Data Provider for .NET Framework 4.0**	サポートされていません。**DB2 Data Server Provider** を代わりに使用してください。
IBM DB2 for i (iSeries)	**.Net Framework Data Provider for i5/OS**	サポートされていません。*IBM i Access Client Solutions - Windows Application Package* の一部として提供されている **.Net Framework Data Provider for IBM i** を代わりに使用してください。
Oracle	**.Net Framework Data Provider for Oracle**	制限つきサポート。NET Framework と共にこのドライバは提供されていますが、Microsoft により使用を奨励されていません。
PostgreSQL	**-**	このベンダーのための ADO.NET ドライバはサポートされていません。
Sybase	**-**	このベンダーのための ADO.NET ドライバはサポートされていません。

ODBC 接続の設定

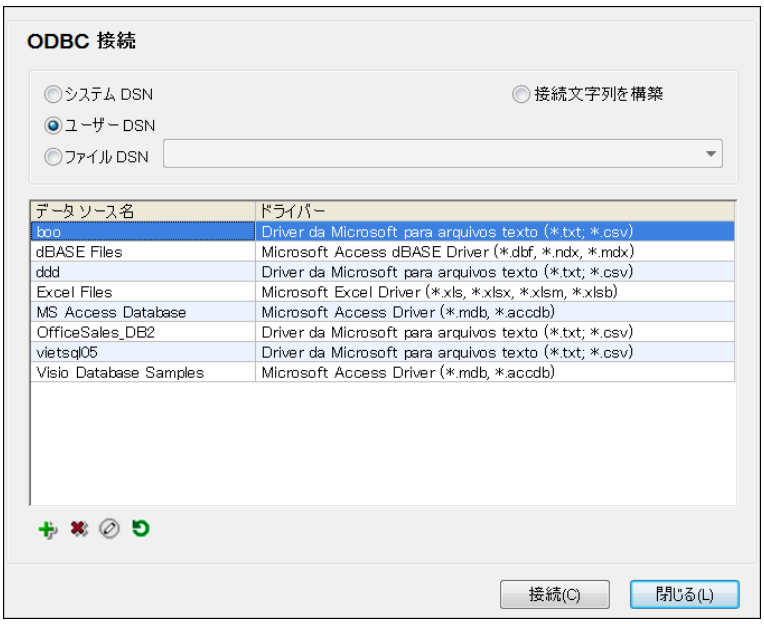
ODBC (Open Database Connectivity) は、一般に使用されているデータアクセス技術で、MobileTogether Designer からデータベースに接続することができます。データベースにアクセスするための主要な手段として、または OLE DB- や JDBC- による接続の代替として使用することができます。

ODBC を使用してデータベースに接続するには、最初に ODBC データソース名(DSN) を作成する必要があります。このステップは、他のユーザーまたはオペレーティングシステムにより DSN が既に作成されている場合必要ありません。DSN は、統一された、MobileTogether Designer を含むオペレーティングシステム上の ODBC に対応したクライアントアプリのデータベース接続方法を表します。DSN には以下の種類があります:

- システム DSN
- ユーザー DSN
- ファイル DSN

オペレーティングシステムで特権を持つすべてのユーザーはシステム データソースにアクセスすることができます。作成したユーザーのみがユーザー データソースにアクセスすることができます。*ファイル DSN* を作成した場合、データソースは dsn 拡張子を持つファイルとして作成され、データソースにより使用されているドライバーがインストールされていることが前提で、他のユーザーと共有することができます。

ODBC 接続ダイアログボックス内の **[ODBC 接続]** をクリックすると、使用中のコンピューターで既に使用することのできる DSN はデータベース接続ダイアログボックスに表示されています。



ODBC 接続ダイアログボックス

データベースにより必要とされる DSN が無い場合、MobileTogether Designer データベース接続ウィザードが作成の手助けをします。しかし、Windows オペレーティングシステムに直接作成することもできます。いずれにせよ、続行する前に、使用中のデータベースに ODBC ドライバーがオペレーティングシステムの ODBC ドライバーリストに適用できることを確認してください。(使用可能な ODBC ドライバーの表示を参照)。

新規 DSN を使用しての接続:

1. データベース接続ウィザードを開始します。
2. データベース接続ダイアログボックスの **[ODBC 接続]** をクリックします。
3. ソースの種類を選択します (ユーザー DSN、システム DSN、ファイル DSN)。

システムDSN を作成するには、オペレーティングシステムの管理者権限が必要です。

4. **「追加」** をクリックします。
5. ドライバーを選択して、(作成するDSN の種類に応じて) **[ユーザーDSN]** または **[システムDSN]** をクリックします。使用中のデータベースに適用することのできるドライバーがリストされていない場合、データベースベンダーからダウンロードして、インストールしてください。(次を参照: データベースドライバー概要)。
6. ポップアップするダイアログボックスに特定の接続情報を入力してセットアップを完了します。

接続を成功させるには、データベースサーバーのホスト名(またはIP アドレス)、とデータベースのユーザー名およびパスワードを提供する必要があります。他の接続パラメーターがある可能性がありますが、これらのパラメーターはデータベースプロバイダーにより異なります。各接続方法に固有のパラメーターについての詳細は、ドライバープロバイダーのドキュメントを参照してください。一度作成されると、DSN はデータソース名のリストで使用することができます。ユーザーDSN はユーザーDSN のリストに追加されますが、システムDSN はシステムDSN のリストに追加されます。

既存のDSNを使用しての接続:

1. データベース接続ウィザードを開始します.
2. **[ODBC 接続]** をクリックします。
3. 既存のデータソース(ユーザーDSN、システムDSN、ファイルDSN) から種類を選択します
4. 既存のDSN をクリックして、**「接続」** をクリックします。

既存の dsn ファイルをベースにして接続文字列を構築する

1. データベース接続ウィザードを開始します.
2. **[ODBC 接続]** をクリックします。
3. **[接続文字列の構築]** を選択して、**[構築]** をクリックします。
4. ファイルDSN を使用して、接続文字列を構築する場合、**[ファイルデータソース]** タブをクリックします。それ以外の場合は、**[マシーンデータソース]** タブをクリックします。(システムDSN とユーザーDSN は "マシン" データソースとして知られています。)
5. 必要とされる dsn ファイルを選択して、**「OK」** をクリックします。

用意された接続文字列を使用して接続する

1. データベース接続ウィザードを開始します.
2. **[ODBC 接続]** をクリックします。
3. **[接続文字列の構築]** を選択しています。
4. 接続文字列を与えられたボックスに貼り付け、**「接続」** をクリックします。

JDBC 接続の設定

JDBC (Java Database Connectivity) は、Oracle のJava ソフトウェアプラットフォームの一部であるデータベースアクセスインターフェイスです。JDBC 接続はODBC 接続に比べ、通常よりもリソースを集中的に使用する接続ですが、ODBC で使用することのできない機能を使用することができます。ODBC コネクタにより使用できないデータベース機能を使用する場合 JDBC 接続を使用することが奨励されています。

前提条件

- Java Runtime Environment (JRE) またはJava Development Kit (JDK) がオペレーティングシステムにインストールされている必要があります。インストールされていない場合はJava の公式 Web サイトからパッケージとインストールの手順をダウンロードしてください。
- データベースベンダーからのJDBC ドライバーがインストールされている必要があります。Oracleデータベースに接続する場合は、一部のOracle ドライバーは特定のJRE バージョンに固有のため、追加コンポーネントと構成が必要な点に注意してください。Oracle 製品のドキュメント (例えば、『JDBC開発者ガイドおよびリファレンス』) には、各 JDBC ドライバーの構成の手続きの詳細が説明されています。
- オペレーティングシステムの`PATH` 環境変数は、JRE またはJDK インストールディレクトリの`bin` ディレクトリへのパスを含む必要があります。例えば、`C:\Program Files (x86)\Java\jre1.8.0_51\bin` など。
- `CLASSPATH` 環境変数は、(.jar ファイルの1つまたは複数の) 使用中のWindows オペレーティングシステムのJDBC ドライバーのパスを含む必要があります。データベースクライアントをインストールする際に、インストーラーはこの変数を自動的に構成する可能性があります。JDBC ドライバーのドキュメントは通常ステップバイステップでCLASSPATH 変数の設定方法を説明しています (CLASSPATH の構成を参照)。

JDBC 接続の設定方法

1. データベース接続ウィザードを開始します。
2. **「JDBC 接続」**をクリックします。
3. 任意で、セミコロンで区切られた jar ファイルパスのリストを「クラスパス」テキストボックスに入力することもできます。ここで入力される.jar ライブラリは、`CLASSPATH` 環境変数内で既に定義されているものに加え、環境にロードされます。「クラスパス」テキストボックスの編集が完了すると、ソース jar ライブラリ内で検索されるJDBC ドライバーは、「ドライバー」リストに自動的に追加されます (次のステップを参照してください)。
4. 「ドライバー」の横でJDBC ドライバーをリストから選択、または、Java クラス名を入力します。`CLASSPATH` 環境変数により構成されたJDBC ドライバーおよび、「クラスパス」テキストボックス内で検出されたドライバーが含まれます(次を参照: CLASSPATH の構成)。

> CLASSPATH 変数内で定義されているJDBC ドライバーのパスを使用することができます。また、データベース接続ダイアログボックスに直接入力された jar パスはJava Virtual Machine (JVM) によりサポートされています。JVM は接続を構築するためのドライバーを決定します。JVM にロードされるJava クラスを確認し、JDBC ドライバーの競合、および、データベースに接続する際に発生する予期されないエラーを回避するようにしてください。

5. 対応するボックスに、ユーザー名とパスワードを入力します。
6. データベースURL テキストボックス内に、JDBC 接続 URL (文字列) をデータベースの型のとおり入力します。次のテーブルは一般のデータベースの型のためのJDBC 接続 URL (文字列) の構文について説明しています。

データベース	JDBC 接続 URL
Firebird	`jdbc:firebirdsql://<host>[:<port>]/<database path or alias>`
IBM DB2	`jdbc:db2://hostName:port/databaseName`
IBM Informix	`jdbc:informix-sqli://hostName:port/databaseName:INFORMIXSERVER=myserver`

データベース	JDBC 接続 URL
MariaDB	`jdbc:mariadb://hostName:port/databaseName`
Microsoft SQL Server	`jdbc:sqlserver://hostName:port;databaseName=name`
MySQL	`jdbc:mysql://hostName:port/databaseName`
Oracle	`jdbc:oracle:thin:@//hostName:port:service`
Oracle XML DB	`jdbc:oracle:oci:@//hostName:port:service`
PostgreSQL	`jdbc:postgresql://hostName:port/databaseName`
Progress OpenEdge	`jdbc:datadirect:openedge://host:port;databaseName=db_name`
Sybase	`jdbc:sybase:Tds:hostName:port/databaseName`
Teradata	`jdbc:teradata://databaseServerName`

メモ 上記のフォーマットに対する構文のバリエーションも使用することができます (例えば、データベースURL はポートを除外、またはデータベースのユーザー名とパスワードを含む可能性があります)。詳細はデータベースのベンダーのドキュメントを参照してください。

7. **接続**をクリックします。

CLASSPATH の構成

CLASSPATH 環境変数はクラスと、オペレーティングシステム上の他のリソースファイルを検索するため Java Runtime Environment (JRE) により使用されます。JDBC を介してデータベースに接続すると、この変数はオペレーティングシステム上の JDBC ドライバーへのパス、または一部の場合、使用中のデータベースの種類に固有の追加ライブラリファイルへのパスを含むように構成されなければなりません。

下のテーブルは、CLASSPATH 変数に通常含まれる必要のあるサンプルファイルパスをリストしています。重要なことは、システム上の JDBC ドライバのロケーション、JDBC ドライバー名、オペレーティングシステム上のドライバーの場所をベースにこの情報を調整する必要があります。接続問題を避けるために、使用中のオペレーティングシステムにインストールされている JDBC ドライバに適用されるインストールの指示、インストール前とインストール後の構成ステップを確認してください。

データベース	サンプル CLASSPATH エントリ
Firebird	`C:\Program Files\Firebird\Jaybird-2.2.8-JDK_1.8\jaybird-full-2.2.8.jar`
IBM DB2	`C:\Program Files (x86)\IBM\SQLLIB\java\db2jcc.jar;C:\Program Files (x86)\IBM\SQLLIB\java\db2jcc_license_cu.jar;`
IBM Informix	`C:\Informix_JDBC_Driver\lib\ifxjdbc.jar;`
Microsoft SQL Server	`C:\Program Files\Microsoft JDBC Driver 4.0 for SQL Server\sqljdbc_4.0\enu\sqljdbc.jar`
MariaDB	**`<installation directory>`**`\mariadb-java-client-2.2.0.jar`
MySQL	**`<installation directory>`**`\mysql-connector-java-`*`version`*`-bin.jar;`
Oracle	**`ORACLE_HOME`**`\jdbc\lib\ojdbc6.jar;`
Oracle (with XML DB)	**`ORACLE_HOME`**`\jdbc\lib\ojdbc6.jar;`**`ORACLE_HOME`**`\LIB\xmlparserv2.jar;`**`ORACLE_HOME`**`\RDBMS\jlib\xdb.jar;`
PostgreSQL	**`<installation directory>`**`\postgresql.jar`
Progress OpenEdge	`%DLC%\java\openedge.jar;%DLC%\java\pool.jar;` メモ: Progress OpenEdge SDK がマシンにインストールされていると想定します。`%DLC%` は OpenEdge がインストールされているディレクトリです。
Sybase	`C:\sybase\jConnect-7_0\classes\jconn4.jar`
Teradata	**`<installation directory>`**`\tdgssconfig.jar;`**`<installation directory>`**`\terajdbc4.jar`

- CLASSPATH 変数の変更は、使用中のコンピュータの Java アプリケーションの振る舞いに影響する可能性があります。続行する前に生じる影響を把握するために、Java ドキュメントを参照してください。
- 環境変数は、ユーザーまたはシステムであることができます。システム環境変数を変更するには、オペレーティングシステムに対する管理者権限が必要です。
- 環境変数を変更後、設定の変更を有効にするには、プログラムを再起動してください。または、オペレーティングシステムをログオフまたは再起動してください。

Windows 7 で CLASSPATH を構成する

1. **[スタート]** メニューを開始して、**[コンピューター]** を右クリックします。

2. **[プロパティ]** をクリックします。
3. **[システムの詳細設定]** をクリックします。
4. **[詳細設定]** タブから**[環境変数]** をクリックします。
5. CLASSPATH 変数を、ユーザーまたはシステム環境変数からコピーして、編集をクリックします。
   CLASSPATH 変数が存在しない場合は、**[新規]** をクリックして作成します。
6. オペレーティングシステム内でJDBC ドライバーがある場所にパスを含むように変数の値を編集します。JDBC ドライバーパスを他の既にCLASSPATH 変数内にあるパスから区別するためにセミコロン(;)で区切ります。

Windows 10 でCLASSPATH を構成する

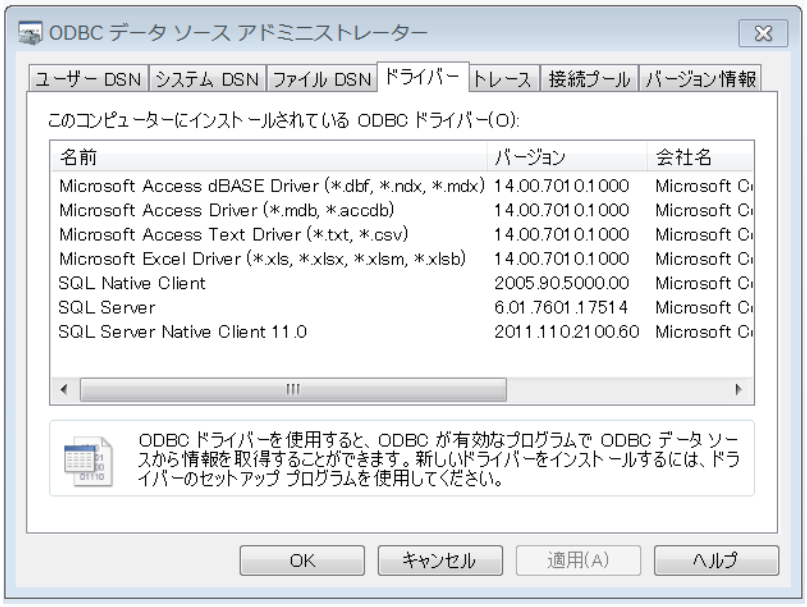
1. Windows キーを押して、環境編集を入力します。
2. **[システム環境変数の編集]** の提案をクリックします。
3. **[環境変数]** をクリックします。
4. CLASSPATH 変数を、ユーザーまたはシステム環境変数からコピーして、編集をクリックします。
   CLASSPATH 変数が存在しない場合は、**[新規]** をクリックして作成します。
5. オペレーティングシステム内でJDBC ドライバーがある場所にパスを含むように変数の値を編集します。JDBC ドライバーパスを他の既にCLASSPATH 変数内にあるパスから区別するためにセミコロン(;)で区切ります。

使用可能な ODBC ドライバーの表示

ODBC データソース管理で使用中のオペレーティングシステムで使用可能な ODBC ドライバーを確認することができます。Windows コントロールパネルの**管理ツール**から ODBC データソース管理 (**Odbcad32.exe**) にアクセスすることができます。64-ビット版オペレーティングシステムには、2 種類のバージョンの実行可能ファイルがあります:

- **C:\Windows\SysWoW64** ディレクトリにある、**Odbcad32.exe** ファイルの 32-ビット版バージョン(**C:** が使用中のシステムドライブとして前提されています)。
- **C:\Windows\System32** ディレクトリにある、**Odbcad32.exe** ファイルの 64-ビット版バージョン

インストールされている 32-ビット版データベースドライバーは、ODBC データソース管理の 32-ビット版バージョンで、64-ビット版バージョン内の 64-ビット版ドライバーで確認することができます。ですから、対応するバージョンの ODBC データソース管理でデータベースドライバーを確認してください。



ODBC データソース管理者

ターゲットデータベースのドライバーがリストに存在しない場合、または、代替ドライバーを追加する場合、データベースのベンダーからダウンロードしてください(データベースドライバー概要 を参照)。ODBC ドライバーがシステムで使用可能になると、ODBC 接続を作成することができます(ODBC 接続のセットアップを参照)。

PostgreSQL 接続のセットアップ

PostgreSQL データベースへの接続は、ネイティブな接続として、または、ODBC、JDBC、および他の ドライバーを使用して確立することができます。ネイティブな接続の利点は、使用中のシステムにインストールする場合、ドライバーを必要としないことです。また、Linux、または、OS X サーバー上での実行のためにファイルをデプロイする場合、ターゲットサーバー上にドライバーをインストールする必要はありません(次も参照してください: Linux とMac 上でのデータベース接続)。

ネイティブでは内ドライバーを使用しての接続を作成する場合は、以下のトピックを参照してください:

- JDBC 接続の設定
- PostgreSQL (ODBC) への接続

それ以外の場合、PostgreSQL へのネイティブの接続をセットアップする場合、以下の手順に従ってください。継続するには、以下の前提条件が必要です: ホスト名、ポート、データベース名、ユーザー名、および パスワード。

ネイティブ PostgreSQL 接続のセットアップ:

1. データベース接続ウィザードを開始します.
2. **「PostgreSQL 接続」**をクリックします。
3. 対応するボックスに、ホスト(PostgreSQL が同じマシンで作動している場合、*ローカルホスト*)、ポート(通常、5432 ですが、これは任意です)、データベース名、ユーザー名とパスワードを入力します。

4. **「接続」**をクリックします。

PostgreSQL データベースサーバーが異なるマシンに存在する場合、次の点に注意してください:

- PostgreSQL データベースサーバーは、クライアントからの接続を受け入れるように構成されなければなりません。具体的には、**pg_hba.conf** ファイルがローカルではない接続を許可するように構成される必要があります。第2に、**postgresql.conf** ファイルは、特定されたIP アドレスとポートをリッスンするように構成される必要があります。詳細に関しては、PostgreSQL ドキュメントを参照してください(https://www.postgresql.org/docs/9.5/static/client-authentication-problems.html)。
- サーバーマシンがファイアウォールを越えてデザインされたポート(通常、5432) 上で接続を受け入れるように構成されている必要があります。例えば、**「コントロールパネル> Windows ファイアウォール> 詳細設定 > 受信の規則」**で設定すると、Windows 上で作動するデータベースサーバーでは、ファイアーウォールを越えてポート

5432 に接続を許可するルールを作成する必要がない場合があります。

SQLite 接続の設定

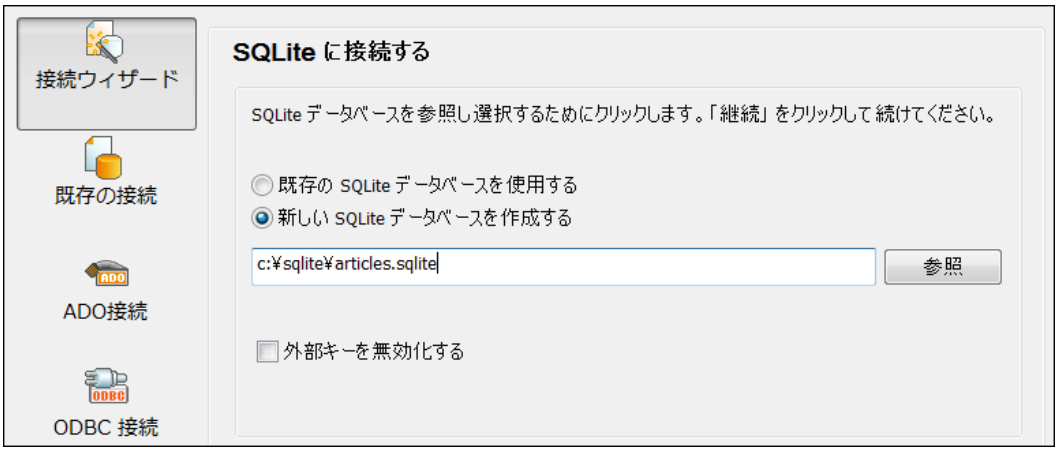
SQLite (http://www.sqlite.org) は、ポータブル性とインストールの簡便性が重要な場合理想的な構成のファイルベースの、内蔵のデータ型です。Since SQLite データベースは、MobileTogether Designer によりネイティブにサポートされるため、接続のためにドライバーをインストールする必要はありません。

新規の SQLite データベースの作成

既存のデータベースファイルに接続する代替として、新規のSQLite データベースファイルを作成して、接続することができます。MobileTogether Designer により作成されるデータベースは、空です。クエリまたはスクリプトを使用して、必要とされるデータベース構造を作成し、データを追加します。

新規のSQLite データベースの作成方法:

1. データベース接続ウィザードを実行します (データベース接続ウィザードの開始を参照)。
2. **SQLite** を選択して、**「次へ」** をクリックします。



3. **[新規 SQLite データベースの作成]** を選択して、作成されるデータベースファイルの(相対または絶対)パスを入力します (例えば、**c:\users\public\products.sqlite**)。または、**[参照]** をクリックして、フォルダーを選択して、ファイル名テキストボックスにデータベースファイルの名前(例えば、**products.sqlite**)を入力し、**[保存]** をクリックします。

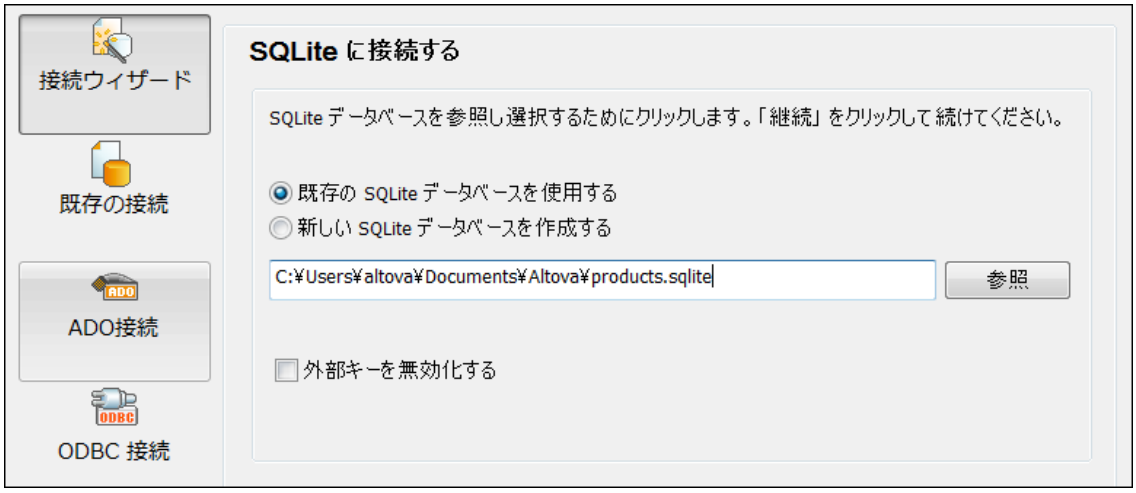
> データベースファイルを作成する場所に対して、書き込みアクセスの許可があることを確認してください。

4. **「外部キーの無効化」** チェックボックスを任意で選択します。次も参照してください外部キーの制約。
5. **「接続」** をクリックします。

外部キーの制約

MobileTogether Designer から既存のSQLite データベースに接続する場合、または新規を作成する場合、デフォルトで、外部キーの制約 が有効化されます。外部キーの制約はデータベース内のデータの整合性を保持する手助けを行います。例えば、外部キーが有効化されていると、他のテーブル内に依存関係がある場合、テーブルからレコードを削除することができません。

特定の場合、一時的にこの振る舞いをオーバーライドし、外部キーを無効化する必要がある場合があります。例えば、データ検証エラー無しにデータの列を更新、または、挿入する場合など。SQLite データベースに接続する前に、明示的に外部キーを無効化するには、データベース接続ウィザードの **「外部キーの無効化」** オプションを選択します。



"SQLite への接続 "ウィザードページ

外部キーが無効化されていると、検証チェックのためチェックすることのできなかったデータに対して操作をおこなうことができます。同時に、データベース内に不正確なデータを入れてしまう危険性、または、「無所属の」行を作成してしまう可能性が出てきます。（「無所属の」行の例は、「address」テーブルにリンクされていない「person」テーブル内のアドレスなどが挙げられます。これは、個人名が削除されているにもかかわらず、この個人に関連した住所が削除されていない場合などにこのような行が発生します。）

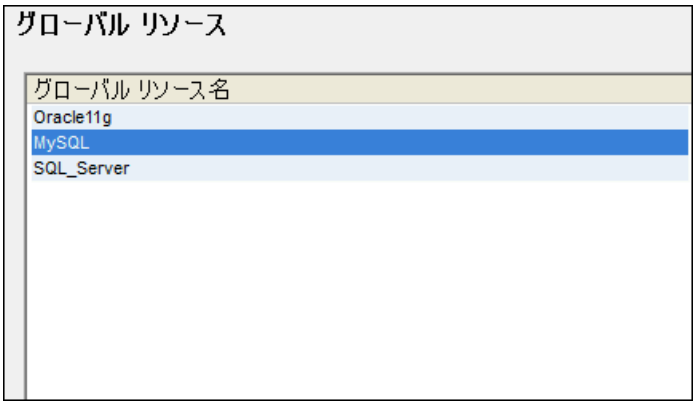
グローバルリソースからの接続の使用

Altova グローバルリソースは、リソースを再利用、構成し複数のAltova アプリケーションで使用するためのファイル、フォルダー、または、データベースを参照する方法です。

以前にデータベースをグローバルリソースとして既に構成している場合、接続をいつでも再使用することができます (異なるAltova アプリケーション間でも使用可能)。

グローバルリソースからのデータベース接続の使用:

1. データベース接続ウィザードを開始します。
2. **[グローバルリソース]** をクリックします。グローバルリソースとして使用可能なデータベース接続がリストされます。



3. データベース接続履歴から **「接続」** をクリックします。

ヒント: 各グローバルリソースの追加情報を取得するために、マウスカーソルをグローバルリソースにポイントします。

データベース接続のサンプル

このセクションは、MobileTogether Designer からデータベースへの接続のサンプル手順を説明します。Windows マシン、ネットワーク環境、とデータベースクライアントまたはサーバーソフトウェアは、次に挙げられるサンプルの構成とは完全に一致しない可能性があります。

メモ ほぼすべてのデータベースの型では、1つ以上のデータアクセステクノロジー (ADO、ADO.NET、ODBC、JDBC) またはドライバーを使用して接続することができます。データベース接続パフォーマンス、およびドライバーの機能と制限は、MobileTogether Designer 外で構成された選択されたドライバー、(該当する場合) データベースクライアントソフトウェア、および追加接続パラメータなどによって異なります。

Firebird (ODBC) への接続

このトピックは、Linux サーバー上で実行される、Firebird 2.5.4 データベースへの接続のサンプルの命令を説明しています。

必要条件:

- Firebird データベースサーバーは、クライアントからのTCP/IP 接続を受け入れるように構成されています。
- Firebird ODBC ドライバーを使用中のオペレーティングシステムにインストールする必要があります。このサンプルは、を使用します。 Firebird ODBC ドライバー バージョン2.0.3.154 は、Firebird ウェブサイト(http://www.firebirdsql.org/) からダウンロードすることができます。
- Firebird クライアントを、使用中のオペレーティングシステムにインストールする必要があります。 Firebird 2.5.4 クライアントのためのスタンドアロンのインストーラーが存在しない点に注意してください。また、クライアントは、サーバーのインストールパックの一部です。Firebird サーバー インストールパックをFirebird ウェブサイト(http://www.firebirdsql.org/) からダウンロードすることができます。ダウンロードの際、[スーパークラシック/クラシック または スーパーサーバーのための、Windows 実行ファイル] インストーラーを選択してください。クライアントファイルのみをインストールする場合、**[最小クライアントインストール -サーバーツール無し]** を選択します。

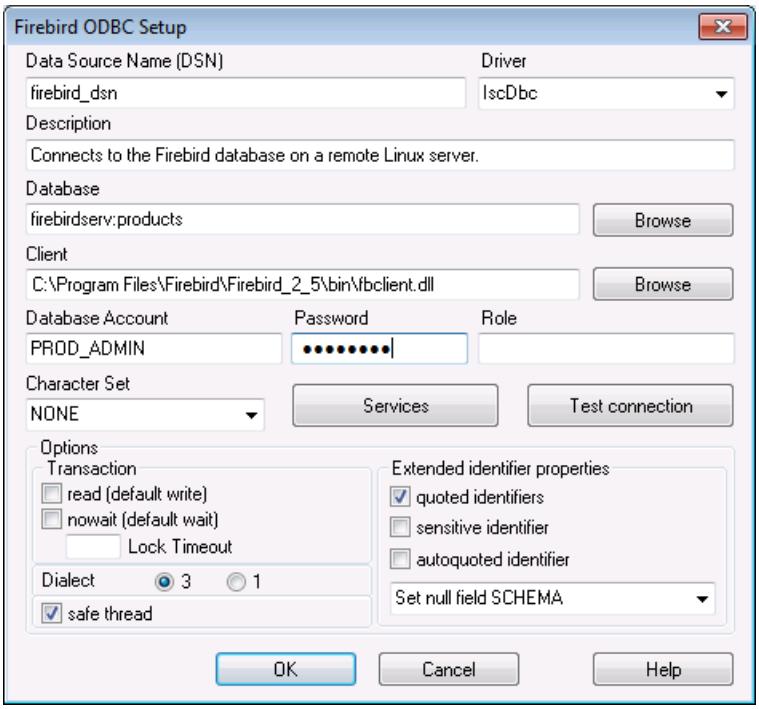
> **重要点:**
>
> - Firebird ODBC ドライバー および クライアント (32 ビット または 64 ビット) のプラットフォームは、MobileTogether Designer のプラットフォームに適応する必要があります。
> - Firebird クライアントのバージョンは、接続されているFirebird サーバーのバージョンに適応する必要があります。

- データベース接続の詳細: サーバー上の サーバー ホスト名 または IP アドレス、データベース パス(または エイリアス)、ユーザー名、および パスワード。

ODBC を使用して Firebird に接続する

1. データベース接続ウィザードを開始します。
2. **[ODBC 接続]** をクリックする。
3. **[ユーザー DSN]** を選択して、(または、管理者の特権がある場合、 **[システム DSN]** を選択して、)**[追加]** をクリックします。

4. Firebird ドライバーを選択して、をクリックします。そして**[ユーザー DSN]** (前のステップでの選択により、または、**[システム DSN]**)。Firebird ドライバーがリスト内に存在しない場合、使用中のオペレーティングシステムにインストールされていることを確認してください (使用可能な ODBC ドライバーの表示 を参照)。



5. データベース接続の詳細を入力します:

データソース名 (DSN)	作成する詳細名とデータソースを入力します。

データベース	サーバー ホスト名 または IP アドレスをコロンとデータベース エイリアス (または パス) の後に入力します。このサンプルでは、ホスト名は `firebirdserv`、データベースエイリアスは、`products` です:
	
	`firebirdserv:products`
	
	データベース エイリアスを使用することにより、サーバー側上では、データベース管理者は、エイリアス製品を構成し、サーバー上の実際の Firebird (.fdb) データベースファイルをポイントします (詳細は、Firebird のドキュメントを参照してください)。
	
	ホスト名の代わりに サーバー IP アドレス、エイリアスの代わりにパスを使用することもできます。ですから、以下のサンプル接続の文字列は有効です:
	
	`firebirdserver:/var/Firebird/databases/butterflies.fdb`
	`127.0.0.1:D:\Misc\Lenders.fdb`
	データベースが Windows マシン上に存在する場合、**[参照]** をクリックして、Firebird (.fdb) データベースファイルを直接選択します。

クライアント	**fbclient.dll** ファイルへのパスを入力します。デフォルトでは、これは Firebird インストールディレクトリのサブディレクトリです。
データベース アカウント	データベース管理者 (このサンプルの場合 `PROD_ADMIN`) により与えられたデータベース ユーザー名を入力します。
パスワード	データベース管理者により与えられたデータベース パスワードを入力します。

6. **「OK」**をクリックします。

Firebird (JDBC) への接続

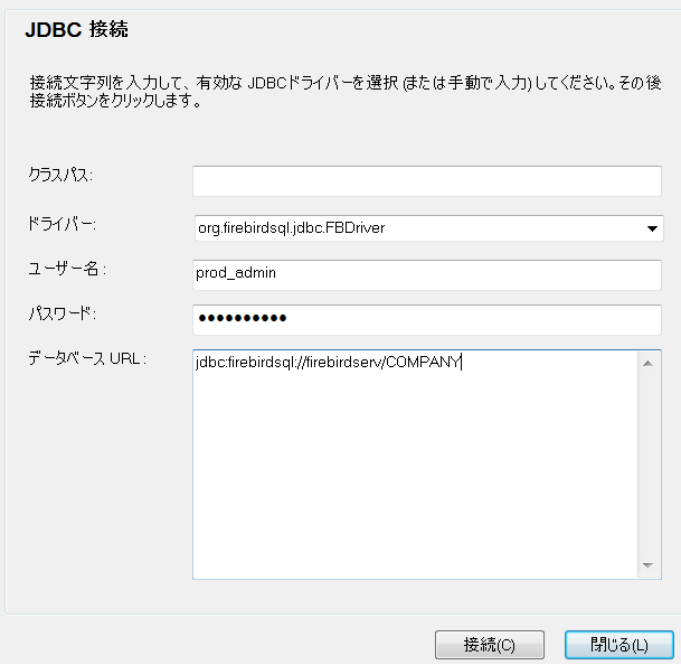
このトピックは、を使用してのFirebird データベースサーバーへの接続のサンプルの命令を説明しています。

必要条件:

- Java Runtime Environment (JRE) またはJava Development Kit (JDK) が使用中のオペレーティングシステムにインストールされている必要があります。
- オペレーティングシステムの`PATH` 環境変数は、JRE またはJDK のインストールディレクトリの`bin` ディレクトリへのパスを含む必要があります。例えば、`C:\Program Files (x86)\Java\jre1.8.0_51\bin`.
- Firebird JDBC ドライバーが使用中のオペレーティングシステムで使用可能である必要があります (データベースへの接続を可能にする jar ファイルのフォームをとります)。ドライバーはFirebird Web サイトからダウンロードすることができます (http://www.firebirdsql.org/)。このサンプルは、*Jaybird 2.2.8* を使用します。
- データベース接続の詳細があります: ホスト、データベースパス、またはエイリアス、ユーザー名、およびパスワード。

JDBC を使用して Firebird に接続する

1. データベース接続ウィザードを開始します.
2. 「**JDBC 接続**」をクリックします。.
3. 「クラスパス」の横にデータベースへの接続を与える jar ファイルへのパスを入力します。必要であれば、セミコロンで区切られた jar ファイルパスのリストを入力することもできます。このサンプルでは、必要とされる jar ファイルは次のパスを使用して見つけることができます: **C:\jdbc\firebird\jaybird-full-2.2.8.jar**。jar ファイルパスをオペレーティングシステムのCLASSPATH 環境変数に追加した場合、「クラスパス」テキストボックスを空白のままにすることができます。(次も参照: CLASSPATH の構成)。
4. 「ドライバー」ボックスから、**org.firebirdsql.jdbc.FBDriver** を選択します。このエントリは、有効な jar ファイルパスが、「クラスパス」テキストボックス内、または、オペレーティングシステムのCLASSPATH 環境変数内で検出された場合のみ使用することができます (前のステップを参照してください)。



5. 対応するテキストボックスに、データベースへのユーザー名とパスワードを入力します。
6. URL テキストボックスに、データベースサーバーに適用することのできるハイライトされた値を置き換える、データベースサーバーの接続文字列を入力します。

```
jdbc:firebirdsql://<host>[:<port>]/<databese path or alias>
```

7. **「接続」**をクリックします。

IBM DB2 (ODBC) への接続

このトピックではODBC を使用した、IBM DB2 への接続のサンプル手順を紹介します。

前提条件:

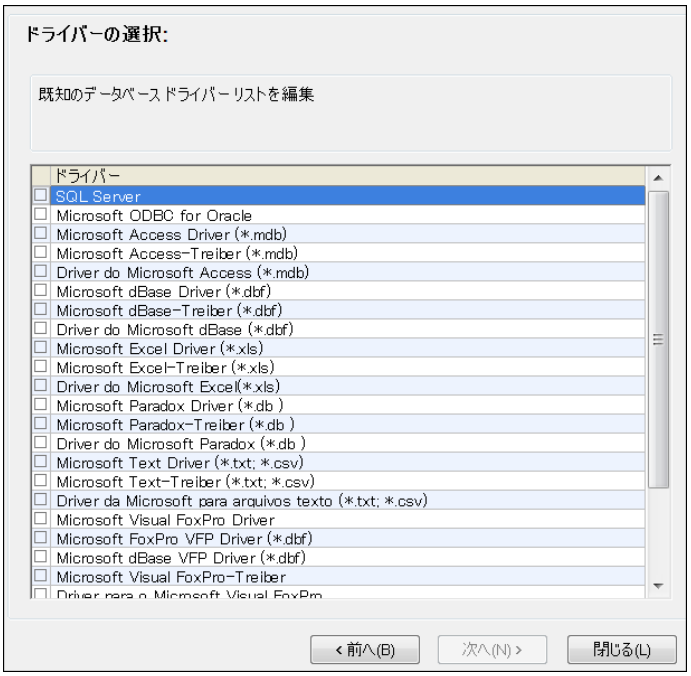
- IBM Data Server Client は、使用中のオペレーティングシステムにインストールおよび構成される必要があります (このサンプルの場合はIBM Data Server Client 9.7 を使用します)。インストールの手順に関しては、IBM DB2 ソフトウェアにより提供されたドキュメントを確認してください。 IBM Data Server Client インストール後、使用中のコンピューターでODBC ドライバーが使用可能か確認してください (使用可能な ODBC ドライバーの表示を参照)。
- データベースエイリアスを作成します。複数の作成方法があります:
  - IBM DB2 構成アシスタントから
  - IBM DB2 コマンドラインプロセッサーから
  - ODBC データソースウィザードから (この場合は、下に表示される手順を参照してください)
- データベース接続の詳細: ホスト、データベース、ユーザー名およびパスワード

IBM DB2 への接続:

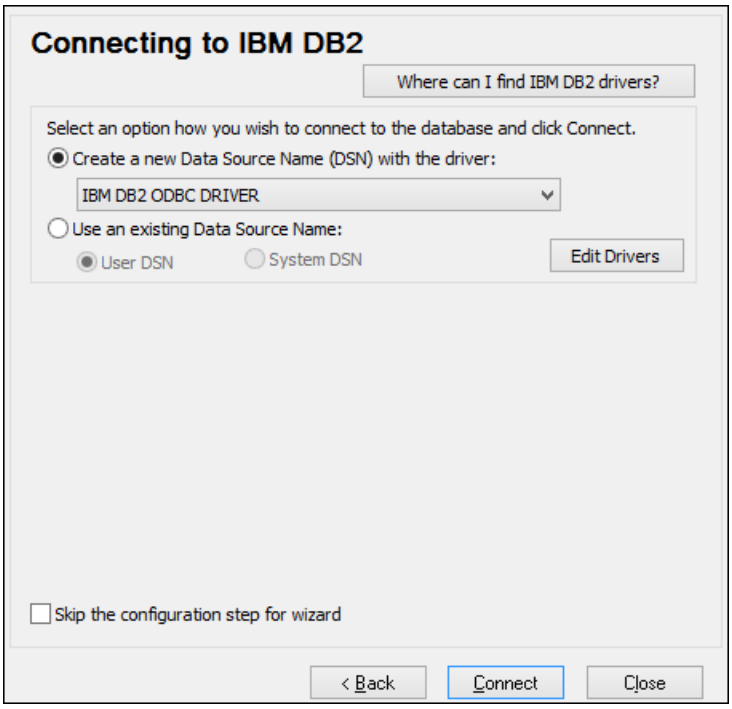
1. データベース接続ウィザードを開始します そして **IBM DB2 (ODBC/JDBC)** を選択します。

2. **「次へ」** をクリックします。

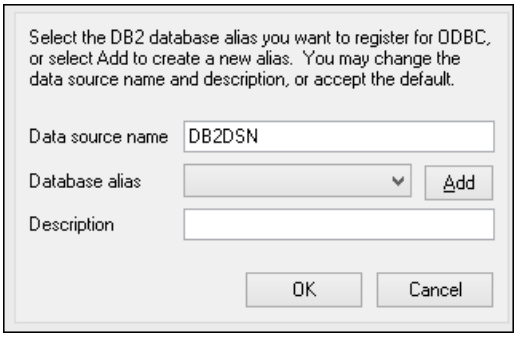
3. **ODBC** を選択して **「次へ」** をクリックします。データベースのための既知のリストを編集するようにプロンプトされると、IBM DB2 に適用することのできるドライバーを選択して(前提条件を参照)、 **「次へ」** をクリックします。



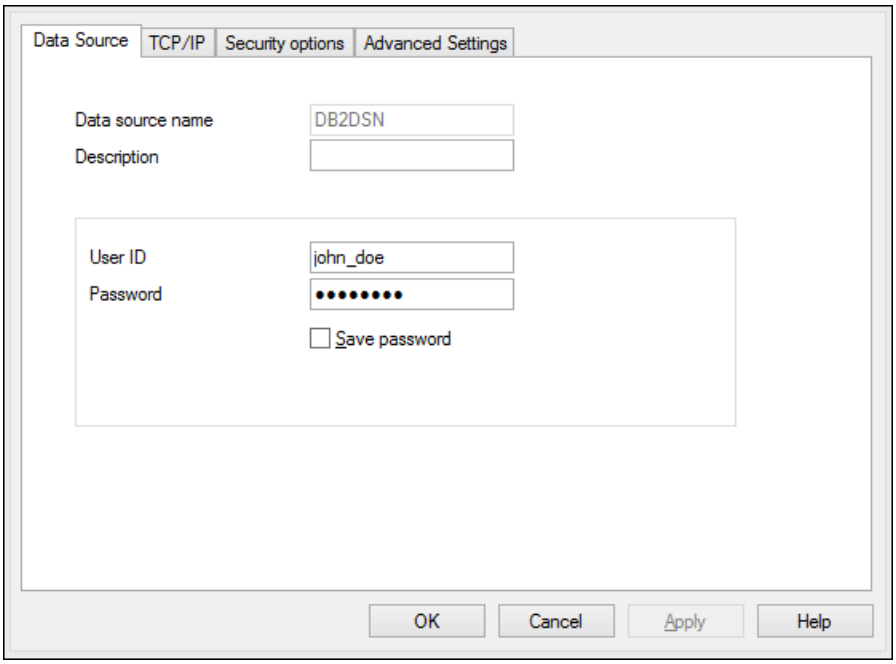
4. リストからIBM DB2 を選択して、 **「接続」** をクリックします。(ドライバーで使用可能なリストを編集するには、**[ドライバーの編集]** をクリックして、使用可能なドライバーのリストを編集します。クリックして追加または削除する IBM DB2 ドライバーをチェックまたはチェックの解除をします。ドライバーを選択または選択の解除をします。)



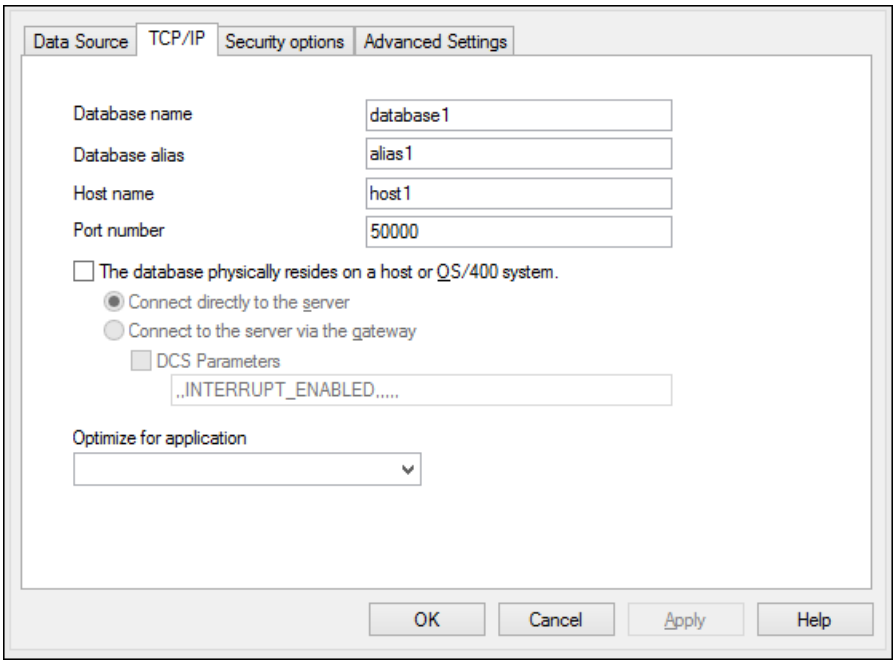
5. ソース名を入力します (このサンプルでは **DB2DSN**)　**「追加」**をクリックします。



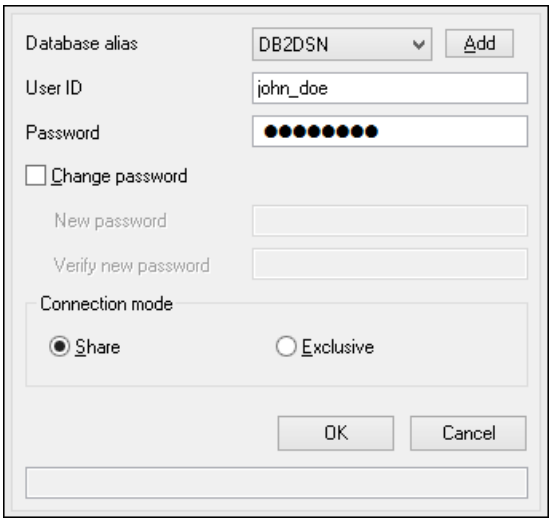
6. **[データソース]** タブで、データベースのユーザー名とパスワードを入力します。



7. **[TCP/IP]** タブで、データベース名、エイリアスの名前、ホスト名とポート名を入力して **「OK」** をクリックします。



8. ユーザ名とパスワードを再入力して **「OK」** をクリックします。

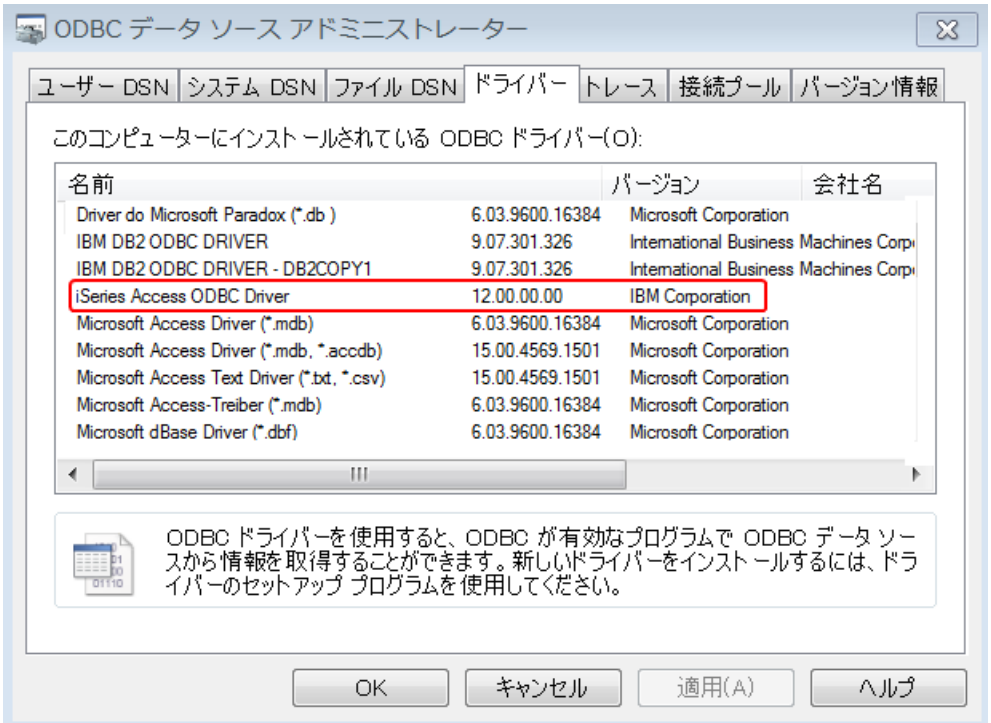

Database alias
DB2DSN
Add
User ID
john_doe
Password
Change password
New password
Verify new password
Connection mode
Share
Exclusive
OK
Cancel

IBM DB2 for i (ODBC) への接続

このトピックは、ODBC を使用して *IBM DB2 for i* データベースへの接続のサンプル手順を説明します。

前提条件:

- *IBM System i Access for Windows* は使用中のオペレーティングシステム (このサンプルでは *IBM System i Access for Windows V6R1M0* を使用しています) にインストールされる必要があります。インストールの手順は、*IBM DB2 for i* ソフトウェアにより提供されているドキュメントを参照してください。インストール後、ODBC ドライバーが使用中のコンピューターで使用可能か確認してください (使用可能なODBC ドライバーの表示を参照)。

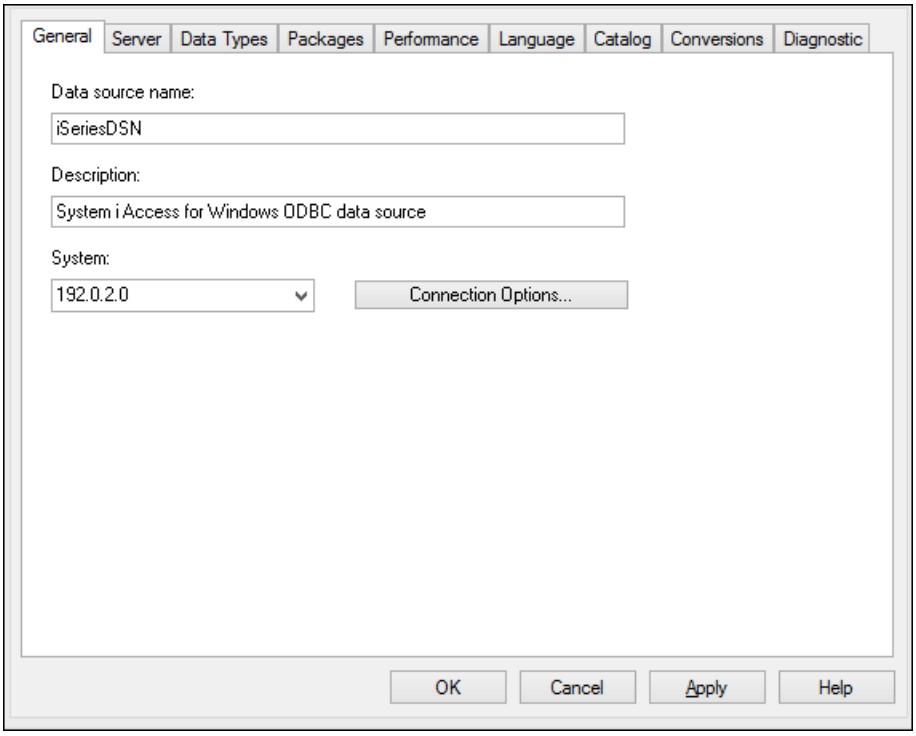


- データベース接続の詳細: データベースサーバーの I.P. アドレス、データベースユーザー名、およびパスワード
- *i Navigator* を実行して、新規接続を作成するウィザードの手順を踏んで新規接続を作成してください。システムの指定をプロンプトされると、データベースサーバーの I.P. アドレスを入力してください。接続を作成した後、検証する (接続をクリックして、**ファイル** > **診断** > **接続の検証** を選択) ことが奨励されます。接続エラーが発生した場合、データベースサーバー管理者に連絡してください。

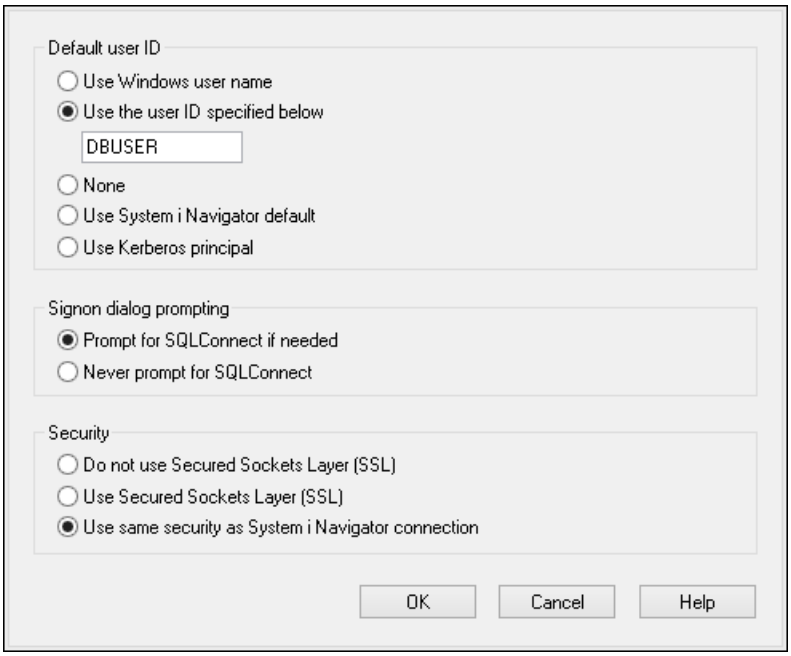
IBM DB2 for i への接続:

1. データベース接続ウィザードを開始します.
2. **[ODBC 接続]** をクリックします。
3. **[ユーザー DSN]** をクリックします。(または、**[システム DSN]** または、後に続く手順が同じ **[ファイル DSN]** をクリックします)。
4. **「追加」** をクリックします。
5. **[iSeries Access ODBC ドライバー]** をリストから選択します。**[ユーザー DSN]** (または、該当する場合 **[システム DSN]**) をクリックします。

6. データソース名を入力して、システムコンボボックスから接続を選択します。このサンプルでは、データソース名は **iSeriesDSN** でシステムは **192.0.2.0** です。



7. **[接続オプション]** をクリックします。**[指定されたユーザー ID を使用] を選択して、**データベースユーザー (このサンプル内では **DBUSER**) の名前を入力します。



8. **「OK」**をクリックします。DSN 内のリストで新しいデータソースを使用することができます。
9. **「接続」**をクリックします。
10. データベースのユーザー名とパスワードをプロンプトされると入力して **「OK」**をクリックします。

IBM Informix (JDBC) への接続

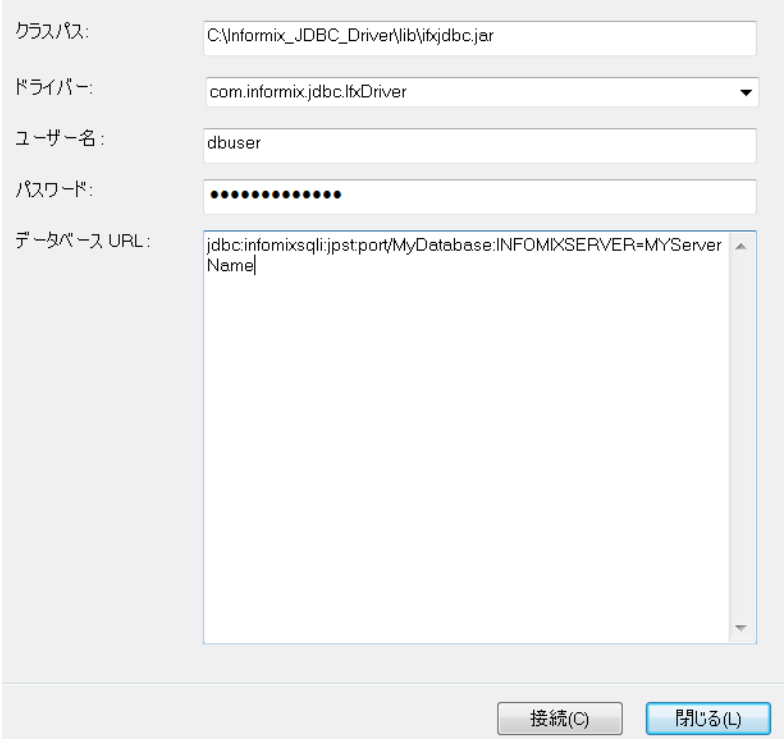
このトピックでは、JDBC を使用して IBM Informix データベースサーバーに接続する手順のサンプルを説明します。

前提条件:

- Java Runtime Environment (JRE) がオペレーティングシステムにインストールされている必要があります。
- (データベースへの接続を与える1つ、または、複数の jar ファイル) JDBC ドライバーが使用中のオペレーティングシステムで使用可能である必要があります。この例では、IBM Informix JDBC ドライバー バージョン3.70 が使用されています。ドライバーのインストールの説明に関しては、ドライバーに付随するドキュメント、または、「IBM Informix JDBC ドライバプログラマーズガイド」を参照してください)。
- データベース接続の詳細: ホスト、サーバーの名前、データベース ポートユーザー名、およびパスワード

JDBC を使用して IBM Informix に接続する

1. データベース接続ウィザードを開始します.
2. **「JDBC 接続」**をクリックします。.
3. 「クラスパス」の横にデータベースへの接続を与える jar ファイルへのパスを入力します。必要であれば、セミコロンで区切られた jar ファイルパスのリストを入力することもできます。このサンプルでは、必要とされる jar ファイルは次のパスを使用して見つけることができます: **C:\Informix_JDBC_Driver\lib\ifxjdbc.jar**。 jar ファイルパスをオペレーティングシステムの CLASSPATH 環境変数に追加した場合、「クラスパス」テキストボックスを空白のままにすることができます。(次も参照: CLASSPATH の構成)。
4. 「ドライバー」ボックスから **com.informix.jdbc.IfxDriver** を選択します。このエントリは、有効な jar ファイルパスが、「クラスパス」テキストボックス内、または、オペレーティングシステムの CLASSPATH 環境変数内で検出された場合のみ使用することができます (前のステップを参照してください)。



5. データベースへのユーザー名とパスワードを対応するテキストボックスに入力します。
6. ハイライトされた値を、使用するデータベースサーバーに適用する値と置き換えて、データベース URL テキストボックス内にデータベースサーバーへの接続文字列を入力します。

```
jdbc:informix-sqli://hostName:port/
```

```
databaseName:INFORMIXSERVER=myserver;
```

7. **接続**」をクリックします。

MariaDB (ODBC) への接続

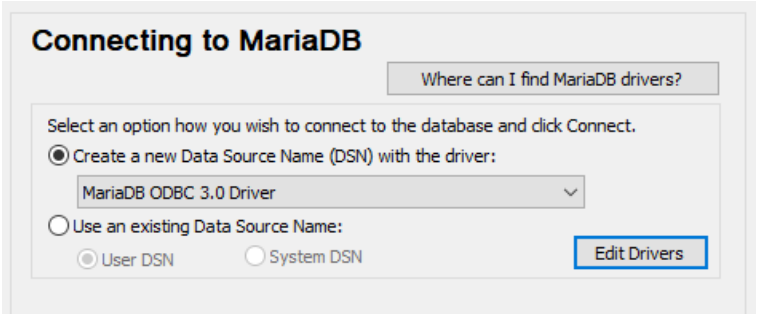
このサンプルでは、ODBC を使用したMariaDB データベースへの接続方法について説明されています。

必要条件:

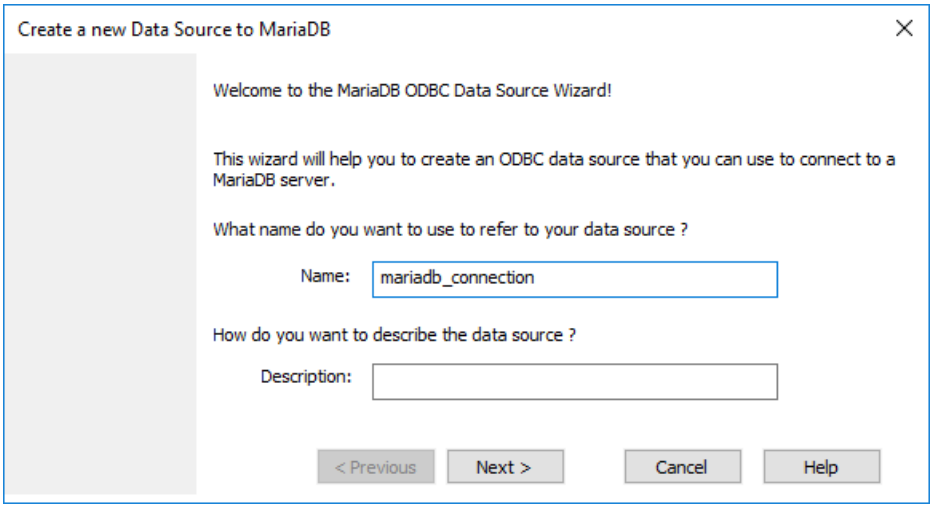
- MariaDB コネクタODBC (https://downloads.mariadb.org/connector-odbc/) がインストールされている必要があります。
- 次のデータベース接続詳細が存在します: ホスト、データベース、ポート、ユーザー名、および、パスワード。

ODBC を使用したMariaDBへの接続の方法:

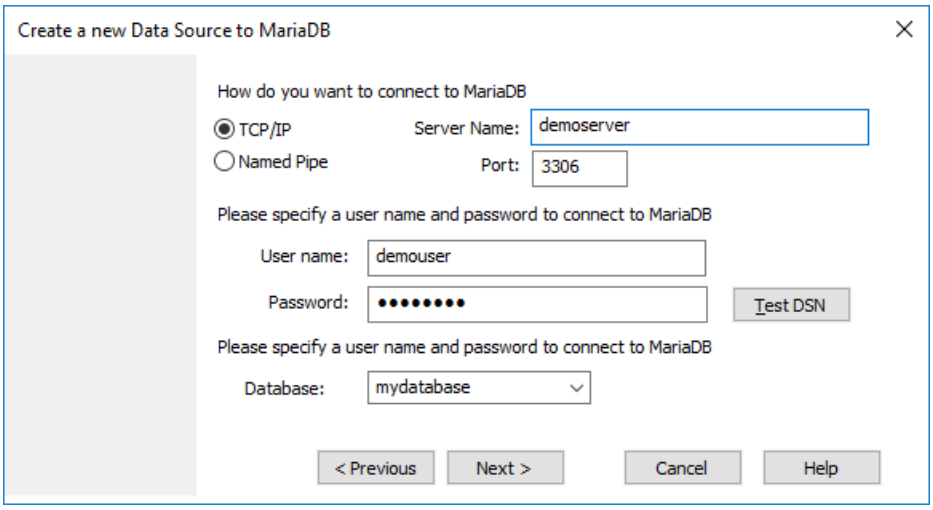
1. データベース接続ウィザードを開始します。
2. 「**MariaDB (ODBC)**」を選択して、「**次へ**」をクリックします。



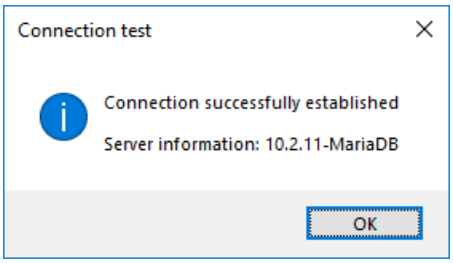
3. 「**ドライバーを使用して新規のデータソース名 (DSN) を作成する**」を選択し、「**MariaDB ODBC 3.0 Driver**」を選択します。このようなドライバーがリスト内で使用できない場合、「**ドライバーの編集**」をクリックし、使用することのできるMariaDB ドライバーを選択します (リストには使用中のオペレーティングシステム上にインストールされているすべてのODBC ドライバーが含まれています)。
4. 「**接続**」をクックします。



5. 名前を入力し、任意で、このODBC データソースを将来識別できる詳細を与えます。



6. データベース接続資格情報(TCP/IP サーバー、ユーザー、パスワード)を入力し、データベースを選択して、**「Test DSN」**をクリックします。接続に成功すると、メッセージが表示されます:



7. **「次へ」**をクリックしてウィザードを完了します。他のパラメータが必要になる場合がありますが(例えば、安全な接続を使用して、MariaDB に接続する場合は、SSL 証明書が必要になります)。

メモ データベースサーバーがリモートの場合、サーバー管理者が遠隔の接続を使用中のマシンの IP アドレスから受け入れるように構成する必要があります。

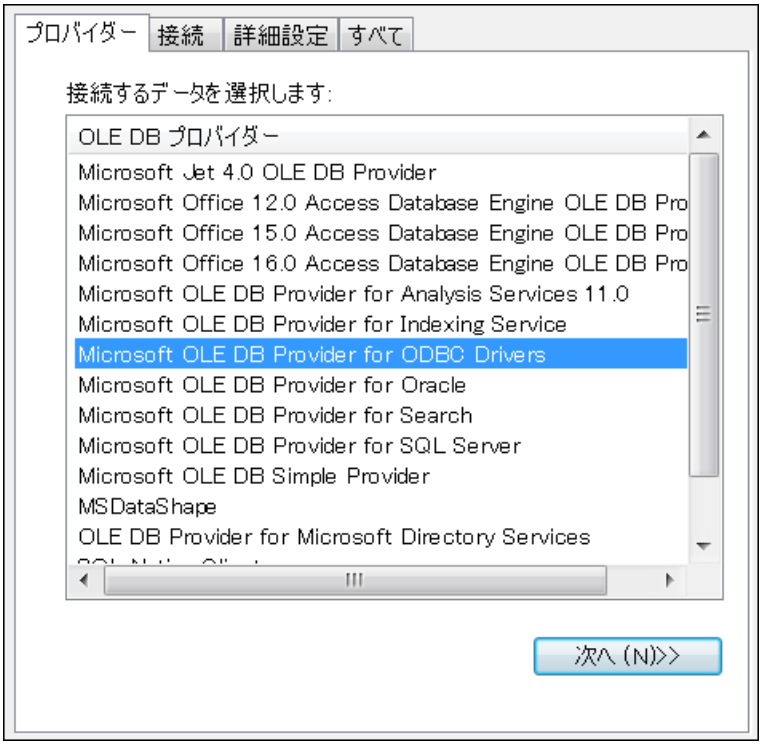
Microsoft Access (ADO) への接続

Microsoft Access データベースに接続するには、既存のMicrosoft Access データベースに接続するで説明されているとおり、データベースファイルのためのウィザードとブラウザーの手順を踏みます。別の方法としては、このトピックで説明されているとおり、明確的にADO 接続をセットアップすることです。このアプローチは、データベースがパスワードで保護されている場合、役に立ちます。

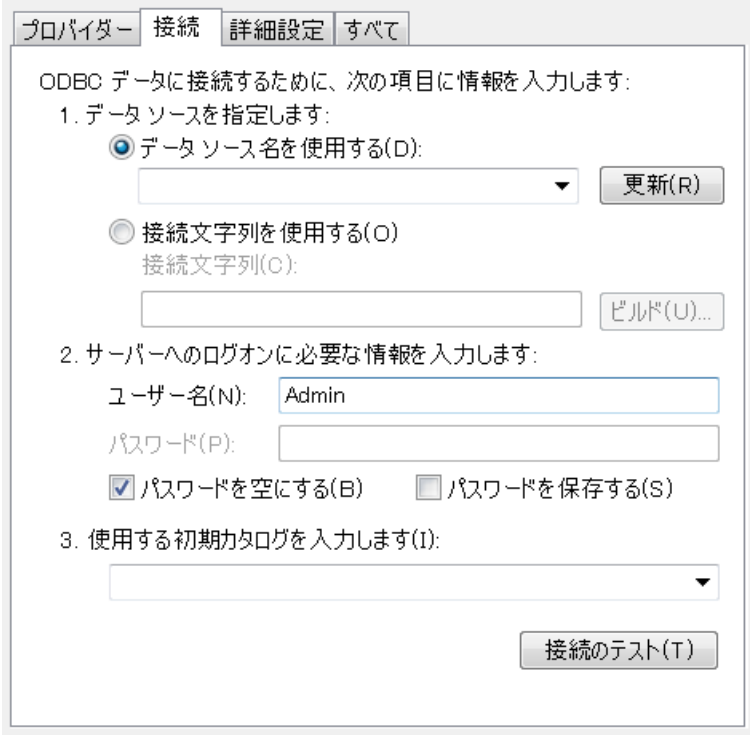
接続を使用して、に接続することもできますが、このシナリオには限度があるため、奨励されません。

パスワードで保護されているMicrosoft Access データベースに接続する

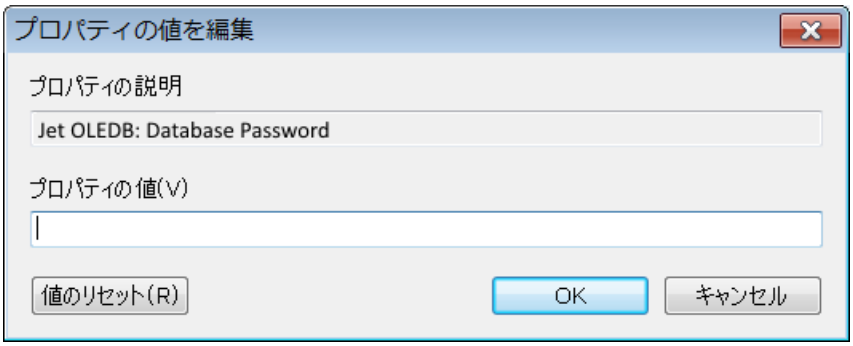
1. データベース接続ウィザードを開始します。
2. **[ADO 接続]** をクリックします。
3. **[構築]** をクリックします。



4. **[Microsoft Office 15.0 Access Database Engine OLE DB Provider]** を選択して、**「次へ」** をクリックします。



5. データソースボックス内で、Microsoft Access ファイルへのパスを入力します。ファイルがローカルネットワーク**U:\Departments\Finance\Reports\Revenue.accdb** を共有するため、ファイルをUNC フォーマットすなわち、**server1** isがサーバーの名前でdfs が共有されるネットワークの名前である、**\\server1\\dfs\Departments\Finance\Reports\Revenue.accdb** に変換します。
6. **全て** タブの**Jet OLEDB:Database パスワード** プロパティをダブルクリックし、データベースパスワードをプロパティの値として入力します。



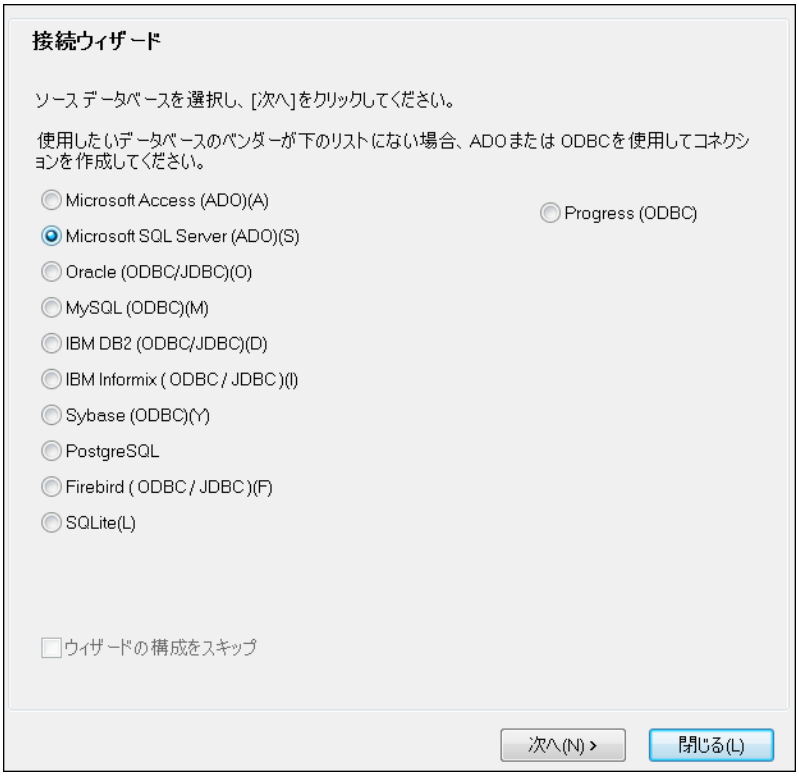
メモ また接続できない場合、ワーキンググループ情報ファイル(**System.MDW**) を検索して、ユーザープロフィールに適用できるようにします。(手順はhttp://support.microsoft.com/kb/305542 を参照してください。) **Jet OLEDB: System database** プロパティの値を**System.MDW** ファイルのパスに設定します。

Microsoft SQL Server (ADO) への接続

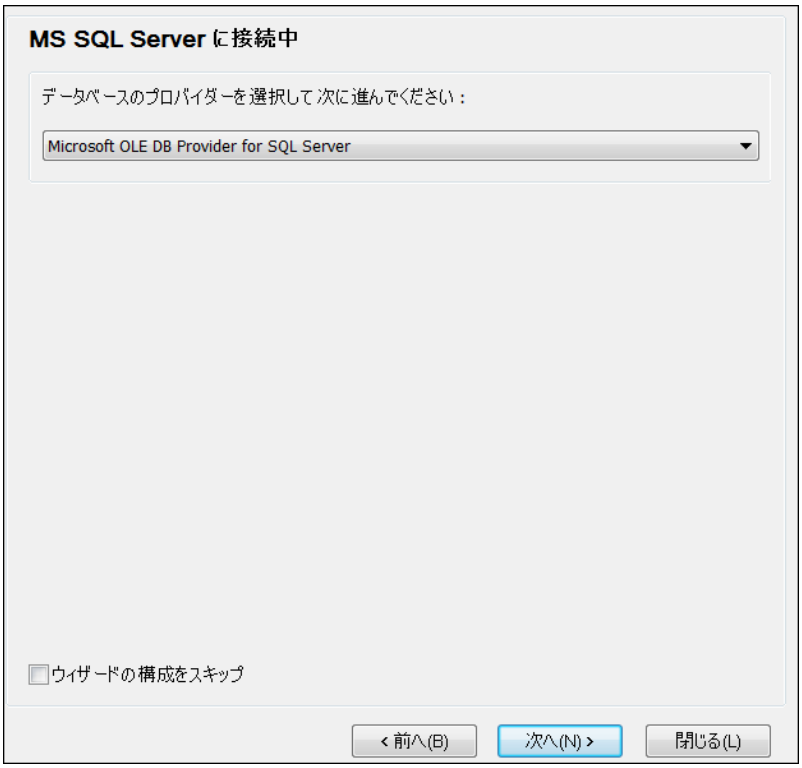
このサンプルでは、ADO を使用したSQL Server データベースへの接続方法について説明されています。

Microsoft OLE DB プロバイダーを使用してのSQL Server への接続 :

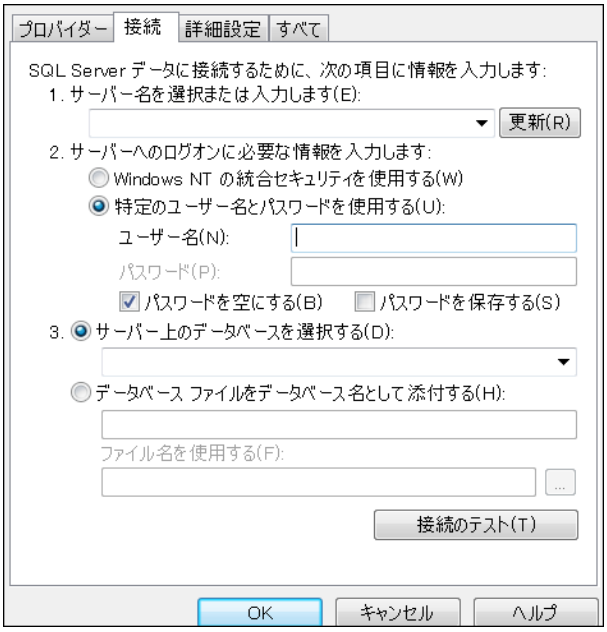
1. データベース接続ウィザードを開始します。



2. **Microsoft SQL Server (ADO)** を選択して、 **「次へ」** をクリックします。使用可能なADO ドライバーが表示されます。



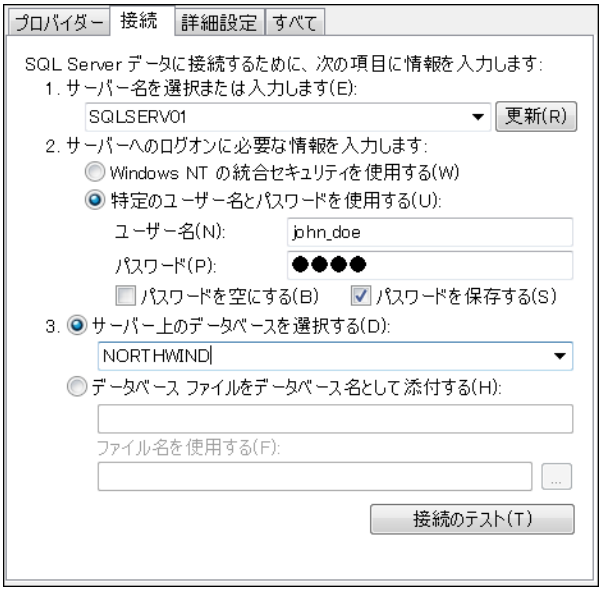
3. **[Microsoft OLE DB Provider for SQL Server]** を選択して、 **「次へ」** をクリックします。



4. データベースサーバーの名前を選択または入力します(このサンプルでは、**SQLSERV01** です)。このネットワーク上の全てのサーバのリストを確認するには、ドロップダウンリストを展開してください。
5. データベースサーバーがWindows ドメイン上の認証されているユーザーの接続を許可している場合、**[Windows NT 統合されたセキュリティの使用]** を選択してください。それ以外の場合は、**[特定の名前とパスワードを使用する]** を選択して、関連するボックスに入力してください。
6. 接続するデータベースを選択します(このサンプルでは、**NORTHWIND** です)。
7. この時点で接続をテストするには、**[接続をテストする]** をクリックします。このステップは任意で奨励されるステップで

す。

8. 以下を行ってください：
   a. **[パスワードの保存を許可]** チェックボックスを選択します。
   b. **すべて** タブで、**[セキュリティ情報を保持する]** プロパティの値を **True** に変更します。



9. **「OK」** をクリックします。

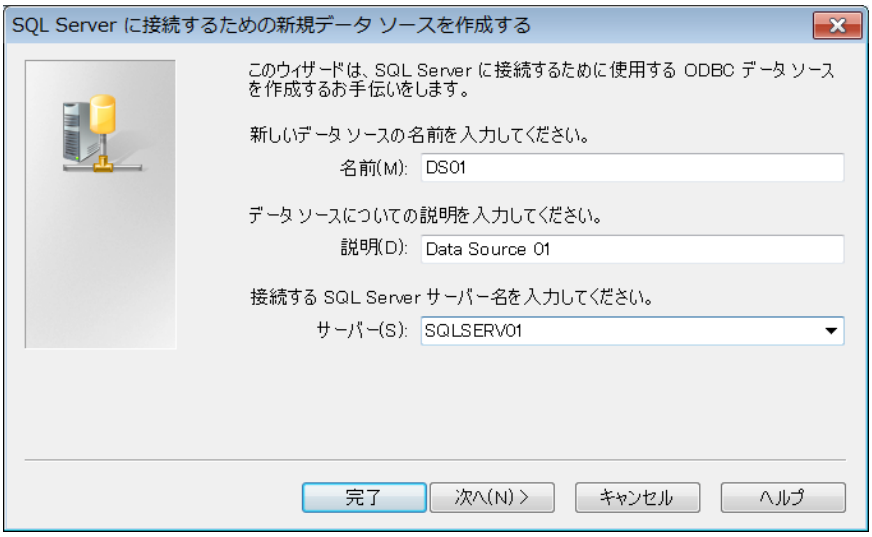
Microsoft SQL Server (ODBC) への接続

このサンプルでは、ODBC を使用したSQL Server データベースへの接続方法について説明されています。

ODBC を使用してのSQL Server への接続:

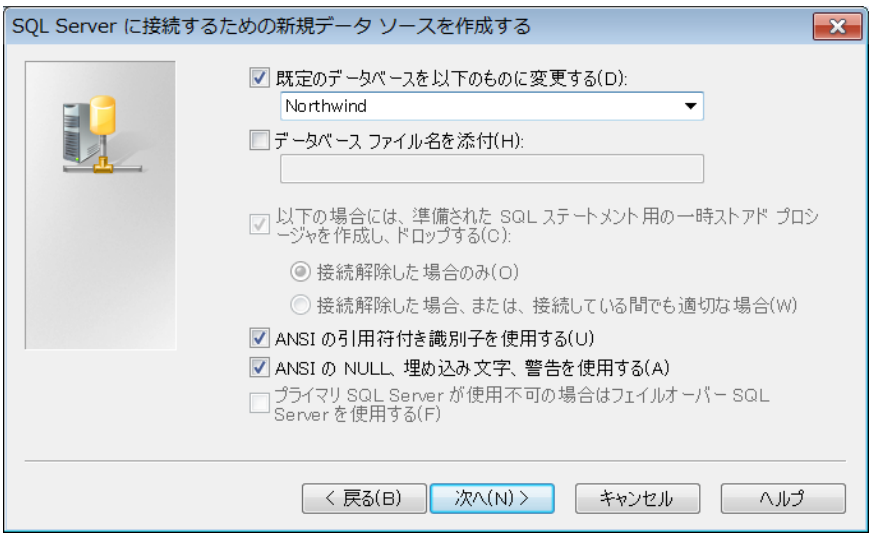
1. データベース接続ウィザードを開始します.
2. **[ODBC 接続]** をクリックします。
3. **[ユーザーDSN]** を選択して、(または管理者の特権を有する場合は、**システムDSN**) **「追加」** をクリックします。

4. **[SQL Server]** (または使用可能な場合は**[SQL Server Native Client]**) を選択して、**[ユーザーDSN]** (またはシステムDSN を作成する場合は、**システム[DSN]**) をクリックします。



5. この接続を認識するための名前と詳細を入力し、接続するをリストから選択します。(このサンプルでは、**SQLSERV01** です。)

6. データベースサーバーがWindows ドメイン上の認証されているユーザーの接続を許可している場合、**[Windows NT 承認]** を選択してください。それ以外の場合は、**[SQL Server 承認 ...]** を選択して、関連するボックスに入力してください。



1. 接続するデータベースを選択します (このサンプルでは、**NORTHWIND** です)。
7. **[完了]** をクリックします。

MySQL (ODBC) への接続

このトピックではODBC ドライバーを使用した、Windows マシンから MySQL データベースへの接続のサンプル手順を紹介します。Windows

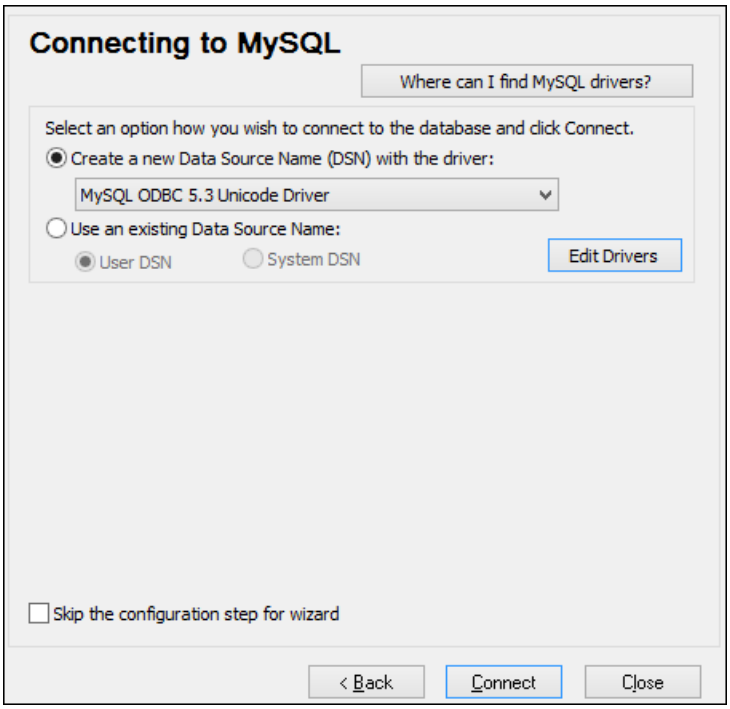
前提条件:

- MySQL ODBC ドライバーを使用中のオペレーティングシステムにインストールする必要があります。(インストールに関しては、ドライバーに付随するドキュメントを参照してください)。
- データベース接続の詳細: ホスト、データベース、ユーザー名およびパスワード

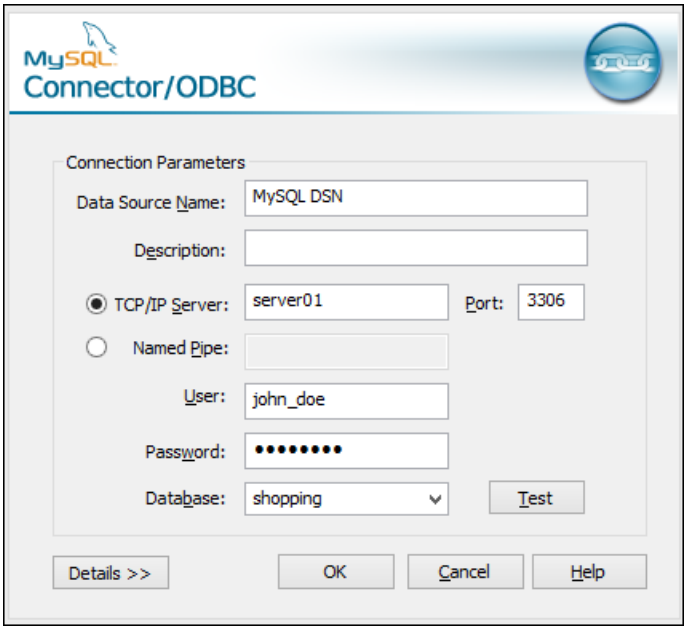
ODBC を使用しての MySQL への接続:

1. データベース接続ウィザードを開始します。

2. **[MySQL (ODBC)]** を選択して、**「次へ」** をクリックします。



3. **[ドライバーを使用して新しいデータソース名 (DSN) を作成]** を選択して、MySQL ドライバーを選択します。MySQL ドライバーがリストにない場合は、**[ドライバーを編集]** をクリックして、使用可能な MySQL ドライバーを選択します (使用中のオペレーティングシステムにインストールされているすべての ODBC ドライバーがリストされています)。
4. **「接続」** をクリックします。



5. データソース名ボックスの中に、今後この ODBC データソースを認識するために役に立つ詳細を入力します。
6. データベース接続資格情報 (TCP/IP サーバー、ユーザー、パスワード) を入力して、データベースを選択し、**「OK」** をクリックします。

メモ　If データベースサーバーがリモートの場合、サーバー管理者が使用中のコンピューターの IP アドレスからのリモート接

続を承諾するように構成される必要があります。また、**[詳細>>]** をクリックすると使用することのできる追加パラメーターが表示されます。デフォルトの値を変更する前に、ドライバーのドキュメントを確認してください。

Oracle (ODBC) への接続

このサンプルは、ローカルオペレーティングシステム上にインストールされているOracle データベースクライアントを介して、ネットワークマシン上でMobileTogether Designer からOracle データベースサーバーへの接続のよく使用されている状況を説明します。

このサンプルでは、MobileTogether Designer 内のデータベース接続ウィザードを使用して、ODBC データソース (DSN) のセットアップについて説明されています。既にDSN が作成されている場合は Windows 内のODBC データソース管理者から直接作成をする事も可能です。ウィザードがプロンプトした場合はそのように選択してください。データソースに関する詳細はODBC 接続のセットアップを参照してください。

前提条件:

- (ODBC Oracle ドライバーを含む) Oracle データベースクライアントが使用中のオペレーティングシステムでインストール、および構成されていなければなりません。 Oracle データベースクライアントのインストールおよび構成方法はOracle ソフトウェアに付随するドキュメントを参照してください。
- Oracle ホームディレクトリにある**tnsnames.ora** ファイルは、データベース接続パラメーターを紹介するエントリを含みます。フォーマットは以下に類似します:

```
ORCL =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = server01)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = orcl)
      (SERVER = DEDICATED)
    )
  )
```

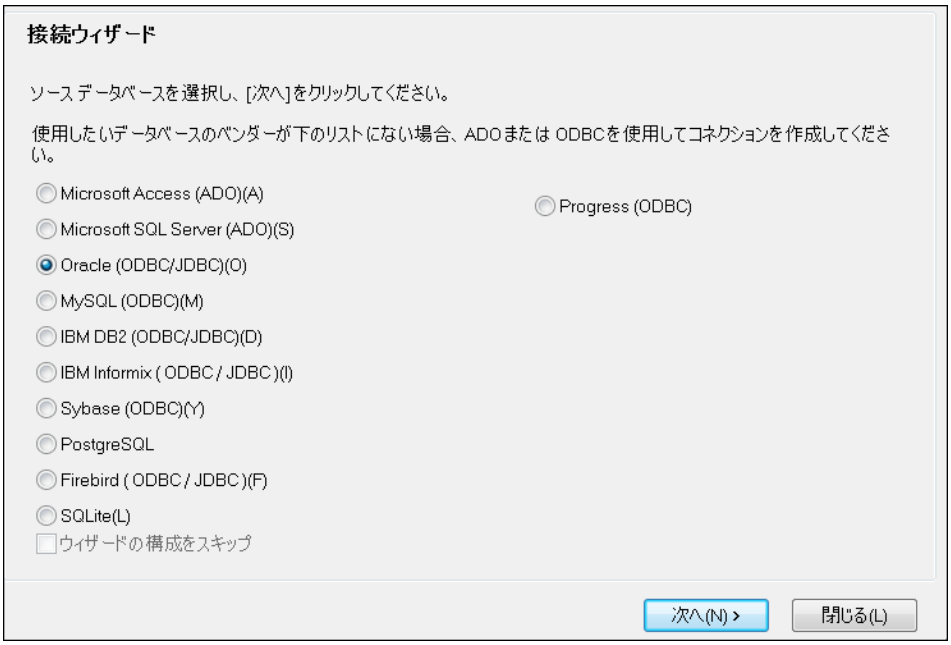
tnsnames.ora ファイルのパスはOracle ホームディレクトリがインストールされている場所によります。Oracle データベースクライアント11.2.0 に関しては、デフォルトOracle ホームディレクトリのパスは以下の通りです:

```
C:\app\username\product\11.2.0\client_1\network\admin\tnsnames.ora
```

接続の詳細を貼り付け保存する、または(使用可能であれば) Oracle *ネット構成アシスタントウィザード* を実行することにより **tnsnames.ora** ファイルに新規エントリを追加することができます。

ODBC を使用してのOracle への接続:

1. データベース接続ウィザードを開始します。

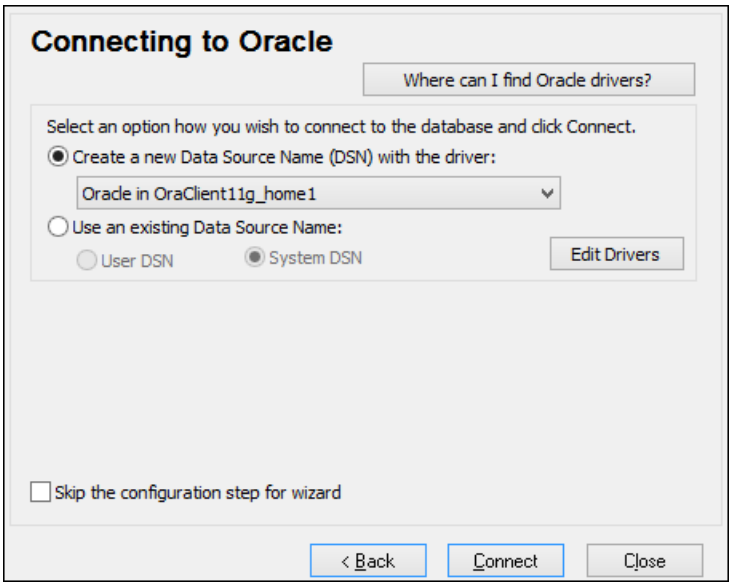


2. **[Oracle (ODBC / JDBC)]** を選択して、 **「次へ」** をクリックします。

3. **[ODBC]** を選択します。

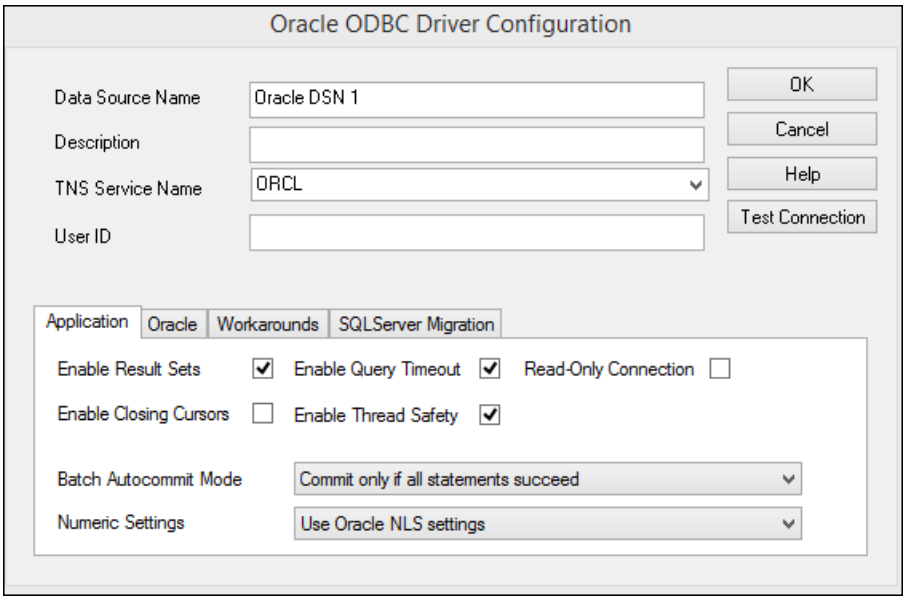
4. **[ドライバーの編集]** をクリックします。

5. 使用するOracle ドライバーをせんたくします (このサンプルでは **Oracle in OraClient11g_home1** です)。Oracle クライアントをインストール後システムで使用可能なOracle ドライバーのリストを表示します。
6. **[戻る]** をクリックします。
7. **[新しいデータソース(DSN) をドライバーを使用して作成]** を選択して、ステップ4 で選択されたOracle ドライバーを選択します。

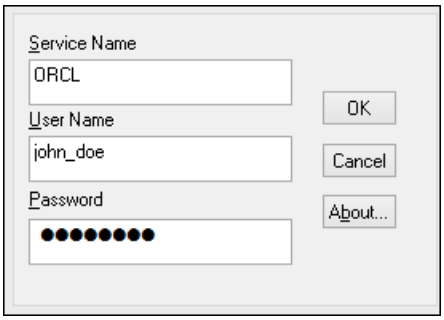


Microsoft ODBC for Oracle という名称のMicrosoft により提供されるドライバーを使用することは避けてください。Microsoft はOracle により提供されるODBC ドライバーを使用することを奨励しています。(以下参照: http://msdn.microsoft.com/en-us/library/ms714756%28v=vs.85%29.aspx)

8. **「接続」**をクリックします。



9. データソース名テキストボックスにデータソースを識別する名前を入力します (このサンプルの場合は**Oracle DSN 1** です)。
10. TNS サービス名ボックスでは、**tnsnames.ora** ファイルで定義されている通りの接続名を入力します。(前提条件を参照してください)。このサンプルでは接続名 **ORCL** はです。
11. **「OK」**をクリックします。



12. データベースのユーザー名とパスワードを入力して、「**OK**」をクリックします。

Oracle (JDBC) への接続

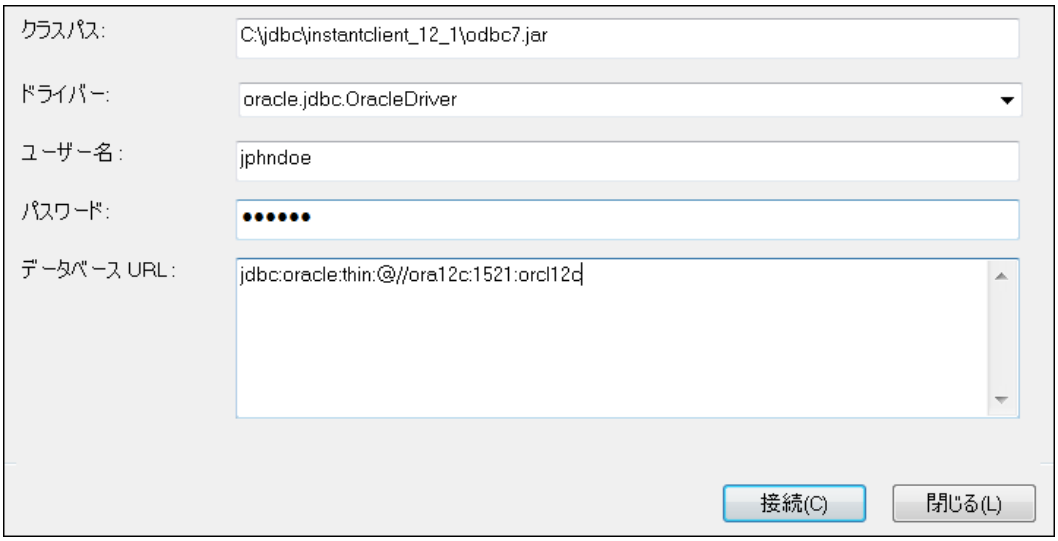
JDBC インターフェイスを使用して、クライアントマシンから、Oracle データベースサーバーへの接続の方法を説明します。接続は、Oracle Web サイトから使用することのできる **Oracle Instant Client Package (Basic)** を使用する Java 接続として作成されます。接続の型の利点は、Oracle Instant Client Package により与えられる Java 環境と jar ライブラリのみを必要とし、更に複雑なデータベースクライアントをインストールする手間を省くことができる必点です。

前提条件:

- Java Runtime Environment (JRE) または Java Development Kit (JDK) が使用中のオペレーティングシステムにインストールされている必要があります。
- オペレーティングシステムの `PATH` 環境変数は、JRE または JDK インストールの `bin` ディレクトリへのパスを含んでいる必要があります。例、`C:\Program Files (x86)\Java\jre1.8.0_51\bin`。
- **Oracle Instant Client Package (Basic)** が使用中のオペレーティングシステムで使用可能である必要があります。パッケージは、Oracle の公式 Web サイトからダウンロードすることができます。このサンプルは、Windows 32-ビットのための Oracle Instant Client Package バージョン 12.1.0.2.0 を使用します。
- データベース接続の詳細があります: ホスト、データベースパス、またはエイリアス、ユーザー名、およびパスワード。

クライアントパッケージを使用して Oracle に接続する

1. データベース接続ウィザードを開始します。
2. **「JDBC 接続」**をクリックします。
3. 「クラスパス」の横にデータベースへの接続を与える jar ファイルへのパスを入力します。必要であれば、セミコロンで区切られた jar ファイルパスのリストを入力することもできます。このサンプルでは、必要とされる jar ファイルは次のパスを使用して見つけることができます: **C:\jdbc\instantclient_12_1\odbc7.jar**。jar ファイルパスをオペレーティングシステムの CLASSPATH 環境変数に追加した場合、「クラスパス」テキストボックスを空白のままにすることができます。(次も参照: CLASSPATH の構成)。
4. 「ドライバー」ボックスから **oracle.jdbc.OracleDriver** または **oracle.jdbc.driver.OracleDriver** を選択します。このエントリは、有効な jar ファイルパスが、「クラスパス」テキストボックス内、または、オペレーティングシステムの CLASSPATH 環境変数内で検出された場合のみ使用することができます (前のステップを参照してください)。
5. データベースへのユーザー名とパスワードを対応するテキストボックスに入力します。



6. ハイライトされた値を、使用するデータベースサーバーに適用する値と置き換えて、データベース URL テキストボックス内にデータベースサーバーへの接続文字列を入力します。

```
jdbc:oracle:thin:@//host:port:service
```

7. **接続**」をクリックします。

PostgreSQL (ODBC) への接続

このトピックではODBC ドライバーを使用した、Windows マシンから PostgreSQL データベースへの接続のサンプル手順を紹介します。Windows からは、 PostgreSQL ODBC ドライバーを入手することができませんので、個別にダウンロードおよびインストールする必要があります。このサンプルでは、公式 Web サイトからダウンロードされたpsqlODBC ドライバー (バージョン09_03_300-1) を使用しています。 (データベースドライバー概要も参照してください)。

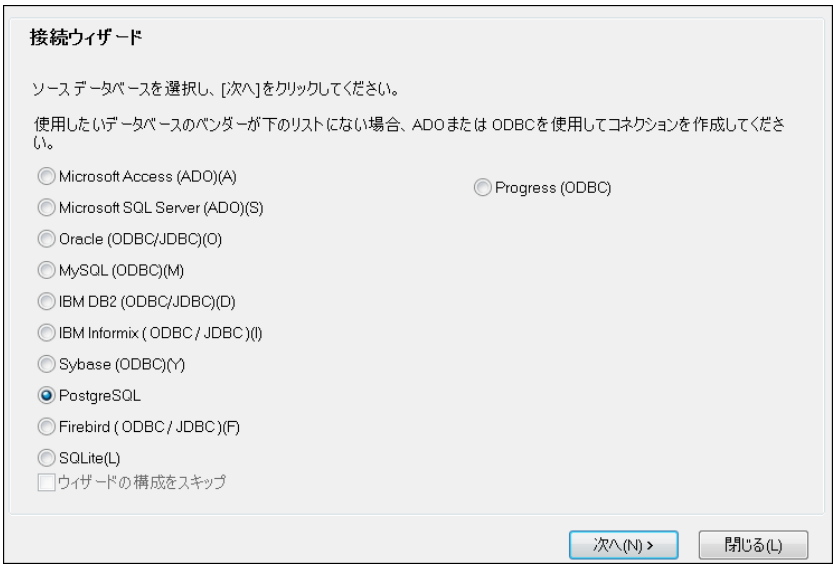
メモ (ODBC ドライバー無しで)直接 PostgreSQL データベースサーバーに接続するには、次を参照してください: PostgreSQL 接続のセットアップ.

前提条件:

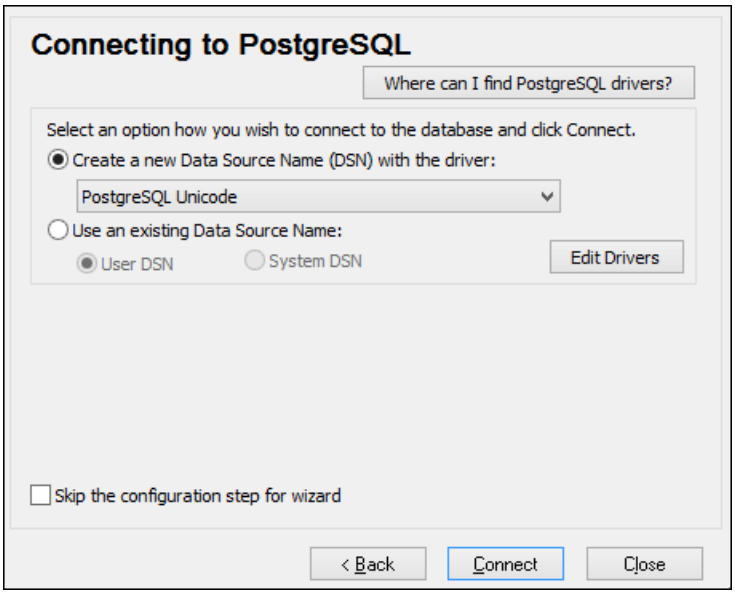
- *psqlODBC* ドライバーを使用中のオペレーティングシステムにインストールする必要があります。(インストールに関しては、ドライバーに付随するドキュメントを参照してください)。
- データベース接続の詳細: ホスト、データベース、ユーザー名およびパスワード

ODBC を使用してのPostgreSQL への接続:

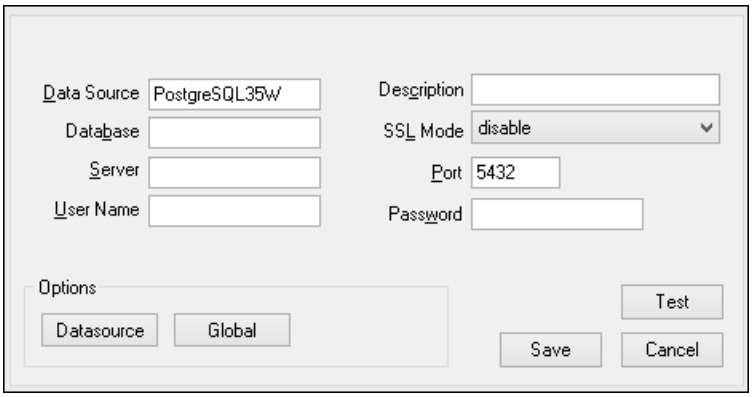
1. データベース接続ウィザードを開始します。



2. **PostgreSQL (ODBC)** を選択して、 **「次へ」** をクリックします。



3. **[ドライバーを使用して新しいデータソース名 (DSN) を作成]** を選択して、 PostgreSQL ドライバーを選択します。MySQL ドライバーがリストにない場合は、**[ドライバーを編集]** をクリックして、使用可能な PostgreSQL ドライバーを選択します (使用中のオペレーティングシステムにインストールされているすべての ODBC ドライバーがリストされています)。
4. **「接続」** をクリックします。



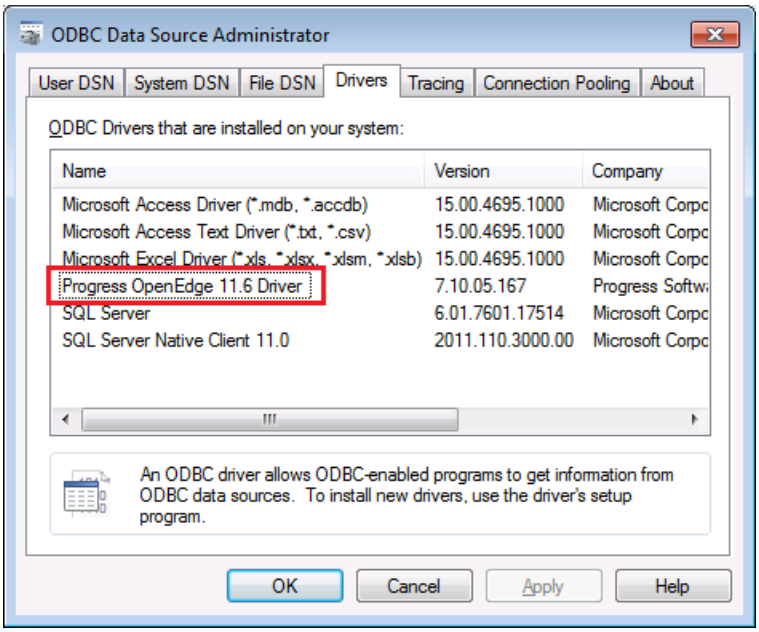
5. データベース接続資格情報 (データベース、サーバー、ポート、ユーザー名、パスワード) を入力して、 **「OK」** をクリックします。

Progress OpenEdge (ODBC) への接続

このトピックは、Progress OpenEdge 11.6 ODBC ドライバーを使用して、Progress OpenEdge データベースサーバーへの接続の簡単な手順について説明しています。

前提条件

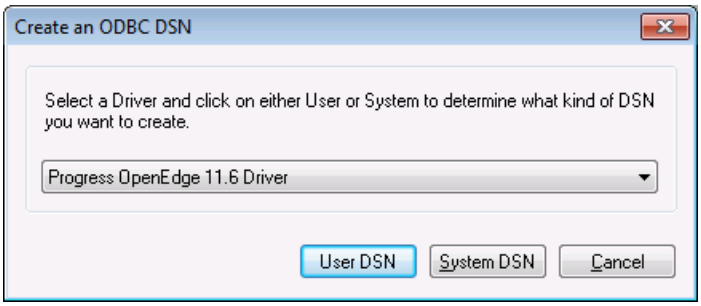
- *Progress OpenEdge ドライバーのための ODBC コネクタが使用中のオペレーティングシステムにインストールされている必要があります。* Progress OpenEdge ODBC ドライバーは、ベンダーの Web サイトからダウンロードすることができます (次も参照: データベース ドライバーの概要)。MobileTogether Designer の 32 ビット バージョンを作動する場合は、32 ビット ドライバーをダウンロードし、64 ビット バージョンを作動する場合は、64 ビット ドライバーをダウンロードしてください。インストールの後、ODBC ドライバーが使用中のマシンで使用することができるかを確認してください (次も参照: 使用することのできる ODBC ドライバーを確認する)。



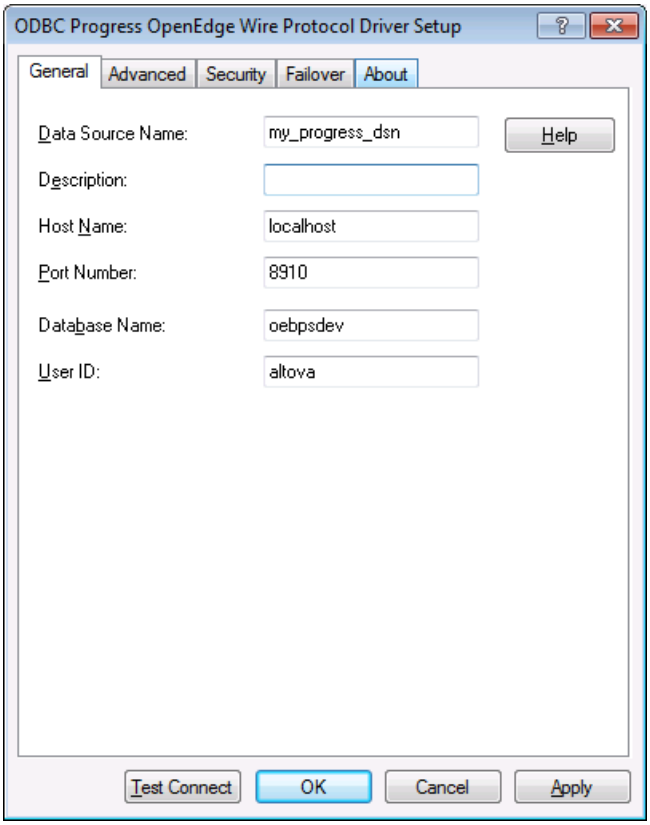
- データベース接続の詳細があります: ホスト名、ポート番号、データベース名、ユーザー ID、および パスワード。

ODBC を使用して Progress OpenEdge に接続する

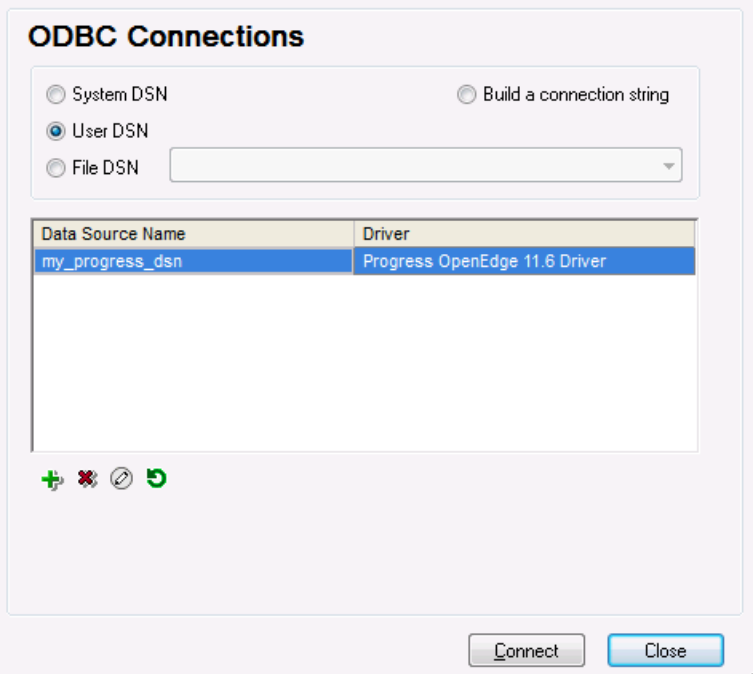
1. データベース接続ウィザードを開始します.
2. **「ODBC 接続」**をクリックします。
3. **「ユーザー DSN」**をクリックします。(または、次の手順が類似している **「システム DSN」**、または、**「ファイル DSN」**をクリックします)。
4. **「追加」** をクリックします。
5. **Progress OpenEdge ドライバー」**をリストから選択し、**「ユーザー DSN」**(または、適用できる場合は、**「システム DSN」**) をクリックします。



6. データベース接続の資格情報 (データベース、サーバー、ポート、ユーザー名、パスワード) を入力し、 **「OK」** をクリックします。入力されたデータを保存する前に、接続を検証するには、 **「接続をテストする」** をクリックします。



7. **「OK」** をクリックします。新規データソースがODBC データソースのリストに表示されます。



8. **接続**をクリックします。

Progress OpenEdge (JDBC) への接続

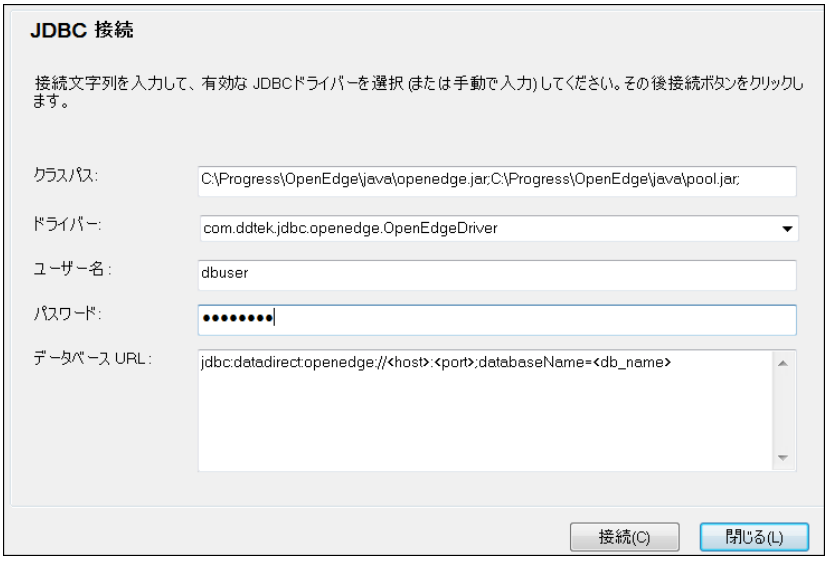
JDBC を使用してProgress OpenEdge 11.6 データベースサーバーへの接続するためのサンプルの手順が説明されています。

前提条件

- Java Runtime Environment (JRE) またはJava Development Kit (JDK) が使用中のオペレーティングシステムにインストールされている必要があります。 MobileTogether Designer (32-ビット、64-ビット) のプラットフォームが、JRE/JDK に一致していることを確認してください。
- オペレーティングシステムの`PATH` 環境変数は、JRE またはJDK インストールの`bin` ディレクトリへのパスを含んでいる必要があります。例、`C:\Program Files (x86)\Java\jre1.8.0_51\bin`.
- Progress OpenEdge JDBC ドライバーが使用中のオペレーティングシステムで使用可能である必要があります。このサンプルでは、JDBC 接続は、**C:\Progress\OpenEdge\java** にOpenEdge SDK のインストールの一部として存在する**openedge.jar** と**pool.jar** ドライバーコンポーネントファイルにより与えられています。 .
- データベース接続の詳細があります: ホスト、データベースパス、またはエイリアス、ユーザー名、およびパスワード。

JDBC を使用して OpenEdge へ接続する

1. データベース接続ウィザードを開始します.
2. 「**JDBC 接続**」をクリックします。.
3. 「クラスパス」の横にデータベースへの接続を与える jar ファイルへのパスを入力します。必要であれば、セミコロンで区切られた jar ファイルパスのリストを入力することもできます。このサンプルでは、必要とされる jar ファイルは次のパスを使用して見つけることができます`C:\Progress\OpenEdge\java\openedge.jar;C:\Progress\OpenEdge\java\pool.jar;`。 jar ファイルパスをオペレーティングシステムの CLASSPATH 環境変数に追加した場合、「クラスパス」テキストボックスを空白のままにすることができます。(次も参照: CLASSPATH の構成)。
4. 「ドライバー」ボックスから、**com.ddtek.jdbc.openedge.OpenEdgeDriver** を選択します。このエントリは、有効な jar ファイルパスが、「クラスパス」テキストボックス内、または、オペレーティングシステムの CLASSPATH 環境変数内で検出された場合のみ使用することができます (前のステップを参照してください)。



5. データベースへのユーザー名とパスワードを対応するテキストボックスに入力します。
6. ハイライトされた値を、使用するデータベースサーバーに適用する値と置き換えて、データベースURL テキストボックス内にデータベースサーバーへの接続文字列を入力します。

```
jdbc:datadirect:openedge://host:port;databaseName=db_name
```

7. **接続**」をクリックします。

Sybase (JDBC) への接続

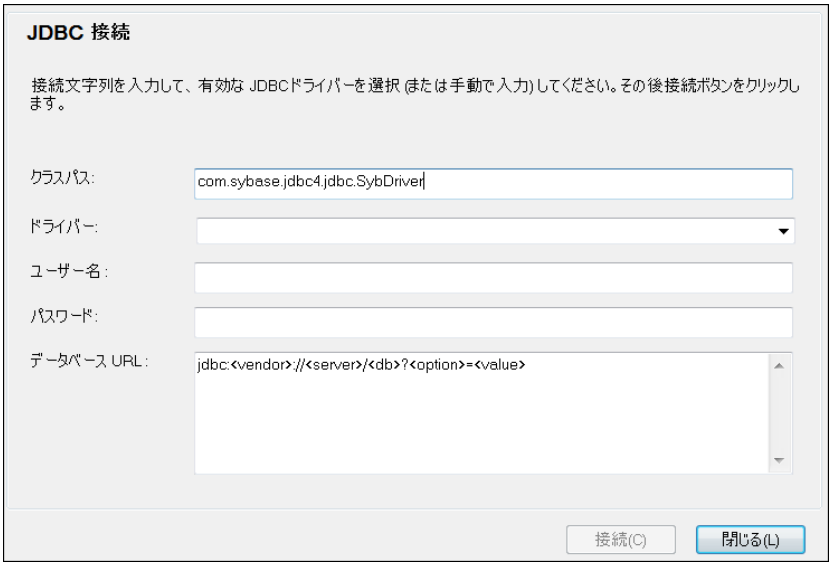
このトピックでは、JDBC を使用して Sybase データベースサーバーに接続する手順のサンプルを説明します。

前提条件:

- Java Runtime Environment (JRE) がオペレーティングシステムにインストールされている必要があります。
- Sybase *jConnect* ドライバーがオペレーティングシステムにインストールされている必要があります (このサンプルでは、*Sybase Adaptive Server Enterprise PC Client* インストールの一部としてインストールされている *jConnect 7.0* が使用されています)。ドライバーのインストールの手順に関しては、ドライバーに付随するドキュメント、または Sybase のドキュメントを参照してください)。
- データベース接続の詳細: ホスト、サーバーの名前、データベース ポートユーザー名、およびパスワード

JDBC を使用して Sybase へ接続する

1. データベース接続ウィザードを開始します。
2. 「**JDBC 接続**」をクリックします。
3. 「クラスパス」の横にデータベースへの接続を与える jar ファイルへのパスを入力します。必要であれば、セミコロンで区切られた jar ファイルパスのリストを入力することもできます。このサンプルでは、必要とされる jar ファイルは次のパスを使用して見つけることができます: **C:\sybase\jConnect-7_0\classes\jconn4.jar**。jar ファイルパスをオペレーティングシステムの CLASSPATH 環境変数に追加した場合、「クラスパス」テキストボックスを空白のままにすることができます。(次も参照: CLASSPATH の構成)。
4. 「ドライバー」ボックスから、 を選択します。「ドライバー」ボックスから **com.sybase.jdbc4.jdbc.SybDriver** を選択します。このエントリは、有効な jar ファイルパスが、「クラスパス」テキストボックス内、または、オペレーティングシステムの CLASSPATH 環境変数内で検出された場合のみ使用することができます (前のステップを参照してください)。



5. データベースのユーザー名とパスワードを対応するテキストボックスに入力します。
6. ハイライトされた値をデータベースサーバーに適用することのできる値と置き換えて、データベース URL テキストボックス内のデータベースサーバーの接続文字列を入力します。

```
jdbc:sybase:Tds:hostName:port/databaseName
```

7. 「**接続**」をクリックします。

Teradata (ODBC) への接続

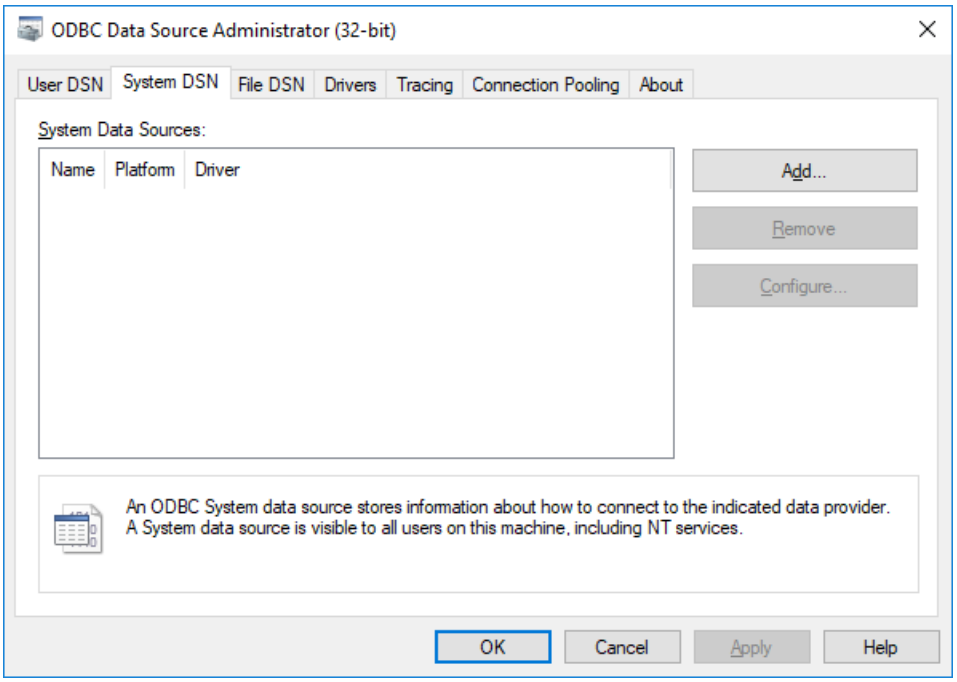
このサンプルでは、ODBC を使用したSQL Teradata データベースへの接続方法について説明されています。

必要条件:

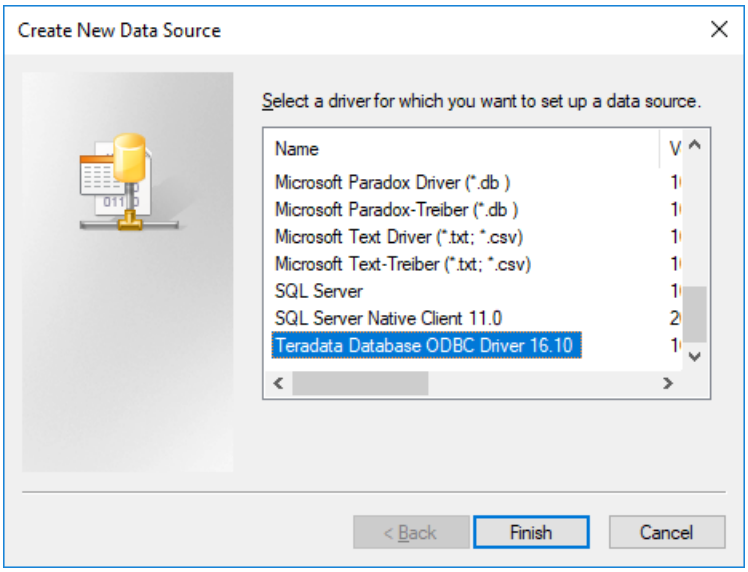
- Teradata ODBC ドライバーがインストールされる必要があります (参照: http://downloads.teradata.com/download/connectivity/odbc-driver/windows。このサンプルは、Windows バージョン16.20.00 のためにODBC ドライバーを使用しています。
- 次のデータベース接続詳細が存在します: ホスト、ユーザー名、および パスワード。

ODBC を使用したTeradata への接続の方法:

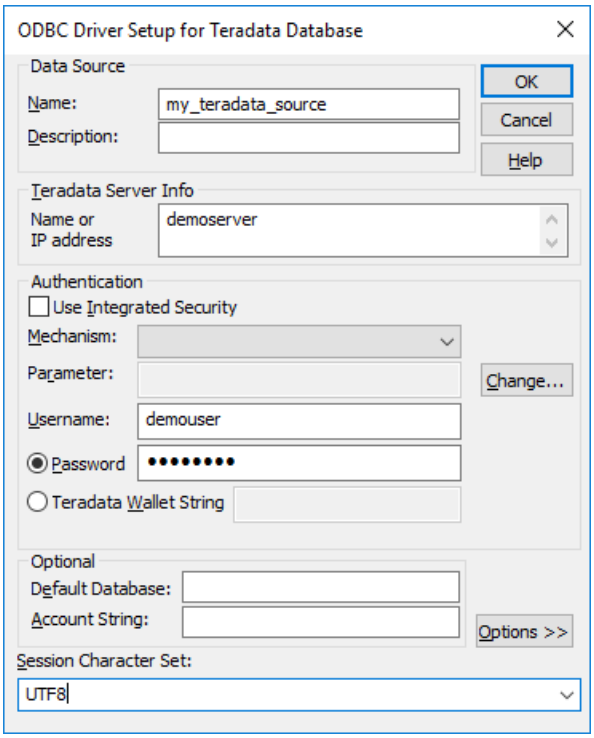
1. **Windows** キーを押して、「ODBC」を入力し、**「ODBC データソースのセットアップ (32-ビット)」**を提案されるリストから選択します。64-ビットODBC ドライバーが存在する場合、**「ODBC データソースのセットアップ (64-ビット)」**を選択し、次のステップ内で64-ビットMobileTogether Designer を使用します。



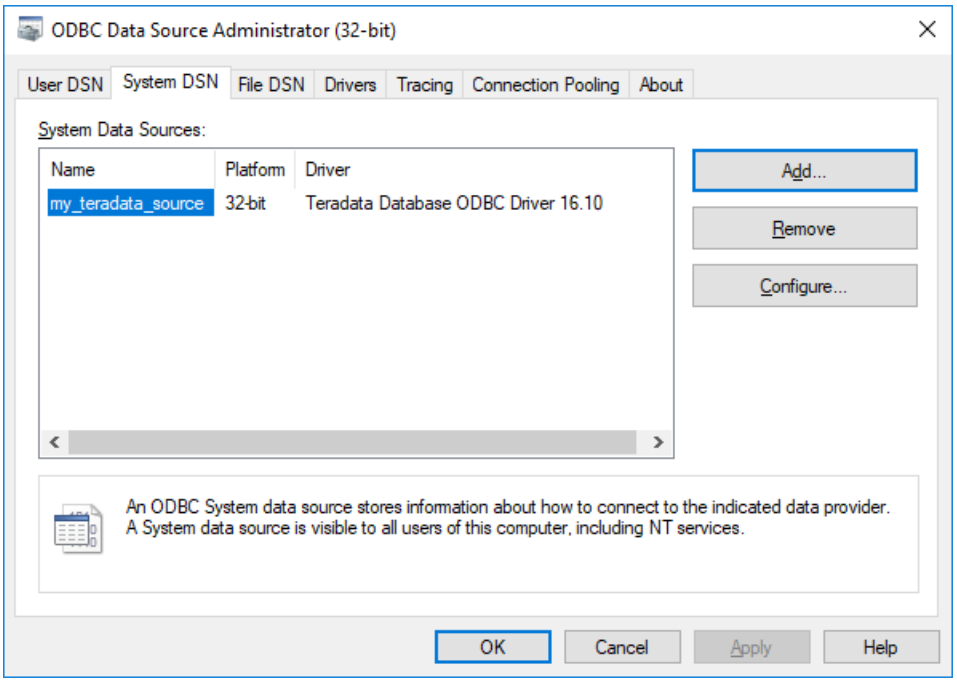
2. **「System DSN」**タブをクリックして、**「追加」**をクリックします。



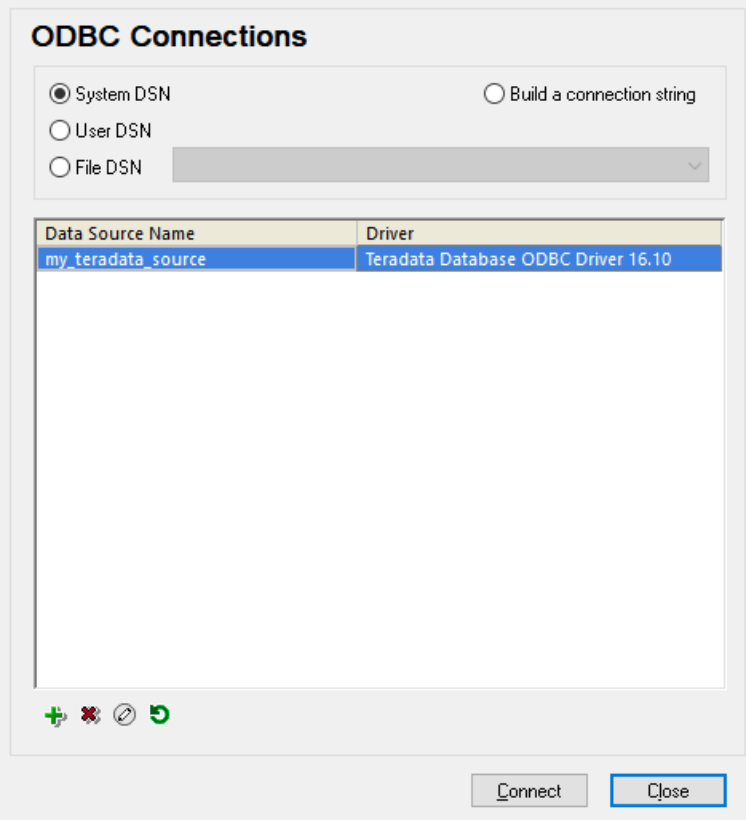
3. **「teradata データベースODBC ドライバー」**を選択して、**「終了」**をクリックします。



4. 名前を入力し、任意で、このODBC データソースを将来識別できる詳細を与えます。また、データベース接続資格情報 (データベースサーバー、ユーザー、パスワード) を入力し、オプションでデータベースを選択します。
5. **「OK」**をクリックします。データソースがリスト内に表示されます。



6. MobileTogether Designer を作動し、データベース接続ウィザードを開始します。
7. **「ODBC 接続」**をクリックします。



8. **「System DSN」**をクリックし、前に作成されているデータソースを選択し、**「接続」**をクリックします。

メモ　次のエラーが発生した場合: 「ドライバーが無効として返されました(または返しませんでした) SQL_DRIVER_ODBC_VER: 03.80」「ドライバーが無効と返されました」。ODBC クライアントへのパスが使用中のシステムの PATH 環境変数内に存在することを確認してください (例えば、このロケーションにインストールされている場合、**C:\Program Files\Teradata\Client\16.10\bin**)。このパスが不足している場合、手動で追加します。

Teradata (JDBC) への接続

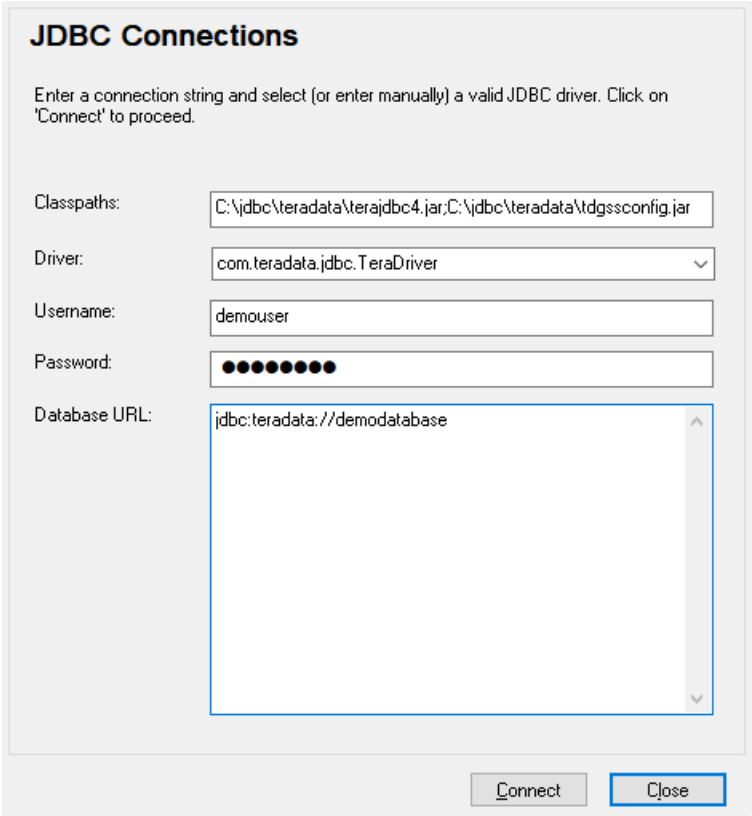
このサンプルでは、JDBC を使用したTeradata データベースへの接続方法について説明されています。

必要条件:

- Java Runtime Environment (JRE) またはJava Development Kit (JDK) が使用中のオペレーティングシステムにインストールされている必要があります。
- JDBC ドライバー(データベースへの接続を提供する1つまたは複数の jar ファイル) がオペレーティングシステム上で使用できる必要があります。このサンプルでは、Teradata JDBC ドライバー16.20.00.02 is used. 詳細に関しては、次を参照してください: http://downloads.teradata.com/download/connectivity/jdbc-driver。
- 次のデータベース接続詳細が存在します: ホスト、データベース、ポート、ユーザー名、および、パスワード。

JDBC を使用したTeradata への接続の方法:

1. データベース接続ウィザードを開始します。
2. **「JDBC 接続」**をクリックします。
3. 「クラスパス」の横に、データベースへの接続を与える jar ファイルへのパスを入力します。必要な場合、セミコロンで区切られた.jar ファイルパスのリストを入力することができます。このサンプルでは、 jar ファイルは、次のパスの場所に存在します: **C:\jdbc\teradata**。オペレーティングシステムのCLASSPATH 環境変数 に.jar ファイルパスが追加されている場合、「クラスパス」テキストボックス空のままにすることができます (次も参照してください: CLASSPATH の構成).
4. 「ドライバー」ボックス内で、**com.teradata.jdbc.TeraDriver** を選択します。有効な jar ファイルパスが「クラスパス」テキストボックスで、または、オペレーティングシステムのCLASSPATH 環境変数内に存在すると、使用することができます (前のステップを参照してください)。



5. ユーザー名とパスワードを対応するテキストボックスのデータベース内に入力します。
6. ハイライトされた値をデータベースサーバーと適用することのできる値を置換え、データベースURL テキスト内にデータ

ベースサーバーに接続文字列を入力します。

```
jdbc:teradata://databaseServerName
```

7. **接続**をクリックします。

13.3 Linux と macOS でのデータベース接続

以下のAltova サーバー製品にライセンスが与えられている場合？MobileTogether Server、MapForce Server またはStyleVision Server、よくあるシナリオとして考えられるのは、MobileTogether デザインのデザイン、MapForce マッピング、またはStyleVision 変換を Windows デスクトップコンピューターでおこない、実行を自動化するためにサーバーマシン(Windows、Linux、またはOS X / macOS) にデプロイすることです。

このドキュメントでは、[サーバー実行ファイル] は以下のファイルの種類を意味します:

- MapForce Server 実行ファイル(.mfx)
- MobileTogether デザインファイル(.mtd)
- Portable XML Forms (.pxf) としてのStyleVision 変換 (.sps) パッケージ

サーバー実行ファイルをデプロイする時以下のシナリオが考えられます:

1. **"Windows上でのデザインと実行"**。 このシナリオでは、MobileTogether デザインのデザイン、MapForce マッピング、StyleVision 変換をWindows 上で行い、対応するサーバ実行ファイルも Windows システムで実行します。(同じWindows マシンまたは、リモートのWindows サーバーで行うことができます)。
2. **"Windows 上でデザイン、Linux またはmacOS上で実行"**。 このシナリオでは、上記のファイルのデザインを Windows 上で行い、対応するサーバー実行ファイルをLinux およびOS X / macOS に実行するためにデプロイします。

> **"Windows上でのデザインと実行"**シナリオでは、使用可能なデータベーステクノロジーとドライバーはADO、ODBC、JDBC およびSQLite 接続 (データベースドライバー概要を参照) から構成されます。
>
> **"Windows 上でデザイン、Linux またはMac上で実行"**シナリオでは、ADO およびODBC 接続はサポートされていません。このシナリオでは、直接的なSQLite 接続 (SQLite 接続を参照) とJDBC 接続 (JDBC 接続を参照) がサポートされます。

サーバーにサーバー実行ファイルをデプロイすると、データベースはデプロイされるパッケージに含まれておらず、(SQLite およびMicrosoft Access等のファイルベースのデータベースでも同様) デプロイサーバー側でも接続がセットアップされていなければなりません。すなわち、同じデータベース構成がデザインを行うオペレーティングシステムとファイルをデプロイするサーバー側で行われていなければならないのです。このルールに対しての例外は、(ドライバーをベースにしない) ネイティブの PostgreSQL 接続です。ネイティブのPostgreSQL 接続は、MobileTogether Designer の外部の構成を必要としません。詳細に関しては、次を参照してください PostgreSQL 接続のセットアップ。

通常、双方のマシンで同じデータベースの構成が必要なため、サーバー実行ファイルを異なるオペレーティングシステムにデプロイするシナリオは若干複雑です。デザインをローカルで行い、デプロイをリモートで行う複雑性をバイパスするため、MapForce、MobileTogether Designer および StyleVision で使用可能なグローバルリソース機能を使用することを検討してください。

例えば、異なる2 つのグローバルリソース構成を定義し、同じデータベースに接続することができます: 1 つはWindows スタイルのパス規約を使用して接続設定を行い、もう1 つは、Linux スタイルのパス規約を使用します。最初の接続を使用して、デザイン中にファイルのテストを行い、第 2 の接続をLinux サーバー上でのファイルの実行に使用することができます。

Linux と macOS 上での SQLite 接続

Altova サーバー製品には、サポートが統合されているため、Linux とmacOS にSQLite を別途にインストールする必要はありません。ですから、サーバー実行ファイルは、SQLite データベースへの呼び出しを含み、SQLite を最初にインストールすることなく実行することができます。ですが、サーバー実行ファイルがLinux またはOS X / macOS マシン上でデータベースファイルへの正しいパスを使用していることを確認する必要があります。すなわち、Linux またはOS X / macOS サーバーでサーバー実行ファイルを実行する前に、SQLite データベースファイルがPOSIX (Portable Operating System Interface) 準拠であるパスを介して参照されていることを確認してください。これは、Windows スタイルではないのドライブ文字がパスに使用されていないこと、また、ディレクトリがスラッシュ(/) で区切られていることを前提にしています。例えば、パス**/usr/local/mydatabase.db** はPOSIX 準拠で、パス**c:\sqlite\mydatabase.db** は準拠ではありません。

Altova 製品の外部で、SQLite データベースと作業するためにシェルが必要な場合、(例えば、データベースの作成など)、コマンドラインシェルを公式のWeb サイト(www.sqlite.org) から、または第三者の代替よりダウンロードする必要があるかもしれません。

Linux と macOS 上での JDBC 接続

Linux または macOS 上での JDBC 接続のセットアップ:

1. データベースベンダーにより提供されるJDBC ドライバーをダウンロードして、オペレーティングシステムにインストールします。オペレーティングシステムが32-ビットで作動する場合は32-ビット版バージョンを、オペレーティングシステムが64-ビットで作動する場合は64-ビット版バージョンを選択してください。
2. 環境変数をJDBC ドライバーがインストールされている場所に設定してください。通常、CLASSPATH 変数と必要があればその他複数を設定します。どの特定の環境変数が構成されるかを検出するには、JDBC ドライバーに付随するドキュメントを確認してください。

メモ macOS では、システムはJDBC ライブラリが**/Library/Java/Extensions** ディレクトリにあることを必要とします。ですから、JDBC ドライバーをこの場所でアンパックすることが奨励されます。それ以外の場合、システムを構成し、JDBC ライブラリがJDBC ドライバーをインストールしたパスを検索するようにする必要があります。

OS X Yosemite 上での Oracle 接続

OS X Yosemite 上では、**Oracle Database Instant Client** を介して Oracle データベースに接続することができます。Java 8 以前の Java バージョンの Mac では **JDBC Thin for All Platforms** ライブラリを介して接続することもできます。この点に関しての手順はこのトピックでは、無視することができます。

Oracle Instant Client を Oracle の公式ダウンロードページからダウンロードすることができます。Oracle ダウンロードページでは複数の Instant Client パッケージがあることに注意してください。Oracle Call Interface (OCI) サポートのあるパッケージが選択されていることを確認してください (例えば Instant Client Basic)。また、オペレーティングシステムが 32-ビットで作動する場合は 32-ビット版バージョンを、オペレーティングシステムが 64-ビットで作動する場合は 64-ビット版バージョンを選択してください。

ダウンロードした後、Oracle Instant Client をアンパックし、インストーラと共にパックされているプロパティリストファイル (.plist) を編集すると、以下の環境変数が対応するドライバーパスをポイントします。例えば

変数	サンプル値
CLASSPATH	/opt/oracle/instantclient_11_2/ojdbc6.jar:/opt/oracle/instantclient_11_2/ojdbc5.jar
TNS_ADMIN	/opt/oracle/NETWORK_ADMIN
ORACLE_HOME	/opt/oracle/instantclient_11_2
DYLD_LIBRARY_PATH	/opt/oracle/instantclient_11_2
PATH	$PATH:/opt/oracle/instantclient_11_2

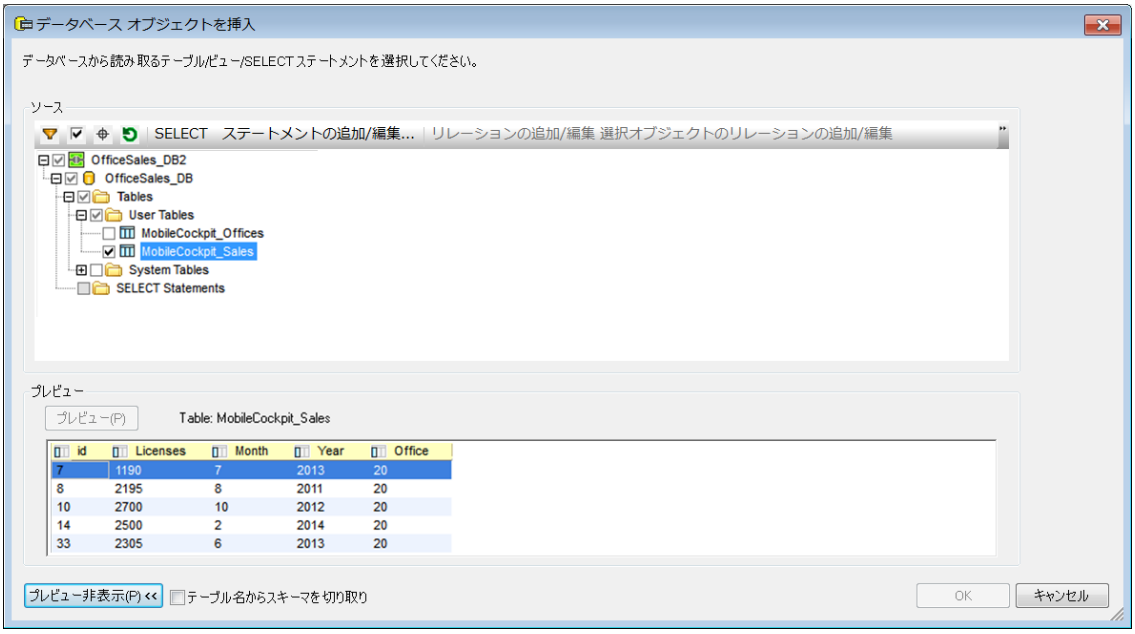
メモ　使用中のオペレーティングシステムで Oracle Instant Client ファイルがインストールされている場所で上記のサンプル値を編集してください。

13.4 データソースとして DB オブジェクトを選択する

DB に接続後、[データベース オブジェクトの挿入] ダイアログが表示されます(*下のスクリーンショット*)。このダイアログで、テーブルを選択または SELECT ステートメントを使用してデータソースとして追加するオブジェクトを選択することができます。

テーブルをデータソースとして選択する

下のスクリーンショットでは、MobileCockpit_Sales テーブルが選択されています。**[プレビューの表示]** をクリックしてダイアログの下の [プレビュー] ペインのテーブルの選択された行を表示します。

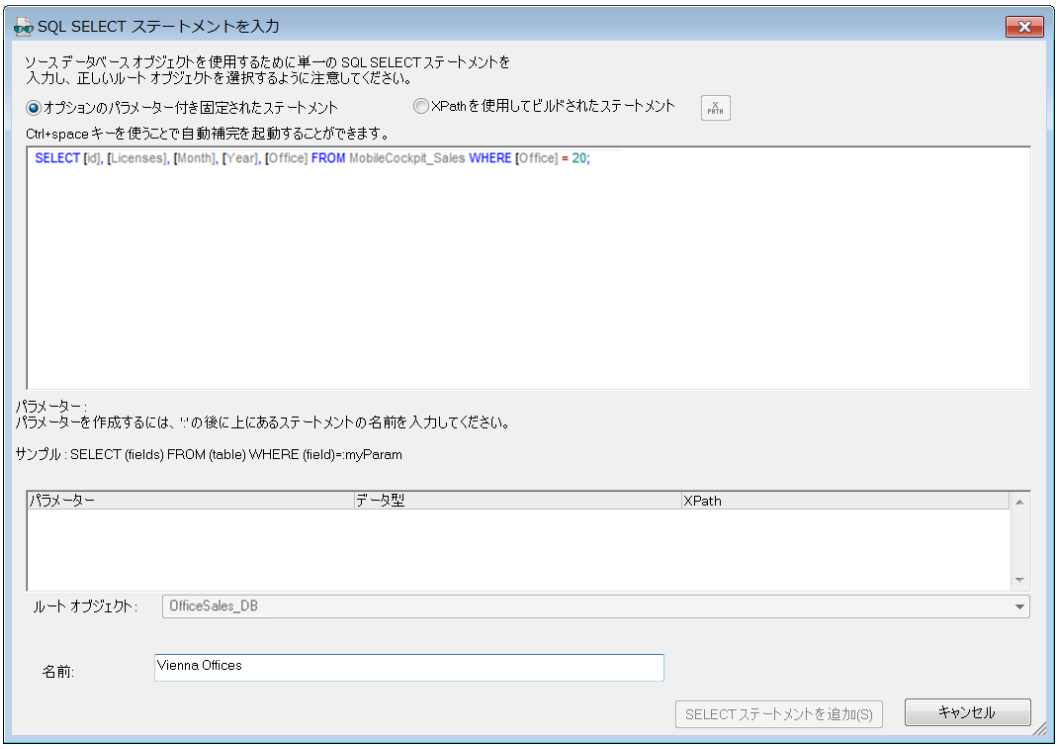


メモ 1 つのテーブルのみを一度にデータソースとして選択することができます。

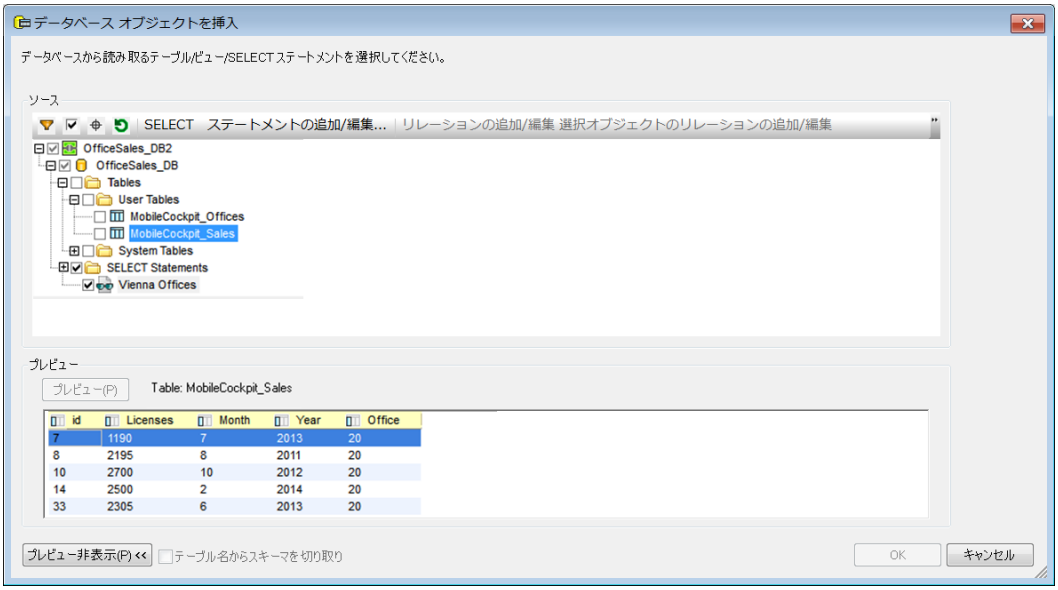
SELECT ステートメントを使用する

SELECT ステートメントを使用してデータソースを選択するには、次を行います:

1. [データベース オブジェクト] ダイアログで(*上のスクリーンショット*) **[追加 編集 SELECT ステートメント]** をクリックします。これにより[SQL SELECT ステートメントの入力] ダイアログが表示されます(*下のスクリーンショット*)。
2. SELECT ステートメントを入力し、名前を付けます。下のスクリーンショットでは、WHERE 句、id=20 を持つ行を選択します。ステートメントに Vienna Offices という名前を与えます。



3. **[SELECT ステートメントの追加]** をクリックしてステートメントを追加します。表示される SELECT ステートメントはソースペインに表示されます (*下のスクリーンショット参照*)。
4. [プレビュー] ペインに選択されたテーブルの行を表示するため **[プレビューの表示]** をクリックします。



5. **[OK]** をクリックしてページソースにデータソースを追加します。

13.5 DB データの編集

このセクション:

- OriginalRowSet について
- テーブル内の DB データおよび他のコントロールの編集
- ノードの更新、挿入、追加、および削除
- DB 実行アクションと $MT_DBExecute_Result 変数
- MobileTogether Designer 内の主キー

OriginalRowSet に関して

データを編集し、保存するには、ツリーはページ ソースに、RowSet 要素のコピーであるOriginalRowSet 要素を含む必要があります。オリジナルのデータはOriginalRowSet 要素に保存され、編集されたデータはRowSet 要素に保存されます。データ ソースが保存されると、OriginalRowSet とRowSet の 2 つのツリーの差異が計算され、ページ ソースは差異を基に更新されます。修正に成功すると、修正されたデータは、OriginalRowSet にコピーされ、OriginalRowSetが新しく保存されたDB データを含み、修正プロセスが繰り返すことができます。

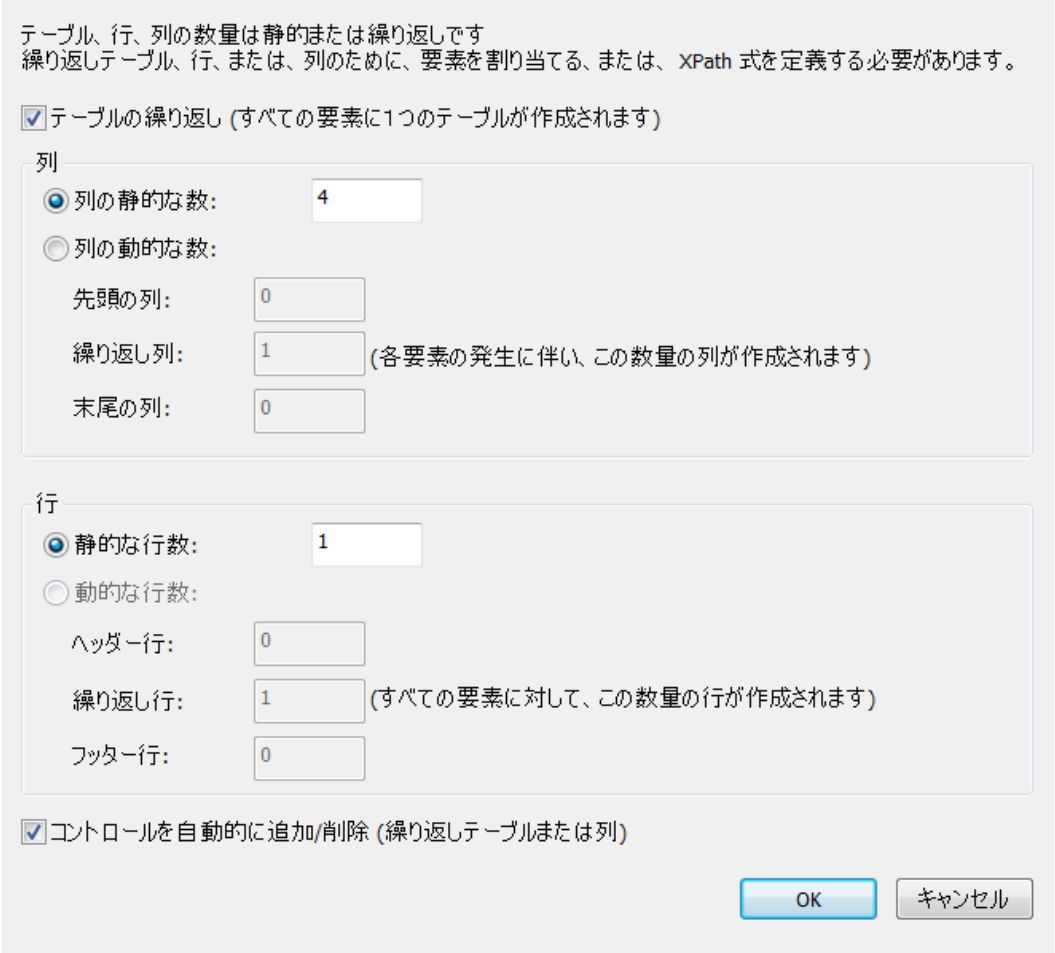
ページ ソースのOriginalRowSet を作成するには、ページ ソースのルート ノードを右クリックして、コマンド **[OriginalRowSet の作成]** を切り替えます。

[OriginalRowSet の作成] コマンドは、データベース型 ($DB) ルート ノードのために有効化されています。ページ ソースの元のデータを含むデータ構造を作成/削除する切り替えコマンドです。ユーザーにより修正されたデータはデータ ソースにより作成されたメインの構造に保存されています。修正されたデータがDB に保存されると、OriginalRowSet 構造は新たにDB に保存されたデータを含むために修正されます。これにより元のDB データはツリーで使用可能となります。

テーブル内の DB データおよび他のコントロールの編集

DB データ内で編集することのできるコントロールを作成するには、次を行います:

- コンボボックスまたは編集フィールドなどエンドユーザーにより編集することのできるコントロールです。ラベルコントロールは編集することができません。テーブルが繰り返し行から生成される場合は、編集可能なコントロールをテーブルのセル内に追加してください。手順はデータベースとチャートチュートリアル内のEdit Offices Table セクションまたはサンプルを参照してください。また、テーブルとの作業方法の詳細を参照してください。
- コントロールで、編集するデータ ソース ノードへのページ ソース リンクを作成します。データ ソース ノードをコントロールにドラッグしてください。
- テーブルを作成する場合、繰り返し行を持つテーブルを使用する場合、自動的にコントロールを追加/削除するオプションを使用してください。(*下のスクリーンショットを参照*)。



DB データソースの繰り返し行要素にリンクされている繰り返し行を持つテーブルを使用する利点は、テーブル行が追加されると、DB 行も自動的に追加されることです。更に詳しい情報は、テーブルとの作業方法の詳細を参照してください。例えば、データベースとチャートチュートリアルとMobileTogether Designer DB ファイルを参照してください。

ノードの更新、挿入、追加、および削除

更新データアクションにより、ページまたはコントロールイベントがトリガーされると内のノードを編集することができます。

- ノードの更新: アクションのXPath 式より生成されたまたは取得されたDB 列などの1つ以上のノードを更新します。
- ノードの挿入: アクションのXPath 式により選択されたノードの前 DB 列などの1つ以上のノードを挿入します。
- ノードの追加: アクションのXPath 式により選択されたノードの最初または最後の子としてDB 列などの1つ以上のノードを追加します。追加されたノードの構成およびコンテンツを編集することができます。
- ノードの削除: アクションのXPath 式により指定されたDB 列などの1つ以上のノードを削除します。

メモ これらのアクションはローカルのデータソリーで実行されます。変更されたデータソリーは、エンドユーザーの変更がDB にパスされるようこのためにDB に保存されなければなりません。

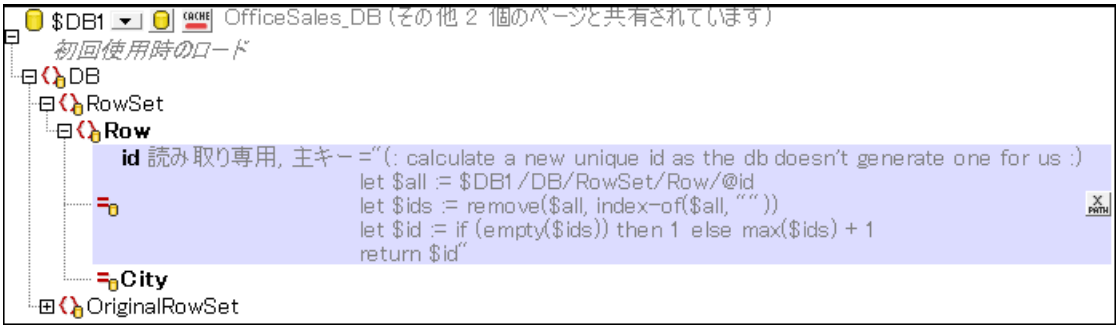
DB 実行アクションと $MT_DBExecute_Result 変数

DB 実行アクションにより、INSERT、APPEND、UPDATE および DELETE を含むパワフルな SQL ステートメントを使用して DB を修正することができます。前のセクションで説明されているアクションとは次の重要な点で異なります: DB 実行の SQL ステートメントにより作成された変更は即時 DB に保存されます。前のセクションで説明されているアクションの場合、変更を保存するには保存メカニズムを使用する必要があります。

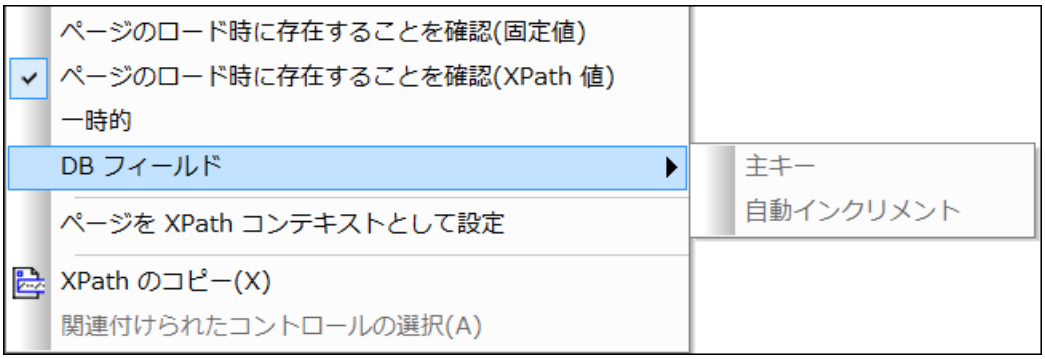
DB 実行アクションが SQL ステートメントを実行した後、結果を変数名 **$MT_DBExecute_Result** に保存します。この変数はプロジェクト内で XPath 式として使用されます。結果、(オプションでパラメーターをベースにした DB からの構造とデータを選択することができ、保管、変更、または他の箇所に挿入することができます。

MobileTogether Designer の主キー

DB の主キーは通常、自動インクリメントです。この場合、新しい行がテーブルに追加され、追加された行の主キー列は自動的にインクリメントされます。MobileTogether Designer では、主キーからテーブルが取得されると、自動インクリメント情報が自動的に取得され[ページ ソース] ペインに表示されます(*下のスクリーンショット参照*)。

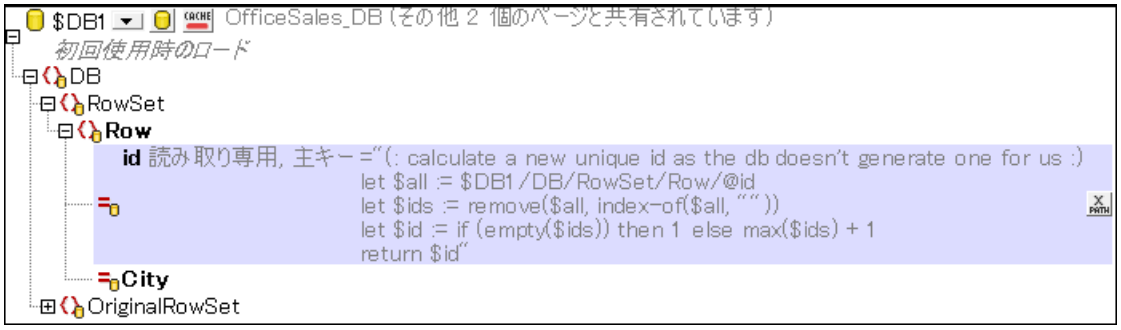


この情報の自動取得が成功しなかった場合、切り替えコマンドを含むツリー ノードのコンテキスト メニューによりノードに正確な注釈をつけることができます (*下のスクリーンショット参照*)。



主キー列が自動インクリメントでない場合、追加された行の新しい主キーの値は XQuery 式を使用して自動的に生成されなければなりません。これは、主キー列を編集することができないからです。XQuery 式は、主キーノードのコンテキストメニューコマンド **[ページのロード時に存在することを確認 (XPath 値)]** を使用して挿入されます。下の例では、次の XQuery 式を使用して新しい値が主キー @id のために生成されます:

```
let $all := $DB1/DB/RowSet/Row/@id
let $ids := remove($all, index-of($all, ""))
let $id := if (empty($ids)) then 1 else max($ids) + 1
return $id
```



13.6 DB にデータを保存する

このセクション:

- ソリューションの進行状況をベースにして保存する
- 保存アクション
- DB 実行アクション
- 保存する列をフィルターする
- OriginalRowSet について
- トランザクションをコミットする

ソリューションの進行状況をベースにして保存する

$DB ルートノードのコンテキストメニューには、ルートノードでのデータソースの入れ替えをしソリューションの進行中の異なる時点で保存することを可能にする **[データの保存]** コマンドがあります (*下のスクリーンショット*)。オプションは下に説明されています。デフォルトのオプションである [自動ではない] が選択されている場合、イベントの保存アクションがトリガーされるとデータが保存されます。

[データの保存] コマンドは、サブメニューのロールアウトで、(一度に 1 つのみ選択可能な) 次の相互に排他的なオプションを表示します:

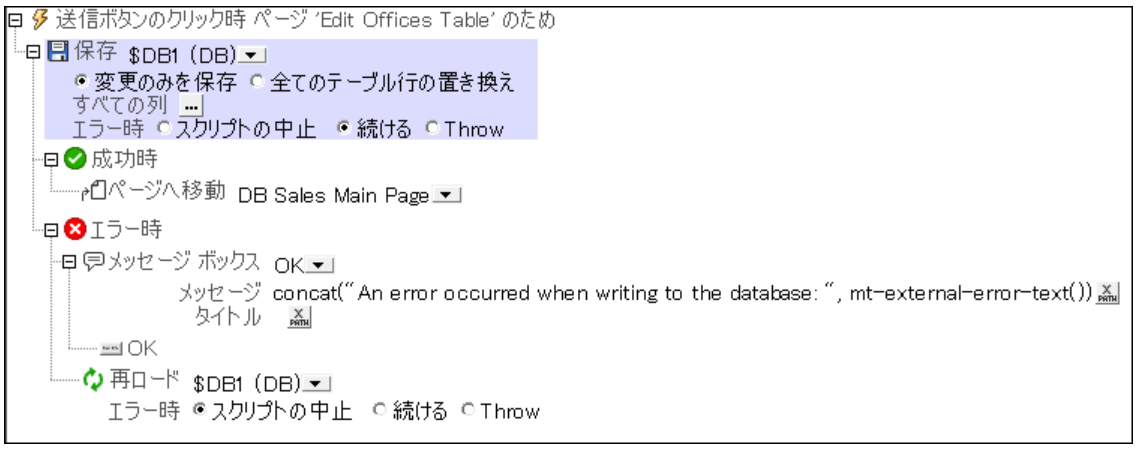
- *すべてのページ移動時*: ツリーのデータはツリーを含むページから移動する際保存されます。
- *ソリューションの完了時*: ソシューションの状態、または終了方法に関わらず、ソリューションを完了するとツリーのデータは保存されます。
- *最後の送信時*: ツリーのデータは、ワークフローの処理がデザインされたとおり、最初のページから最後のページまで、最後の **[送信時]** ボタンがタップされた時まで保存されます。もしこのオプションが選択されていて、ソリューションが最後の **[送信時]** ボタンがタップされる前に終了した場合、ツリーのデータは保存されません。
- *自動ではない*: ツリーは自動的に保存されません。保存、ファイルに保存 、または HTTP/FTP をコード保存 アクションを使用してデータを保存します。

デフォルトの設定は、*自動ではない*。

保存アクション

保存アクションが定義されるページイベントまたはコントロールイベントがトリガーされると、DB にデータを保存することができます。イベントの例としては、エンドユーザーが **[送信]** ボタンをクリックすることがあげられます。下のスクリーンショットでは、**[送信]** ボタンは Edit Offices Table バーにあります。

保存アクションはページ アクションまたはコントロール アクションで定義することができます。対応するアクション ダイアログには、すべてのアクション ダイアログ [**ページ | アクションの概要**] からアクセスすることができます。下のスクリーンショットは、**送信ボタンのクリック時** イベントに対して定義されている保存アクションを持つページ アクション ダイアログを表示しています。



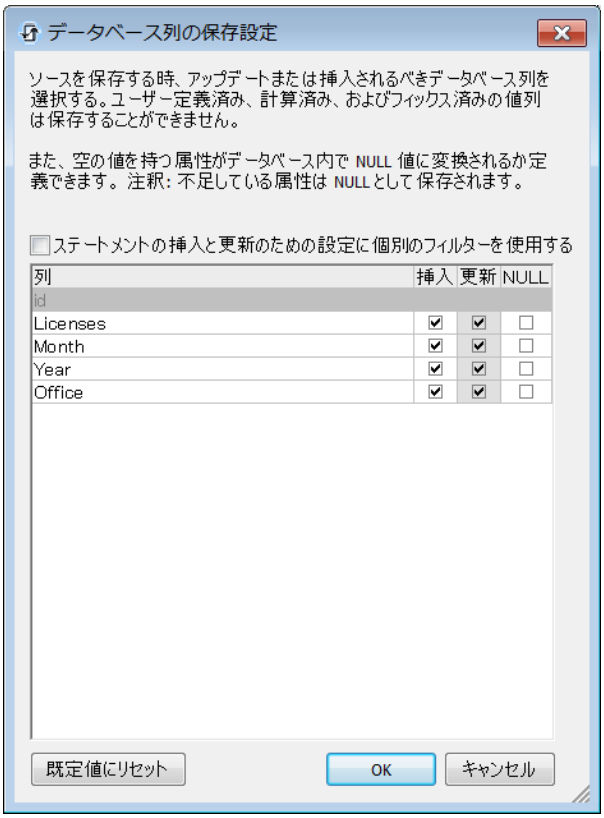
メモ　DB に秘密キーがある場合、秘密キーは変更、追加、削除された記録のみを保存するために使用されます。DB に秘密キーが無い場合、変更されたテーブル全体が、元のテーブルを置換、DB に保存されます。

DB 実行アクション

DB 実行アクションは、DB データを変更することのできる強力なメカニズムです。このアクションとSQL ステートメントを使用して、データを更新し保存することができます。このアクションの使用に関しては、[ページデザイン| データベース| DB 実行アクション]のセクションを参照してください。

保存する列をフィルターする

$DB ルートノードのコンテキスト メニューで、[**列をフィルターする**] コマンドを選択して[データベース列の保存設定] ダイアログを表示し(*下のスクリーンショット参照*)、更新または挿入する列を選択します。



ダイアログは、DB データソースのための列を表示します。その列が更新され、または、挿入された値を取ることができるかを指定することができます。(更新は、既存の行要素内の変更されたデータを指します。挿入された値は、新規に追加された行要素を指します)。デフォルトでは、各列の挿入と更新のオプションはペアとして選択されます。ですが、異なるオプションを列の挿入と更新オプションのために指定する場合、挿入慮更新ステートメントのために個別のフィルター設定を使用するチェックボックスをチェックしてください。DB のカラム内の *NULL* チェックボックスをチェックすることで、空の値を持つ属性は、内で`NULL` 値に変換されることができます。不足する属性は常に`NULL` として保存できることに注意してください。

(ユーザーにより定義されているため、固定されているため、または計算されているため) 更新することのできない列は、挿入、更新またはオプションチェックボックスを使用することができません。 上のスクリーンショットでは、固定値を持つため、`ID` 列は、更新されることができません。更新しない列の選択を解除します。全ての列を更新するために、保存設定をリセットする場合は、**[デフォルトにリセット]** をクリックします。

`OriginalRowSet` について

データを編集し、保存するには、ツリーはページ ソースに、`RowSet` 要素のコピーである`OriginalRowSet` 要素を含む必要があります。オリジナルのデータは`OriginalRowSet` 要素に保存され、編集されたデータは`RowSet` 要素に保存されます。データ ソースが保存されると、`OriginalRowSet` と`RowSet` の 2 つのツリーの差異が計算され、ページソースは差異を基に更新されます。修正に成功すると、修正されたデータは、`OriginalRowSet` にコピーされ、`OriginalRowSet`が新しく保存されたDB データを含み、修正プロセスが繰り返すことができます。

ページソースの`OriginalRowSet` を作成するには、ページソースのルート ノードを右クリックして、コマンド **[OriginalRowSet の作成]** を切り替えます。

[OriginalRowSet の作成] コマンドは、データベース型 (`$DB`) ルート ノードのために有効化されています。ページソースの元のデータを含むデータ構造を作成/削除する切り替えコマンドです。ユーザーにより修正されたデータはデータ ソースにより

作成されたメインの構造に保存されています。修正されたデータがDB に保存されると、 `OriginalRowSet` 構造は新たにDB に保存されたデータを含むために修正されます。これにより元のDB データはツリーで使用可能となります。

トランザクションをコミットする

データをDB に保存する他の方法は、独立したトランザクションを開始してコミットすることです。ページとコントロールイベント DB トランザクションはアクションとして使用することができます。

DB トランザクションについて

トランザクションが必要な各 DB アクセスには、自動的にトランザクションが作成され、後に自動的に閉じられます。これは一部のセットアップでは理想的です。例えば、同時に自動的に更新したい2 つのDB ページソースが存在する場合、両方のテーブルが保存されると、トランザクションがコミットされます。その他の場合はロールバックされます。このようなシチュエーションに対応するために、接続ベースのトランザクションを作成することができます。

トランザクションの開始 が始まると、同じDB 接続に属するすべてのDB オペレーションはこのトランザクションを使用します。

トランザクションのコミット により、トランザクションの外界で変更が可視化されます。変更はロールバックされることができます。この場合、ページソースに保存した場合でも、ロールバックの後で変更は可視化されません!閉じられていない(コミットされたまたはロールバックされた) トランザクションは、アクションツリーの最後に到達すると自動的にロールバックされます!この効果についてメッセージ ウィンドウに警告が表示されます。

上記の振る舞いはトランザクション アクションについて参照していますが、振る舞いはトランザクションと同じ接続を使用するDB オペレーションにも適用されます。

13.7 DB 実行アクション

[DB 実行] アクション (*下のスクリーンショット参照*) はDB データを修正するためのパワフルなメカニズムです。SQL ステートメントを使用してデータを挿入、削除、更新および保存することができます。これにより、ソリューションの進行中にSQL 言語のパワーをイベントが発生する際に使用することができます。

このセクションでは、DB 実行 を使用してデータの挿入、更新、削除と保存方法について説明されます。 DB データの変更を行うコマンドは、アクションのSQL ステートメントで指定されています。(*上のスクリーンショットを参照してください*) DB 実行アクションの詳細については、ページデザイン> アクション> データベース> DB 実行のセクションを参照してください。 DB 実行のSQL ステートメントは、パラメータの使用を許可することにより、柔軟性を向上します。これらのパラメーターの値は、XPath 式により生成されます。詳細に関しては、[DB 実行] アクションのセクションを参照してください。

DB データが、アクションが定義されているページと同じページで表示されている場合、再ロードアクションを使用して変更されたDB の表示を更新することができます (*上のスクリーンショット参照*)。上のスクリーンショットでは、`$DB1` ツリーがデータベーステーブル`OfficeSales_DB` のルートノードです。`INSERT` ステートメントと共に、`OfficeSales_DB` が変更された後、デザインページの`$DB1` ツリーは再ロードされ、このため、DB への変更がすぐに反映されます。

挿入: DB 実行 とSQL を使用しての行の挿入

SQL の`INSERT` ステートメントはデータベーステーブルに行を挿入するために使用することができます。`INSERT INTO` ステートメントは特定の値を持つ行を挿入するために使用されますが、`INSERT SELECT` ステートメントは、`SELECT` ステートメントの結果をテーブルに挿入する時に使用されます。`SELECT INTO` などの他のSQL ステートメントを使用して、テーブルに行を挿入することができます。

- *単一の行全体または行の一部を挿入する*

  `INSERT INTO` を使用してテーブルに単一行を挿入する。SQL 構文は以下の通りです:

```
INSERT INTO DestinationTable (ID,City)VALUES ('ID-Value','City-Value');
```

上のステートメントは2 つの列 (ID および City) を含む行を DestinationTable テーブルに挿入します。以下の点に注意してください:

- SQL ステートメントで指定された列のみが新しい行(上のサンプル内の ID,City)に挿入されます。
- (すべてのテーブル列を含む) 行全体を挿入するには、SQL ステートメント内のすべてのテーブル列を指定します。
- SQL ステートメント内の列の名前と列の値は、各ポジションで互いに対応する必要があります。この列の順序は、DB テーブル内の列の順序に対応する必要はありません。これは、DB テーブルのレイアウトが後に変更し、SQL ステートメントは正しく変更されたレイアウトを反映するために更新される必要がないからです。
- 各列の名前に対して、列の値は存在する必要があります。それ以外の場合、エラーが発生し、行は挿入されません。
- SQL ステートメントで、列が省略されている場合、DB 内で定義されている列は NULL 値 (または空の) またはデフォルトの値を持つことを許可されていなければなりません。それ以外の場合、エラーが生成され、行は挿入されません。
- 複数の行を挿入する場合は、複数の INSERT INTO ステートメントを指定する必要があります。

SELECT ステートメントの結果の挿入

INSERT SELECT を使用して、テーブルに SELECT ステートメントの結果を挿入します。通常、INSERT SELECT はテーブルから他のテーブルに行のセットをコピーするために使用されます。SQL 構文は以下の通りです:

```
INSERT SELECT Offices (ID,City,Country) SELECT ('ID', 'Stadt', 'Land') FROM Offices_DE ;
```

上のステートメントは Offices_DE テーブルの全ての行を Offices テーブルに挿入します。以下の点に注意してください:

- SQL ステートメント内で指定された列のみが新しい行(上のサンプル内 ID,City,Country) に挿入されます。
- SELECT ステートメントにより返された列は、目的のテーブルの対応する列に挿入されます。列の対応は、位置により決定されます。上のサンプルでは、たとえば、Stadt 列は SELECT ステートメントのポジション2 にあり、目的テーブルのポジション2 にある City 列に対応します。目的テーブルにある2 つの列の名前は一致する必要がありません。対応はポジションにより固定されます。
- WHERE 句で、SELECT ステートメントを使用して挿入されたデータをフィルターすることができます。

更新:DB 実行とSQL を使用しての行の更新

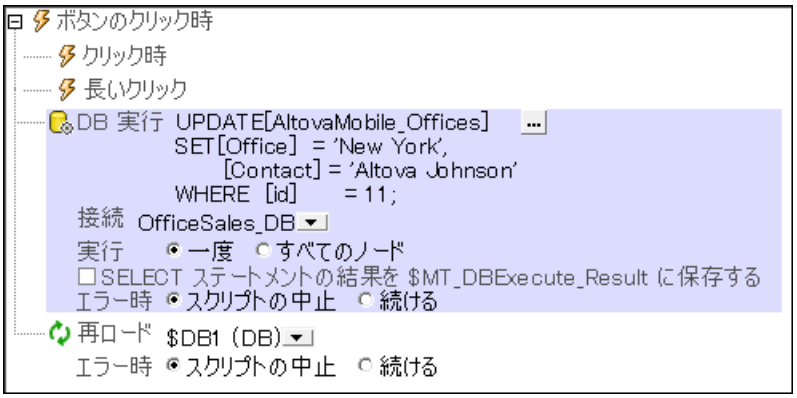
SQL の UPDATE ステートメントはデータベーステーブル内の行を更新するために使用することができます。UPDATE ステートメントには以下の3 つのパートがあります:

- 更新される DB テーブルの名前
- 更新される列の名前とその値
- 更新される行をフィルターする WHERE 句

SQL UPDATE ステートメントのサンプルは以下の通りです:

```
UPDATE [AltovaMobile_Offices]
SET    [Office] = 'New York',
       [Contact] = 'Altova Johnson'
WHERE  [id]      = 11;
```

このステートメントは、id=11 を持つ行を更新します。例えば、Office='USA' からOffice='New York' へと Contact=NULL からContact='Altova Johnson' へ更新するとします。下のスクリーンショットは、[DB 実行] アクションのSQL ステートメント設定内のこのUPDATE ステートメントのサンプルを表示しています。



以下の点に注意してください:

- 更新される列はname=value の組み合わせにより与えられます。各、name=value 組み合わせは次の組み合わせとカンマで区切られています。最後のname=value 組み合わせの後にはカンマはありません。
- 更新される列は単一のSET 句により指定されています。
- 列の値は、NULL 値がその列に対して使用することを許可されていることを前提にして、A値をNULL に設定することで削除することができます。例えば、: SET [Contact] = NULL。

再ロードアクションは、変更が加えられたあと変更されたDBをすぐに再ロードします。再ロードアクションを使用しないと、ページに変更は表示されません。

削除:DB 実行 とSQL を持つ行の削除

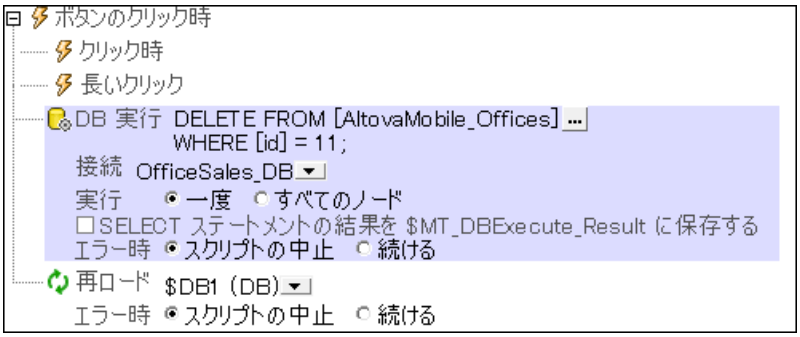
SQL のDELETE ステートメントは以下を削除するために使用されます:

- テーブルの特定の行 (WHERE 句を指定して削除する行を選択します。)
- テーブルの全ての行 (WHERE 句を省略することによりテーブルの全ての行を選択します。)

以下はSQL DELETE ステートメントのサンプルです:

```
DELETE FROM [AltovaMobile_Offices]
WHERE [id] = 11;
```

上のSQL DELETE ステートメントはid=11 を持つ行を削除します。WHERE 句が省略される場合、AltovaMobile_Offices テーブルのすべての行は削除されます。



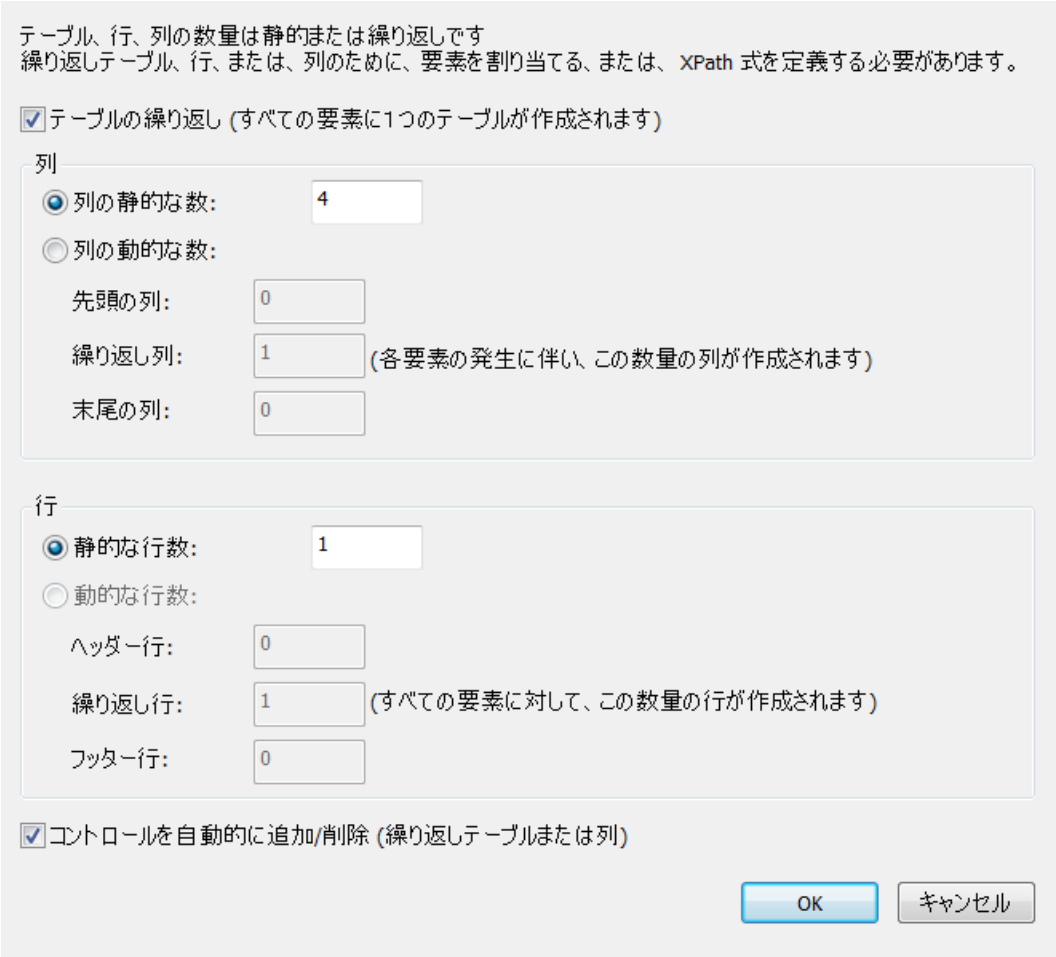
DB 実行アクション内のSQL DELETE ステートメント

再ロードアクションにより変更されたDB を変更が実行された後すぐに再ロードすることができます。再ロードアクションを使用しない場合、変更はページに表示されません。

13.8 DB データの表示

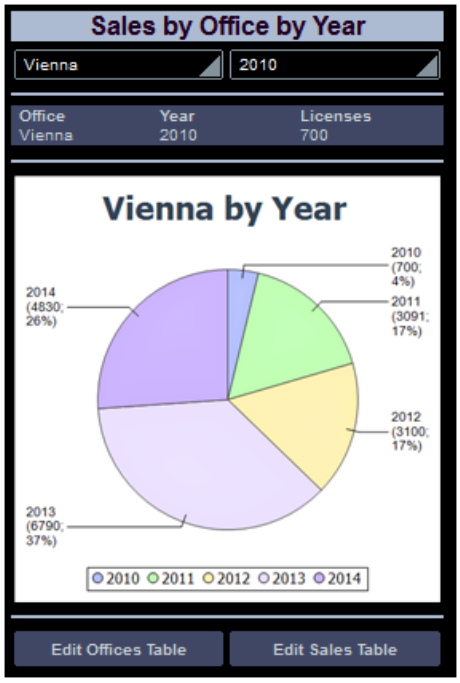
DB データをテーブルと他のコントローで表示する

コントロールからデータソース ノードへのページ ソース リンクを作成して、コントロールにDB データを表示することができます。データを表示するには、通常繰り返し行のテーブルが最も確実な方法です。デザインにテーブルコントロールをドラッグして、繰り返しテーブルとして新しいテーブルを作成します (*下のスクリーンショット参照*)。テーブルのセルにコントロールをドラッグして、DB 行の行ノードにページソースリンクを作成します。どの様に作動するかのデモンストレーションは、データベースとチャートのチュートリアルを参照してください。を参照してください。



DB データをチャートとして表示する

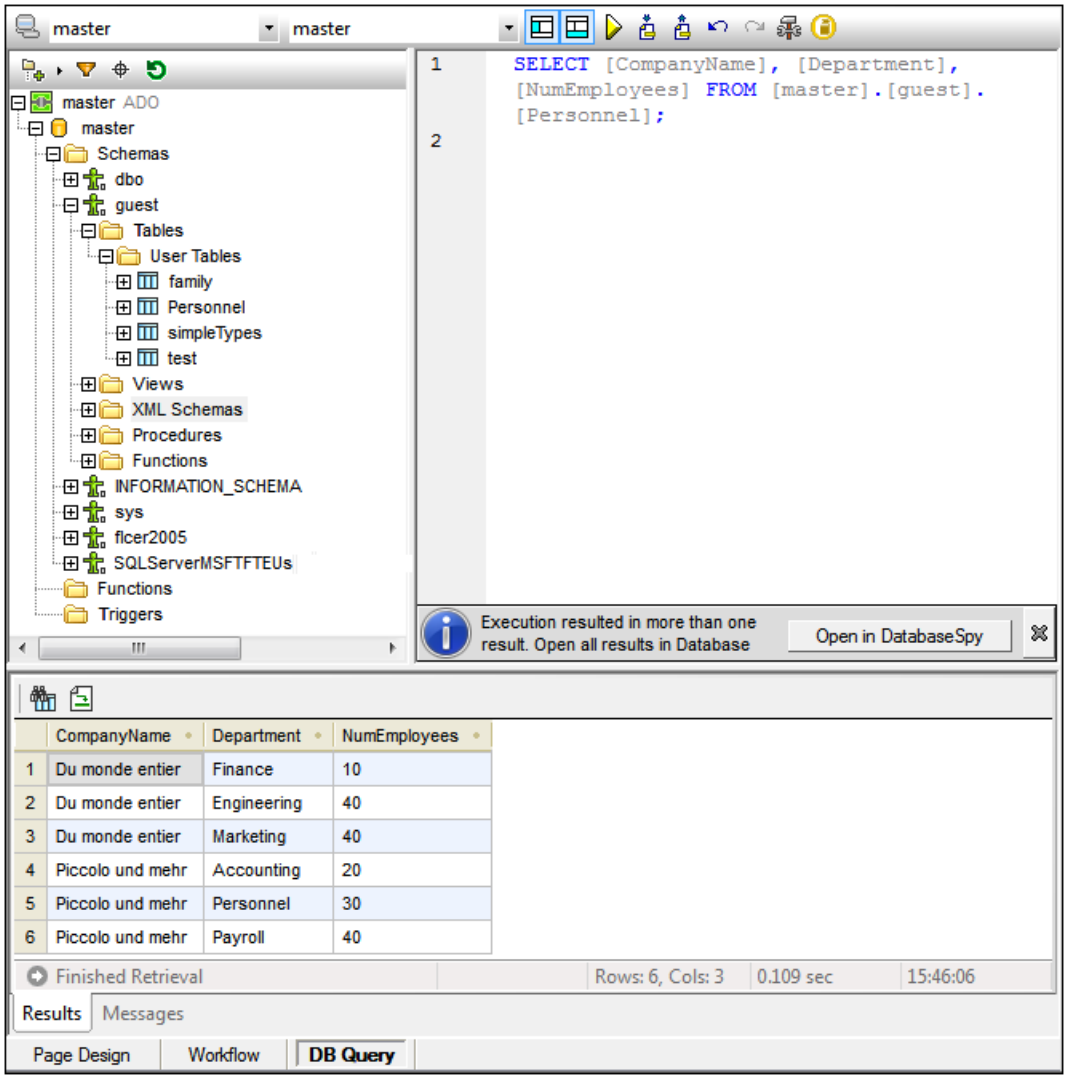
DB データを直接表示できる他、DB データをベースにしてチャートを作成することができます。



この点に関してはデータベースとチャートのチュートリアルを参照してください。

13.9 データベース クエリ

[DB クエリビュー] (*下のスクリーンショット*) により、MobileTogether Designer GUI 内部から主要なデータベースに直接クエリすることができます。データベースは、アクティブなドキュメント内または外部データベースで参照されるデータソースであることができます。各 DB クエリポイントは、現在アクティブなデザインと関連付けられることに注意してください。デザインの DB クエリポイントは、一度に複数のデータベース接続することができます。MobileTogether Designer 内で一度に複数のデザインを開くこともできます。DB クエリビュー内で定義されたクエリとアクションは、他の MobileTogether Designer タブとは独立していて、デザインファイルの一部としては保存されません。



データベース クエリのメカニズム

データベースクエリのメカニズムは以下の通りです。(このセクションのサブセクションに詳細付きで説明されています。)

1. [データベースクエリ] データソースへ接続] ウィンドウを使用して、データベースへの接続が確立されます。
2. ビューの必要条件に合わせて構成することのできる、ブラウザーペインに接続されたデータベースまたはその一部が表示されます。
3. データベースがクエリされるために、適切な構文で書かれたクエリはクエリペインに入力され、クエリが実行されます。

4. クエリの結果 は多種のフィルターで表示することができます。

サポートされるデータベース

次のデータベースはサポートされています。各データベースの使用できるルート オブジェクトもリストされています。Altova は他のODBC/ADO データベースへのサポートの向上のための努力をしていますが、以下にリストされたデータベースのみで接続とデータ処理のテストに成功しています。使用中のAltova アプリケーションが、64 ビット バージョンの場合、接続されているデータベース特定の64 ビットデータベース ドライバーにアクセスしてください。Altova アプリケーションはIBM iSeries データベースのロジカル ファイルをサポートし、ロジカル ファイルを表示します。

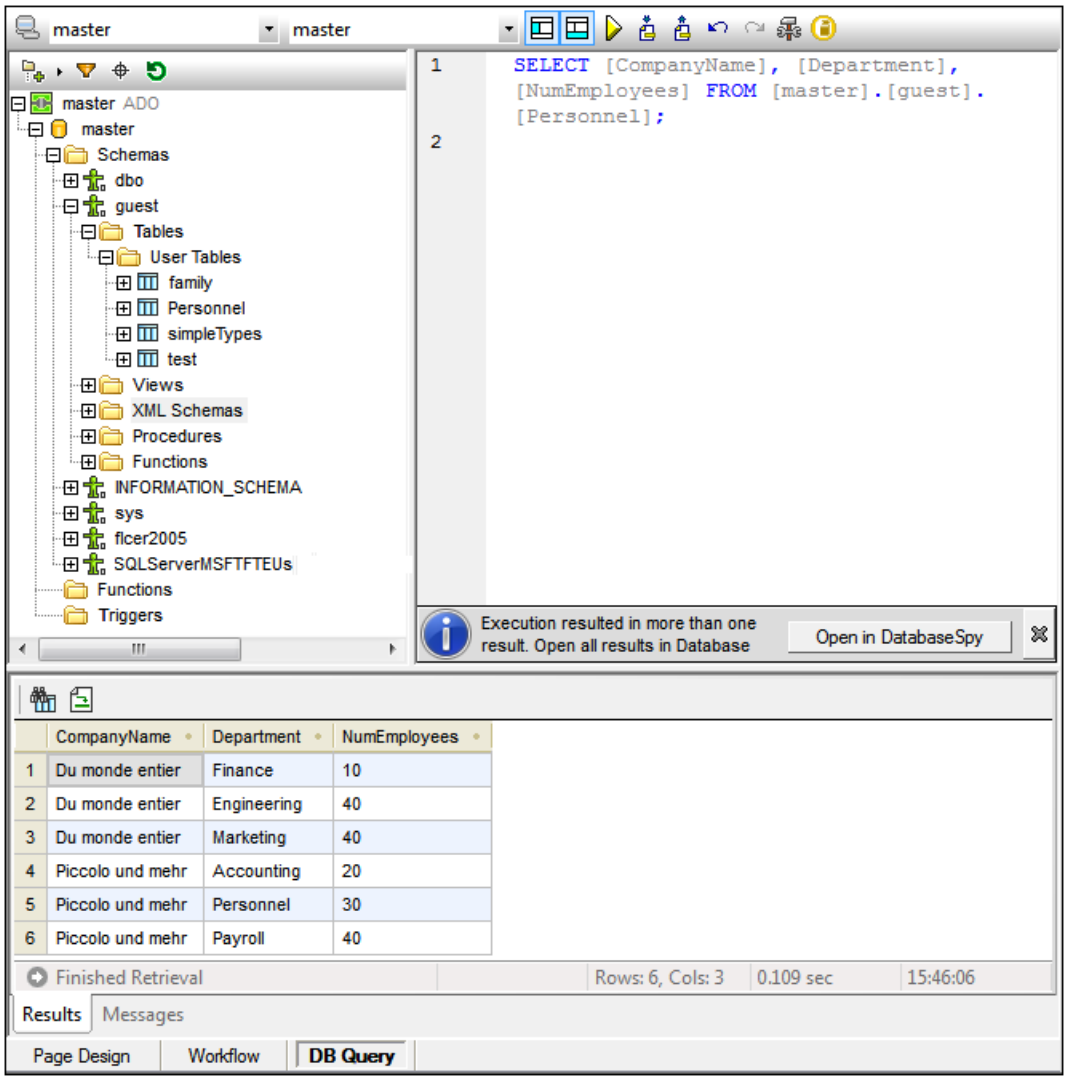
データベース	ルートオブジェクト	メモ
Firebird 2.5.4	データベース	
IBM DB2 8.x, 9.1, 9.5, 9.7, 10.1, 10.5	スキーマ	
IBM DB2 for i 6.1, 7.1	スキーマ	ロジカルファイルはサポートされ、ビューとして表示されています。
IBM Informix 11.70	データベース	
MariaDB 10	データベース	
Microsoft Access 2003, 2007, 2010, 2013	データベース	
Microsoft Azure SQL Database	データベース	SQL Server 2016 codebase
Microsoft SQL Server 2005, 2008, 2012, 2014, 2016	データベース	
MySQL 5.0, 5.1, 5.5, 5.6, 5.7	データベース	
Oracle 9i, 10g, 11g, 12c	スキーマ	
PostgreSQL 8.0, 8.1, 8.2, 8.3, 9.0.10, 9.1.6, 9.2.1, 9.4, 9.5, 9.6	データベース	PostgreSQL 接続は、ODBC またはJDBC などのインターフェイス (ドライバ) によりネイティブとドライバーベースの接続でサポートされています。ネイティブな接続はドライバを必要としません。
Progress OpenEdge 11.6	データベース	
SQLite 3.x	データベース	SQLite 接続は、ネイティブとしてサポートされます。SQLite データベースファイルへの直接の接続です。別途のドライバーは必要ありません。
Sybase ASE 15, 16	データベース	
Teradata 16	データベース	

GUI 概要とツールバー

DB クエリビュー (*下のスクリーンショット*) は以下のパーツに分割されています:

- ブラウザー ウィンドウ 左側にあり、接続情報とデータベーステーブルを表示します。
- SQL エディター ウィンドウ (クエリウィンドウ) ブラウザーウィンドウの右側にあり、SQL クエリを含みます。
- 結果 メッセージウィンドウの結果タブ クエリの結果を表形式で表示します。
- 結果 メッセージウィンドウのメッセージタブ 警告またはエラーメッセージを表示します。

ウィンドウの詳細はこのセクションのサブセクションに説明されています。



DB クエリビュー ツールバー

DB クエリビュー ツールバー (*下のスクリーンショット*) は、ビューのトップにあり、ビューに関連した重要なコマンドのアイコンを表示しています。

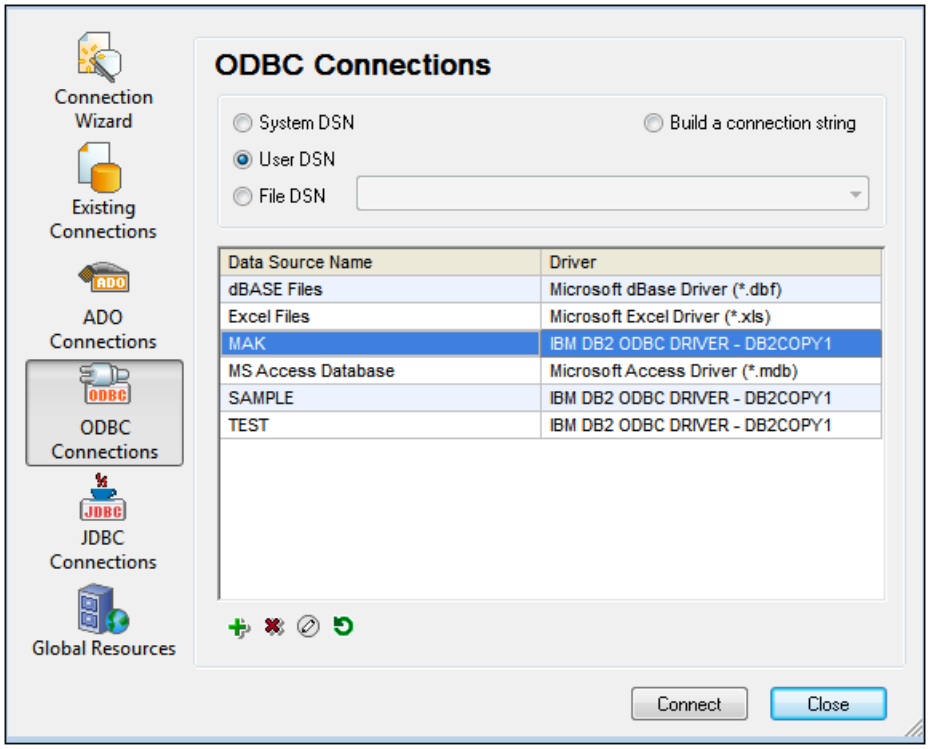
アイコン	コマンド	機能
	クイック接続	データベース接続ウィザードを開始します。
	ブラウザーの切り替え	ブラウザーウィンドウのオンとオフを切り替えます。
	結果の切り替え	結果 /メッセージウィンドウのオンとオフを切り替えます。
	クエリの実行	現在選択されているSQL ステートメントを実行します。スクリプトが複数のステートメントを含んでいて、選択されていない場合、すべてのステートメントが実行されます。
	SQL ファイルのインポート	SQL ファイルをSQL エディターで開きます。
	SQL ファイルのエクスポート	SQL クエリをSQL ファイルに保存します。
	元に戻す	SQL エディター内で無制限に編集を元に戻す。
	やり直し	SQL エディター内で無制限に編集をやり直す。
	オプション	SQL エディターのダイアログのオプションを開く。
	DatabaseSpy でクエリを開く	Altova DatabaseSpy アプリケーション内でSQL スクリプトを開く。

データソースへの接続

データベースをクエリするには、最初に必要とされるデータベースに接続し、必要とされるデータソースとルートオブジェクトを複数の既存の接続から選択します。

データベースに接続する

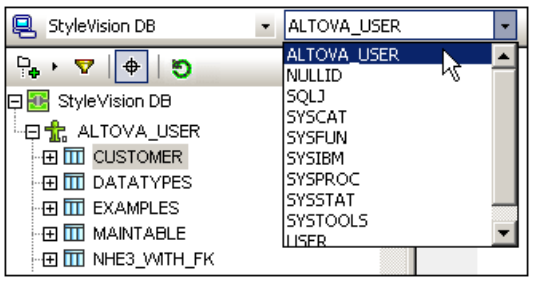
DB クエリビューに切り替えると、アクティブデザインがデータソースとして使用するすべてのデータベースがブラウザーペインに表示されます。他のデータベースをクエリするには、クエリビューツールバー内の[クイック接続] をクリックしてください。



サポートされるデータベースのリストは、データベースクエリ| サポートされるデータベースを参照してください。

必要なデータソースの選択

既存のすべての接続およびルートオブジェクトは、データベース クエリウィンドウのツールバー内の 2 つのコンボボックスにそれぞれリストされています (下のスクリーンショット)。左側のコンボボックスから必要なデータソースを選択した後、右側のコンボボックスから必要なルートオブジェクトを選択することができます。



上のスクリーンショットでは、`StyleVision DB` という名前のデータベースが選択されています。このデータベースの使用できるルートオブジェクトに関しては、ルートオブジェクト `ALTOVA_USER` が選択されています。データベースとルートオブジェクトはブラウザーペインに表示されます。

ブラウザー ペイン

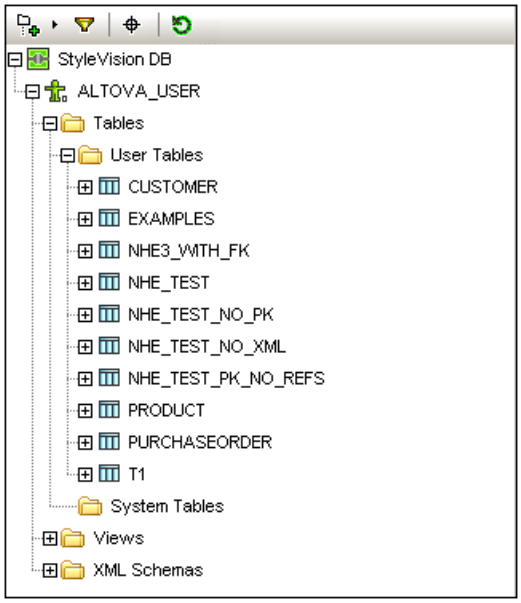
ブラウザーペインは、選択されたデータベースのオブジェクトの概要を提供します。この概要は、列が主キーまたは外部キーであるかなどのデータベースの制約に関する情報を含みます。IBM DB2 バージョン9 または以降のデータベースでは、ブラウザーは登録されたXML スキーマを個別のフォルダーで表示します (*下のスクリーンショット参照*)。

このセクションでは以下について説明されています:

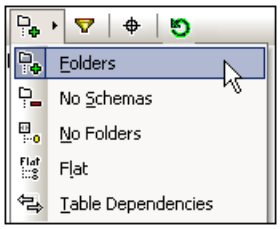
- ブラウザーペインで使用することのできる レイアウト。
- データベースオブジェクトのフィルター方法。
- データベースオブジェクトの検索方法。
- アクティブなデータソースのルートオブジェクトの更新方法。

ブラウザーペインのレイアウト

デフォルトのフォルダーのレイアウトは、データベースオブジェクトを階層表示します。選択されたオブジェクトにより、アイテムを右クリックすると、異なるコンテキストメニューのオプションを使用することができます。



ブラウザーのレイアウトを選択するには、ブラウザーペインのツールバー内のレイアウトアイコンをクリックし、ドロップダウンリストからレイアウトを選択します (*下のスクリーンショット*)。選択されたレイアウトによりアイコンが変更されることに注意してください。



使用可能なレイアウトは以下の通りです:

- **フォルダー レイアウト**: データベースオブジェクトを、階層ツリー内のオブジェクトの型をベースに、フォルダーに整理します。デフォルトの設定です。

- **スキーマをレイアウトしない**: フォルダーレイアウトに類似していますが、データベース スキーマフォルダーは存在しません。ですから、テーブルは、データベーススキーマにより整理されません。
- **フォルダーをレイアウトしない**: フォルダーを使用せずにデータベースオブジェクトを階層順に表示します。
- **フラット レイアウト**: データベースオブジェクトを最初の階層レベルで分けます。例えば、列を対応するテーブルに包含する代わりに、すべての列は個別の列フォルダーに表示されます。
- **テーブル依存関係のレイアウト**: テーブルを他のテーブルとの関係により整理します。外部キーを持つテーブルおよび他のテーブルとの関係を持たない外部キーを持つテーブルにより参照されるテーブルのカテゴリーがあります。

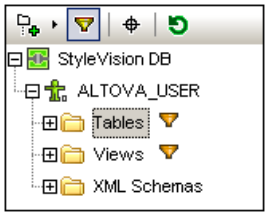
ユーザーとシステムテーブルにテーブルを並べ替えるには、フォルダー、フォルダーをレイアウトしない、フラットレイアウトに切り替え、テーブルふぉふだーを右クリックして、**[ユーザーとシステム テーブルの並べ替え]** を選択します。テーブルは、ユーザーとシステムテーブルフォルダー内でアルファベット順に並べ替えられます。

データベースオブジェクトのフィルター

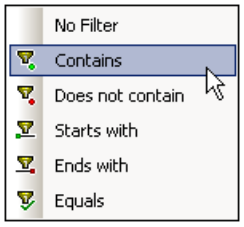
ブラウザーペイン内で（[フォルダー をレイアウトしない] および [テーブル依存関係のレイアウト] 以外の）全てのレイアウト、スキーマ、テーブル、およびビューは名前および名前の一部でフィルターすることができます。オブジェクトは文字を入力中にフィルターされ、フィルタリングはデフォルトで大文字と小文字を区別します。

ブラウザー内でオブジェクトをフィルターするには、以下を行います:

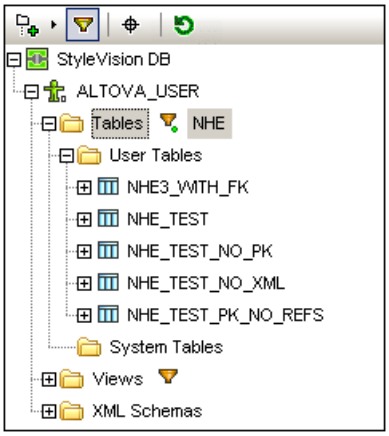
1. ブラウザーペインのツールバー内の [フォルダーコンテンツのフィルター] アイコンをクリックします。現在選択されているレイアウトのテーブルとビューフォルダーの横にフィルターアイコンが表示されます（*下のスクリーンショット*）。



2. フィルターするフォルダーの横のフィルターアイコンをクリックして、ポップアップメニューからフィルターのオプション、たとえば、`Contains` を選択します。



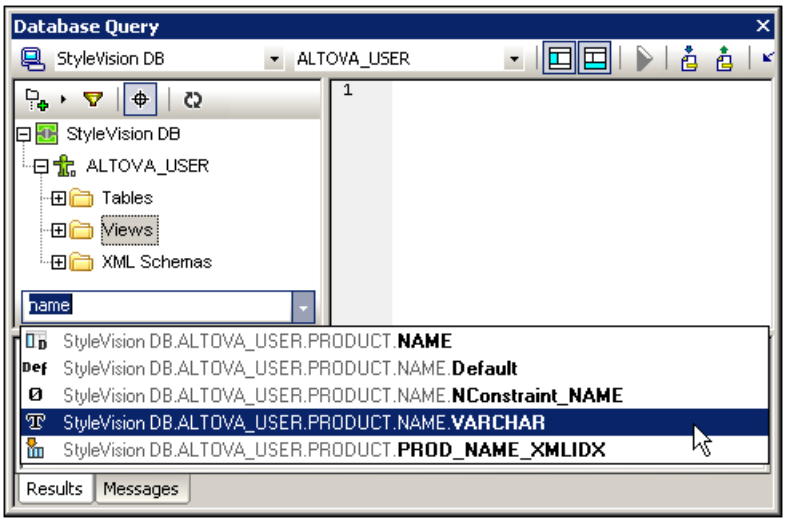
3. 表示されるエントリフィールドに、フィルターする文字列を入力します（下のスクリーンショットでは、`Tables` フォルダーのフィルターする文字列は `NHE` です。）フィルターは入力と同時に表示されます。



データベースオブジェクトの検索

名前検索で特定のデータベースアイテムを検索する場合、ブラウザーペインのオブジェクト ロケーターを以下の通り使用することができます:

1. ブラウザーペインのツールバー内で、[オブジェクト ロケータ] アイコンをクリックします。ブラウザーペインの下にドロップダウンのリストが表示されます。
2. このリストの入力フィールドに検索する文字列を入力します。例えば、name と入力します(*下のスクリーンショット*)。ドロップダウンの矢印は検索する文字列を含むすべてのオブジェクトを表示します。



3. リスト内のオブジェクトをクリックしてブラウザーペイン内に表示します。

ルートオブジェクトの更新

アクティブなデータソースのルートオブジェクトはブラウザーペインのツールバーの**[更新]** ボタンをクリックすることで更新できます。

クエリ ペイン: 説明

クエリペインは、選択されたデータベースへクエリを入力するためのインテリジェントな SQL エディターです。クエリを入力後、データベースクエリウィンドウの実行コマンドをクリックすると、クエリを実行して、結果 メッセージ ペイン内に結果と実行メッセージが表示されます。クエリとの作業方法は次のクエリペイン: クエリとの作業方法セクションで説明されています。このセクションでは、クエリペインの主な機能について説明されています:

- データベースクエリツールバー内の SQL エディター アイコン
- SQL エディター オプション
- SQL ステートメントの自動補完
- SQL スクリプト内の地域の定義
- SQL スクリプト内へのコメントの挿入
- ブックマークの使用

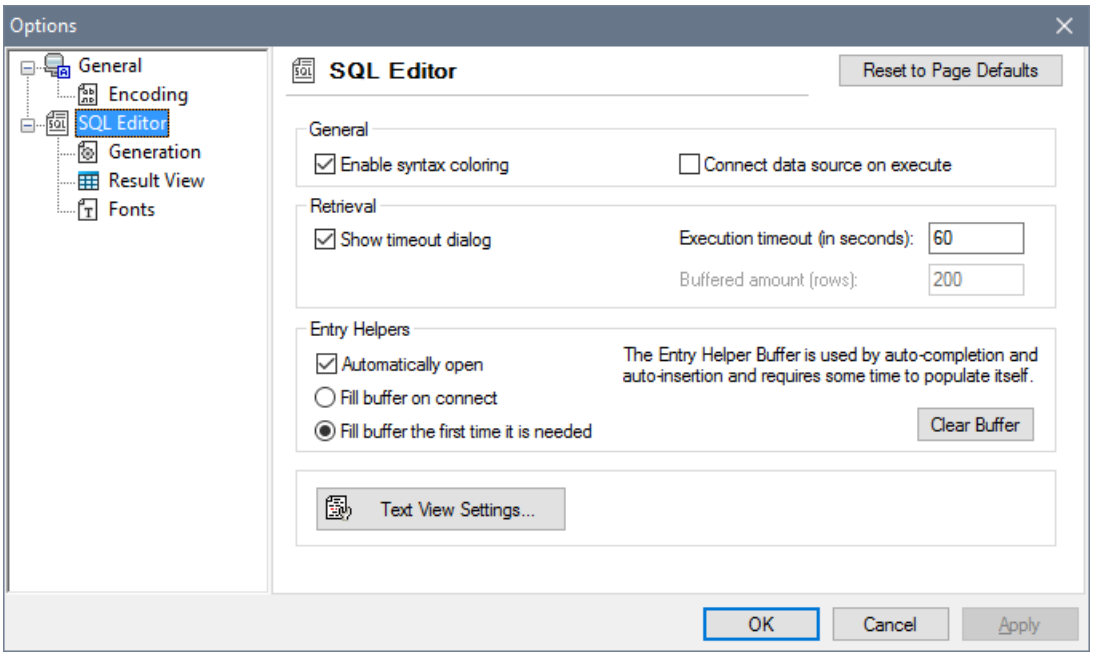
データベースクエリツールバー内の SQL エディター アイコン

データベースクエリウィンドウのツールバーの以下のアイコンが、SQL エディターとの作業時に使用されます:

アイコン	コマンド	機能
	クエリの実行	現在選択されている SQL ステートメントを実行します。スクリプトが複数のステートメントを含んでいて、選択されていない場合、すべてのステートメントが実行されます。
	SQL ファイルのインポート	SQL ファイルを SQL エディターで開きます。
	SQL ファイルのエクスポート	SQL クエリを SQL ファイルに保存します。
	元に戻す	SQL エディター内で無制限に編集を元に戻す。
	やり直し	SQL エディター内で無制限に編集をやり直す。
	オプション	SQL エディターのダイアログのオプションを開く。
	DatabaseSpy でクエリを開く	Altova DatabaseSpy アプリケーション内で SQL スクリプトを開く。

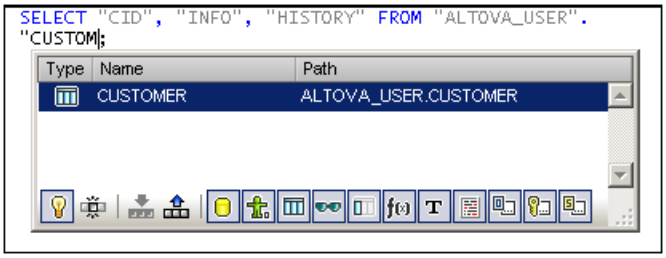
SQL エディター オプション

データベースクエリツールバー内の **[オプション]** アイコンをクリックすると、オプションダイアログがポップアップします (下のスクリーンショット)。左側のペイン内で設定のページ、またそのページでのオプションを選択することができます。 **[ページデフォルトにリセット]** ボタンをクリックして、そのページのオプションを元の設定にリセットすることができます。



キーの設定は以下の通りです:

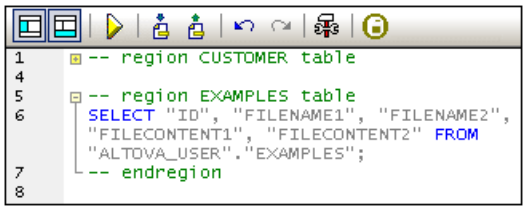
- **全般 | エンコード**: 新規 SQL ファイル、エンコードが検出されない既存の SQL ファイル、バイト オーダー マーク (BOM) の設定、のためのオプション。(既存の SQL ファイルのエンコードが検出される場合、ファイルは開かれエンコードを変更することなく保存されます。)
- **SQL エディター**: 構文の色分けとデータソースへの接続の実行のオン/オフの切り替えのオプション。クエリの実行に対してタイムアウトを設定することができ、指定された時間が経過すると、タイムアウトを変更するダイアログを表示することもできます。入力ヘルパーは、自動補完機能の一部として表示されます。SQL ステートメントを入力すると、エディターはコンテキストを考慮した自動補完の提案を表示します。自動表示のスイッチがオフに切り替えられている場合、 **Ctrl+スペースキーSpacebar** を押して、SQL エディターに自動補完の提案をリクエストすることができます。入力ヘルパー情報のバッファーは、データソース、または、最初の必要時に入力されます。テキストビュー設定ボタンは、SQL エディター内のタブテキストのサイズとインデントを指定することのできるダイアログを開きます。
- **SQL エディター | SQL 生成**: アプリケーションは、ブラウザーペインからオブジェクトをドラッグしてクエリペインにドロップすると、SQL ステートメントを生成します。SQL ステートメント生成のオプションは、SQL 生成タブで設定することができます。データベースリストボックスを使用して、データベースの種類を選択して、ステートメント生成オプションを個別に作業中の異なるデータベースの種類に対して設定します。すべてのデータベースに適用チェックボックスをアクティブ化することにより、現在選択されているすべてのデータベースに対してオプションを設定することができます。オプションは、ステートメントにセミコロンを追加、識別子をエスケープ文字で囲むなどが含まれます。
- **SQL エディター | 自動補完**: 自動補完機能は、タイプ中に、関連するエントリの候補を多種の SQL 構文カテゴリから提案します。以下のデータベースで使用することが可能です: MS SQL Server 2000、2005、および 2008、MS Access 2003 および 2007、および IBM DB2 v.9。自動補完がオンに切り替えられていると、自動補完の候補を提案する自動補完ウィンドウが表示されます(*下のスクリーンショット*)。自動補完ポップアップが、遅延の後に自動的にトリガーされるか、自動補完のトリガーペインで設定または、ポップアップが手動でトリガーされるか定義することができます。選択された補完を挿入するために使用されるキーを選択することができます。SQL エディターは言語統計をベースにして、自動補完エントリを知的に提案することができます。この機能がアクティブ化されていると、提案されるエントリのリストの上に頻繁に使用されるアイテムが表示されます。自動補完ウィンドウ内ではボタンはウィンドウの下に表示されます (*下のスクリーンショット*) 。状況依存の提案ボタンにより、コンテキストに関連したエントリのみを表示するか、またはそのスペルを持つすべての考えられるコンテキストを表示するかを設定することができます。[シングルモード] ボタンにより、カテゴリをクリックして、そのカテゴリのみを選択することができます。[全てのカテゴリを選択] ボタンによりすべてのカテゴリを選択することができます。カテゴリの選択を解除するには、ボタンをクリックします。[全てのカテゴリをクリア] ボタンによりすべてのカテゴリの選択を解除します。他のボタンは多種のカテゴリのボタンです。



- ***SQL エディター| 結果ビュー:*** 結果タブのオプションの構成。
- ***SQL エディター| フォント:*** テキストエディターと結果ビュー内のテキストのフォントスタイルの設定のオプション。

SQL スクリプト内のリージョンの定義

リージョンは、マークおよび宣言されてユニットになるSQL スクリプト内のセクションです。リージョンは、スクリプトの一部を非表示または表示するために縮小または展開することができます。また、リージョンを他のリージョンとネストすることもできます。リージョンは、リージョンの前後で、 **`--region`** および **`--endregion`** コメントにより区切られることができます。 **`--region`** の後に区切りを入力することにより、リージョンにオプションで名前を与えることができます (*下のスクリーンショット参照*)。



リージョンを挿入するには、リージョンとするステートメントを選択して、右クリックした後に**[リージョンの挿入]** を選択します。展開可能な縮小可能なリージョンが作成されます。名前を与えることもできます。上のスクリーンショットでは、行の付番にも注意してください。リージョンを削除するには、 **`--region`** および **`--endregion`** の2 つの区切りを削除してください。

SQL スクリプト内へのコメントの挿入

SQL スクリプト内のテキストをコメントアウトすることができます。スクリプトの一部の箇所はスクリプトが実行されるとき実行されません。

- ブロックをコメントアウトするには、ブロックをマークした後、右クリックして**[ブロックコメントの挿入 /削除]** を選択します。ブロックコメントを削除するには、ブロックをマークした後、右クリックして、**[ブロックコメントの挿入 /削除]** を選択します。
- 行または行の一部をコメントアウトするには、行コメントが開始される箇所にカーソルをポイントした後、右クリックして、**[行コメントの挿入 /削除]** を選択します。行コメントを削除するには、コメントをマークした後、右クリックし**[行コメントの挿入 /削除]** を選択します。

ブックマークの使用

特定の行にブックマークを挿入することができ、ドキュメント内をブックマークを使用してナビゲートすることができます。ブックマー

クを挿入するには、ブックマークされる行にカーソルをポイントした後、右クリックして、**[ブックマークの挿入 /削除]** を選択します。次のまたは前のブックマークに移動するには、右クリックした後、**[次のブックマークへ移動]** または **[前のブックマークへ移動]** を選択します。ブックマークを削除するには、カーソルを削除するブックマークのある行にポイントして、右クリックした後 **[ブックマークの挿入 /削除]** を選択します。すべてのブックマークを削除するには、右クリックした後、**[すべてのブックマークを削除]** を選択します。

クエリ ペイン: 作業方法

データベースに接続後、SQL エディターにSQL スクリプトを入力し実行することができます。このセクションでは以下が説明されています:

- SQL エディターへのSQL スクリプトの入力方法
- データベース クエリウィンドウ内でのスクリプトの実行

このセクションでは以下のアイコンが参照されています:

	クエリの実行	現在選択されているSQL ステートメントを実行します。スクリプトが複数のステートメントを含み、いずれも選択されていない場合、すべてが実行されます。
	SQL ファイルのインポート	SQL エディターでSQL ファイルを開きます。

SQL ステートメントの作成とSQL エディター内のスクリプト

以下のGUI メソッドを使用してSQL ステートメントまたはスクリプトを作成することができます:

- *ドラッグ アンド ドロップ:* ブラウザー ペインからオブジェクトをドラッグしてSQL エディターにドロップします。SQL ステートメントが生成され、そのオブジェクトのためにクエリが生成されます。
- *コンテキスト メニュー:* ブラウザーペイン内のオブジェクトを右クリックして、**[SQL エディターで表示 | 選択]** を選択します。
- *手動エントリ:* SQL エディターにSQL ステートメントを直接 タイプします。自動補完機能が編集を手助けします。
- *SQL スクリプトのインポート:* データベースクエリウィンドウのツールバー内の**[SQL ファイルのインポート]** アイコンをクリックします。

SQL ステートメントの実行

SQL エディター内のSQL スクリプトが1 つ以上存在する場合、実行するステートメントを選択して、データベース クエリウィンドウのツールバー内の**[実行]** アイコンをクリックします。スSQL クリプト内でステートメントが選択されていない場合、スクリプト内の全てのステートメントが実行されます。データベースデータが取得され結果タブでグリッドとして表示されます。実行に関するメッセージはメッセージタブに表示されます。

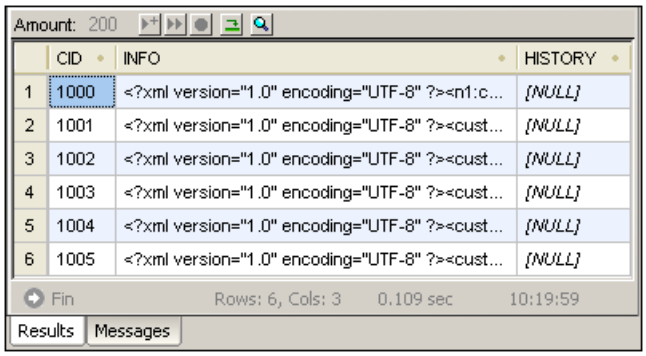
結果とメッセージ ペイン

結果 /メッセージは2 つのタブがあります:

- 結果タブ は、クエリにより取得されたデータを表示します。
- メッセージタブ クエリ実行に関するメッセージを表示します。

結果タブ

クエリにより取得されたデータは、結果タブにグリッドのフォームで表示されます (*下のスクリーンショット*)。



次の操作は、コンテキスト結果タブ内の適切な場所を右クリックするとポップアップする、コンテキストメニューを使用して、結果タブで実行することができます。

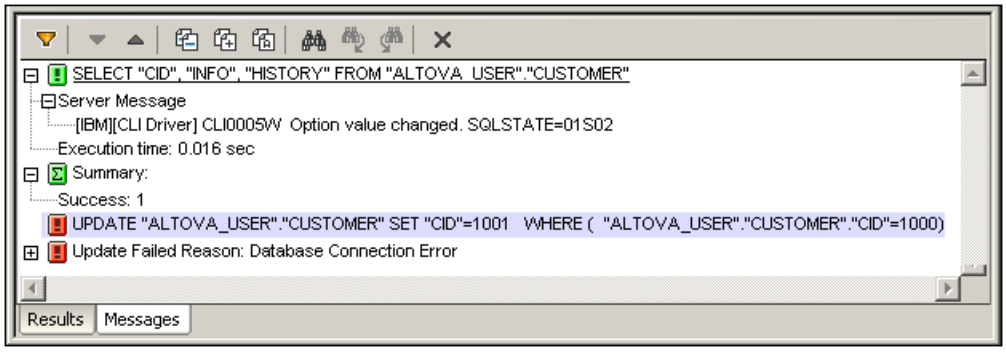
- *列の並べ替え* 記録が並べ替えられる列を右クリックし、**[並べ替え | 昇順 /降順 /既定値に復元]** を選択します。
- *クリップボードにコピー:* この手順には2 つのステップがあります: (i) データ範囲の選択、(ii) 選択範囲のコピー。データは複数の方法で選択することができます。: (i) 列のヘッダーまたは行番号をクリックしてれつまたは行を選択、(ii) 個別のセルを選択 (**Shift** キーおよび /または **Ctrl** キーを使用して複数のセルを選択します); (iii) セルを右クリックして、**[選択 | 行 /列 /すべて]** を選択します。選択が決定されると、右クリックして **[選択されたセルのコピー]** を選択します。これにより、他のアプリケーションに貼り付けすることのできるクリップボードに選択範囲をコピーすることができますセルと共にヘッダーをコピーするには、**[ヘッダーと選択対象のセルのコピー]** コマンドを選択します。

結果タブには以下のツールバーアイコンがあります:

	ステートメントへ移動	現在の結果を作成したSQL エディター内のステートメントをハイライトします。
	検索	結果ペイン内のテキストを検索します。XML コンテンツも検索されます。

メッセージタブ

メッセージタブは、以前実行されたSQL ステートメント、エラーの報告および警告メッセージ関する情報を提供します。



メッセージタブのツールバーは、ビューをカスタム化、移動、またはメッセージをクリップボードにコピーするアイコンを含んでいます。**[フィルター]** アイコンにより、特定の種類のメッセージの表示の切り替えをオンまたはオフします。**[次へ]** および **[前へ]** アイコンにより、リストをそれぞれ上下に一覧することができます。メッセージはクリップボードに子コンポーネント共にまたは無しでコピーし、ドキュメントに貼り付けることができます。**[検索]** 機能により、検索用語を指定し、この用語をリスト内上下に検索することができます。 **[クリア]** アイコンにより、レポートペインの内容をクリアすることができます。

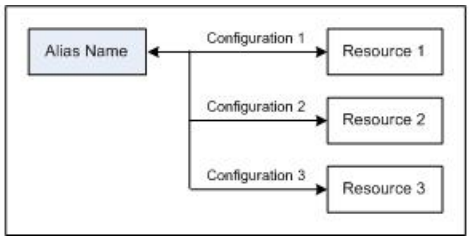
メモ　ツールバー アイコン コマンドは、コンテキスト メニュー コマンドとしても使用することができます。

チャプター 14

Altova グローバルリソース

14 Altova グローバルリソース

Altova グローバルリソースはファイル、フォルダー、そしてデータベースリソースのエイリアス集合のことです。各エイリアスは複数の構成を持ち、各構成を1つのリソースにマッピングすることができます (下のスクリーンショット参照)。そのため、グローバルリソースを入力として使用することで、グローバルリソースを構成とともに切り替えることができます。この操作は、アクティブな構成を選択することのできるGUI 上で簡単に行うことができます。例えば、グローバルリソースをページデータソースのデフォルトのファイルとして指定することができ、リソースをGUI 内のアクティブ構成をスイッチすることで切り替えることができます。



Altova グローバルリソースの使用には、2 つのステップがあります:

- グローバルリソースの定義: XML ファイルによりリソースが定義されます。これらのリソースは複数のAltova 製品間で共有することができます。
- グローバルリソースの使用: MobileTogether Designer 内部ではファイルパスの代わりにグローバルリソースによりファイルを指定することができます。グローバルリソースを使用すると、MobileTogether Designer にてアクティブな構成を切り替えることで、使用するリソースを簡単に変更することができるようになります。

その他の Altova 製品におけるグローバルリソース

現在のところ、以下のAltova 製品でグローバルリソースを使用することができます: XMLSpy、StyleVision、MapForce、Authentic Desktop、MobileTogether Designer およびDatabaseSpy。

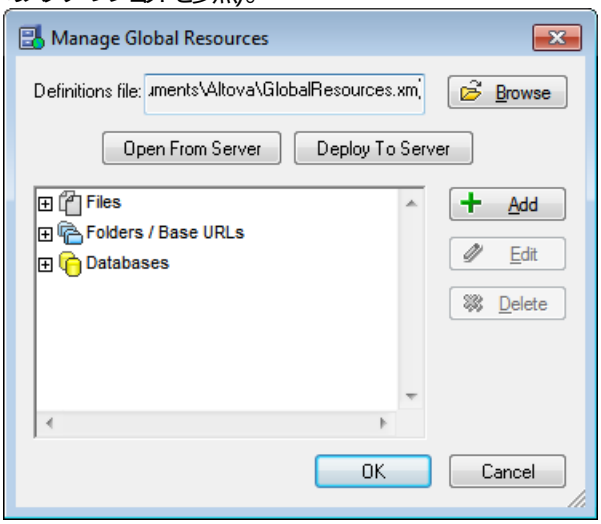
14.1 グローバル リソースの定義

Altova グローバル リソースは、以下の方法によりアクセスすることのできるグローバル リソースの管理ダイアログで定義されます:

- メニュー コマンドの **「ツール| グローバル リソース」** を選択する。
- グローバル リソース ツールバーにある **「グローバル リソースの管理」** アイコンをクリックする(下のスクリーンショット)。

グローバル リソース定義ファイル

グローバル リソースに関する情報はグローバル リソース定義ファイルと呼ばれるXML ファイルに収められています。このファイルは、グローバル リソースの管理ダイアログにて初めてグローバル リソースが定義ならびに保存された際に作成されます (下のスクリーンショットを参照)。



グローバル リソースの管理ダイアログを初めて開くと、グローバル リソース定義ファイルのデフォルトの場所と名前が、定義ファイル テキスト ボックスにて指定されます (上のスクリーンショットを参照):

```
C:\Users\<username>\My Documents\Altova\GlobalResources.xml
```

このファイルは、すべてのAltova アプリケーションに対するデフォルトのグローバル リソース定義ファイルとしてセットされます。従って、任意のAltova アプリケーションからこのファイルに対して保存されたグローバル リソースは、その他すべてのAltova アプリケーションにて直ちにグローバル リソースとして使用することができることになります。グローバル リソースを定義して、グローバル リソース定義ファイルに保存するには、グローバル リソースの管理ダイアログにてグローバル リソースを追加し、**「OK」** ボタンをクリックしてください。

既に存在するグローバル リソース定義ファイルを、実行中のAltova アプリケーションにおけるアクティブな定義ファイルとして選択するには、定義ファイル テキスト ボックスの隣にある **「参照」** ボタンから、そのファイルを選択してください。(上のスクリーンショットを参照)

メモ: グローバル リソース定義ファイルの名前は自由に変更することができ、Altova アプリケーションからアクセスできる任意の場所へ配置することができます。この場合、各アプリケーションにて (定義ファイル テキスト ボックスから) このファイルをアプリケーションのグローバル リソース定義ファイルとして指定する必要があります。複数のAltova 製品にて同一の定義ファイルを指定することで、これらのAltova 製品にてリソースが利用できるようになります。

メモ: 複数のグローバル リソース定義ファイルを作成することもできます。この場合、各 Altova アプリケーションにて同時に選択できるアクティブな定義ファイルは 1 つだけで、そのファイルの中に含まれている定義だけがアプリケーションでは利用できるということに注意してください。そのため、必要に応じて同じリソースを複数の定義ファイルで記述する必要が生じることもあります。

グローバル リソースの管理: 追加、編集、削除、保存

グローバル リソースの管理ダイアログでは、選択されたグローバル リソース定義ファイルへグローバル リソースを追加することができるほか、選択されたグローバル リソースを編集または削除することができます。ファイル、フォルダー、そしてデータベースのグループに追加されたグローバル リソースは、グローバル リソース定義ファイルにより管理されます (上のスクリーンショットを参照)。

グローバル リソースを追加するには、「追加」ボタンをクリックして、表示される「**グローバル リソース**」ダイアログにてグローバル リソースを定義します (このセクション以下のサブセクションにある「ファイル」、「フォルダー」、「データベース」の記述を参照ください)。グローバル リソースを定義して (グローバル リソースの管理ダイアログにある「**OK**」ボタンをクリックすることで) 保存すると、選択されたグローバル リソース定義ファイルのグローバル リソース定義ライブラリに、そのグローバル リソースが追加されます。グローバル リソースはエイリアスにより識別されます。

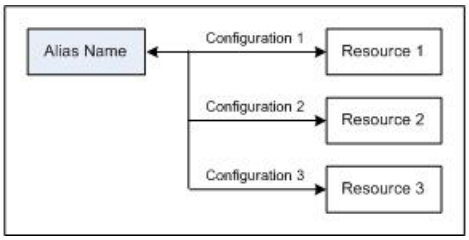
グローバル リソースを編集するには、「**編集**」ボタンをクリックします。関連するグローバル リソース ダイアログが表示され、必要な変更を行うことができます (「ファイル」、「フォルダー」、「データベース」サブセクションの記述を参照ください)。

グローバル リソースを削除するには、「**削除**」ボタンをクリックします。

グローバル リソースの追加、編集、または削除を終えた後には、グローバル リソースの管理ダイアログにある「**OK**」ボタンをクリックして、グローバル リソース定義ファイルに**変更点を保存**するようにしてください。

構成によりグローバル リソースをエイリアス名へ関連付ける

グローバル リソースを定義すると、エイリアス名をリソース (ファイル、フォルダー、データベース) へマッピングすることになります。1 つのエイリアス名から複数のリソースへのマッピングを行うことができ、各マッピングは構成と呼ばれます。そのため、単一のエイリアス名から、複数の構成を介して複数のリソースへの関連付けを行うことができます (以下の図を参照)。



Altova アプリケーションではファイルの代わりにエイリアスを割り当てることができます。アプリケーションのアクティブなグローバル リソース構成を切り替えることで、各エイリアスに割り当てられたリソースを切り替えることができます。例えば、MobileTogether Designer 内で、2 つ以上の代替デフォルトファイルを持つデータソースの場合、グローバルリソース エイリアスをデフォルトファイルとして割り当てることができます。MobileTogether Designer 内で、異なるデフォルトファイルを使用するために、アクティブな構成を変更することができます。`Configuration-1` が `FirstDefault.xml` をグローバルリソースエイリアスにマップする場合、そして、`Configuration-1` がアクティブな構成として選択されている場合、`FirstDefault.xml` がデフォルトファイルとして使用されます。この様にして、単一のエイリアスを介して複数の構成にアクセスすることができます。このメカニズムはリソースをテストまたは比較する際に役に立ちます。更に、グローバルリソースは Altova 製品全てで使用することができるため、複数の Altova 製品でもリソースをテストおよび比較することができます。

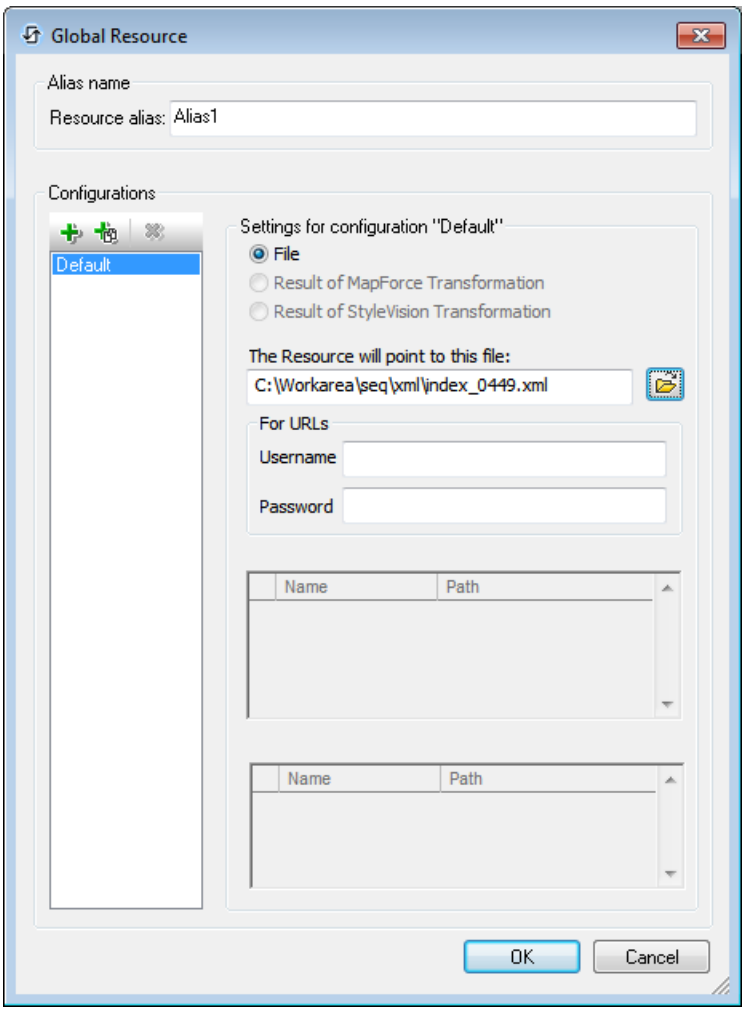
ファイル

「グローバル リソースの管理」ダイアログにある **追加 | ファイル**」コマンドにより、ファイルのグローバル リソース ダイアログへアクセスすることができます (以下のスクリーンショットを参照)。グローバル リソース ダイアログでは、リソース エイリアス テキスト ボックスにて入力されたエイリアスの構成を定義することができます。以下に記される方法により構成のプロパティを指定して、**「OK」**をクリックしてエイリアスの定義を保存してください。

エイリアスの定義を保存した後には上記の操作を (「グローバル リソースの管理」ダイアログにある「追加 | ファイル」コマンドを選択するところから) 再度行うことで、別のエイリアスを追加することができます。

グローバル リソース ダイアログ

エイリアスはグローバル リソース ダイアログにて定義されます (以下のスクリーンショットを参照)。



グローバル リソース ダイアログのアイコン

構成の追加: 追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。

構成をコピーして追加: 選択された構成のコピーとして追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。

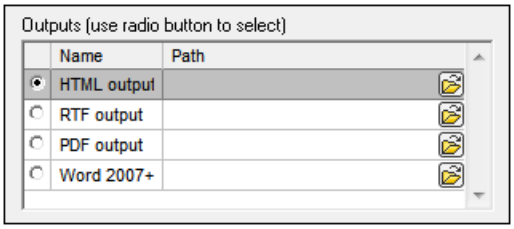
削除: 選択された構成を削除します。

開く: グローバル リソースとして作成されるファイルを参照します。

エイリアスの定義

以下の操作により、エイリアスの名前と構成を定義することができます:

1. **エイリアスに名前を付ける**: リソース エイリアス テキスト ボックスにエイリアスの名前を入力します。
2. **構成を追加する**: 構成ペインには、デフォルトで(名前変更や削除を行うことができない) Default という名前の構成が表示されます。(i) 構成の追加または構成のコピーを追加アイコンをクリックして、(ii) 表示されるダイアログにて**構成名を入力**することで、必要なだけの**構成を追加**することができます。追加された各構成は構成リストに表示されます。上のスクリーンショットでは、Long とShort という名前の構成が構成リストに追加されています。構成のコピーを追加コマンドにより、選択されている構成をコピーして、修正することもできます。
3. **各構成に対してリソースの種類を選択する**: 構成のリストから目的の構成を選択し、構成の設定ペインにて、(i) ファイル、(ii) MapForce 変換の結果、または(iii) StyleVision 変換の結果から、構成のリソースを指定します。対応するラジオ ボタンを選択してください。MapForce または StyleVision 変換の結果が選択された場合、mfd または sps ファイルと対応する入力ファイルを使って MapForce または StyleVision により変換が行われ、その変換結果がリソースとなります。
4. **リソースの種類に対するファイルを選択する**: リソースの種類にファイルが選択された場合、リソースのファイル テキスト ボックスにてファイルを選択してください。リソースの種類に変換の結果が選択された場合、ファイルの選択テキスト ボックスにて(MapForce 変換の場合は) .mfd ファイルまたは(StyleVision 変換の場合は) .sps ファイルを選択してください。変換に対して複数の入力または出力が可能な場合、選択するためのオプションが表示されます。例えば、StyleVision 変換の出力オプションが、インストールされている StyleVision のエディションに合わせて表示されます(以下のスクリーンショットでは、Enterprise Edition の出力が示されています)。



目的のオプションに対応するラジオ ボタンを選択します(上のスクリーンショットでは、「HTML 出力」が選択されています)。変換の結果をリソースとする場合、出力をファイルまたはグローバル リソースとして保存することができます。 アイコンをクリックして、「他のグローバル リソースを選択」または、「参照」を選択してください。どちらのオプションも選択されていない場合、グローバル リソースが使用された際に変換の結果が一時ファイルとしてロードされます。

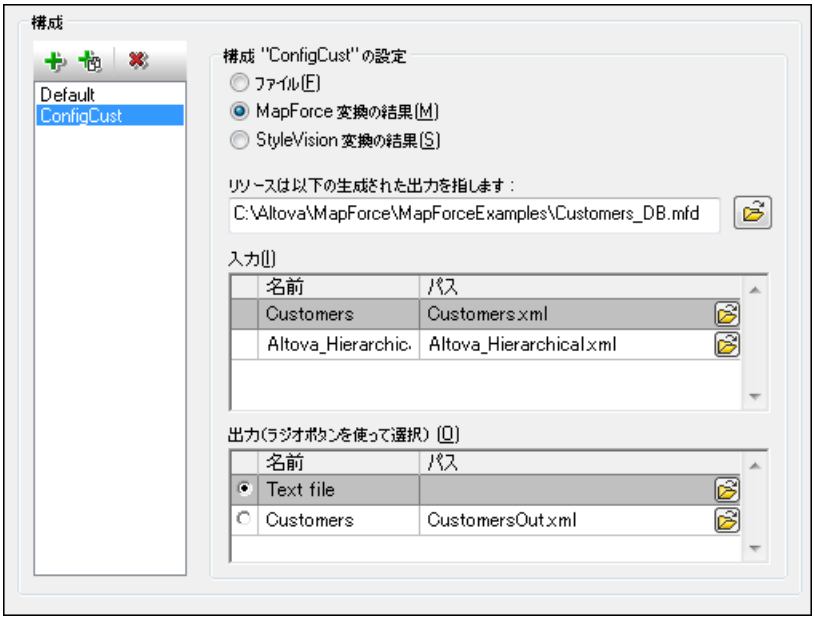
5. **必要に応じて複数の構成を定義する**: その他の構成を必要な数だけ追加して、各構成に対してリソースを指定します。各構成に対して上記のステップ3 と4 を実行してください。エイリアスに対する新たな構成はいつでも追加することができます。

6. **エイリアスの定義を保存する**: 「OK」をクリックしてエイリアスと、すべての構成をグローバル リソースとして保存します。グローバル リソースが、「グローバル リソースの管理」ダイアログにあるファイル以下に表示されます。

MapForce 変換の結果

Altova MapForce では、(既存の) 入力ドキュメント スキーマから出力ドキュメント スキーマへのマッピングが行われます。MapForce ユーザーにより作成されたこのマッピングは、MapForce Design (MFD) と呼ばれます。入力スキーマに関連付けられた XML ファイル、テキスト ファイル、データベースは、データ ソースとして使用することができ、MapForce により出力ドキュメント スキーマに対応した出力データ ファイルが生成されます。「MapForce 変換の結果」が出力ドキュメントで、グローバル リソースになります。

MapForce により生成されたデータ ファイルをグローバル リソースとしてセットするには、グローバル リソース ダイアログにて、以下の設定を行う必要があります (以下のスクリーンショットを参照)。



- **.mfd (MapForce Design) ファイル**: 「リソースは以下の生成された出力を指します」テキスト ボックスにて、mfd ファイルを指定する必要があります。
- **入力データ ファイル**: MFD ファイルが指定されると、そのファイルが解析され、中に含まれる入力スキーマ情報をベースにデフォルトのデータ ファイルが入力ペインに入力されます (上のスクリーンショットを参照)。他のファイルを指定することで、各入力スキーマに対するデフォルト ファイルの選択を変更することができます。
- **出力ファイル**: MFD ドキュメントに複数の出力スキーマが含まれる場合、出力ペインにそれらの出力すべてが表示されます (上のスクリーンショットを参照)。個々の出力スキーマに関連付けられた出力ファイルの場所が MFD ドキュメントにて指定されている場合、その出力スキーマに対するファイルの位置が出力ペインに表示されます。上のスクリーンショットからは、Customers 出力スキーマにデフォルトの XML データ ファイル (CustomersOut.xml) が与えられており、Text ファイル出力 スキーマにはそのような割り当てが行われていないことが理解できます。出力ペインに表示あるデフォルトのファイルの場所を使用することができるほか、自分で指定することができます。MapForce 変換の結果が、選択された出力スキーマがある場所へ保存され、そのファイルがグローバル リソースとして使用されます。

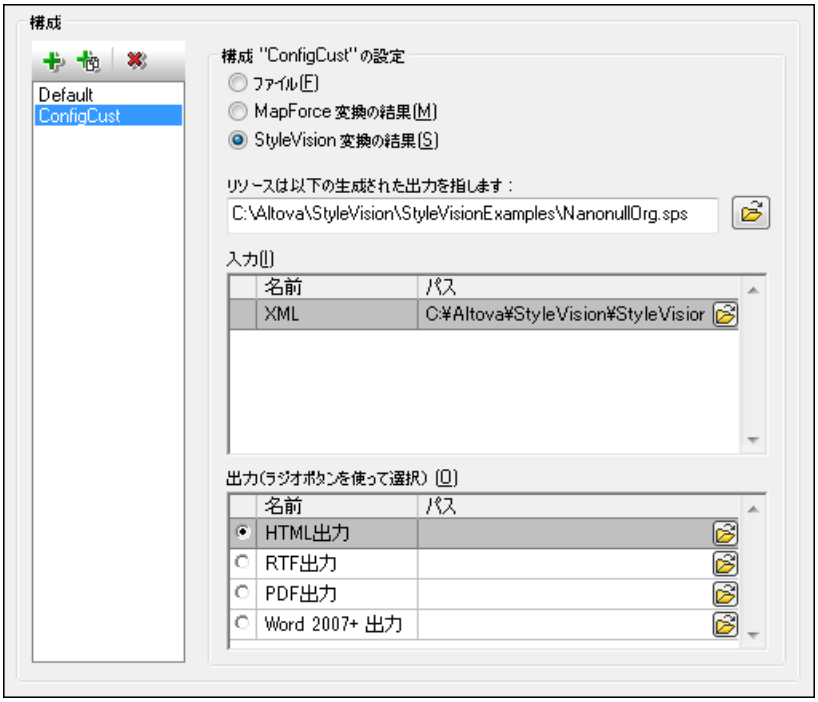
メモ　このオプション (MapForce の結果) により、グローバル リソースが呼び出された時に変換が行われるようになります。これで (入力ファイルから得られた) 最新のデータがグローバル リソースに含まれることが保証されます。

メモ　変換を行うためには MapForce が使用されるため、この機能を使用するには Altova MapForce をインストールする必要があります。

StyleVision 変換の結果

Altova StyleVision はStyleVision Power Stylesheet (SPS) ファイルを作成するために使用され、これらSPS ファイルにより XML ドキュメントを様々なフォーマット (HTML、PDF、RTF、Word 2007+ など) の出力ドキュメントへ変換するためのXSLT スタイルシートが生成されます。「StyleVision 変換の結果」オプションを選択すると、StyleVision により作成された出力ドキュメントが、選択された構成に関連付けられたグローバル リソースとなります。

グローバル リソースダイアログにて「StyleVision 変換の結果」オプションを選択すると、以下のファイルを指定する必要があります。



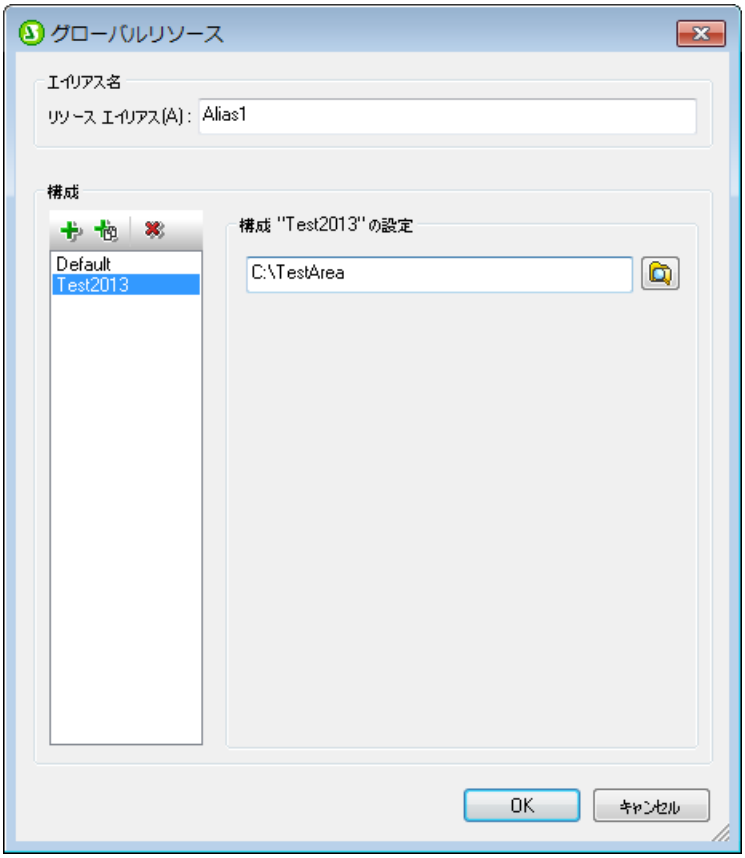
- ***.sps* (SPS) ファイル**: 「リソースは以下の生成された出力を指します」テキスト ボックスにて、ファイルを指定する必要があります (上のスクリーンショットを参照)。
- **入力ファイル**: 入力ファイルがSPS ファイル内部で既に指定されている場合、SPS ファイルが選択された段階で、そのファイルが入力ペインに表示されます。このエントリーは変更することができるほか、エントリーが指定されていない場合、追加する必要があります。
- **出力ファイル**: 出力ペインにて出力フォーマットを選択し、そのフォーマットに対する出力ファイルの場所を指定します。

メモ このオプション (StyleVision の結果) により、グローバル リソースが呼び出された時に変換が行われるようになります。これで、(入力ファイルから得られた) 最新のデータがグローバル リソースに含まれることが保証されます。

メモ 変換を行うためにはStyleVision が使用されるため、この機能を使用するにはAltova StyleVision をインストールする必要があります。

フォルダー

フォルダーに対するグローバル リソース ダイアログでは、以下の操作によりフォルダー リソースを追加することができます (以下のスクリーンショットを参照)。



グローバル リソース ダイアログのアイコン

- **構成の追加**: 追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。
- **構成をコピーして追加**: 選択された構成のコピーとして追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。
- **削除**: 選択された構成を削除します。
- **開く** グローバル リソースとして作成されるファイルを参照します。

エイリアスの定義

以下の操作によりエイリアスを定義することができます::

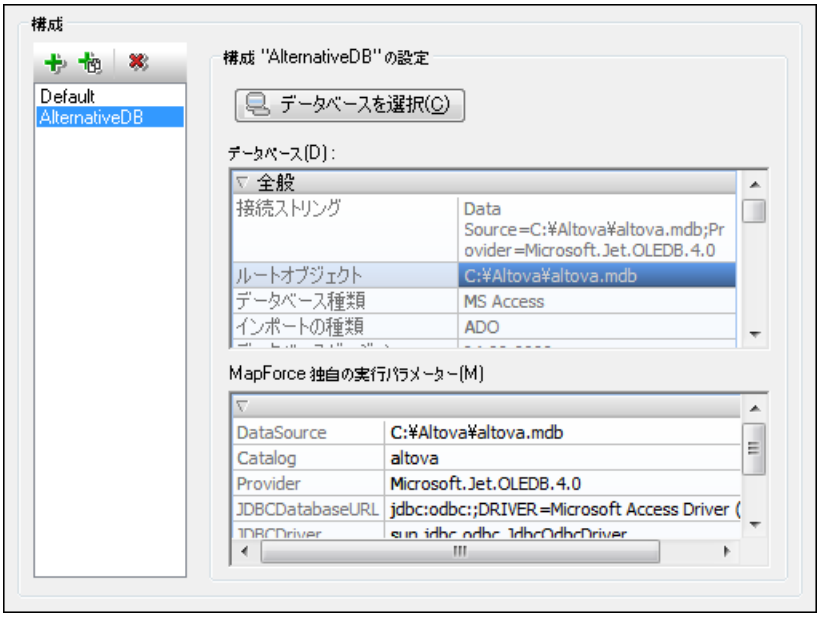
1. **エイリアスに名前を付ける** リソース エイリアス テキスト ボックスにエイリアスの名前を入力します。
2. **構成を追加する** 構成ペインには、デフォルトで (名前変更や削除を行うことができない) Default という名前の構成が表示されます。**構成の追加**または**構成のコピー**を追加アイコンをクリックして、表示されるダイアログにて構

成名を入力することで、必要なだけの構成を追加することができます。構成の追加または構成のコピーを追加アイコンをクリックして、表示されるダイアログで構成の名前を入力することで、構成が追加されます。「**OK**」をクリックすることで新たな構成が構成ペインに表示されます。同じ作業を必要な数だけ繰り返してください。

3. 構成のリソースとなるフォルダーを選択する　構成ペインに表示された構成を選択し、グローバルリソースとして作成するフォルダーを参照します。
4. 必要に応じて複数の構成を定義する　その他の構成を必要な数だけ追加して、各構成に対してリソースを指定します。各構成に対して上記のステップ2 と3 を実行してください。エイリアスに対する新たな構成はいつでも追加することができます。
5. エイリアスの定義を保存する　「**OK**」をクリックしてエイリアスと、すべての構成をグローバルリソースとして保存します。グローバルリソースが、「グローバルリソースの管理」ダイアログにあるファイル以下に表示されます。

データベース

データベースに対するグローバル リソースダイアログでは、以下の操作によりデータベースのリソースを追加することができます:

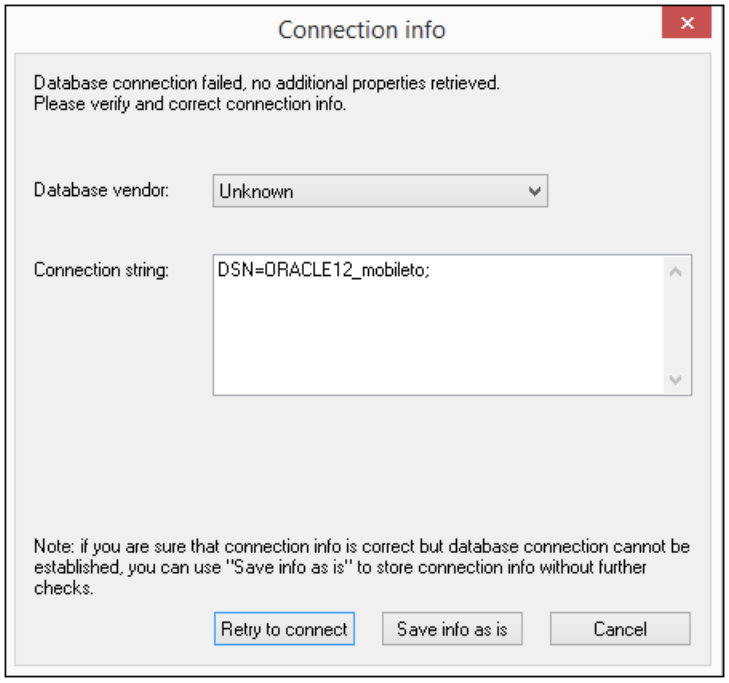


グローバル リソース ダイアログのアイコン

- **構成の追加**: 追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。
- **構成をコピーして追加**: 選択された構成のコピーとして追加する構成の名前を入力するための、構成の追加ダイアログが表示されます。
- **削除**: 選択された構成を削除します。

検証されていない接続情報の保存

入力された接続情報で正しくデータベースに接続できない場合、これはサーバーにソリューションのサーバーのデプロイ時の接続情報は正確でも、サーバーがローカルマシンで作動しない場合があります。接続が(グローバルリソースを定義する)ローカルマシンから作動しない場合、接続情報ダイアログ(*下のスクリーンショット*)が表示されます。



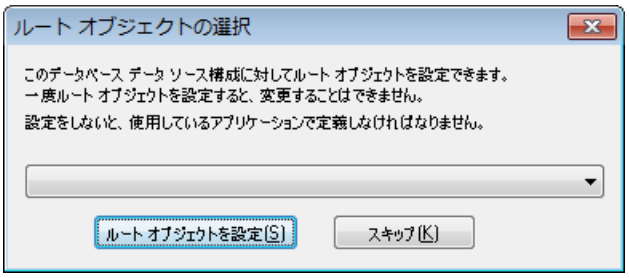
以下のオプションがあります:

- *接続を再度試みる* ローカルマシンからの接続を有効化する接続情報を入力し、 **「接続を再度試みる」** をクリックします。は、接続を試みます。
- *情報をそのまま保存する* または接続または接続情報を検証することなく接続情報を保存します。この接続情報はソリューションがサーバーにデプロイされる時に使用されます。
- *キャンセル:* グローバルリソースとしてデータベースを定義することをキャンセルする。

エイリアスの定義

以下の操作によりエイリアスを定義することができます:

1. *エイリアスに名前を付ける* リソース エイリアス テキスト ボックスにエイリアスの名前を入力します。
2. *構成を追加する* 構成ペインには、デフォルトで(名前変更や削除を行うことができない) Default という名前の構成が表示されます。構成の追加または構成のコピーを追加アイコンをクリックして、表示されるダイアログにて構成名を入力することで、必要なだけの構成を追加することができます。構成の追加または構成のコピーを追加アイコンをクリックして、表示されるダイアログで構成の名前を入力することで、構成が追加されます。「OK」をクリックすることで新たな構成が構成ペインに表示されます。同じ作業を必要な数だけ繰り返してください。
3. *構成のリソースとなるデータベースを選択する* 構成ペインに表示された構成を選択し、 **「データベースを選択」** ボタンをクリックします。グローバル リソース接続を作成ダイアログが表示されます。
4. *データベースへ接続する* 接続ウィザード、既存の接続、 ADO 接続、ODBC 接続、または JDBC 接続を使用して、データベースへの接続を作成します。
5. *ルート オブジェクトの選択:* ルート オブジェクトを選択することのできるデータベースへ接続した場合、サーバー上にあるルート オブジェクトを選択するためのダイアログが表示されます (以下のスクリーンショットを参照)。ルート オブジェクトを選択して、 **「ルート オブジェクトを設定」** をクリックしてください。選択されたルート オブジェクトが、この構成が使用された時にコードされるルート オブジェクトとなります。



ルートオブジェクトを選択しなかった場合 (**「スキップ」**ボタンをクリックした場合)、グローバルリソースがロードされた際にルートオブジェクトを選択することになります。

6. *複数の構成を必要に応じて定義する:* 作成されたその他の構成に対してデータベースリソースを指定します (作成された構成に対して上記のステップ3 から5 までを繰り返します)。新たな構成をエイリアス定義へ追加することもできます。
7. *エイリアスの定義の保存:* **「OK」**をグローバルリソースダイアログ内でクリックし、エイリアスとその構成をグローバルリソースとして保存します。グローバルリソースは、グローバルリソースの管理ダイアログ内でデータベースの下にリストされています。

14.2 グローバルリソースを使用する

グローバルリソースには、ファイル型、フォルダー型、そしてデータベース型と、複数の種類が存在します。MobileTogether Designer で使用することのできる様々なシナリオについて以下で紹介されています: ファイルとフォルダー および データベース。

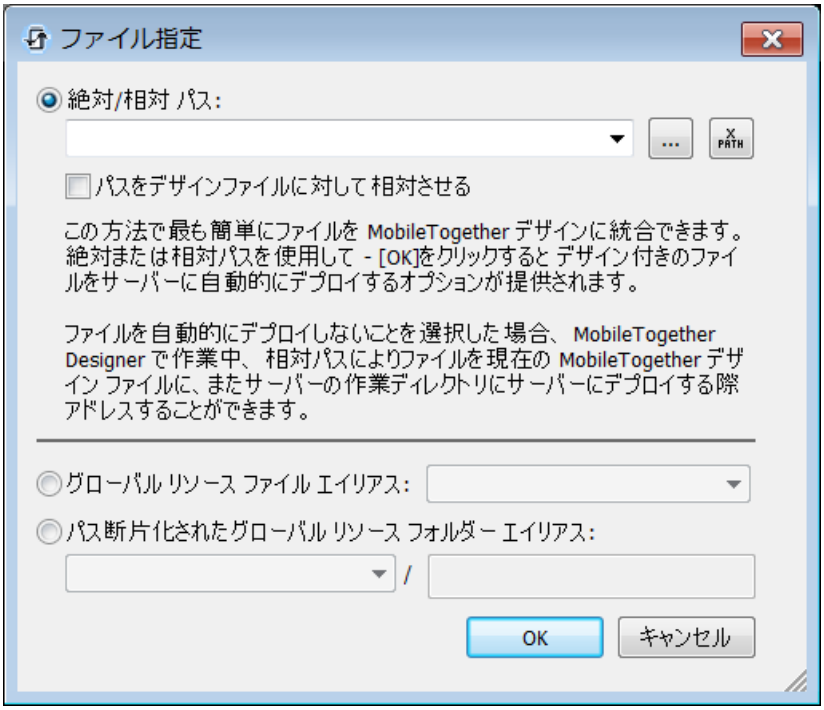
使用するリソースの選択

アプリケーション全体で使用することのできるグローバルリソースの選択、ならびに実際に使用されるグローバルリソースの選択を行うには方法が2つあります:

- 「グローバルリソース」ダイアログにてアクティブなグローバルリソース XML ファイルを選択します。アクティブなグローバルリソース XML ファイル内に収められている利用可能なグローバルリソースの定義は、アプリケーション内の開かれているファイル全てで使用することができます。使用することができるのは、アクティブなグローバルリソース XML ファイルにある定義だけとなります。アクティブなグローバルリソース XML ファイルはいつでも変更することができ、新しいファイルに収められたグローバルリソース定義が、それまで使用されていたアクティブなファイルの内容に置き換えられます。従って、アクティブなグローバルリソース XML ファイルにより、(i) 割り当て可能なグローバルリソースと、(ii) 検索可能なグローバルリソースが決定されます（例えば、あるグローバルリソース XML ファイル内のグローバルリソースに割り当てが行われているが、現在アクティブなグローバルリソース XML ファイルにて、対応する名前のリソースがない場合、割り当てられたグローバルリソースの検索を行うことはできなくなります）。
- **「ツール」 「アクティブな構成」** または グローバルリソース ツールバーから アクティブな構成 を選択します。メニューコマンド（または ツールバーのドロップダウンリスト）をクリックすることで、全てのエイリアスに対する構成のリストが表示されます。構成を選択することで、その構成がアプリケーションでアクティブな構成となります。この操作により、グローバルリソース（または エイリアス）が使用される際には、各エイリアスに対して対応するアクティブな構成が適用されます。アクティブな構成は使用される全てのエイリアスに対して適用されます。アクティブな構成と同じ名前の構成をエイリアスが持たない場合、そのエイリアスに対してはデフォルトの構成が適用されます。アクティブな構成はリソースの割り当てに影響を与えません。アクティブな構成は、実際にリソースが使用されるときに重要になります。

ファイルとフォルダーの割り当て

特定の使用シナリオでは、ファイル型ならびにフォルダー型のグローバルリソースをファイルまたはフォルダーを指定するために使用することができます。例えば、「ページソース」ペイン内で、データソースのデフォルトファイルをグローバルリソースを使用して割り当てることができます。ファイルの指定が表示されます（以下のスクリーンショットを参照）。

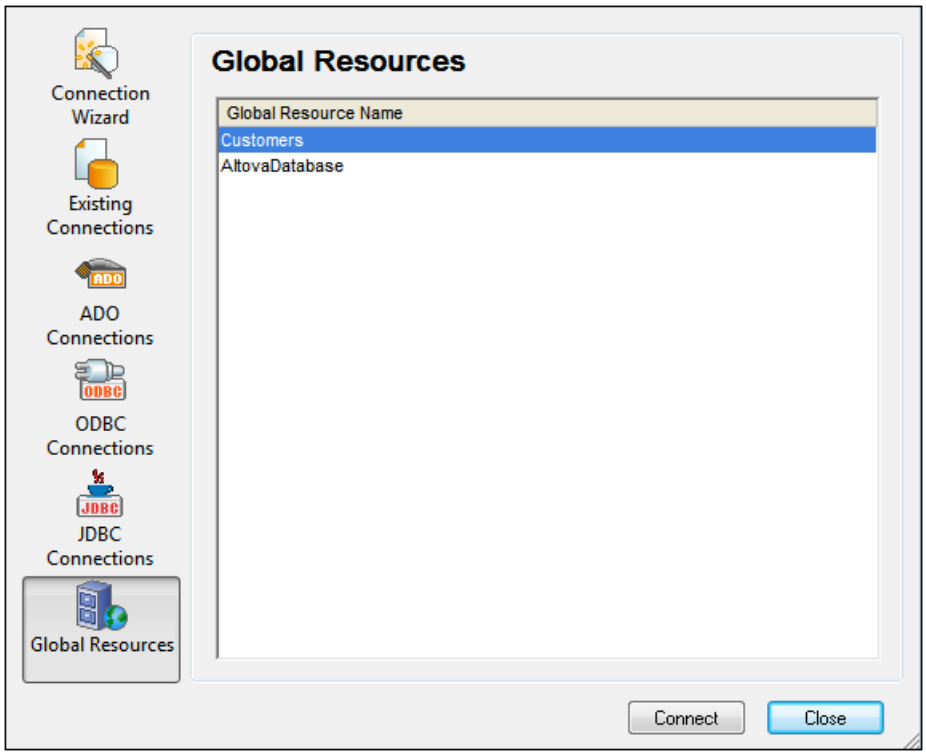


ファイル型またはフォルダー型のグローバルリソースを選択します。現在アクティブなグローバルリソースXML ファイルで定義された、すべてのファイル型グローバルリソースとすべてのフォルダー型グローバルリソースを表示するコンボボックスが表示されます。必要なグローバルリソースを選択します。ファイル型グローバルリソースの場合、選択されたエイリアスがファイルにマップされます。フォルダー型のグローバルリソースの場合、選択されたエイリアスがフォルダーにマップされ、リソースを検索するためのパスの残りを入力することができます。**「OK」**をクリックします（上のスクリーンショットを参照）。

選択されたグローバルリソースが1つ以上の構成を持つ場合、現在アクティブな構成のデータベースリソースが使用されます（**「ツール| アクティブな構成」**またはグローバルリソース ツールバーをチェックしてください）。

データベースの割り当て

データまたはデータ構成をDB からインポートするコマンドを実行する場合、グローバルリソースを使用するオプションを使用することができます（以下のスクリーンショットを参照）。



データベースを開くダイアログにてグローバルリソースアイコンをクリックすると、現在アクティブなグローバルリソースXML ファイルに定義されている全てのデータベース型グローバルリソースが表示されます。目的のグローバルリソースを選択して、**「接続」**をクリックします。選択されたグローバルリソースに2つ以上の構成が割り当てられている場合、現在アクティブな構成に対応したグローバルリソースが使用され（**「ツール | アクティブな構成」**またはグローバルリソースツールバーを確認してください）、接続が確立されます。

構成の切り替え

同時にアクティブにすることのできる構成は1つだけで、その設定はアプリケーション全体に対して適用されます。これは、アクティブな構成が、現在開いている全てのファイルやデータソース接続のエイリアスに対して有効であることを意味します。エイリアスにアクティブな構成と同じ名前の構成が割り当てられていない場合、デフォルトの構成がそのエイリアス対して使用されます。構成を変更する例として、XSLT ファイルが複数の構成を持つグローバルリソースを介して XML ドキュメントに割り当てられたと仮定しましょう。各構成は異なるファイルにマップされます。ですから、ファイルの選択はアプリケーションのアクティブな構成としてどの構成が選択されているかにより決まります。

構成の切り替えるには2つの方法があります：

- メニューコマンド **「ツール | アクティブな構成」** を選択すると、グローバルリソース XML ファイル内にある全ての構成がサブメニューに表示されます。目的の構成を選択します。
- グローバルリソースコンボボックスのツールバーから、目的の構成を選択します（以下のスクリーンショットを参照）。

アクティブな構成を切り替えるという方法により、グローバルリソースを介して割り当てられたソースファイルを変更することができます。

チャプター 15

シミュレーション

15 シミュレーション

ソリューションを実行するには次のシミュレーションの方法を使用することができます:

- MobileTogether Designer でのシミュレーション: デザインにエラーがないか素早く確認することができます。
- サーバーでのシミュレーション: デザインにエラーがないか追加のテストを実行することができます。このシミュレーションにより、データソースが正確に検索されているか、また、URL が正確であるか、現在のサーバー設定が適切であるか、使用されるDB、URL、およびファイルへアクセスするための全てのパーミッションがあるかを追加確認することができます。
- クライアントでのシミュレーション: 試行で実際のクライアントとサーバーのインタラクションのテストケースの試行することができます。このシミュレーションでは、MobileTogether Designer がサーバーの役割を果たします。

メモ ソリューションをサーバーにデプロイすることも可能です、また、MobileTogether Server のWeb UI のワークフロータブをクリックして、ソリューションをブラウザーで実行することもできます。

シミュレーションの実行中 MobileTogether Designer GUI の[メッセージ] ペインで、実行中のすべてのアクティビティのステップバイステップの詳細レポートが表示されます。これは、デザインファイルをテストしデバッグするためのとても重要な機能です。

メモ

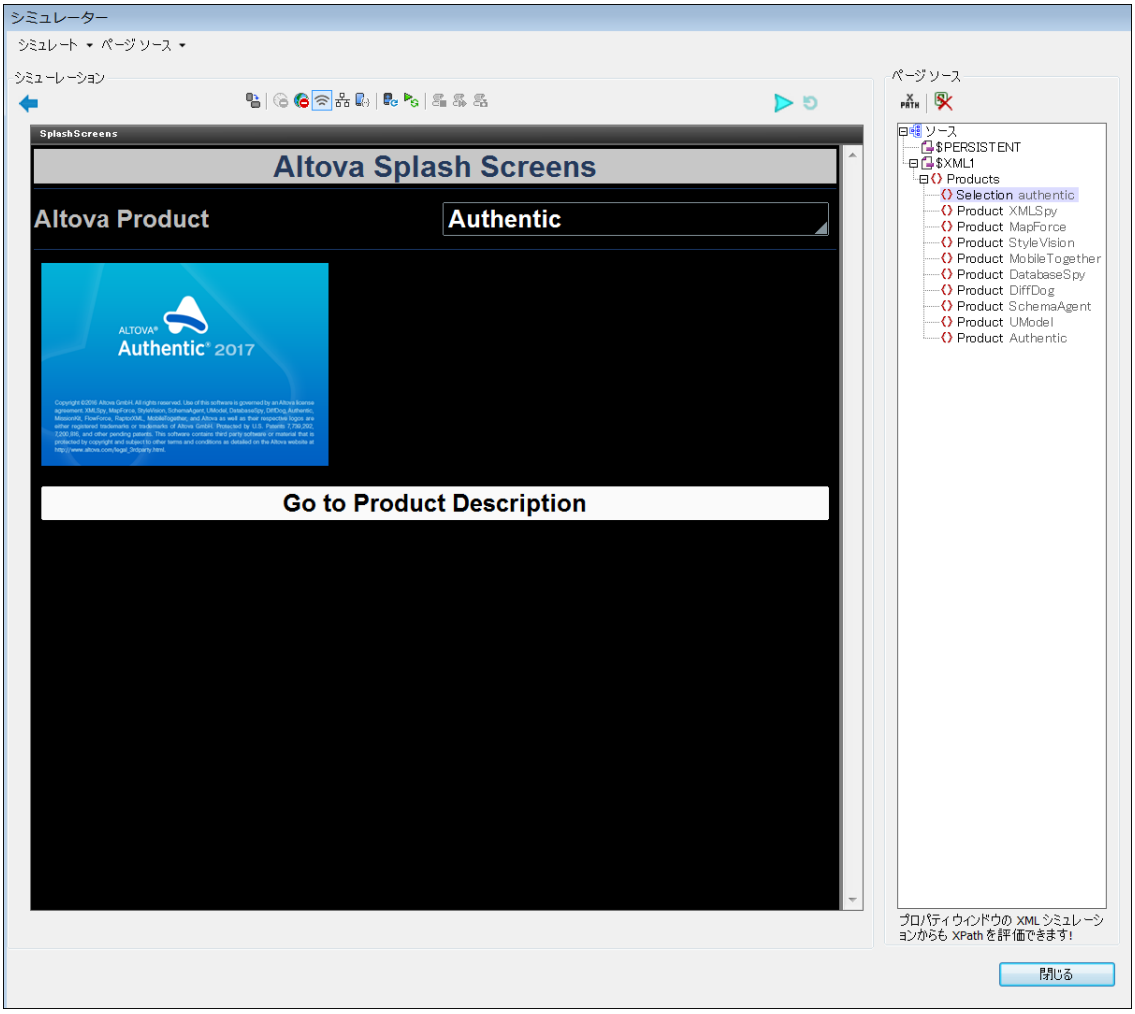
- 位置情報アクションを含むデザインをデザイナーまたはサーバでシミュレートする場合は、シミュレーション内で使用する位置情報を定義する必要があります。操作方法は位置情報の設定のセクションに説明されています。
- デスクトップで実行されるシミュレーターはNFC が有効化されておらず、NFC サンプルファイルはNFC タグをシミュレートするために使用することができます。サンプルファイルのテンプレートに関して、NFC サンプルファイルのセクションを参照してください。
- 送信および受信ソリューション内のプッシュ通知のシミュレーションが異なる場合、シミュレーションされ、送信されるプッシュ通知内のデータをファイルに保存することができます。このデータは、プッシュ通知が受信されたものとして、ソリューションの受信のシミュレーションにロードされることができます。
- デバイスのアドレス帳をシミュレートするには、アドレス帳の連絡先のサンプルXML ファイルを使用することができます。このようなファイルの作成および使用に関する詳細は、次を参照してください: 連絡先のサンプルファイル。

シミュレーションする言語

デザイナー とサーバー のためのシミュレーションする言語は **[プロジェクト] シミュレーションする言語]** コマンドで選択します。クライアント シミュレーション の言語は、シミュレーションが実行されるモバイルデバイスの言語と同じです。

15.1 MobileTogether Designer でのシミュレーション

MobileTogether Designer 内で直接プロジェクト ワークフローのシミュレーションを実行することができます。シミュレーションのデバイスはメイン ツールバーのプレビューデバイスのコンボ ボックスで現在選択されているデバイスです。異なるデバイスでシミュレーションを確認するためにプレビュー デバイスを変更することができます。シミュレーションを実行するには、[プロジェクト | ワークフローのシミュレーション] または [F5] を選択します。これによりシミュレーターが開かれ、シミュレーションが開始されます。デザイナーのシミュレーションは [メッセージ] ペイン] のサーバーとクライアントのメッセージを報告します。



シミュレーションする言語

デザイナー とサーバー のためのシミュレーションする言語は [**プロジェクト**] [**シミュレーションする言語**] コマンドで選択します。クライアント シミュレーション の言語は、シミュレーションが実行されるモバイルデバイスの言語と同じです。

ファイルの場所

MobileTogether Designer 内でシミュレーションが直接実行される場合、ファイルのロケーションはデザイン内で指定された通り解決されます。相対パスはデザインファイルのロケーションと相対します。[ワークフローのシミュレーションにサーバーを使

用] を使用する場合ファイルのロケーションがどのように解決されるか、これらのロケーションを比較します。

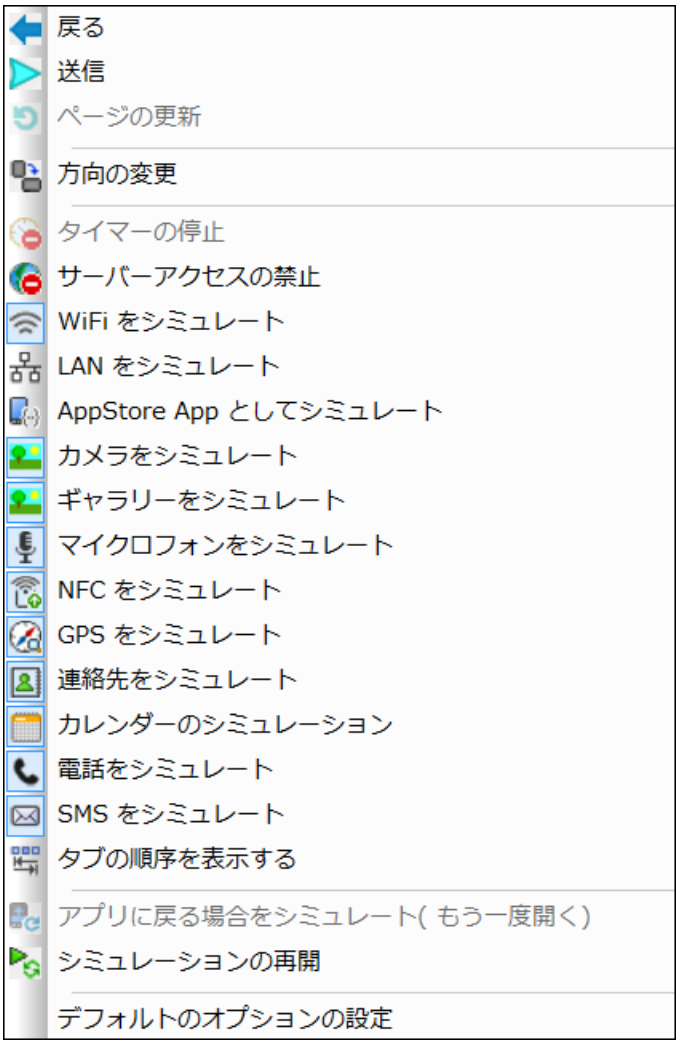
シミュレーターの機能

[シミュレーター] ウィンドウには次の機能があります:

- 左側の (シミュレーション) ペインはシミュレーションを表示します。シミュレーションペインのためのオプションは下で説明されています。
- 右側の (ページソース) ペインはシミュレーションの展開中に変化するページソースを表示します。XML データペインのためのオプションは下で説明されています。
- シミュレーションの展開中 MobileTogether Designer GUI の [メッセージ] ペインは、発生しているすべてのアクティビティのステップ バイ ステップのレポートを作成します。ですから、ワークフローの展開の各ステップで何がおこっているのか確認することができます。これはデザインファイルのテストおよびデバッグを行うとても有益な機能です。
  - *デザイナーのシミュレーションがサーバーとクライアントの両方のアクティビティに関してレポートを行う。*
  - *サーバーのシミュレーションはクライアントメッセージに関してのレポートを行う。*
  - *クライアントのシミュレーションはサーバーメッセージに関してのレポートを行う。*
- ユーザー インタラクションの有効化を必要とするコントロール。上のスクリーンショットでは、例えば、コンボ ボックスが有効化されています。

シミュレーションペインのオプション

[シミュレート] メニュー内とシミュレートツールバーアイコン でこれらのオプションを使用する事ができます (*下のスクリーンショット*) 。



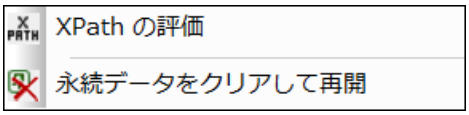
- *戻る* ページがサブページの場合、**[戻る]** をクリックするとサブページが閉じられます。ページがトップページの場合は、シミュレーターが閉じられます。 **「戻る」ボタンのクリック時**も参照してください。
- *送信:* ページが最後のページではない場合、**[送信]** をクリックすると、次のページに移動します。ページが最後のページの場合は、ワークフローが終了されます。 **送信ボタンのクリック時** も参照してください。
- *ページの更新:* 基になるデータが変更されると、**[ページの更新]** が有効化されます。**[ページの更新]** をクリックすると、ページが更新されたデータと共に更新されます。
- *方向の変更:* ビューを横長と縦長の方向に切り替えることができます。
- *タイマーの停止:* タイマーが間隔で実行するに設定されており、アクションが指定された間隔で実行されるように定義されている場合、**[タイマーの停止]** をクリックすることによりタイマー(結果的にアクション) を停止する事ができます。これによりこれらのアクションによるメッセージの多くをクリアする事ができ、他のメッセージやワークフローの他の部分を更に簡単に分析する事ができます。
- *サーバーアクセスの禁止:* サーバーへのアクセスを無効化し、サーバーへの接続エラーの状況下でのソリューションの振る舞いをテストする事ができます。クリックされると、ボタンは **[サーバーへのアクセスの有効化ボタン]** に切り替えられます。詳細に関してはサーバー接続エラーを参照してください。
- *WiFi が音の状態でシミュレート:* オンに切り替えられた場合、`mt-connected-via-wifi` は、XPath 拡張関数を `true()` に設定し、オフに切り替えられた場合は、`false()` を返します。これにより、シミュレータは WiFi へのアクセスが使用できる状態でのシミュレーションを行います。このように、WiFi へのアクセスを必要とするデザインのシナリオをシミュレートすることができます。
- *LAN がオンの状態でシミュレート:* オンに切り替えられた場合、`mt-connected-via-lan` は、XPath 拡

張関数を`true()` に設定し、オフに切り替えられた場合は、`false()` を返します。これにより、シミュレータは LAN へのアクセスが使用できる状態でのシミュレーションを行います。このように、LAN へのアクセスを必要とするデザインのシナリオをシミュレートすることができます。

- *AppStore App としてシミュレート:* 有効化されると、静的グローバル変数 **`MT_IsAppStoreApp`** を `true()` に設定し、無効化されると`false()` に設定します。これによりシミュレーションはこの変数の値の条件により実行されます。
- *カメラがオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスのカメラ機能が使用できるモノとして振る舞います。これによりカメラ機能を必要とするデザインシナリオをシミュレートすることができます。
- *ギャラリーがオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスのギャラリー機能が使用できるモノとして振る舞います。これによりギャラリーへのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *マイクがオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスのマイクロフォン機能が使用できるモノとして振る舞います。 これによりマイクへのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *NFC がオンの状態をシミュレートする* オンに切り替えられると、**`mt-nfc-started`** XPath 拡張関数が `true()` に設定され、オフに切り替えられると、`false()` に設定されます。NFC が有効化され、NFC アクションを実行することができることをシミュレーターに示します。実際のNFC データはNFC サンプルファイルを介してシミュレーターに提供されます。
- *GPS がオンの状態をシミュレートする* オンに切り替えられると、**`mt-geolocation-started`** XPath 拡張関数が`true()` に設定され、オフに切り替えられると、`false()` に設定されます。シミュレーターがデバイスの GPS ロケーター開始の通知を受け取ると、位置情報機能が使用できるものとしてソリューションを実行します。シミュレーションのために位置情報を提供するために位置情報 XML ファイルを使用してダミーの位置情報を提供することができます。
- *連絡先がオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスのアドレス帳機能が使用できるモノとして振る舞います。これによりアドレス帳へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。連絡先はサンプルファイル またはMicrosoft Outlook 連絡先 からシミュレートされます。オプションダイアログのシミュレーションタブ内で使用するオプションを指定することができます。
- *カレンダーのシミュレーション:* オンに切り替えられると、シミュレーターはデヴァイスのカレンダーが使用できるかのように振る舞います。これにより、カレンダーへのアクセスを必要とするデザインのシナリオをシミュレートすることができます。カレンダーは サンプルファイル またはMicrosoft Outlook 連絡先 からシミュレートされます。オプションダイアログのシミュレーションタブ内で使用するオプションを指定することができます。
- *電話がオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスの電話機能が使用できるモノとして振る舞います。 これにより電話へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *SMS がオンの状態をシミュレートする* オンに切り替えられると、シミュレーターはデバイスのSMS 機能が使用できるモノとして振る舞います。これによりSMS へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *タブの順序の表示:* オンに切り替えられていると、タブ順序のシーケンスの一部になるように定義されている コントロールは、番号を含む青い丸でマークされています。番号はタブ順序シーケンス内のコントロールの位置を示しています。
- *アプリに戻る場合をシミュレート(もう一度開く):* ページが開かれるとページの更新が行われるように定義される と有効化されます。シミュレーション中にページが更新されます。
- *シミュレーションの再開:* シミュレーションを再開します。
- *デフォルトオプションの設定:* シミュレーションペインのオプションをデフォルトの設定にリセットします。

ページソースペインオプションと機能

これらのオプションは**[ページソース]** メニュー内とページソースツールバーで使用する事ができます。



- **[XPath の評価]** をクリックすると、 XPath 式を評価することのできるXPath/XQuery 式の編集ダイアログを

開きます。XPath 式はシミュレーションの実行中、[スタイル & プロパティ] ペインからも編集することができます。
- **[永続データをクリアして再開する]** をクリックすると、永続データをクリアして、シミュレーションを再開することができます。
- ページソースペインでは、ツリーの一部をコピーしてツリーの他の箇所に貼り付けることができます。この機能は、DB 記録などのデータをコピーして、シミュレーションに更なるデータを追加するために役にたちます。コピーされたノードはシミュレーションの間のみ使用することができます。
- ページソースのノードを右クリックすると、そのノードの XPath ロケーター式を **[XPath をコピーする]** コンテキストメニューコマンドを選択して、クリップボードにコピーすることができます。

シミュレーター内のページソースの XML ツリーを編集する

シミュレーター内のツリーは、多種のページソースの XML データを表示し、シミュレーションが展開中にこれらの値がどのように変化するかを表示します。シミュレーター内で切り取り/コピー/貼り付け/削除とドラッグアンドドロップを使用して、XML ツリーを直接編集します。(XML ツリーのコンテキストメニュー内の編集コマンドを使用することができます。) シミュレーションペインは即時変更されたデータを表示します。この機能により、異なる手動で入力されたデータを含む変更された XML データ構造と共にソリューションをテストします。結果、ソリューションはより徹底してテストされ、テストはシミュレーション内で素早く実行することができます。

シミュレーター内の XML ツリーのコンテキストメニューは、以下の機能を提供します:

- **XML のロード**: (XML ツリーと同じ構造と要素を持つ) 外部 XML ファイルを XML ツリーにロードします
- **XML の保存**: XML ツリーの構造とデータを希望する場所に保存します。
- **XMLSpy 内での表示**: Altova XMLSpy プログラム内で XML ツリーを開きます。
- **このツリーをベースにした $XML 構造**: ページソースの構造をシミュレーター内の XML ツリーの構造に上書きします。

15.2 サーバーでのシミュレーション

サーバーでのシミュレーションはMobileTogether Server **[プロジェクト | ワークフローのシミュレーションのためにサーバーを使用する]** を使用してシミュレーションを実行します。[メッセージ] ペイン] にメッセージをレポートします。デザインのエラーをテストするための追加機能として、このシミュレーションによりデータソースが正確な場所にあるかどうか、URL が正確であるか、現在のサーバー設定が適切であるか、サーバーが使用される DB、URL、およびファイルにアクセスするパーミッションがあるかを確認することができます。

サーバーでのシミュレーションのワークフローは以下の通りおこなわれます:

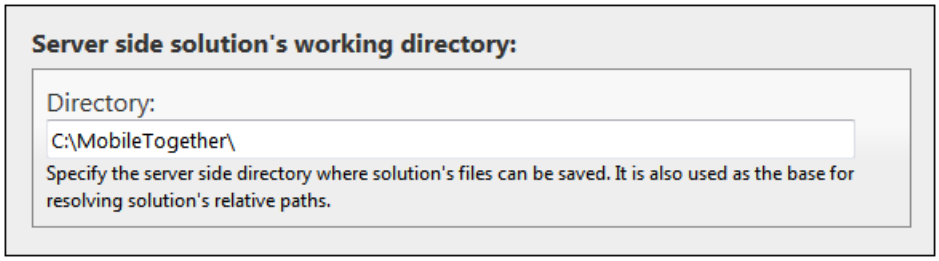
1. MobileTogether Designer 内のアクティブなデザインファイルのワークフローは一時的にMobileTogether Server にパスされます。このようにすると、サーバーからデザインが機能するか確認するためにデザインをサーバにデプロイする必要がなくなります。
2. サーバーがMobileTogether Designer のシミュレーターのワークフローとして機能します。このようにシミュレーターがクライアントの役割をはたします。

シミュレーションする言語

デザイナー とサーバー のためのシミュレーションする言語は **[プロジェクト] シミュレーションする言語]** コマンドで選択します。クライアント シミュレーション の言語は、シミュレーションが実行されるモバイルデバイスの言語と同じです。

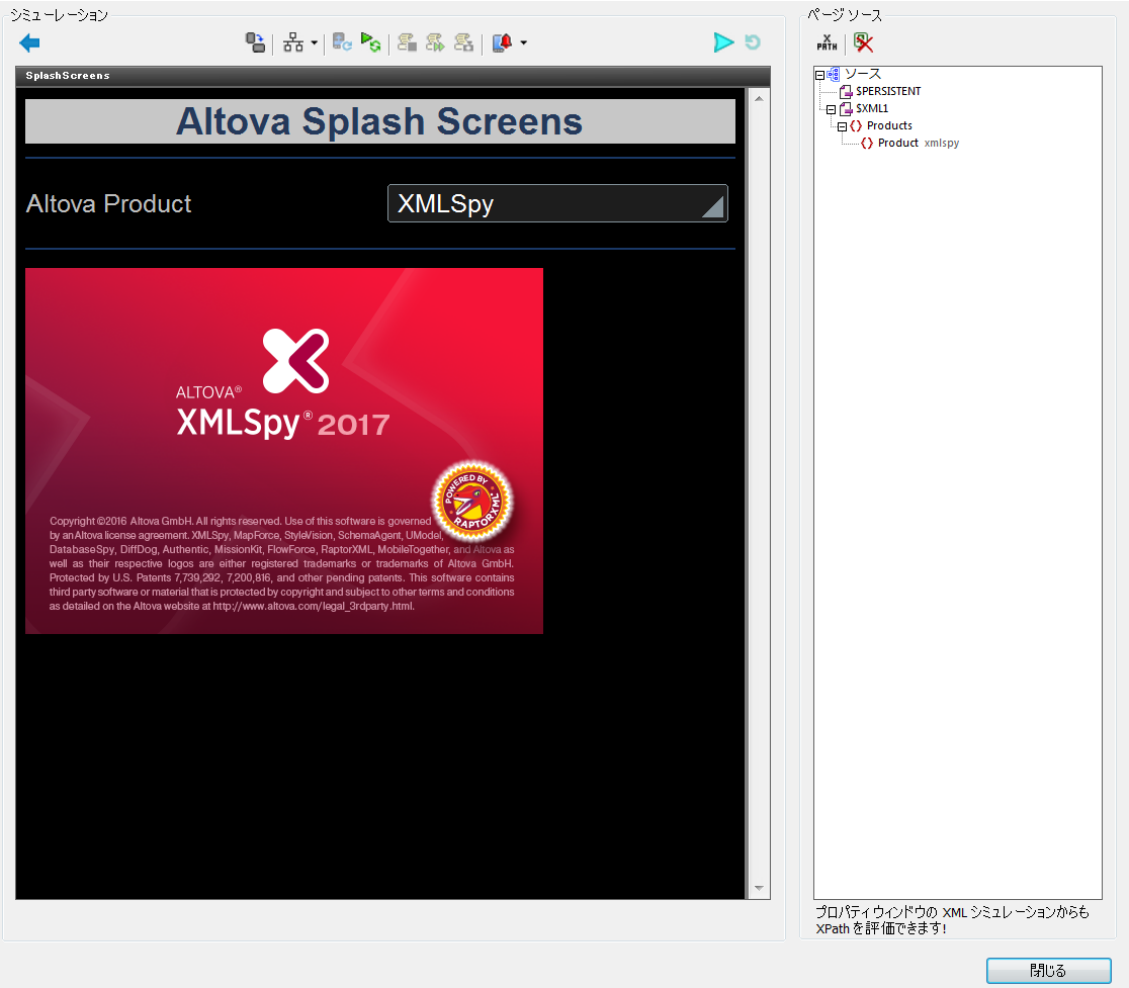
シミュレーションの実行

1. MobileTogether Server を開始します。実行するには 「MobileTogether Server ユーザー マニュアル」 を参照してください。
2. MobileTogether Server のWeb UI でソリューションの作業用ディレクトリを設定します。**[設定 | サーバー側ソリューション作業ディレクトリ]** *下のスクリーンショット参照*)。デザインのすべての相対パスはこの設定で指定されたディレクトリに相対して解決されます。サーバーでのシミュレーションが正しく作動するためには、参照されるファイルが保存されるディレクトリのパスを入力しなければなりません。



3. MobileTogether Designer でサーバー設定 が正確に設定されている。
4. MobileTogether Designer で **[プロジェクト | ワークフローのシミュレーションのためにサーバーを使用する]** を選択する。
5. サーバーにアクセスするために必要な資格情報を確認された場合、`user-name/password` に`root/root` の組み合わせを入力するか、サーバーでのシミュレーションを実行するために設定した特権のある他のユーザーの資格情報を入力します。ユーザーへの特権の割り当てに関しては 「MobileTogether Server ユーザー マニュアル」 を参照してください。

シミュレーションウィンドウが開かれワークフローが実行されます。



ファイルの場所

MobileTogether Server がシミュレーションに使用される場合、デザインにより参照されるファイルは、サーバー側ソリューション作業ディレクトリとして指定されているディレクトリかこのディレクトリの子孫ディレクトリである必要があります。(作業用ディレクトリの設定は MobileTogether Server 設定ページで行うことができます。)

- 絶対パスが使用される場合、ファイルは作業用ディレクトリまたは作業用ディレクトリの子孫ディレクトリに存在しなければなりません。
- 相対パスが使用される場合、パスは作業用ディレクトリに相対して解決されます。

シミュレーターの機能

[シミュレーター] ウィンドウには次の機能があります:

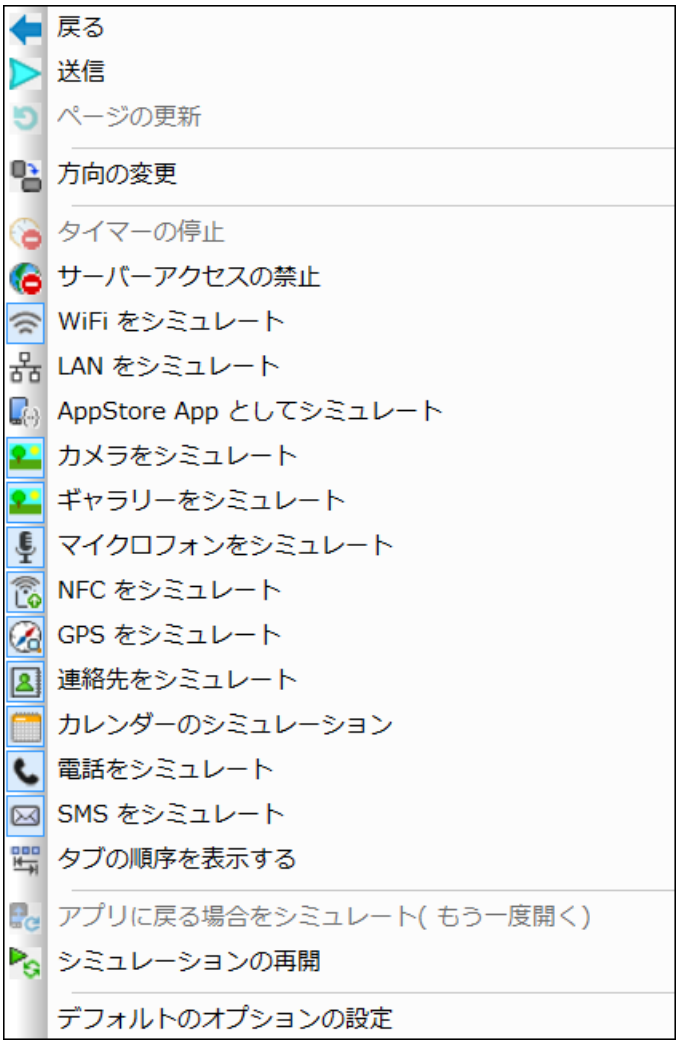
- 左側の (シミュレーション) ペインはシミュレーションを表示します。シミュレーションペインのためのオプションは下で説明されています。
- 右側の (ページソース) ペインはシミュレーションの展開中に変化するページソースを表示します。XML データペイ

ンのためのオプションは下で説明されています。

- シミュレーションの展開中 MobileTogether Designer GUI の[メッセージ] ペインは、発生しているすべてのアクティビティのステップ バイ ステップのレポートを作成します。ですから、ワークフローの展開の各ステップで何がおこっているのか確認することができます。これはデザインファイルのテストおよびデバッグを行うとても有益な機能です。
  - *デザイナーのシミュレーションがサーバーとクライアントの両方のアクティビティに関してレポートを行う*
  - *サーバーのシミュレーションはクライアントメッセージに関してのレポートを行う*
  - *クライアントのシミュレーションはサーバーメッセージに関してのレポートを行う*
- ユーザー インタラクションの有効化を必要とするコントロール。上のスクリーンショットでは、例えば、コンボ ボックスが有効化されています。

シミュレーションペインのオプション

[シミュレート] メニュー内とシミュレートツールバーアイコン でこれらのオプションを使用する事ができます (*下のスクリーンショット*) 。



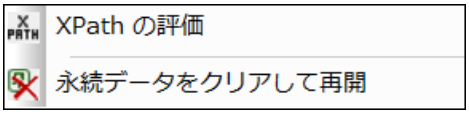
- *戻る* ページがサブページの場合、**[戻る]** をクリックするとサブページが閉じられます。ページがトップページの場合は、シミュレーターが閉じられます。**「戻る」ボタンのクリック時**も参照してください。
- *送信:* ページが最後のページではない場合、**[送信]** をクリックすると、次のページに移動します。ページが最後のページの場合は、ワークフローが終了されます。**送信ボタンのクリック時** も参照してください。

- *ページの更新:* 基になるデータが変更されると、**[ページの更新]** が有効化されます。**[ページの更新]** をクリックすると、ページが更新されたデータと共に更新されます。
- *方向の変更:* ビューを横長と縦長の方向に切り替えることができます。
- *タイマーの停止:* タイマーが間隔で実行するに設定されており、アクションが指定された間隔で実行されるように定義されている場合、**[タイマーの停止]** をクリックすることによりタイマー(結果的にアクション) を停止する事ができます。これによりこれらのアクションによるメッセージの多くをクリアする事ができ、他のメッセージやワークフローの他の部分を更に簡単に分析する事ができます。
- *サーバーアクセスの禁止:* サーバーへのアクセスを無効化し、サーバーへの接続エラーの状況下でのソリューションの振る舞いをテストする事ができます。クリックされると、ボタンは **[サーバーへのアクセスの有効化ボタン]** に切り替えられます。詳細に関してはサーバー接続エラーを参照してください。
- *WiFi が音の状態でシミュレート:* オンに切り替えられた場合、**`mt-connected-via-wifi`** は、XPath 拡張関数を`true()` に設定し、オフに切り替えられた場合は、`false()` を返します。これにより、シミュレータは WiFi へのアクセスが使用できる状態でのシミュレーションを行います。このように、WiFi へのアクセスを必要とするデザインのシナリオをシミュレートすることができます。
- *LAN がオンの状態でシミュレート:* オンに切り替えられた場合、**`mt-connected-via-lan`** は、XPath 拡張関数を`true()` に設定し、オフに切り替えられた場合は、`false()` を返します。これにより、シミュレータは LAN へのアクセスが使用できる状態でのシミュレーションを行います。このように、LAN へのアクセスを必要とするデザインのシナリオをシミュレートすることができます。
- *AppStore App としてシミュレート:* 有効化されると、静的グローバル変数 **`MT_IsAppStoreApp`** を`true()` に設定し、無効化されると`false()` に設定します。これによりシミュレーションはこの変数の値の条件により実行されます。
- *カメラがオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスのカメラ機能が使用できるモノとして振る舞います。これによりカメラ機能を必要とするデザインシナリオをシミュレートすることができます。
- *ギャラリーがオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスのギャラリー機能が使用できるモノとして振る舞います。これによりギャラリーへのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *マイクがオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスのマイクロフォン機能が使用できるモノとして振る舞います。これによりマイクへのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *NFC がオンの状態をシミュレートする:* オンに切り替えられると、**`mt-nfc-started`** XPath 拡張関数が`true()` に設定され、オフに切り替えられると、`false()` に設定されます。NFC が有効化され、NFC アクションを実行することができることをシミュレーターに示します。実際のNFC データはNFC サンプルファイルを介してシミュレーターに提供されます。
- *GPS がオンの状態をシミュレートする:* オンに切り替えられると、**`mt-geolocation-started`** XPath 拡張関数が`true()` に設定され、オフに切り替えられると、`false()` に設定されます。シミュレーターがデバイスの GPS ロケーター開始の通知を受け取ると、位置情報機能が使用できるものとしてソリューションを実行します。シミュレーションのために位置情報を提供するために位置情報 XML ファイルを使用してダミーの位置情報を提供することができます。
- *連絡先がオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスのアドレス帳機能が使用できるモノとして振る舞います。これによりアドレス帳へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。連絡先はサンプルファイルまたはMicrosoft Outlook 連絡先からシミュレートされます。オプションダイアログのシミュレーションタブ内で使用するオプションを指定することができます。
- *カレンダーのシミュレーション:* オンに切り替えられると、シミュレーターはデヴァイスのカレンダーが使用できるかのように振る舞います。これにより、カレンダーへのアクセスを必要とするデザインのシナリオをシミュレートすることができます。カレンダーはサンプルファイルまたはMicrosoft Outlook 連絡先からシミュレートされます。オプションダイアログのシミュレーションタブ内で使用するオプションを指定することができます。
- *電話がオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスの電話機能が使用できるモノとして振る舞います。これにより電話へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *SMS がオンの状態をシミュレートする:* オンに切り替えられると、シミュレーターはデバイスの SMS 機能が使用できるモノとして振る舞います。これによりSMS へのアクセス機能を必要とするデザインシナリオをシミュレートすることができます。
- *タブの順序の表示:* オンに切り替えられていると、タブ順序のシーケンスの一部になるように定義されているコントロールは、番号を含む青い丸でマークされています。番号はタブ順序シーケンス内のコントロールの位置を示してい

す。

- *アプリに戻る場合をシミュレート(もう一度開く)*: ページが開かれるとページの更新が行われるように定義されると有効化されます。シミュレーション中にページが更新されます。
- *シミュレーションの再開*: シミュレーションを再開します。
- *デフォルトオプションの設定*: シミュレーションペインのオプションをデフォルトの設定にリセットします。

ページソースペインオプションと機能
これらのオプションは**[ページソース]** メニュー内とページソースツールバーで使用する事ができます。



- **[XPath の評価]** をクリックすると、XPath 式を評価することのできるXPath/XQuery 式の編集ダイアログを開きます。XPath 式はシミュレーションの実行中、[スタイル & プロパティ] ペインからも編集することができます。
- **[永続データをクリアして再開する]** をクリックすると、永続データをクリアして、シミュレーションを再開することができます。
- ページソースペインでは、ツリーの一部をコピーしてツリーの他の箇所に貼り付けることができます。この機能は、DB 記録などのデータをコピーして、シミュレーションに更なるデータを追加するために役にたちます。コピーされたノードはシミュレーションの間のみ使用することができます。
- ページソースのノードを右クリックすると、そのノードのXPath ロケーター式を **「XPath をコピーする」** コンテキストメニューコマンドを選択して、クリップボードにコピーすることができます。

メモ サーバーに接続する際に問題が生じる場合、サーバーの設定を確認してください。手順については「MobileTogether Server ユーザー マニュアル」を参照してください。

シミュレーター内のページソースの XML ツリーを編集する

シミュレーター内のツリーは、多種のページソースのXML データを表示し、シミュレーションが展開中にこれらの値がどのように変化するかを表示します。シミュレーター内で切り取り/コピー/貼り付け/削除とドラッグアンドドロップを使用して、XML ツリーを直接編集します。(XML ツリーのコンテキストメニュー内の編集コマンドを使用することができます。) シミュレーションペインは即時変更されたデータを表示します。この機能により、異なる手動で入力されたデータを含む変更されたXML データ構造と共にソリューションをテストします。結果、ソリューションはより徹底してテストされ、テストはシミュレーション内で素早く実行することができます。

シミュレーター内のXML ツリーのコンテキストメニューは、以下の機能を提供します:

- **XML のロード**: (XML ツリーと同じ構造と要素を持つ) 外部 XML ファイルをXML ツリーにロードします
- **XML の保存**: XML ツリーの構造とデータを希望する場所に保存します。
- **XMLSpy 内での表示**: Altova XMLSpy プログラム内でXML ツリーを開きます。
- **このツリーをベースにした $XML 構造**: ページソースの構造をシミュレーター内のXML ツリーの構造に上書きします。

サーバー IP アドレスとネットワーク ファイアーウォールの設定
使用中のサーバーは(インターネットを介してアクセスすることのできる) 公開されたIP アドレス、および/または(例えば、企業内ネットワーク内のWiFi など、プライベートのネットワーク内でアクセスすることのできる) プライベートのIP アドレスを持つことができます。もし、モバイルクライアントデバイスがインターネットにサーバーのプライベートのIP アドレスを使用して、アクセスした場合、アクセスはうまくいきません。これは、プライベートのIP アドレスは、インターネット上では知

られておらず、解決することができないからです。クライアントデバイスが、プライベートのIP アドレスを使用する場合、クライアントデバイスはプライベートネットワークへのアクセスを既に取得している必要があります。

サーバーへのアクセスを確実にするために、以下を行ってください：

- インターネットを介してアクセスできるように、サーバーに公開されたIP アドレスを与えます。クライアントデバイスでは、この公開されたIP アドレスを使用してサーバーにアクセスします。
- ファイアーウォールを使用しており、MobileTogether Server をサーバーにプライベートIP アドレスを使用して（プライベートのネットワーク内に）インストールする場合、ネットワークのファイアーウォールを使用して、公開されたIP アドレス/ポートの組み合わせに送信されたリクエストを使用中のMobileTogether Server サーバーに転送します。クライアントデバイスでは、公開されたIP アドレスを使用してください。

MobileTogether Client 通信のために使用されるサーバーポートが使用されてサーバーへのアクセスが許可されるようにファイアーウォールが構成されていることを確認してください。MobileTogether Server で使用されるポートは、MobileTogether Server のWeb UI の設定ページで指定することができます。(*MobileTogether Server ユーザーマニュアルを参照してください*)。クライアントデバイスでは、これはアクセスするサーバーポートとして指定されている必要があります。

ヒント: ポート**80** はデフォルトで通常ほとんどのファイアーウォールに対して開かれています。ですから、ファイアーウォールの設定で問題があり、ポート**80** が他のサービスにバインドされていない場合、ポート**80** をクライアントとの通信用のMobileTogether Server ポートとして指定することができます。

15.3 クライアントでのシミュレーション

このシミュレーションメソッドでは、MobileTogether Designer をサーバーとして使用し、モバイルデバイスでワークフローを実行します。[メッセージ] ペインにサーバーメッセージをレポートします。クライアントでのシミュレーションでは、ワイアレスLAN で PC に接続できることを前提としています。以下の手順でクライアント(クライアント上でのシミュレーション) でトライアルランを行ってください:

1. クライアント: 新しいサーバーの設定
   1. MobileTogether をクライアントで開始して、**[設定 | サーバー]** アイコンをクリックします。
   2. 新しいサーバーを追加して、すべての関連するデータを入力します。(サーバーアドレス、ポート番号、(`8083` はクライアントへのアクセスのデフォルトのサーバーポートです)、ユーザー名およびパスワード)。
   3. サーバー設定を保存します。

2. PC: シミュレーションの開始
   1. MobileTogether Server がPC で実行されていて、デザイナーと同じポートを使用される場合、サービスを停止します。
   2. MobileTogether Designer で **[ツール | オプション]** を選択して、[クライアントのトライアルラン] タブをクリックし、クライアントが使用するポート番号を入力します(`8083` がデフォルトです)。
   3. クライアントでテストする(`.mtd`) ファイルを開きます。
   4. **[プロジェクト | クライアント上のトライアルラン]** を選択します。これによりページソースとデータ構造を表示するダイアログボックスが開かれます。

3. クライアント: シミュレーションの実行
   1. まだ開始されていない場合、モバイルデバイスでMobileTogether Client アプリを開始します。 PC で現在開かれているデザインファイルのみがこの種類のシミュレーションを使用するモバイルクライアントで(ソリューションとして) 閲覧することができます。
   2. PC でソリューションを開始するか確認するプロンプトが表示されます。**[はい]** をクリックします PC のダイアログボックスにツリー構造が表示されます。少しすると、ソリューションがクライアントで実行されます。
   3. **[前へ戻る]** をクリックして現在のソリューションを中止することができます。ソリューションの中止を確認するプロンプトが表示されます。**[はい]** をクリックします。

メモ クライアントでのシミュレーションで、デザインファイルは1 つのクライアントでのみで使用できます。

シミュレーションする言語

デザイナー とサーバー のためのシミュレーションする言語は **[プロジェクト | シミュレーションする言語]** コマンドで選択します。クライアント シミュレーション の言語は、シミュレーションが実行されるモバイルデバイスの言語と同じです。

シミュレーター内のページソースの XML ツリーを編集する

シミュレーター内のツリーは、多種のページソースのXML データを表示し、シミュレーションが展開中にこれらの値がどのように変化するかを表示します。シミュレーター内で切り取り/コピー/貼り付け/削除とドラッグアンドドロップを使用して、XML ツリーを直接編集します。(XML ツリーのコンテキストメニュー内の編集コマンドを使用することができます。) シミュレーションペインは即時変更されたデータを表示します。この機能により、異なる手動で入力されたデータを含む変更されたXML データ構造と共にソリューションをテストします。結果、ソリューションはより徹底してテストされ、テストはシミュレーション内で素早く実行することができます。

シミュレーター内のXML ツリーのコンテキストメニューは、以下の機能を提供します:

- **XML のロード**: (XML ツリーと同じ構造と要素を持つ) 外部 XML ファイルを XML ツリーにロードします
- **XML の保存**: XML ツリーの構造とデータを希望する場所に保存します。
- **XMLSpy 内での表示**: Altova XMLSpy プログラム内で XML ツリーを開きます。
- **このツリーをベースにした $XML 構造**: ページソースの構造をシミュレーター内の XML ツリーの構造に上書きします。

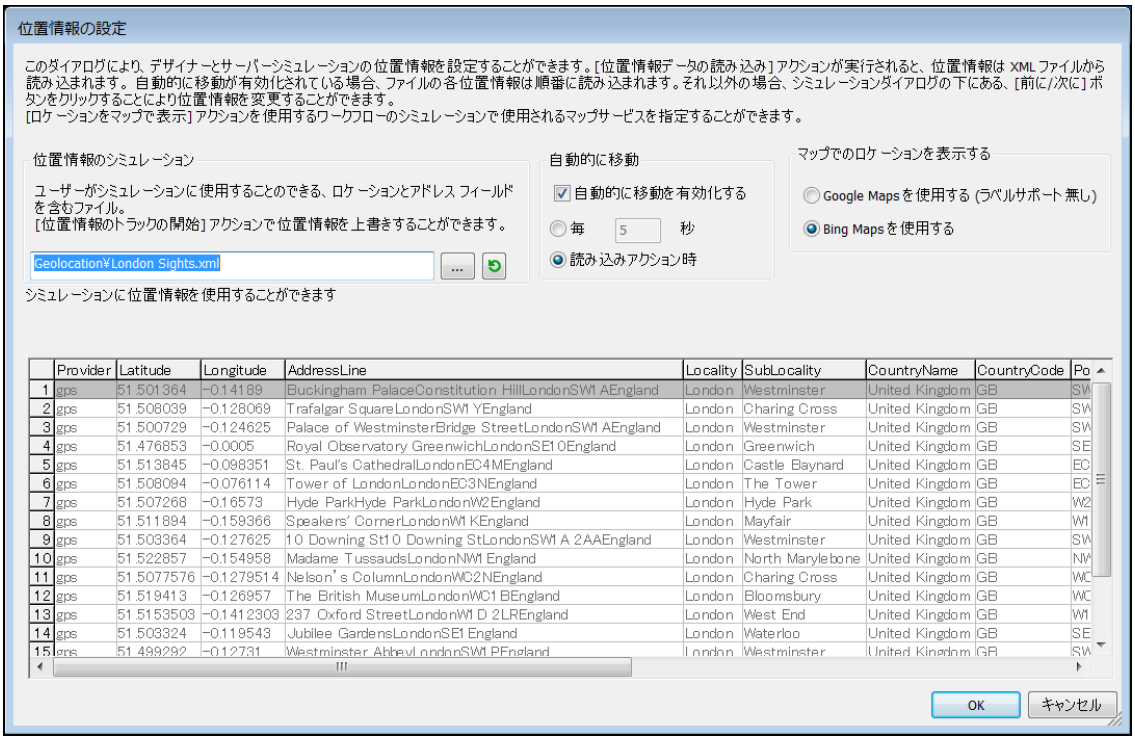
15.4 位置情報の設定

位置情報設定ダイアログ (*下のスクリーンショット*) により、デザイナーとサーバーシミュレーションのためのXML ファイルから位置情報を指定することができます。シミュレーションはモバイルデバイスを使用せず、使用できる位置情報がないため、位置情報はこの方法で指定される必要があります。

位置情報設定ダイアログはシミュレーターダイアログからアクセスすることができます:

1. **[プロジェクト | ワークフローのシミュレーション (F5)]** または **[プロジェクト | ワークフローのシミュレーションにサーバーを使用 (Ctrl+F5)]** をクリックして、シミュレーターダイアログにアクセスします。
2. シミュレーションダイアログ内で、左下にある **[位置情報]** ボタンをクリックします (*下のスクリーンショット*)。ここで指定された設定はデザイナーおよびサーバーのシミュレーション両方に使用されます。

メモ 位置情報アクションがデザインで使用されない場合、**[位置情報]** ボタンは**表示されません**。



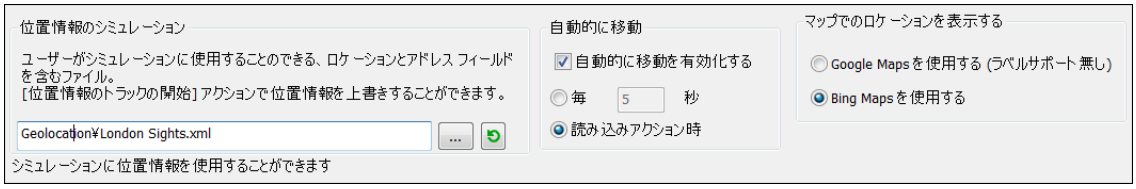
位置情報の設定

以下の位置情報の設定を行うことができます:

- *位置情報 XML ファイル:* シミュレーションのためのデフォルトの位置情報を含みます。XML ファイルは下にリストされる構造を持つ必要があります。**[参照]** ボタンにより、ファイルを参照することができます。**[更新]** ボタンにより XML ファイルの位置情報データを読み込み、ダイアログの下の部分の位置情報値ペインに入力することができます。デフォルトのファイルは、位置情報のトラッキングの開始 アクションにより指定された位置情報シミュレーションファイルによりオーバーライドされます。
- *自動的に移動:* 自動的に移動が有効化されていると、XML ファイル内の各位置情報ファイルは順番にシミュレーション中に読み込まれます。各位置情報の読み込みの間隔を指定することができます。自動的に移動が指定されていない場合シミュレーションダイアログの左下の、**[前に]** または **[次に]** ボタンをクリックして、シミュレーション中に

位置情報を変更することができます。これらの位置情報の値はシミュレータのみで実行することができます。これらの値は、デザイン内でアクションが明示的に指定されている場合、($GEOLOCTION ツリーを含む) データソースツリーノードにパスされます。

- *マップでロケーションを表示する* マップでの位置情報の表示アクションが実行されると、どのマップアプリケーションが Web ブラウザーで開かれるかを選択します。



位置情報 XML ファイル構造

MobileTogether Designer が、正確に位置情報を読み込むために、XML ファイルは下にリストされている構造に類似している構造を持つ必要があります。属性は省略または属性の値は空の文字列である場合があります。しかし、`//Location/Latitude` と`//Location/Longitude` 属性は存在しなくてはなりません。

位置情報 XML ファイルのサンプル構造

このサンプルは展開された**Geolocation** 要素を表示しています。**Location** と**Address** 要素の全ての属性が使用されているわけではありません。属性の完全なリストは次のリストを参照してください。緯度と経度の値の構文フォーマットに関しては、下の*位置情報入力文字列フォーマット*のセクションを参照してください。

```
<Root>
  <Geolocations>
    <Geolocation name="Buckingham Palace">
      <Location
        Latitude="51.501364"
        Longitude="-0.14189"
        Provider="gps"
      />
      <Address
        AddressLines="Buckingham Palace
        Constitution Hill
        London
        SW1A
        England"
        Locality="London"
        SubLocality="Westminster"
        CountryName="United Kingdom"
        CountryCode="GB"
        PostalCode="SW1A"
        AdminArea="England"
        SubAdminArea="London"
        FeatureName="Buckingham Palace"
        Thoroughfare="Constitution Hill"
      />
      </Geolocation>
      <Geolocation/>
       ...
      <Geolocation/>
  </Geolocations>
</Root>
```

位置情報 XML ファイルの全ての属性

```
<Root>
```

```
<Geolocations>
  <Geolocation name="">
    <Location
     AccuracyHorizontal=""
     AccuracyVertical=""
     Altitude=""
     Latitude=""
     Longitude=""
     MagneticHeading=""
     Provider=""
     Speed=""
     Time=""
    />
    <Address
     AddressLines=""
     AdminArea=""
     CountryCode=""
     CountryName=""
     FeatureName=""
     Locality=""
     Phone=""
     PostalCode=""
     Premises=""
     SubAdminArea=""
     SubLocality=""
     SubThoroughfare=""
     Thoroughfare=""
     Url=""
    />
  </Geolocation>
</Geolocations>
</Root>
```

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は `+90` から `-90` (北 から南) です。経度の値の範囲は `+180` から `-180` (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(**"**) でハイライトされており、エスケープした単位インジケーターは青い(**""**) でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(`N/S`, `W/E`)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(`+/-`); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(`N/S`, `W/E`)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(`+/-`); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:

```
33.33N  -22°44'66.66"
33.33  22°44'66.66"W
33.33  22.c
```

15.5 NFC サンプルファイル

NFC タグの検出をシミュレートする場合、(デスクトップマシンがのためにNFC が有効化されていないため) NFC タグの代わりにNFC サンプルファイルを使用する必要があります。シミュレーションペインの下の **「NFC サンプル」** ボタンをクリックして、NFC サンプルファイルを参照します。NFC サンプルファイルは、以下に表示される構造が必要です。

NFC サンプルファイルのためのテンプレート

```
<Root>
  <NFCS>
    <NFC name="Text" tooltip="sends a wellknown text 'This is my text">
          <Root>
             <Tag Id=""/>
             <NdefMessage
                CanMakeReadOnly=""
                IsWriteable=""
                MaxSize=""
                Type="">
                <NdefRecord
                   Id=""
                   TypeNameField=""
                   RecordTypeDefinition=""
                   Type=""
                   Text=""
                   Language=""
                   URI=""
                   Payload=""
                   MimeType=""
                   ExternalDomain=""
                   ExternalPackageName="">
                   <NdefRecord />
                </NdefRecord>
                <NdefRecord />
                    ...
                <NdefRecord />
             </NdefMessage>
          </Root>
    </NFC>
    <NFC/>
     …
    <NFC/>
  </NFCS>
</Root>
```

上のリストに表示されているとおり、各要素は複数のレコード(**NdefRecord** 要素)を含む単一メッセージ (**NdefMessage** 要素)に対応しています。

各サンプルファイルは複数のメッセージを持つことができます。NFC サンプルファイルがシミュレーションで使用するために選択されると、現在選択されているメッセージのインデックスがシミュレーターに表示されます (*下のスクリーンショット参照*)。更に、メッセージ番号にカーソルをポイントすると、ヒントが表示されます (*スクリーンショットを参照してください*)。このヒントは、メッセージの **NFC/@tooltip** 属性の値です。新規のメッセージを現在のメッセージとして設定するには、「前へ」と「次へ」ボタンを使用してください (*スクリーンショットを参照してください*)。

NFC サンプルファイルからメッセージデータを読み取る

NFC サンプルファイルから特定のメッセージを `$MT_NFC` ツリーに読み取るには、以下をおこないます:

1. (NFC を開始する アクションをトリガーして) NFC を開始します。
2. **NFC サンプル** をクリックして、ファイルを参照して NFC サンプルファイルを指定します。
3. NFC サンプルファイルから読み取るメッセージの番号が現在表示されているメッセージ番号 (*上のスクリーンショット、番号 2*) であることを確認してください。
4. このメッセージ番号をクリックして、ファイル内のメッセージデータが読み取られ、`$MT_NFC` ツリーにロードされます。

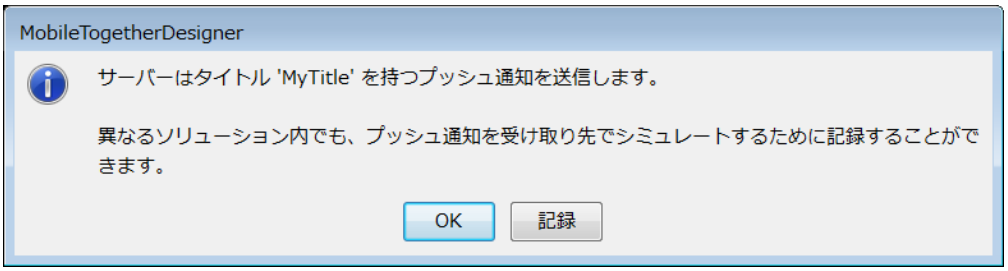
15.6 プッシュ通知のシミュレーション

プッシュ通知 (PN) には以下に関連下データが含まれています: (i) プッシュ通知の短いメッセージ。(ii) プッシュ通知の長いメッセージ。 (iii) プッシュ通知のペイロード。ソリューションの送信と受信は1つで同じソリューションです。シミュレーション内での送信と受信間のデータの転送は1つのシミュレーション内で行われます。この場合のシミュレーションはわかりやすいです。

しかしながら、ソリューションの送信と受信が異なる場合、シミュレーションのメカニズムは以下のとおりです: ソリューションの送信のシミュレーション中に送信されるプッシュ通知データはMT プッシュ通知シミュレーションファイル (`.mtpnsim` 拡張子を持つ) に記録されます。ソリューションの受信がシミュレートされる場合、`.mtpnsim` ファイルをロードすることができます。シミュレーターは `.mtpnsim` ファイル内にプッシュ通知データのセットを表示します。また、シミュレートすることを希望するプッシュ通知を選択することができます。

プッシュ通知シミュレーションデータの記録

ソリューションの送信をシミュレート中にプッシュ通知を送信するイベントをトリガーする場合、下に表示されるダイアログが表示されます。

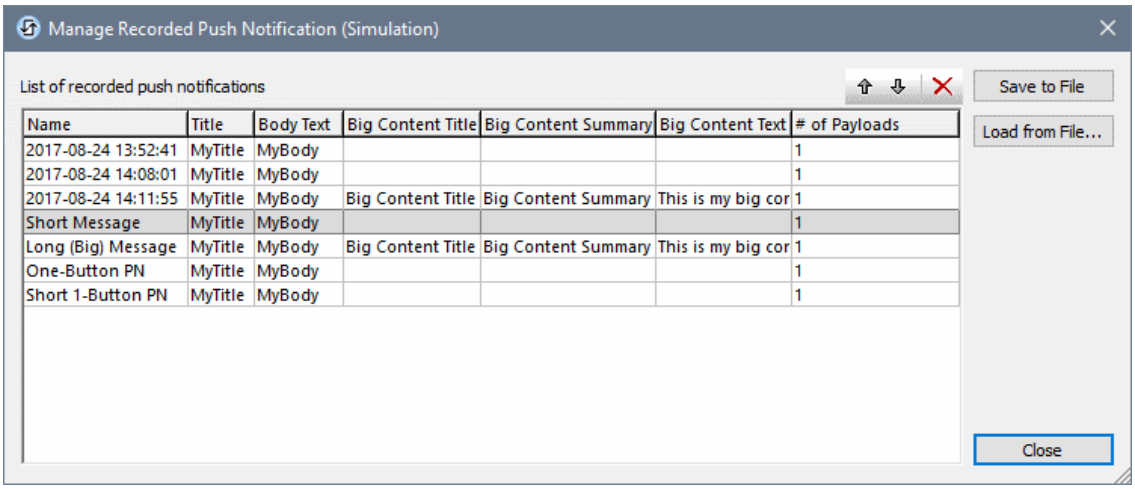


メモリにプッシュ通知データを記録するには、 **「記録」**をクリックします。メモリに複数のプッシュ通知をこの方法で記録することができます。シミュレーションを閉じる場合、記録されたプッシュ通知はメモリ内に存在しますが、ファイルに保存されていないことに注意してください。

ソリューションファイルを保存、または、閉じる場合、メモリ内に保存されていない記録されたプッシュ通知データが存在することを通知され、このデータをファイルに保存するかどうかが問われます。 **「はい」**を選択すると、メモリ内の記録されたプッシュ通知データの異なるセットがソリューションとして同じフォルダー内のMT プッシュ通知シミュレーションファイルに保存されます。このファイルは次のパターンで名前が付けられます: `YourSolutionName.mtpnsim`。追加のプッシュ通知は現在の、または、シミュレーションの次のシミュレーション中に同じファイル内に保存されます。ファイル内のプッシュ通知データのセットはプッシュ通知の記録日時である名前と共に識別されます。

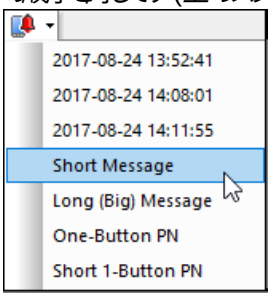
シミュレーターに記録されたシミュレーションデータをロードする

記録されたプッシュ通知シミュレーションデータをMT プッシュ通知シミュレーションファイル (`.mtpnsim` ファイル) からロードするには、ソリューションの受信 (MTD ファイル) を開始し、シミュレーターのツールバー内の **プッシュ通知** アイコン を クリックします。記録されたプッシュ通知 (シミュレーション) の管理ダイアログが表示されます(*下のスクリーンショットを参照*)。 **「ファイルからロードする」**をクリックしロードする `.mtpnsim` ファイルを参照し、 **開く**をクリックします。このファイル内の記録されたプッシュ通知データはダイアログとメモリ内にロードされます (*スクリーンショットを参照*) 。 MTD ファイルに自動的に保存されないことに注意してください。MTD ファイルを保存せずに再ロードした場合、記録されたプッシュ通知データを `.mtpnsim` ファイルから再ロードする必要があります。 **「ファイルに保存する」**をクリックして、MTD ファイルに記録されたプッシュ通知データを保存します。これにより `.mtpnsim` ファイルに再ロードすることを回避できます。プッシュ通知データを他の `.mtpnsim` ファイルからロードした場合、メモリ内のプッシュ通知データを新しいデータが上書きします。MTD ファイル内の記録されたプッシュ通知データを上書きするには、このダイアログ内の **「ファイルに保存する」**をクリックします。



Name	Title	Body Text	Big Content Title	Big Content Summary	Big Content Text	# of Payloads
2017-08-24 13:52:41	MyTitle	MyBody				1
2017-08-24 14:08:01	MyTitle	MyBody				1
2017-08-24 14:11:55	MyTitle	MyBody	Big Content Title	Big Content Summary	This is my big cor	1
Short Message	MyTitle	MyBody				1
Long (Big) Message	MyTitle	MyBody	Big Content Title	Big Content Summary	This is my big cor	1
One-Button PN	MyTitle	MyBody				1
Short 1-Button PN	MyTitle	MyBody				1

記録されたプッシュ通知 (シミュレーション) の管理ダイアログ内で (*上のスクリーンショット*)、各記録されたプッシュ通知は個別のラインに名前、短いメッセージデータ 長いメッセージデータ ペイロードの情報別に表示されています。プッシュ通知の順序を(右上の) **「上に移動」**と**「下に移動」**ツールバーアイコンをクリックすることにより変更することができます。 **「削除」**ツールバーアイコンをクリックしてプッシュ通知を削除することもできます。簡単に識別するためにプッシュ通知の名前を編集することもできます。これを行うには、名前をダブルクリックして編集します。ダイアログ内での変更はメモリ内で行われます。MTD ファイルに変更を保存するには、**ファイル「ファイルに保存する」**をクリックします。
シミュレーション内で使用されているプッシュ通知を選択するには、シミュレーターツールバー内の **プッシュ通知** アイコン のドロップダウン矢印をクリックします。これにより現在メモリ内に存在する全てのプッシュ通知のリストが表示されます (*下のスクリーンショットを参照*)。プッシュ通知の表示の順番は記録されたプッシュ通知 (シミュレーション) ダイアログの現在の順序と同じです (*上のスクリーンショットと下のスクリーンショットを比較してください*)。



このリストからプッシュ通知を選択すると、選択されたプッシュ通知が受信された状態をソリューションはシミュレートします。他のプッシュ通知の受信をシミュレートするには、ドロップダウンリストから新規のプッシュ通知を選択してください。

15.7 連絡先のサンプルファイル

デバイスのアドレス帳をシミュレートするには、下にリストされるXML 構造を持つ連絡先ファイルのサンプルを使用することができます。オプションダイアログのシミュレーションペイン内で、シミュレーションのために、連絡先の読み取りアクション、または、シミュレーションのためにMicrosoft Outlook の連絡先を使用することができます。どちらかの設定を使用するか指定する設定は、オプションダイアログのシミュレーションペイン内です。

サンプルアドレス帳のためのテンプレート

```
<?xml version="1.0" encoding="UTF-8"?>
<Root>
  <Contact Id="">
    <Name Prefix="" First="" Middle="" Last="" Suffix=""/>
    <Image Image=""/>
    <Address Description="" Country="" PostalCode="" City="" Street=""/>
    <JobInfo Title="" Company="" Department=""/>
    <Phone Description="" Number=""/>
    <Email Description="" Address=""/>
    <Website Description="" URL=""/>
    <Note Note=""/>
  </Contact>
  <Contact/>
    …
  <Contact/>
</Root>
```

15.8 カレンダーのサンプルファイル

デバイスのカレンダーをシミュレートするには、下のリストで示されるXML 構造を持つサンプルカレンダーファイルを使用することができます。 オプションダイアログのシミュレーションペイン内でカレンダーへのアクセスアクションを含むシミュレーションのためにこのファイルが使用されるように指定することができます。オプションダイアログのシミュレーションペイン内で使用される2つの選択肢を指定する設定です。

カレンダーファイルのためのテンプレート

```
<?xml version="1.0" encoding="UTF-8"?>
<Root>
  <Calendar Id="1" Name="Business">
    <Event Id="1" Title="Quarterly Meeting" Start="2018-04-04" End="2018-04-04" AllDay="1"
Location="Meeting Room 2">
      <Attendee Name="Bob" Status="Accepted" Type="Required" Relationship="Speaker"/>
    </Event>
    <Event Id="1" Title="New Customer Lunch" Start="2018-05-14T12:30:00" End="2018-05-
14T14:00:00" Location="Sushi Restaurant">
      <Attendee Name="Alice" Status="Accepted" Type="Optional" Relationship="Attendee"/>
    </Event>
  </Calendar>
  <Calendar Id="2" Name="Private">
    <Event Id="1" Title="Family Dinner" Start="2018-05-18T19:00:00" End="2018-05-18T23:00:00"
Location="Home"/>
    <Event Id="2" Title="Summer Vacation" Start="2018-07-09" End="2018-07-22" AllDay="1"
Location="Home"/>
  </Calendar>
</Root>
```

メモ: Microsoft Outlook カレンダーでは、参加者のステータスはイベントの企画者のみが閲覧することができます。イベントを企画しない限り、参加者のステータスを確認することができないことに注意してください。

15.9 サービストリガーのシミュレーション

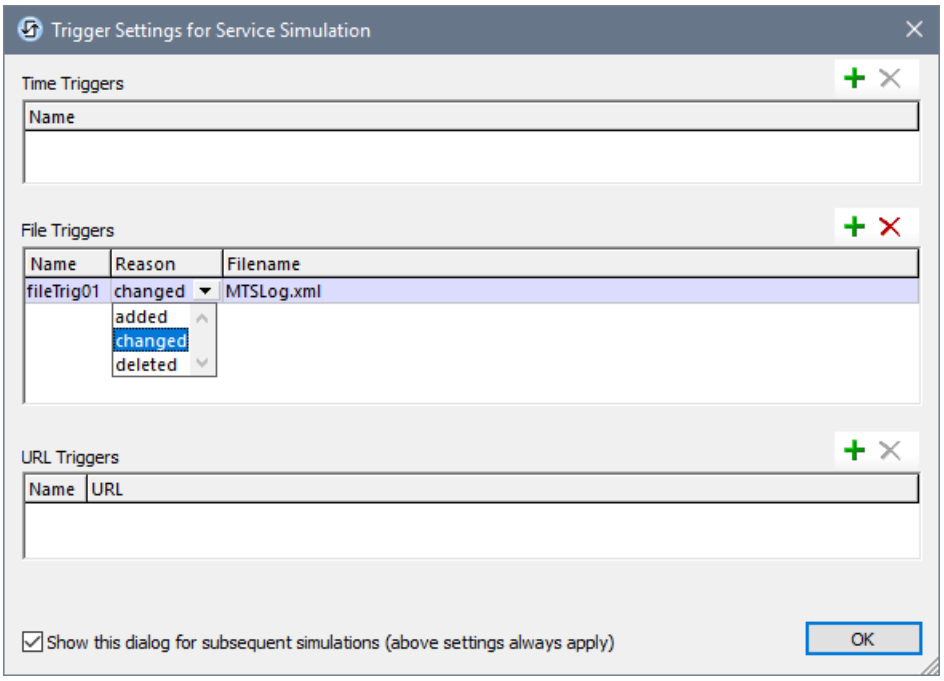
サーバーサービスがシミュレートされると、サービスのために定義されているアクションが実行されます。**$MT_SERVICE** ページソースツリーをアクションが使用する場合、このシミュレーションのために、ツリー内にデータを手動で作成する必要があります。これは、実際のしなりをでは、**$MT_SERVICE** ページソース内でデータはランタイムにサービスのトリガー情報から生成されます。データはシミュレーション中には生成されません。

$MT_SERVICES ページソースの構造

```
<Root>
   <Triggers>
      <File name="" filename="" reason=""/>
      <URL name="" url=""/>
      <Timer name=""/>
   </Triggers>
</Root>
```

$MT_SERVICE ツリー内のデータを以下のようにシミュレーションのために作成することができます:

1. 通常通りシミュレーションを開始します: デザイナー または サーバー。トリガー設定のためのサービスのシミュレーションダイアログが表示されます。
2. このダイアログには、トリガーの各型のためのペインが含まれています。(*下のスクリーンショット内では、ファイルシステムトリガーのためのペインが表示されています*)。(**追加** アイコンをクリックして) シミュレートする個別のトリガーのためにエントリを入力します。(エントリを追加しない場合、**$MT_SERVICE** ページソースのノードはシミュレーション中は空白になります。サービスアクションの定義の XPath 式がページソースからノードにアクセスする場合、**$MT_SERVICE** ツリーにデータが含まれているかは関連しません)。
3. トリガーのアクションと共にシミュレートする値を入力します。各トリガーは **Name** 属性により識別されます。例えば、下のスクリーンショット内では、ファイルシステムトリガーのために3つの値を入力することができます。これらは **$MT_SERVICE** ツリーの **File** 要素の属性の値に伝達されます (*上の $MT_SERVICE ツリーの構造を参照してください*)。シミュレーションのためにサーバーを使用する 場合、トリガーのために入力する名前がサーバー上のトリガーの名前に一致することを確認してください。



4. ダイアログの表示無しにこれらの値を次のサービスのシミュレーションで使用するには、ダイアログの下にある *このダイ*

アログの表示 オプションのチェックを解除します。(シミュレーションの開始時にこのダイアログを表示するには、オプションダイアログのシミュレーションタブに移動し(**「ツール」 オプション」**)、*サービスのシミュレーション時にトリガーのためのダイアログを表示* オプションをチェックします。.

5. **「OK」**をクリックします。(*上記の*)サービスはダイアログ内で指定された**$MT_SERVICE** ページソースの値を使用して、シミュレートされます。

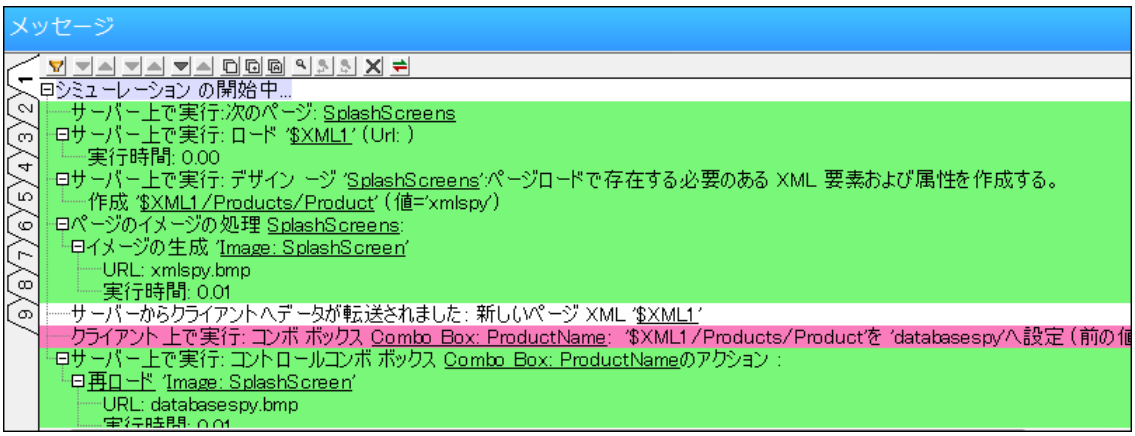
サービスの作成方法に関する説明に関しては、サーバーサービスのセクションを参照してください。

15.10 メッセージ ペイン

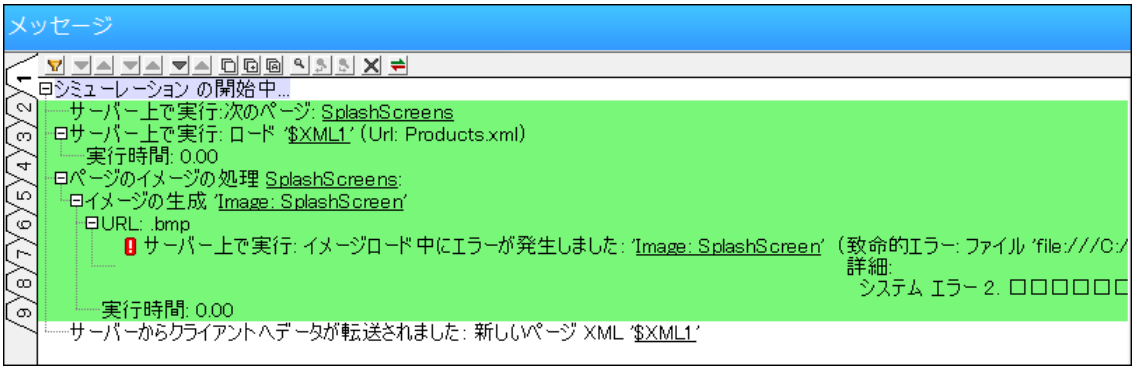
ソリューションの進行中に、MobileTogether Designer GUI の[メッセージ] ペインにソリューションに関連するすべてのアクティビティに関するステップ バイ ステップのレポートが表示されます。ワークフローの進行の各ステップで何が起こっているかを明確に確認することができます: サーバーで何が実行されているか、クライアントで何が起こっているか、アクションのフローの順序、また、アクションの実行の理由など。サーバーとクライアントアクションは異なる背景色で表示されます。エラーはフラッグされ、また警告およびトレースも表示されます。これらはすべて必須であり、デザインファイルをテストおよびデバッグする際にとても役に立ちます。

- デザイナーでのシミュレーションはサーバーとクライアントのアクティビティについてレポートします。
- サーバーでのシミュレーションはクライアントメッセージをレポートします。
- クライアントでのシミュレーションはサーバーメッセージをレポートします。

下のスクリーンショットはエラー無く終了したシミュレーションを表示しています。ワークフローの各ステップで、サーバーおよび/またはクライアントで起っているアクションの詳細についてのメッセージおよび異なる実行にかかった時間が表示されます。サーバーとクライアントのログメッセージは異なる背景色で表示されており、(下のスクリーンショットでは、それぞれ緑とピンクでハイライトされています)。多種のアクティビティがどこで起きているかを素早く確認することができます。(背景色は好みによって変更することができます)。[メッセージ] ペインのリンクから、デザインまたは[ページソース] ペインの関連するコンポーネントに移動することができます。メッセージの上にマウスをポイントすると、特定のワークフローのアクティビティについての追加の情報がポップアップします。シミュレーションが終わると、エラーと警告の要約が表示されます(下のスクリーンショットでハイライトされています)。



エラーが発生した場合、赤いアイコンでフラグが付けられます(*下のスクリーンショット参照*)。



詳細はユーザーインターフェイス> [メッセージ] ペインのセクションを参照してください。

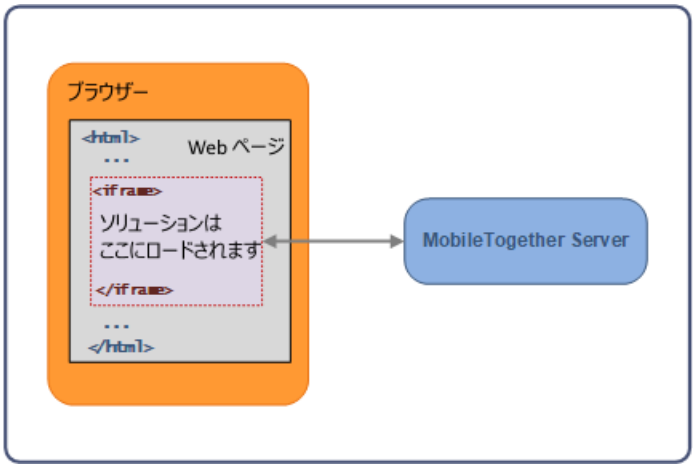
チャプター 16

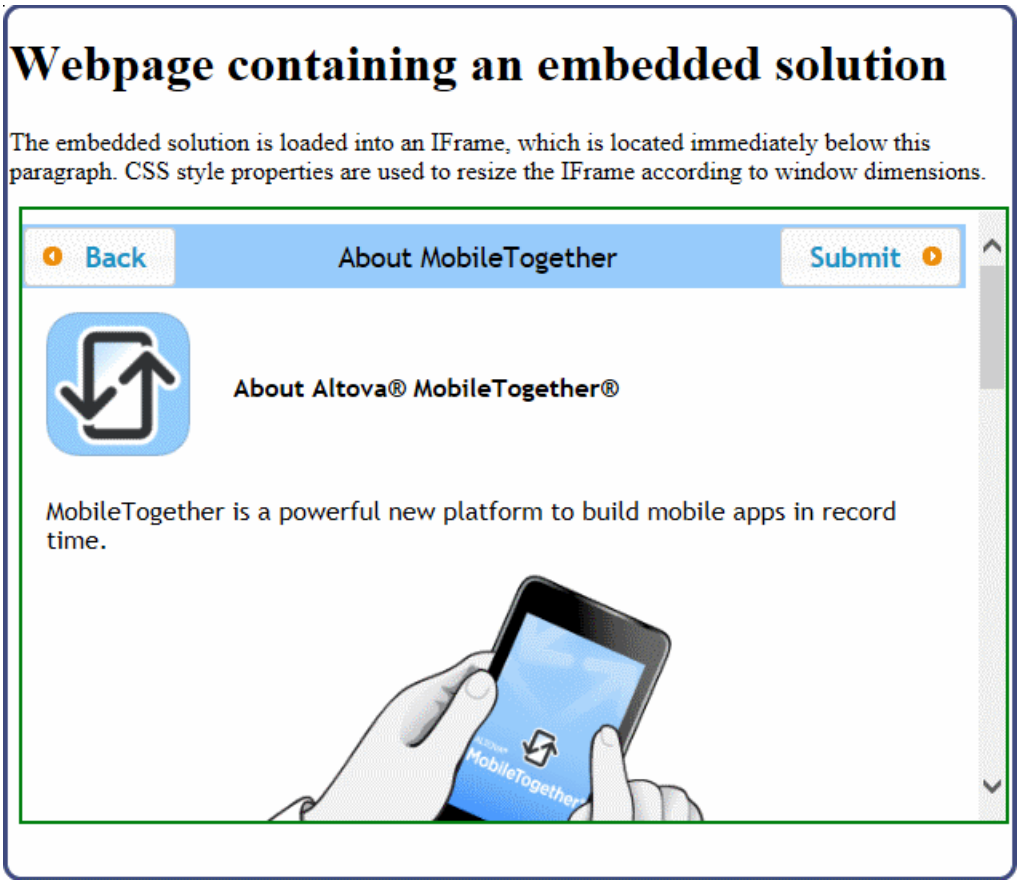
埋め込まれた Web ページソリューション

16 埋め込まれた Web ページソリューション

埋め込まれた Web ページソリューションは Web ページ内で埋め込まれるソリューションです。事実上、Web ページの HTML コードには、ソリューションがロードされる **IFrame** 要素が含まれています。Web ページと埋め込まれたソリューション間で、JavaScript を使用してデータを転送することができます。ソリューションは MobileTogether Server と通常通り相互作動し、JavaScript メカニズムを介して、Web ページに通信を返すことのできる MobileTogether Server からデータを受信します。

下の左下の図は、どのように埋め込まれたソリューションが埋め込まれた Web ページと MobileTogether Server と相互作用するかを示しています。右下のスクリーンショットは、埋め込まれたソリューションが含まれる(緑の枠で囲まれた) Web ページを表示しています。





このセクションは以下のように整理されています:

- Web ページにソリューションを埋め込む では、**iframe** 要素にソリューションをロードする方法。について説明されています。また、自身で試すための完全なHTML コードリスティングが含まれています。
- ページとソリューション間の通信: Web ページ内の埋め込まれたソリューションに加え、通信と異なるコンポーネント間の通信も有効化する必要があります。Web ページはIFrame 内のソリューションとの通信が可能である必要があります。JavaScript を使用して、メッセージをIFrame にポストし、メッセージからIFrameへ返すためリッスンします。IFrame 自身内では、データはソリューションとMobileTogether Server 間で通信されます。セクション ポスティング: Web ページからソリューションへ では、Web ページから、IFrame へ、MobileTogether Server への通信について説明されています。セクション リッスン: ソリューションからWeb ページへ は、MobileTogether Server からIFrame へ、Web ページへの通信の方法について説明しています。
- 認証: MobileTogether Server 上のワークフローにアクセスを試みる通信はすべて認証される必要があります。このセクションでは、埋め込まれたソリューションが使用することのできる認証の型について説明されています。JSON Web Tokens (JWT) を介した認証 は埋め込まれたソリューション特有の認証の種類です。これにより、これらのソリューションが既存のネットワークとシステムに手軽に統合されます。
- サンプル: このセクションには、埋め込まれたWeb ページソリューションを使用するHTML ページのコードリストが含まれており、Web ページとサーバー間の通信の詳細が説明されています。

役に立つデザインの仕組み

次のデザインの仕組みは、Web ページソリューション機能のための重要な機能のため重要な機能を提供します:

- ソリューションのページの**埋め込まれたメッセージ** ページイベントはWeb ページからメッセージをピックアップします。
- **$MT_EMBEDDEDMESSAGE** JSON ページソースは、取得されたデータを構造化された書式に保存します。
- 文字列にコード アクションは、シリアル化された文字列を解析し、ページソース内にシリアル化が解除された構造を配置します。JSON ノード内のXML 文字列のシリアル化を解除し、ページソースを作成する際に役に立ちます。
- 文字列に保存 アクションはページソースをシリアル化し、ページソース内に結果する文字列を他のページソースノードに配置します。
- 埋め込まれたメッセージをバックする アクションは、シリアル化されたJSON 文字列を現在のソリューションにコードしたIFrame に送信します。

(C) 2018 Altova GmbH

16.1 Web ページにソリューションを埋め込む

Web ページ内に1つ、または、複数のMobileTogether ソリューションを埋め込むことができます。各ソリューションは HTML **IFRAME** 要素内に埋め込まれています。 埋め込むには、以下を行ってください:

1. ソリューションを表示するWeb ページ内の場所にHTML **IFRAME** 要素を追加します(*下のサンプルを参照してください*)。
2. IFRAME の**src** 属性をMobileTogether Server 上で埋め込むためのソリューションのURL に設定します(*下のリスティングを参照してください*)。

```
<iframe src="http://localhost:8083/run?d=/public/my-mt-solution" frameborder="0"></iframe>
```

HTML ページがロードされると、その**IFRAME** 要素 はターゲットとされたソリューションをロードします。IFrame は MobileTogether Server にアクセスするため3つの認証に関連したシナリオを使用することができます:

- *匿名のアクセス*: MobileTogether Server 上でソリューションへの匿名のアクセスが有効化されている場合、IFrame 上でソリューションは直接表示されます。 ユーザーの認証は必要とされません。匿名のアクセスを設定するには、サーバーに対して以下を行う必要があります (i) 匿名ログインの許可 (*サーバー モバイルクライアントポートのための設定*を参照)、(ii) (*ワークフローのコンテナーのパーミッションを設定* を container=read と workflow=read,use の最小値に設定することで) ソリューションの匿名の使用を許可します。
- *ユーザーログイン*: 匿名のユーザーログインがサーバー上で有効化されていない場合、ユーザーは MobileTogether Server ログイン資格を提供するように問われます。ユーザーが*ターゲットとされたソリューションにアクセスするパーミッション*を有する場合、ソリューションMobileTogether Server からWeb ページの IFrame にダウンロードされます。
- *JWT 認証*: MobileTogether Server 認証システム外のシステム内でユーザーが認証され、JSON Web Tokens (JWT) を介して、MobileTogether 固有の認証を必要とせず、JWT MobileTogether Server に認証の情報がパスされます。認証はロードするためのターゲットとされたソリューションを有効化します。

サンプル: 埋め込まれたソリューションを含むシンプルな Web ページ

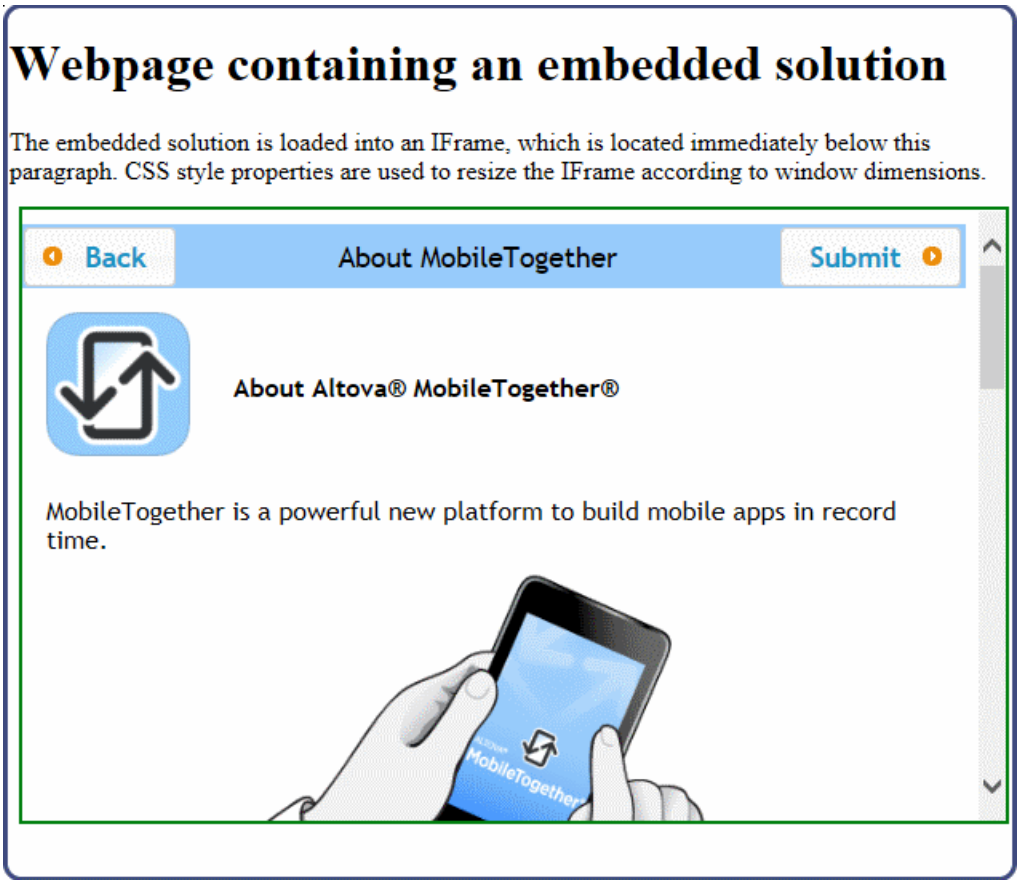
下にリストされるHTML コードは、ソリューションを埋め込むためにどのように**IFRAME** 要素が使用されるかを示しています。埋め込まれたソリューションを含むHTML ページのスクリーンショットがその下に表示されています。ソリューションは(緑の枠で)囲まれています。このリスティングで使用されるサンプルは About という名前のサンプルです。これは、MobileTogether Server 内にパッケージされており、サーバーの Public コンテナーにデフォルトで存在しています。 このソリューションをソリューション内に正確に埋め込むために、About ワークフローの*匿名のアクセスを許可*するようにサーバーをセットアップする必要があります。この機能を試すには、下のHTML コードをコピーし、ファイルに保存し、ファイルをブラウザーから開きます。

```
<!DOCTYPE html>
<html>
  <head>
    <title>Webpage containing an embedded solution</title>
  </head>
  <body>
    <h1>Webpage containing an embedded solution</h1>
    <p>The embedded solution is loaded into an IFrame, which is located
immediately below this paragraph.</p>
    <div class="resize">
      <iframe src="http://localhost:8083/run?d=/public/About"
frameborder="0"></iframe>
    </div>
  </body>
</html>
```

src 属性内の URL は、次のパターンを解決する必要があります: **http://**<serveraddress>**:**<serverport-for-client>**/run?d=/**<path-to-container>**/**<solution-name>

完全なリスティング (IFrame のサイズを調整するための CSS スタイルが含まれます)

```
<DOCTYPE html>
<html>
  <head>
    <title>Webpage containing an embedded solution</title>
    <style>
      .resize {
        position:relative;
        padding-bottom:56.25%;/* proportion value to aspect ratio 16:9 (9/16 = 0.5625 or 56.25%) */
        padding-top:30px;
        height:auto;        /* alternatively, try a value of 0 */
        overflow:hidden;
      }

      .resize iframe {
        position:absolute;
        top:0;
        left:0;
        width:100%;
        height:100%;
      }
    </style>
  </head>
  <body>
    <h1>Webpage containing an embedded solution</h1>
    <p>The embedded solution is loaded into an IFrame, which is located immediately below this paragraph.
CSS style properties are used to resize the IFrame according to window dimensions so that the IFrame's
aspect ratio is maintained at 16:9.</p>
    <div class="resize">
      <iframe src="http://localhost:8083/run?d=/public/About" frameborder="0"></iframe>
    </div>
  </body>
</html>
```


Webpage containing an embedded solution
The embedded solution is loaded into an IFrame, which is located immediately below this paragraph. CSS style properties are used to resize the IFrame according to window dimensions.
Back
About MobileTogether
Submit
About Altova® MobileTogether®
MobileTogether is a powerful new platform to build mobile apps in record time.

16.2 Web ページとサーバー間の通信

Web ページにソリューションが埋め込まれる理由のひとつは、Web ページと埋め込まれたソリューションの間でデータを交換し、MobileTogether Server を使用してソリューションが入力データを処理し、Web ページに結果を返すためにです。典型的なシナリオは以下のとおりです:

1. ユーザーは HTML Web ページのフォームにデータを入力します。
2. このデータは(Web ページの IFrame に現在ロードされている)ソリューションに通信されます。
3. ソリューションでは、通常データの処理が行われるソリューションの **MobileTogether Server 上のワークフロー**にデータが送信されます。
4. (i) ソリューションの一部として表示することのできる、および/または (ii) 表示のために、または、更なる処理のために Web ページに返す IFrame に結果が返されます。

メモ この機能の詳細では、用語 (IFrame 内に表示される) *ソリューション* と(サーバー上にデプロイされるデザイン) *ワークフロー* が区別されていることに注意してください。

段階は次の段階により構成されています: Web ページ?ソリューション?ワークフロー?ソリューション?Web ページ。 Web ページとワークフロー間でデータを送信するために使用されているメカニズムは、このセクションのサブセクションで説明されています:

- ポスティング: Web ページからサーバーへ
- リッスン: ソリューションから Web ページへ

データ転送メカニズム

Web ページとサーバー間のデータの転送は2つの段階で行われます: Web ページ?ソリューションとソリューション?ワークフロー。2つの段階は次のメカニズムをそれぞれ使用します:

Web ページ?ソリューション

Web ページとソリューション間の通信は JavaScript により行われます:

- **`Window.postMessage()`** メソッドは Web ページから埋め込まれた IFrame へデータを送信するために使用されます(メッセージは自動的にソリューションからワークフローに送信されます)。
- **`Window.addEventListener()`** メソッドは、ワークフローから IFrame に送信されるをリッスンする メッセージイベント のために Web ページ内で使用されます。IFrame によりメッセージが受信されると、JavaScript 関数がメッセージを処理し、Web ページ内で表示するために使用される Web ページに転送されます。

上記の2つのメソッドは、W3C 仕様です。詳細に関しては、次を参照してください: Mozilla Developer Network での説明 : **`PostMessage`** と **`AddEventListener`**。または、イベントリスナーはメッセージイベントをリッスンするので、次を参照してください: **`MessageEvent`**。

ソリューション?ワークフロー

ソリューションとワークフロー間の通信は、(**`$MT_EMBEDDEDMESSAGE`** という名前の) JSON ページソースからワークフロー内でデータがアクセスされるためです。JSON ページソースは JSON 構造から作成される必要があります。ソリューションは以下を行います : (i) IFrame 内で受信されるメッセージを JSON 文字列にシリアル化します。 (ii) シリアル化された JSON 文字列を(**`$MT_EMBEDDEDMESSAGE`** JSON ページソースとして作成することのできる) ワークフローに自動的に送信します。

(ワークフローからソリューションへ)逆方向に通信が発生する場合、JSON 文字列として送信されます (**`$MT_EMBEDDEDMESSAGE`** JSON ページソースと通常シリアル化されます)。

ポスティング: Web ページからサーバーへ

(Web ページに埋め込まれたIFrame を使用して) Web ページからサーバーへ送信されるメッセージは **埋め込まれたメッセージ** と呼ばれます:

1. Web ページからIFrame へ埋め込まれたメッセージを送信するためにJavaScript が使用されます。埋め込まれたメッセージは、`postMessage()` メソッドの最初のパラメーターであるJSON オブジェクトとして送信されます。
2. 埋め込まれたメッセージがIFrame に到達すると、ソリューションはメッセージをJSON 文字列にシリアル化し、MobileTogether Server 上のワークフローのソリューションにこの文字列を送信します。このステップは自動的に実行され、プロセスを指定する必要はありません。
3. ワークフロー内では、埋め込まれたメッセージは **埋め込まれたメッセージ** イベントによりピックアップすることができます。アクションがこのイベントのために指定されていると、`$MT_EMBEDDEDMESSAGE` JSON ページソースは自動的に作成されます。必要に応じて追加アクションを定義することができます。

このセクションではステップ1 の`postMessage()` メソッドについて説明されています。JSON とXML データをプロセス全体のための詳細をポストしているHTML Web ページサンプルを参照してください。ステップ2 は自動的に実行され、説明を必要としません。ステップ3 はJSON とXML サンプル内で詳しく説明されています。詳細に関しては、**埋め込まれたメッセージ** イベントを参照してください。

---

postMessage メソッドについて

次の主要な点について注意してください:

- 埋め込まれたメッセージは`postMessage()` の最初のパラメーターとしてパスされます。
- 埋め込まれたメッセージはJSON オブジェクトとしてパスされます。
- メッセージ内のデータは、構成されたクローンアルゴリズムを使用してシリアル化されます。これは、メッセージがデータオブジェクトの範囲を含むことができることを意味します。シリアル化は自動的に行われます。しかしながら、全てのデータ型がMobileTogether Server に送信されるとは限りません。

例えば、JSON フォーマットでデータが送信される場合、`postMessage()` メソッドは変更すること無しにJSON データ構造を送信することができます。JSON オブジェクトが`myJSONData` という名前の変数に割り当てられると、このデータ構造は`postMessage()` メソッドを使用して以下のように送信することができます:

```
function sendMyMessage() {
      document.querySelector('iframe').contentWindow.postMessage(myJSONData
, '*');
      }
```

このサンプルでは、JSON データ構造がIFrame により受け取られます。IFrame 内のソリューションは、データをワークフローにJSON 文字列として送信します。ワークフロー内では、データ**埋め込まれたメッセージ** ページイベントによりピックアップされることができ、`$MT_EMBEDDEDMESSAGE` JSON ページソース内に保管されます。

`postMessage()` に関する詳細は、Mozilla Developer Network ドキュメント を参照してください。JSON と XML サンプルも参照してください。

リッスン: ソリューションから Web ページへ

MobileTogether Server からWeb ページへメッセージを次の段階で送信することができます:

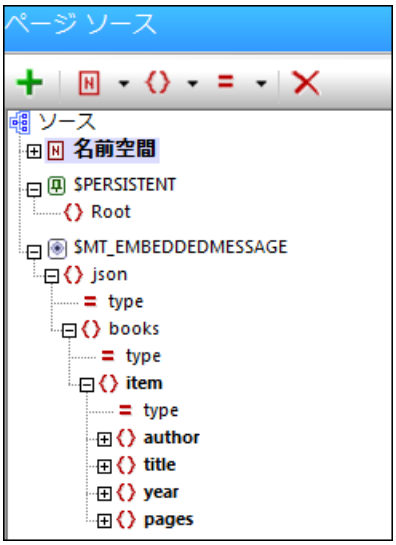
1. ワークフローの埋め込まれたメッセージをバックアクションは、メッセージをシリアル化されたJSON 文字列のフォーマットでIFrame に送信します。アクション内で送信するメッセージを指定することができます。
2. IFrame にメッセージが達すると、イベントリスナーが**メッセージ** イベントをピックアップし、メッセージを処理する関数を呼び出し、Web ページに転送されます。**メッセージ** イベントのためにイベントリスナーを以下のとおり登録することができます: `window.addEventListener('message', ProcessReturnMsg)`
3. 上のサンプル内の関数 (`ProcessReturnMsg`) は、シリアル化が解除されたJSON オブジェクトの書式の**メッセージ** イベントをパラメーターとして取ります。通常通りオブジェクトにアクセスすることができ、HTML ページ内で使用することができます。例:

```
function ProcessReturnMsg(m) {
msgVar = m.data.json.books
...
}
```

詳細に関しては、Mozilla Developer Network Web サイトで`AddEventListener` と`MessageEvent` の詳細を参照してください。

サンプル

下のスクリーンショットに示されるように、デザインには構造を持つ`$MT_EMBEDDEDMESSAGE` ページソースが含まれています。このページソースのルート要素は(JSON ページソースであるため) `json` という名前が与えられます。



ページソースの全てのコンテンツ(またはその一部) をIFrame 内のソリューションに`message` イベントとして送信することができます。これを行うには、例えば、デザイン内に、**クリック時** イベントのためにが埋め込まれたメッセージをバックするアクション設定されているボタンがある場合などです (*下のスクリーンショットを参照*)。下のXPath 式では、`$MT_EMBEDDEDMESSAGE` ノードのコンテンツがメッセージイベントとして送信されます。すなわち、`json` ノードとのコンテンツが送信されます。

HTML ページ内では、イベント リスナーを登録することができます: `window.addEventListener('message', ProcessReturnMsg)`

オブジェクトにアクセスし、HTML ページ内で使用することができます。例:

```
function ProcessReturnMsg(m) {
   msgVar = m.data.json.books
   /* 'm' is the HTML message event that is passed to ProcessReturnMsg
*/
   /* 'data' belongs to the event and holds the message returned by the
MT action */
   /* 'json' is the JSON object that is contained in the message */
   ...
   }
```

上のサンプルでは、**books** のコンテンツは **msgVar** に保存されます。

16.3 認証

ソリューションを使用するために、ユーザーは認証される必要があります。認証は2つのレベルで実行されます: (i) ユーザーがサーバーにログインする必要がある場合、ログインの資格情報が検証されます。(ii) 特定のワークフローにアクセスする際にユーザーに異なる許可が与えられ、異なるユーザーに異なるパーミッションが与えられ、認証されます。

ユーザー認証の3つの種類は、埋め込まれたWeb ページソリューションのために使用することができます:

- *匿名のユーザー*: ユーザーはログインする必要はありません。
- *ユーザーログイン*: ソリューションがロードされると、ソリューション内でMobileTogether Server ログインページが表示され、ユーザーはMobileTogether Server に登録されている資格を利用してログインすることができます。
- *JWT 認証*: 認証は、MobileTogether システム外で定義され、MobileTogether Server にユーザーがログインすることなく静的におこなうことができます。

異なる認証メソッドの長所と短所

埋め込まれたWeb ページソリューションのための承認メソッドを決定する際に、次の点に注意してください:

- ソリューションがシンプルなデータの処理である場合、ユーザーを匿名にし、重要なデータベースの変更を許可せず、データベースからの機密名情報の表示を許可しないことは安全です。
- ユーザーログインは、ユーザーがログインの詳細をMobileTogether Server に登録し、ログインの詳細が既知であることを必要とします。
- ユーザーログインは、ユーザーとソリューション間のインタラクションに不必要なレイヤーを追加する可能性があります。
- ユーザーログインは、ユーザーの個人的な認証を有効化します。
- JWT 認証は、Web ページ内のコードによりトリガーされる通信により静的に実行されます。実行者は認証プロセスの処理方法を決定することができます。これによりシステムデザイン内に柔軟性が与えられます。

1つのセッション、1つの認証の種類

Web ページとサーバー間のセッションが1つの認証の種類を使用する場合、セッションが終了、または、再起動されるまで認証メソッドを使用し続けます。ユーザーがログアウトした場合、または、サーバーがタイムアウトするとセッションは終了します (サーバーの設定 内でセッションのタイムアウトは指定することができます)。

匿名ログイン

匿名ログインにより、ユーザーがログイン、または、認証を行うことなく、ソリューションがIFrame 内で表示されるようになります。匿名ログインは最もシンプルな認証メソッドですが、ソリューションが(i) 重要なデータベースの変更を許可されていない場合、(ii) 機機密名情報が表示されないように設定されている場合のみ設定されるべきです。

匿名のアクセスを設定するには、以下を行ってください：

- 匿名ログインを許可するように設定してください(*次を参照してください：モバイルクライアントポートのためのサーバー設定*)
- (*ワークフローのコンテナーのパーミッションを設定* を`container=read` と`workflow=read,use` の最小値に設定することにより) ソリューションの匿名の使用を許可してください。

ユーザーログイン

この認証の種類では、ソリューションがロードされる前に、ユーザーは正確な資格を使用して、MobileTogether Server にログインする必要があります。ソリューションへの呼び出しがトリガーされると、MobileTogether Server ログインページが表示されます。ログが認証されるには、2つの条件が満たされるひつようがあります:

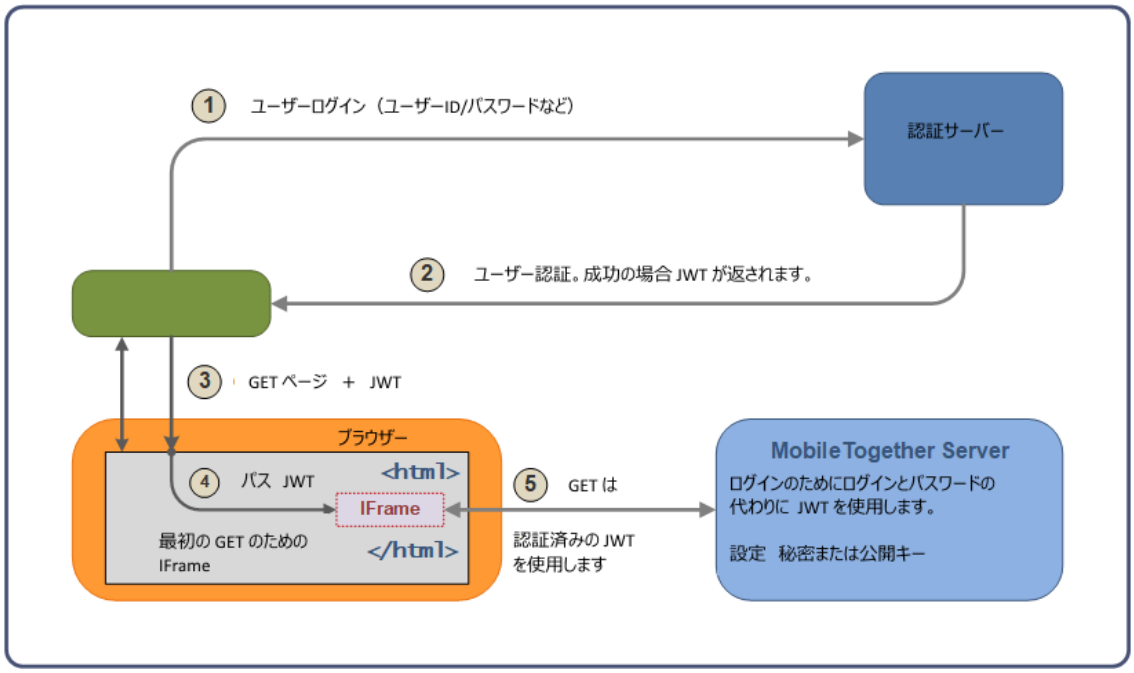
- ユーザーは、MobileTogether Server のユーザーとして登録される必要があります。
- ユーザーは、適切なターゲットとされたソリューションへアクセスするパーミッションを有する必要があります。

MobileTogether Server へのユーザーの登録と、パーミッションのセットアップの詳細に関しては、次を参照してください: MobileTogether Server ドキュメント。

JWT 認証

埋め込まれたWeb ページソリューションの認証は、JSON Web Tokens (JWT、奨励される発音: *jŏt*) を使用して行うこともできます。基本的には、JWT 認証は、MobileTogether Server 認証システム外で処理されます。認証されたユーザーは、MobileTogether Server のWeb ページにパスされるJWT と共に発行されます。また、ユーザーを検索するために解析され、JWT が有効な場合、JWT 内でユーザーのために指定されている埋め込まれたソリューションとサーバー間の通信が実行されます。

下の図は、JWT 認証の仕組みを示しています。



以下の点に注意してください：

- 認証サーバーとアプリケーション(www) サーバーを分ける必要はありません。
- 埋め込まれたWeb ページソリューションと認証システムが使用されるシナリオでは、実行者がどのように定義するかを決定することができます
- (ユーザーの認証に成功し) JWT が作成されると、ユーザーを指定するパラメータが定義されます。MobileTogether Server 上で構成されるユーザーにユーザーが対応するように指定されると、このユーザーのために構成されたパーミッションによりサーバーへのアクセスが決定されます。他のユーザーは構成されたユーザーのリストに自動的にインポートされます。しかしながら、このユーザーのパーミッションは、MobileTogether Server 上で構成されます。
- 実行者はMobileTogether Server 設定内に、共有される秘密、または、公開キーを入力します。認証サーバー上にJWT を生成するため、実行者が使用する文字の文字列と共有する秘密は同じです。JWT を暗号化するために使用された秘密キーに公開キーは対応します。
- JWT はIFrame にパスされ、ソリューションのMobileTogether Server への最初の呼び出しと共にパスされます。
- MobileTogether Server では、共有される秘密、または、MobileTogether Server 設定内の公開キーが使用され、受信されたJWT を検証するために使用されます。
- JWT が検証されると、埋め込まれたIFrame とMobileTogether Server のソリューションの間に認証された通信セッションがセットアップされます。
- セッションのためのユーザーは、JWT 内で指定されたユーザーです。

JWT とは?

JSON オブジェクトを構成するJWT は要求 (JSON プロパティ?値ペア) のペアです。これは3つの部分から構成されます:

- ヘッダー: 2つのプロパティにより構成されています: `{ "alg": "HS256", "typ": "JWT" }`. `alg` は、JWT を暗号化するために使用されるアルゴリズムです。
- *ペイロード*: 送信されるデータはここに保存され、JSON プロパティ?値ペアとしてこのデータは保管されます。
- *署名*: ヘッダー内で指定されるアルゴリズムを使用して暗号化することにより作成されます: (i) base64Url-encoded ヘッダー、(ii) base64Url-encoded ペイロード (iii) 秘密 (または、秘密キー):
  `HMACSHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload), secret|privateKey)`

最終のJWT は3つの部分から構成されます。それぞれは base64Url-encoded で、点により区切られています。次を参照してください: `openid.net` と`jwt.io` Web サイト。

対称キーと非対称キー

対称キー (共有された秘密) 、または、非対称キー (秘密 ?公開ペアの秘密キー) を使用することによりJWT は暗号化されます。

- *対称キー*: (JWT が作成されると) 同じキーが両方の暗号化、および、暗号化解除に使用されます (MobileTogether Server はキーを使用して JWT を検証します)。共有される秘密として知られる対象キーは、MobileTogether Server 内の設定として保管されています。対象キーと作業するための詳細に関しては、次を参照してください: *対称キー: 共有される秘密* 。
- *非対称キー*: 暗号化 (秘密キー) 、および、暗号化解除 (公開キー) のために異なるキーが使用されています。JWT を検証できるように公開キーは、MobileTogether Server 内の設定として保管されています。JWT のために非対称暗号化を使用するための詳細は、次を参照してください: *非対称キー: 公開キー*。

JWT を試す

JWT を以下のように試すことができます:

1. ジェレミー クリッツのオンライン JWT ビルダー で自身の JWT を作成します。概要に関しては次を参照してください: 対称キー: 共有される秘密。
2. `jwt.io` Web サイトで、以下のようにキーを検証します: (i) 暗号化されたペイン内に暗号化された JWT を入力します。(ii) 署名の検証ペインに、JWT を作成する際に使用した秘密を入力します。Web サイトのデバッガーは署名が検証されたことを通知します。

サンプル JWT 認証 も参照してください。

対称キー: 共有された秘密

埋め込まれたWeb ページソリューション内でJWT を使用する方法:

1. 対称的な暗号化を使用したJWT の作成。JWT は以下をベースとします (i)(要求と呼ばれる) プロパティ値の情報。 (ii) ランダムな文字列 (共有される秘密)。
2. Web ページから送信されたを検証するためのJWT のためのMobileTogether Server のセットアップ。2つの情報を提供します: (i) JWT を生成するために使用された秘密。(ii) オーディエンスの要求の値 (JWT を生成するために使用されたものと同じである必要があります)。
3. Web ページ内では、JWT がIFrame にパスされます。

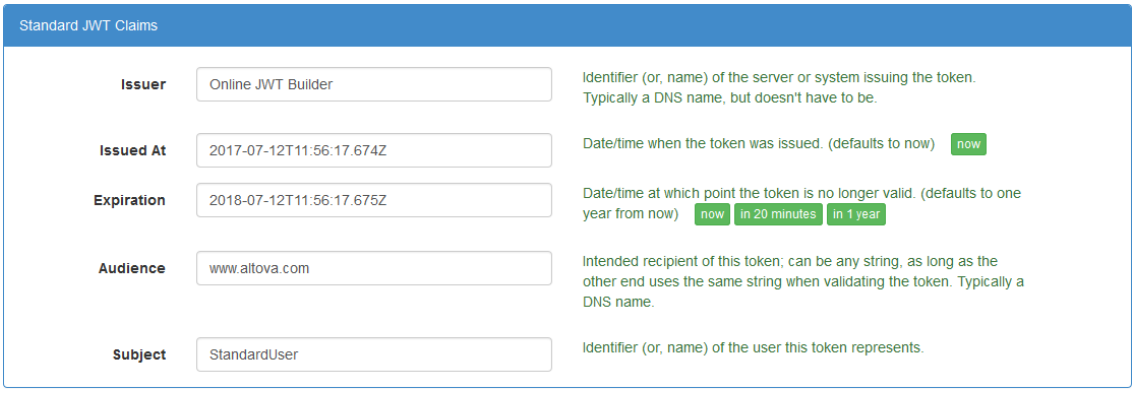
JWT がサーバーにパスされると、オーディエンスの情報とJWT を生成するために使用した設定内に入力した秘密を使用してサーバーは検証します。

JWT の作成

説明ではジェレミー クリッのオンラインJWT ビルダーを使用して対称キー (共有される秘密) を用いたのJWT 作成のプロセスが説明されています。ここではMobileTogether の埋め込まれたWeb ページソリューション内でのJWT の使用に関連した要求 (JSON プロパティ?値ペア) について説明されています。

標準の要求

標準の要求はコアの要求を構成します (*下のスクリーンショットを参照*) :



- MobileTogether Server は、サーバーへのアクセスの時刻がJWT の有効期間内に存在するかをチェックします。有効期限を適切に設定してください。
- *オーディエンス* パラメーターは、MobileTogether Server 上で設定する設定の1つです。ですから (JWT を設定する際の) *オーディエンス* パラメーターとMobileTogether Server *オーディエンス* 設定 内で同じ値を設定してください (*次を参照してください: MobileTogether Server 設定*)。
- サブジェクトパラメーターは、MobileTogether Server へログインするユーザーを指定する場所です。ユーザー名がMobileTogether Server を使用して登録されているユーザーの場合、ログインはそのユーザーが有するパーミッションを使用して行われます。ユーザーがMobileTogether Server に登録されておらずMobileTogether Server にログインした場合、関連するワークフローにアクセスできるようこの新規のユーザーのためにパーミッションを設定する必要があります。

JWT対称キーのための (または、共有される秘密)

キー (または、共有される秘密) は、他の入力されたデータと共に、JWT を生成するために使用されます。この秘密は、Web ページから受信されるJWT を非暗号化し認証するために使用されます。MobileTogether Server により使用されます。秘密は、(JWT の) 暗号化と非暗号化に使用されます。JWT を生成する際、共有される秘密として任意の文字列を指定することができます。MobileTogether Server *秘密* 設定として同じ文字列が入力されなければなりませ

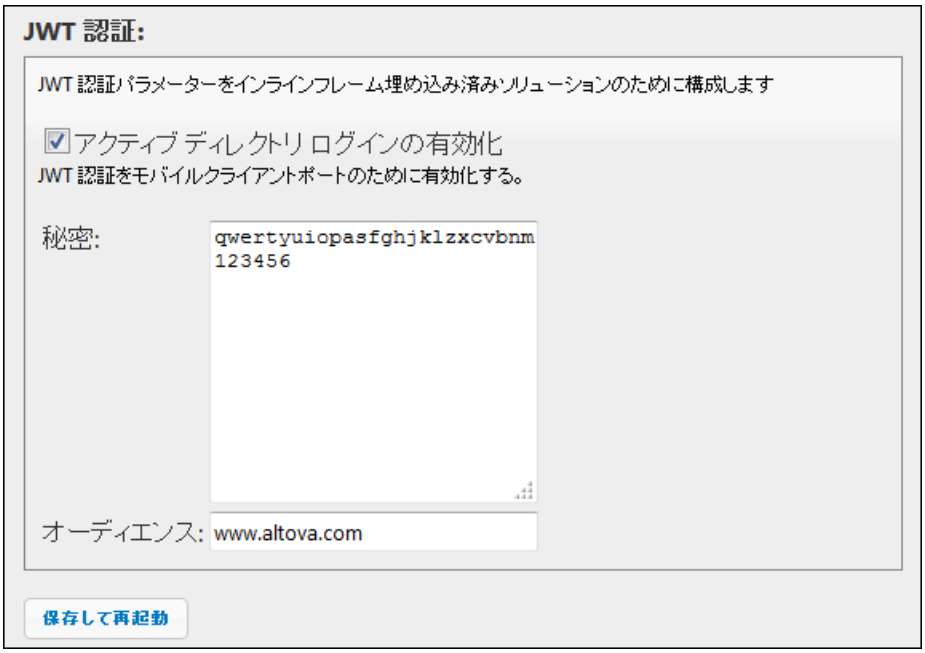
ん (下の MobileTogether Server 設定 を参照してください)。

下のスクリーンショットでは、32-文字の長さの秘密が入力され、暗号化アルゴリズム **HS256** が選択されます。 **署名されたJWT の作成** をクリックすると、JWT が作成され、テキストボックス内に表示されます。

MobileTogether Server 設定

MobileTogether Server の設定タブ内で、JWT 認証を有効化する必要があり、次の2つの設定を入力する必要があります (*下のスクリーンショットを参照*):

- *秘密*: これは JWT を作成するために使用された対称キー (共有された秘密) です。この情報を用いて、サーバーは JWT を検証することができます (非対称暗号化 を使用する場合、このフィールド内に秘密 ?公開ペアの公開キーを入力します)。
- *オーディエンス*: JWT を作成する際にオーディエンスの要求のために入力された文字列と同じ文字列を入力します。

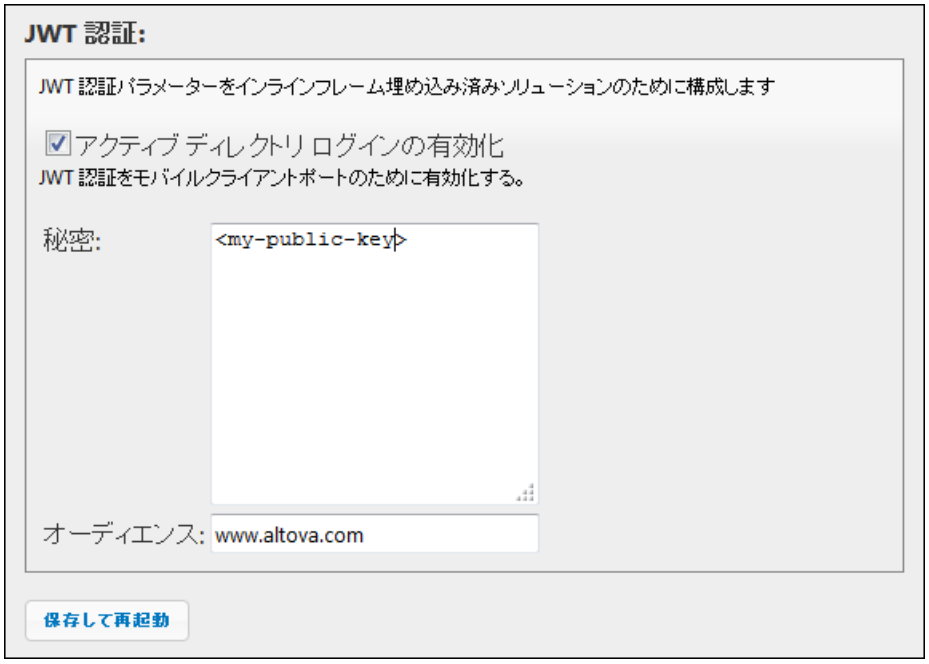


非対称キー: 公開キー

JWT のために非対称暗号化が使用されている場合、暗号化 (JWT 署名) は秘密キーを使用して行われ、検証は公開キーを使用して行われます。MobileTogether Server が JWT を検証できるようにするには、以下を行ってください:

MobileTogether Server の設定タブ内で、JWT 認証を有効化し、以下の設定を入力します (*下のスクリーンショットを参照*):

- *秘密*: 秘密 ?公開ペアの公開キーを入力します (対称暗号化 が使用される場合、共有される秘密を入力してください)。
- *オーディエンス*: JWT を作成する際にこの要求のために入力された文字列と同じ文字列を入力します。



16.4 サンプル

このセクション内のサンプルは、簡単なサンプルから始まり、JWT 認証を使用するサンプルで終わる埋め込まれたWeb ページソリューションのセットアップ方法、について説明しています。このセクションで使用されるサンプルファイルは、マイドキュメントMobileTogether フォルダー内にあります: **MobileTogetherDesignerExamples\Tutorials\EmbeddedWebpageSolutions**。

このセクション:

- ソリューションの埋め込み
- JSON データの送信 受信
- XML データの送信 受信
- JSON ページソースの事前設定
- JWT 認証

ソリューションの埋め込み

下にリストされるHTML コードは**IFRAME** 要素がどのようにソリューションを埋め込むために使用されるかを示しています。HTML コードをコピーして、ファイルに保存し、ブラウザーで開くことによりこの機能を試すことができます。

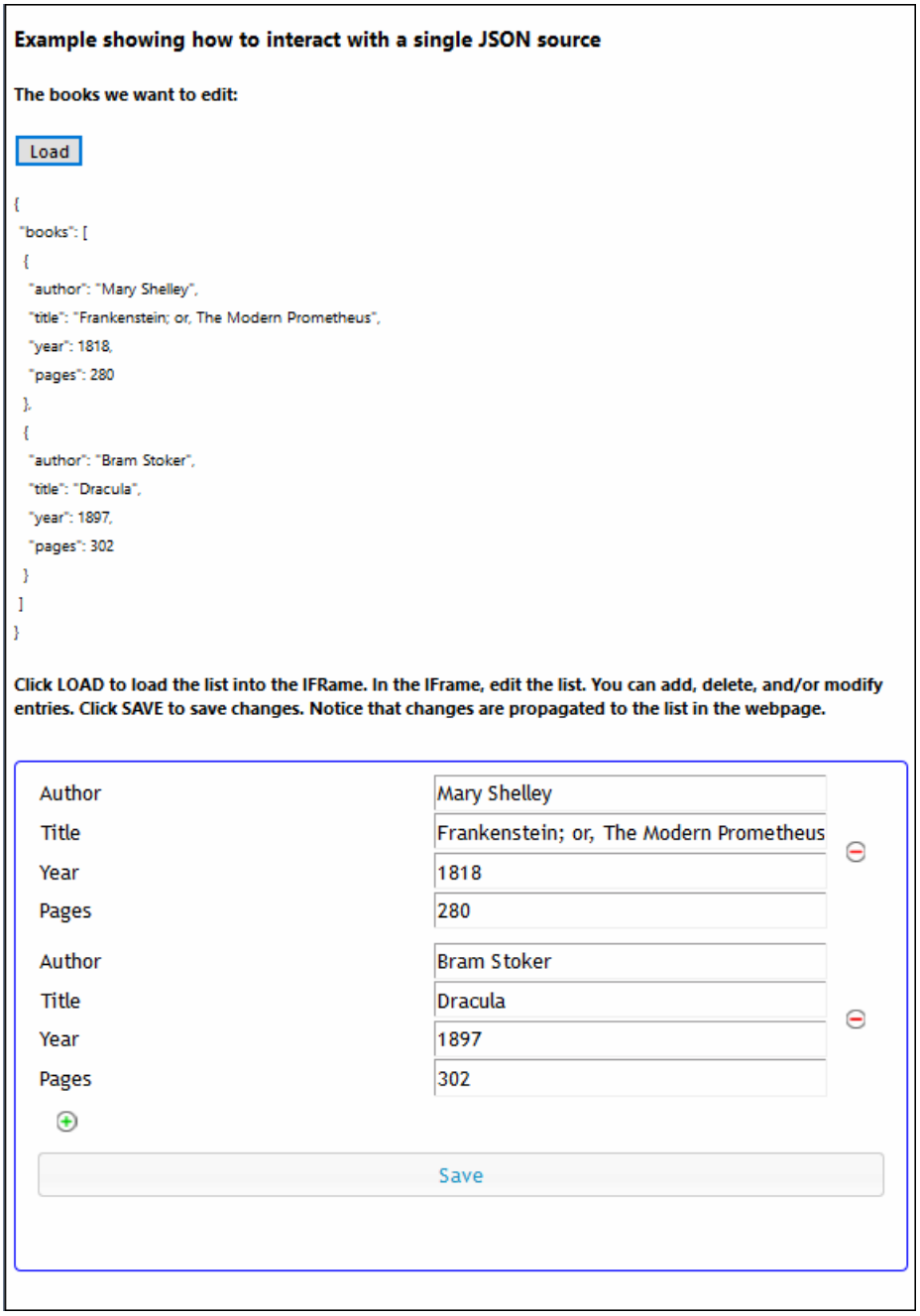
```
<!DOCTYPE html>
<html>
  <head>
    <title>Webpage containing an embedded solution</title>
  </head>
  <body>
    <h1>Webpage containing an embedded solution</h1>
    <p>The embedded solution is loaded into an IFrame, which is located
immediately below this paragraph.</p>
    <div class="resize">
      <iframe src="http://localhost:8083/run?d=/public/About"
frameborder="0"></iframe>
    </div>
  </body>
</html>
```

このリスティングで使用されているサンプルソリューションは`About` と名前が付けられており、MobileTogether Server 内にパッケージされており、サーバーの`Public` コンテナー内のデフォルトの場所に保存されています。このソリューションを正確に埋め込むために、`About` ワークフローの匿名のアクセスが許可されるようにセットアップされているようにしてください。

次を参照: Web ページにソリューションを埋め込む

JSON データの送信 /受信

このセクションでは、JSON ソースデータを使用する埋め込まれたWeb ページの仕組みが説明されています。Web ページからJSON 書式の本のリストが埋め込まれた(青い枠で囲まれている) IFrame に送信されます (下のスクリーンショット)。 MobileTogether ソリューションを使用してリストを編集することができます。IFrame 内に変更を保存すると、編集された本のリストがWeb ページに送信されます。



埋め込まれたWeb ページソリューションは(`jsonBooks.html`) とMobileTogether デザイン(`jsonBooks.mtd`) から構成されています。両方のファイルは次の場所にあります: マイドキュメントMobileTogether フォルダー: `MobileTogetherDesignerExamples\Tutorials\EmbeddedWebpageSolutions`。ファイルを試すには、ファイルをサーバーにデプロイし、匿名でアクセスできるようにします。必要な場合は、IFrame がサーバー上のフークフロ

ーを正確にターゲットできるように HTML コードを変更します。Web ページをブラウザー内で開き、開始するために **ロード** ボタンをクリックします。

下の説明には、Web ページの完全な HTML コードリスティングと、コードとソリューションのインタラクションの方法について説明されています。

HTML コードリスティング

ファイル **jsonBooks.html** の HTML コードリスティング。コードの説明は次のセクションで行われています。このサンプルで使用される JavaScript 機能は、全てのブラウザーで使用できない場合があることに注意してください。この場合、ブラウザーに合うように JavaScript を変更してください。

Web ページコードリスティング

```
<DOCTYPE html>
<html>
  <head>
    <style>
      * {
        font-family:Segoe UI,Tahoma,Arial,Helvetica,sans-serif;
      }
      iframe {
        width:100%;
        height:400px;
        border:2px solid blue;
        border-radius:5px;
        margin:10px 0px;
      }
      code {
        font-size:small;
      }
    </style>
    <script>
      // The initial book list is stored in a variable in JSON format
      // It can be easily handled in JavaScript
        var books = {
            "books":[
              {
                "author":"Mary Shelley",
                "title":"Frankenstein; or, The Modern Prometheus",
                "year":1818,
                "pages":280
              },
              {
                "author":"Bram Stoker",
                "title":"Dracula",
                "year":1897,
                "pages":302
              }
            ]
          };

        // Posts variable 'books' to IFrame (from where 'books' is forwarded to MT Server)
        function sendbooks() {
            document.querySelector('iframe').contentWindow.postMessage(books, '*');
          }

       // Contents of variable 'books' converted to string and displayed inside HTML element CODE
        function showbooks() {
            document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
          }

        // m = HTML message event; data = container for message from server
        // m.data.json = contents of the 'json' object that was sent from the server
```

```
        function receivebooks(m) {
            books = m.data.json;
            showbooks();
          }

        // Handler to receive messages from server via solution in IFrame
        window.addEventListener('message', receivebooks);

        // Handler to show initial book list in webpage on page load
        document.addEventListener('DOMContentLoaded', showbooks);

    </script>
  </head>


  <body>
    <h4>Example showing how to interact with a single JSON source</h4>
    <h5>The books we want to edit:</h5>
    <!-- Send the JSON book list from the webpage to the IFrame -->
    <button onclick="sendbooks()">Load</button>
    <pre><code><!-- The SHOWBOOKS function displays the book list here --></code></pre>
    <h5>
      Click LOAD to load the book list into the IFrame. In the IFrame, edit the list. You can add, delete,
and/or modify entries. Click SAVE to save changes. Notice that changes are propagated to the list in the
webpage.
    </h5>
    <iframe src="http://localhost:8083/run?d=/public/jsonBooks" frameborder="0"></
iframe>
  </body>

</html>
```

しくみ

以下の説明では、異なる背景色が使用され仕組みのそれぞれの部分で起こっていることが説明されています (*Web ページ?ソリューション?ワークフロー*):

	Web ページ: *ユーザーアクションとHTML/JavaScript コードの作動方法*
	ソリューション: *IFrame 内のソリューションにより実行されるアクション*
	ワークフロー: *(MT デザインをベースにした) サーバー上の処理*

HTML ページのロード時:

books という名前のJavaScript 変数が読み込まれます。**books** という名前のJSON オブジェクトが含まれています。

```
var books = {
  "books":[
    {
      "author":"Mary Shelley",
      "title":"Frankenstein; or, The Modern Prometheus",
      "year":1818,
      "pages":280
    },
    {
      "author":"Bram Stoker",
      "title":"Dracula",
```

```
      "year":1897,
      "pages":302
    }
  ]
};
```

DOMContentLoaded イベントをリッスンするイベントリスナーとJavaScript 関数 (**showbooks**) を使用して**books** 変数のコンテンツがHTML **code** 要素内で表示されます:

```
document.addEventListener('DOMContentLoaded', showbooks);

function showbooks() {
  document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
}

<pre><code></code></pre>
```

HTML ドキュメントがロードされた時、**books** 変数のコンテンツを確認することができます (後に、同じ関数を**books** 変数の更新を確認するために使用します。)

IFrame は(**iframe** 要素の**src** 属性内をターゲットにして) **jsonBooks** ソリューションをロードします:

```
<iframe src="http://localhost:8083/run?d=/public/jsonBooks"
frameborder="0"></iframe>
```

***ロード**ボタンのクリック時:*

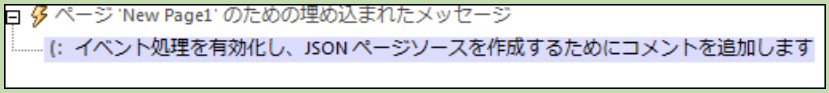
```
<button onclick="sendbooks()">Load</button>
```

JavaScript 関数は、**postMessage()**を使用して、**books** 変数のコンテンツを IFrame に送信します。

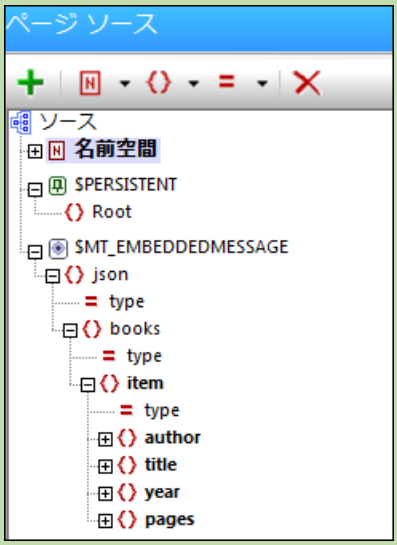
```
function sendbooks() {
document.querySelector('iframe').contentWindow.postMessage(books, '*');
}
```

{books} は(シリアル化されたJSON フォームで)自動的にサーバー上のワークフローに送信されます。

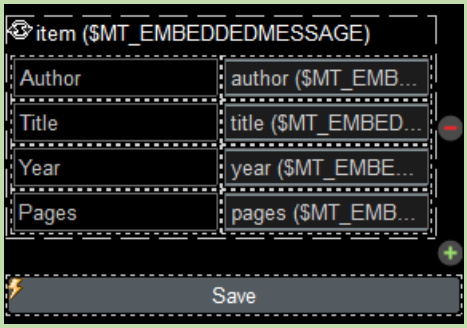
ページイベント**埋め込まれたメッセージ** には ..と定義されているアクションが存在するため、



... ページソース**$MT_EMBEDDEDMESSAGE** は**{books}** データと共に自動的にロードされます。デザイン内で目的どおり作動するには、(デザイン時に定義される) ページソースの構成が受信される JSON データの構造に対応する必要があります。ページソース内の**item** ノードは、JSON 配列内の各アイテムに対応します。(受信されるJSON データがページソースのために定義されている構造に一致しない場合、自身の構造を用いてロードされます。デザイン内のXPath 式が定義された構成を参照するため、ランタイムにロードされた構造にXPath 式を使用してたどり着くことはできません。)

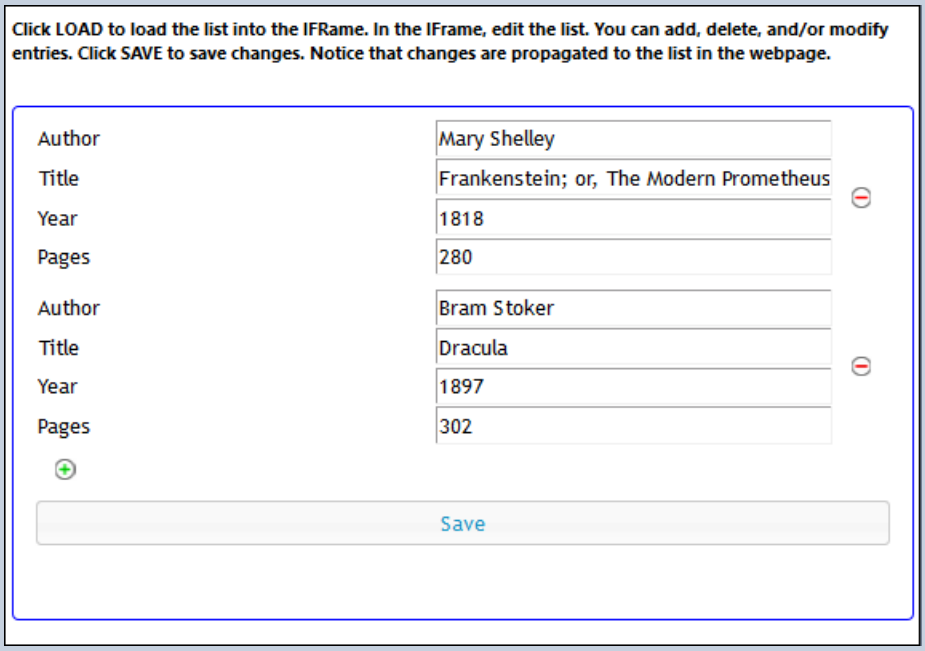


デザインは**item** ノードの繰り返しテーブルを含んでいます。テーブルのセルはそれぞれ**author**、**title**、**year**、および**pages** ページソースノードにリンクされています。



この結果、...

...IFrame 内のソリューションが更新されます。ワークフローの**$MT_EMBEDDEDMESSAGE** ページソース内のデータと共に繰り返しテーブルが作成されます。



ソリューション内で本のデータを編集する際、**$MT_EMBEDDEDMESSAGE** ページソースが継続的に更新されます。 **「保存」**をクリックすると、**ボタンのクリック時** イベントハンドラーがトリガーされます。

ボタンのクリック時 イベントは埋め込まれたメッセージをバックアクションが実行されるように指定します。これはページソース**$MT_EMBEDDEDMESSAGE** 全体のコンテンツを**message** へイベントとして IFrame に送信されます (**$MT_EMBEDDEDMESSAGE** には編集された本のリストが含まれていることに注意してください。)

message イベントをリッスンするためにイベントリスナーを登録することができます。**message** イベントをピックすると、JavaScript 関数 (**receivebooks**) が呼び出されます:

```
window.addEventListener('message', receivebooks);
```

receivebooks 関数は(*下を参照*) **message** イベント(**m**) をパラメータとして取り(**data** は **message** イベントのデータ)、受信されたメッセージ内の**json** オブジェクトのコンテンツを**books** 変数に割り当てます。 **books** 変数は、サーバーからの更新された本のリストである**json** オブジェクトのコンテンツを含みます。更新されたブックリストの構造は元の本のリストの構造と同じです。(しかしながら、**books** 配列内には数の多い、または、少ない本のアイテムが含まれています)。

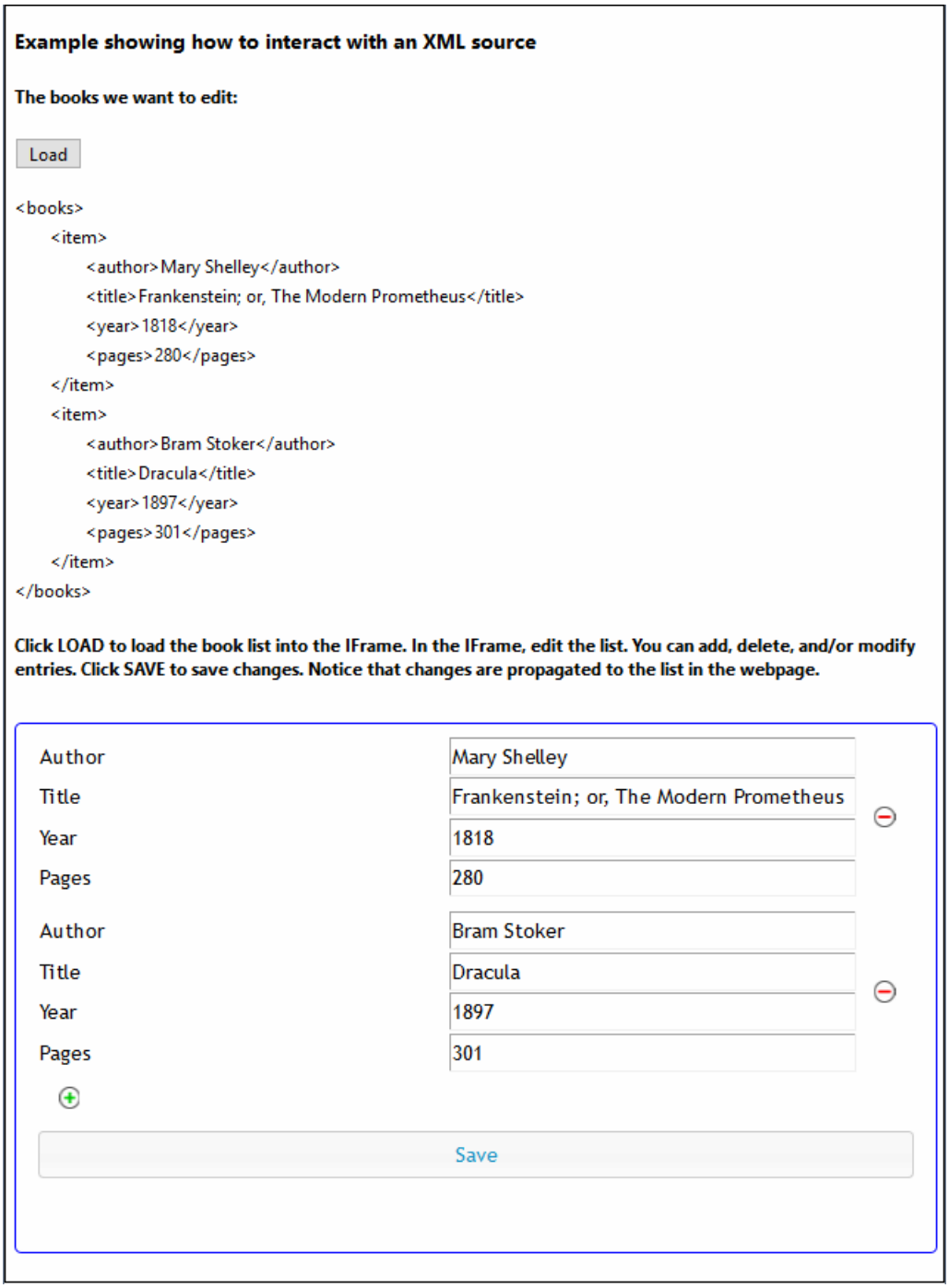
```
function receivebooks(m) {
  books = m.data.json;
  showbooks();
}
```

showbooks 関数は Web ページ内の更新されたブックリストを表示しています:

```
function showbooks() {
  document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
}

<pre><code></code></pre>
```

XML データの送信 /受信

このセクションでは、 XML ソースデータ(ブックリスト)を使用する埋め込まれたWeb ページの仕組みが説明されています。Web ページからJSON 書式の本のリストが埋め込まれた(*青い枠で囲まれている*) IFrame に送信されます (*下のスクリーンショット*)。 MobileTogether ソリューションを使用してリストを編集することができます。IFrame 内に変更を保存すると、編集された本のリストがWeb ページに送信されます。



埋め込まれたWeb ページソリューションは(`xmlBooks.html`) とMobileTogether デザイン(`xmlBooks.mtd`) か

ら構成されています。両方のファイルは次の場所にあります: マイドキュメントMobileTogether フォルダー: **MobileTogetherDesignerExamples\Tutorials\EmbeddedWebpageSolutions**。ファイルを試すには、ファイルをサーバーにデプロイし、匿名でアクセスできるようにします。必要な場合は、IFrame がサーバー上のワークフローを正確にターゲットできるようにHTML コードを変更します。開始するために **[ロード]** ボタンをクリックします。

下の説明には、Web ページの完全な HTML コードリスティングと、コードとソリューションのインタラクションの方法について説明されています。

HTML コードリスティング

ファイル **xmlBooks.html** のHTML コードリスティング。コードの説明は次のセクションで行われています。このサンプルで使用されるJavaScript 機能は、全てのブラウザーで使用できない場合があることに注意してください。この場合、ブラウザーに合うようにJavaScript を変更してください。

Web ページコードリスティング

```
<DOCTYPE html>
<html>
  <head>
    <style>
      * {
        font-family:Segoe UI,Tahoma,Arial,Helvetica,sans-serif;
      }
      iframe {
        width:100%;
        height:400px;
        border:2px solid blue;
        border-radius:5px;
        margin:10px 0px;
      }
      code {
        font-size:small;
      }
    </style>
    <script>
      // The book list in XML format
      var books = '
      <books>
        <item>
          <author>Mary Shelley</author>
          <title>Frankenstein; or, The Modern Prometheus</title>
          <year>1818</year>
          <pages>280</pages>
        </item>
        <item>
          <author>Bram Stoker</author>
          <title>Dracula</title>
          <year>1897</year>
          <pages>302</pages>
        </item>
      </books>
      ';

      // This is the XML DOM tree (initialized in showbooks)
      var books;

      function sendbooks() {
        document.querySelector('iframe').contentWindow.postMessage({
        "books":books.childNodes[0].outerHTML
        }, '*');
      }

      // This is the function that receives the updated books
```

```
      function receivebooks(m) {
        books = m.data.jsonbooks;
        showbooks();
      }

      // Contents of variable 'books' converted to string and displayed inside HTML element CODE
      function showbooks() {
        // Create a DOM tree from the XML
        books = new DOMParser().parseFromString(books, 'text/xml');
        // Manipulate the DOM and show the result
        document.querySelector('code').innerText = books.childNodes[0].outerHTML;
      }

      // Handler to receive messages from server via solution in IFrame
      window.addEventListener('message', receivebooks);

      // Handler to show initial list of books on page load
      document.addEventListener('DOMContentLoaded', showbooks);
    </script>
  </head>
  <body>
    <h4>Example showing how to interact with an XML source</h4>
    <h5>The books we want to edit:</h5>
    <button onclick="sendbooks()">Load</button>
    <pre><code></code></pre>
    <h5>
      Click LOAD to load the book list into the IFrame. In the IFrame, edit the list. You can add, delete,
and/or modify entries. Click SAVE to save changes. Notice that changes are propagated to the list in the
webpage.
    </h5>
    <iframe src="http://localhost:8083/run?d=/public/xmlBooks" frameborder="0"></iframe>
  </body>
</html>
```

しくみ

以下の説明では、異なる背景色が使用され仕組みのそれぞれの部分で起こっていることが説明されています (*Web ページ?ソリューション?ワークフロー*):

	Web ページ: *ユーザーアクションとHTML/JavaScript コードの作動方法*
	ソリューション: *IFrame 内のソリューションにより実行されるアクション*
	ワークフロー: *(MT デザインをベースにした) サーバー上の処理*

HTML ページのロード時:

books という名前のJavaScript 変数が読み込まれます。XML 構造を含む文字列が含まれています:

```
var books = '
  <books>
    <item>
      <author>Mary Shelley</author>
      <title>Frankenstein; or, The Modern Prometheus</title>
      <year>1818</year>
      <pages>280</pages>
    </item>
    <item>
```

```
      <author>Bram Stoker</author>
      <title>Dracula</title>
      <year>1897</year>
      <pages>302</pages>
    </item>
  </books>
';
```

(**DOMContentLoaded** イベントのをリッスンする)イベントリスナーとJavaScript 関数 (**showbooks**) を使用して**books** 変数のコンテンツがHTML **code** 要素内で表示されます:

```
document.addEventListener('DOMContentLoaded', showbooks);
```

showbooks 関数: (i) DOM ツリーを **books** 変数内の XML 構造から作成します。(ii) HTML **code** 要素内に希望する XML 構造を配置します。

```
function showbooks() {
  books = new DOMParser().parseFromString(books, 'text/xml');
  document.querySelector('code').innerText = books.childNodes[0].outerHTML;
}
```

```
<pre><code></code></pre>
```

HTML ドキュメントがロードされた時、**books** 変数のコンテンツを確認することができます (後に、**showbooks** 関数を**books** 変数の更新を確認するために使用します)。

IFrame はIFrame は(**iframe** 要素の**src** 属性内をターゲットにして) **jsonBooks** ソリューションをロードします: ソリューション**xmlBooks** をロードします:

```
<iframe src="http://localhost:8083/run?d=/public/xmlBooks"
frameborder="0"></iframe>
```

***ロード**_ボタンのクリック時*:

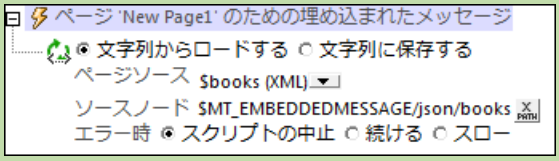
```
<button onclick="sendbooks()">Load</button>
```

JavaScript 関数 (**sendbooks**) は、**postMessage()**を使用して、**books** 変数のコンテンツを IFrame に送信します。XML コンテンツは JSON オブジェクト内に配置されルことに注意してください (これはワークフローが JSON を受信することを期待するからです)。

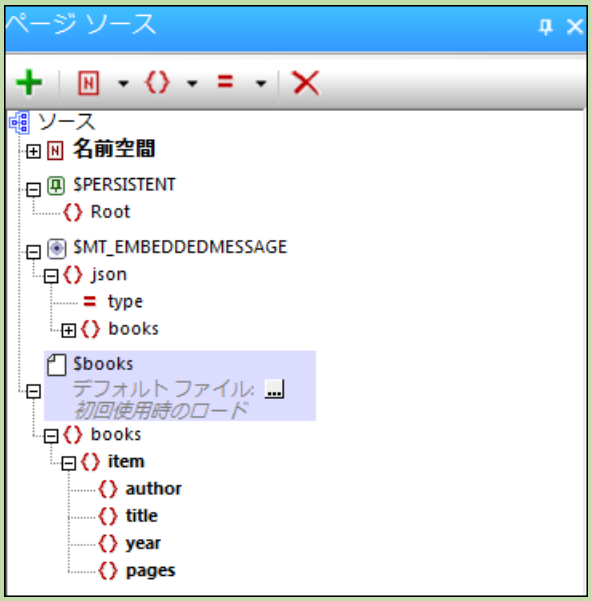
```
function sendbooks() {
  document.querySelector('iframe').contentWindow.postMessage({
    "books": books.childNodes[0].outerHTML
  }, '*');
}
```

{books} は(シリアル化されたJSON フォームで)自動的にサーバー上のワークフローに送信されます。

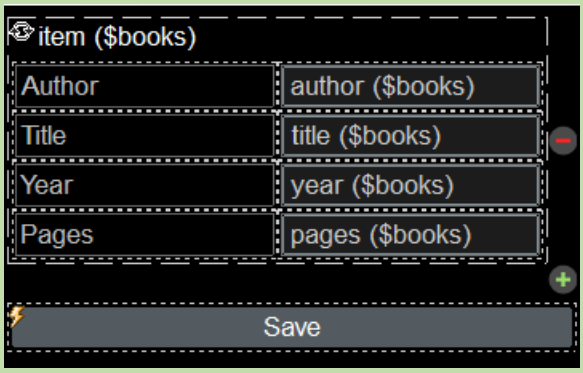
アクションが定義されているため、ページイベント**埋め込まれたメッセージ** は有効化されます (*下のスクリーンショットを参照*)。この結果、ページソース**$MT_EMBEDDEDMESSAGE** は**{books}** データと共に自動的にロードされます。



文字列からロードアクションは **$MT_EMBEDDEDMESSAGE/json/books** のコンテンツを XML ページソース **$books** として作成します (*上のスクリーンショット*)。これらのページソースは既に作成されています (*下のスクリーンショットを参照*)。

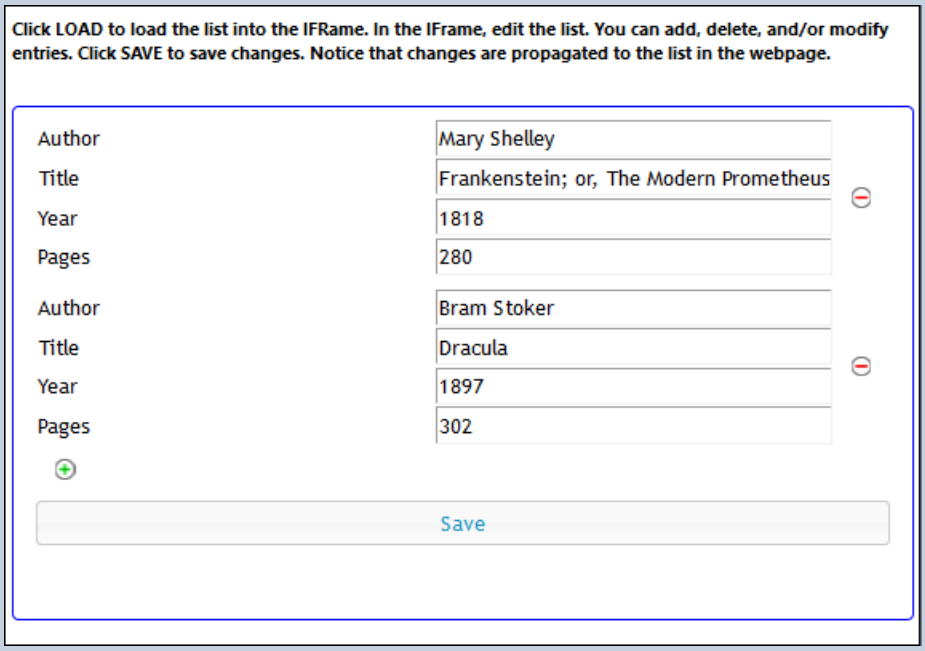


デザインコンポーネントに XML データをリンクし、XML データを編集できるように、XML ページソースが作成されます。デザインは **$books** ページソースの **item** ノードの繰り返しテーブルを含んでいます (*下のスクリーンショットを参照*)。デザインは **item** ノードの繰り返しテーブルを含んでいます。テーブルのセルはそれぞれ **author**、**title**、**year**、および **pages** ページソースノードにリンクされています。



ページソースがロードされるため、ワークフローの **$MT_EMBEDDEDMESSAGE** ページソース内のデータを使用して繰り返しテーブルが作成されます。

IFrame 内のソリューション内でアップデートは表示されています。



ソリューション内で本のデータを編集する際、**$books** ページソースが継続的に更新されます。**「保存」**をクリックすると、**ボタンのクリック時** イベントハンドラーがトリガーされます。

ボタンのクリック時 イベントは次の2つのアクションを指定します: (i) **$books** ページソースを保存する「文字列に保存」アクション(編集された本のリストを含む) **$MT_EMBEDDEDMESSAGE/json/books**。(ii) **$MT_EMBEDDEDMESSAGE** のコンテンツを**message** イベントとして IFrame に送信する「埋め込まれたメッセージをバックする」アクション(**$MT_EMBEDDEDMESSAGE/json/books** には編集された本のリストが含まれていることに注意してください)。

message イベントをリッスンするためにイベントリスナーが登録されます。**message** イベントをピックアップする JavaScript 関数 (**receivebooks**) が呼び出されます:

```
window.addEventListener('message', receivebooks);
```

receivebooks 関数 (*下を参照*) は **message** イベント (**m**) をパラメーターとして取り (**data** は**message** イベントのデータです)、受信されたメッセージ内の**json/books** オブジェクトのコンテンツを**books** 変数に割り当てます。**books** 変数には、サーバーから更新された本のリストが含まれるようになります。

```
function receivebooks(m) {
  books = m.data.json.books;
```

```
  showbooks();
}
```

showbooks 関数: (i) **books** 変数内の XML 構造から DOM ツリーを作成します。(ii) 希望する XML 構造を HTML **code** 要素内に配置します。

```
function showbooks() {
  books = new DOMParser().parseFromString(books, 'text/xml');
  document.querySelector('code').innerText = books.childNodes[0].outerHTML;
}
```

```
<pre><code></code></pre>
```

Web ページに更新された本のリストが表示されます。

JSON ページソースの事前設定

このセクション内のサンプルでは、HTML ページが開かれると、自動的にWeb ページからのJSON ソースデータが自動的にワークフローに送信されます。

HTML コードはJSON データの送信 受信 サンプル内で使用されるものをベースにしています。差異は、以下のとおりです: 前のサンプルでは、Web ページ内でボタンをクリックして、最初の本のリストをIFrame に送信する必要があります。このサンプルでは、データはページが開かれると自動的にロードされます。(**「ロード」** ボタンとその関数は削除され、新しい関数がデータをロードするために使用されます)。

このサンプルで使用されるファイルは **jsonBooksOnStart.html** と **jsonBooks.mtd** です。両方のファイルは次の場所にあります: マイドキュメントMobileTogether フォルダー: **MobileTogetherDesignerExamples\Tutorials\EmbeddedWebpageSolutions**. ファイルを試すには、MTD ファイルをサーバーにデプロイし、匿名でのアクセスを有効化してください。必要な場合は、正確なワークフローをターゲットするためにHTML コードを変更します。

下の説明はJSON データの自動ロードに関連した点についてのみ行われています。仕組みの他の点については、次を参照してください: JSON データの送信 受信。

HTML コード リスティング

ファイル **jsonBooksOnStart.html** のHTML コード リスティング。コードの説明は次のセクションで行われています。このサンプルで使用されるJavaScript 機能は、全てのブラウザーで使用できない場合があることに注意してください。この場合、ブラウザーに合うようにJavaScript を変更してください。

Web ページ コード リスティング

```
<DOCTYPE html>
<html>
  <head>
    <style>
      * {
        font-family:Segoe UI,Tahoma,Arial,Helvetica,sans-serif;
      }
      iframe {
        width:100%;
        height:400px;
        border:2px solid blue;
        border-radius:5px;
        margin:10px 0px;
      }
      code {
        font-size:small;
      }
    </style>
    <script src="http://localhost:8083/js4.0/WebAppIFrame.js"></script>
    <script>
      // The initial book list stored in a variable in JSON format
        var books = {
            "books":[
              {
                "author":"Mary Shelley",
                "title":"Frankenstein; or, The Modern Prometheus",
                "year":1818,
                "pages":280
              },
              {
                "author":"Bram Stoker",
                "title":"Dracula",
                "year":1897,
                "pages":302
              }
```

```
            ]
        };

        // Contents of variable 'books' converted to string and displayed inside HTML element CODE
        function showbooks() {
            document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
        }

        // m = HTML message event; data = container for message from server
        // m.data.json = contents of the 'json' object that was sent from the server
        function receivebooks(m) {
            books = m.data.json;
            showbooks();
        }

        // Handler to show books in webpage on page load
        document.addEventListener('DOMContentLoaded', showbooks);

        // Handler to receive messages from server via solution in IFrame
        window.addEventListener('message', receivebooks);

        // Handler to send data to IFrame on page load
        document.addEventListener('DOMContentLoaded', function() {
          var embedded = new WebAppIFrame(document.querySelector('iframe'));

            embedded.start('http://localhost:8083/run?d=/public/jsonBooks', books);
        });
    </script>
  </head>


  <body>
    <h4>An editable list of books in JSON format</h4>
    <h5>The book list, stored as a JSON object in the webpage:</h5>
    <pre><code><!-- The SHOWBOOKS function displays the book list here --></code></pre>
    <h5>The book list is displayed in the Iframe as soon as the HTML page is opened:</h5>
    <iframe frameborder="0"></iframe>
  </body>


</html>
```

しくみ

以下の説明では、異なる背景色が使用され仕組みのそれぞれの部分で起こっていることが説明されています (*Web ページ?ソリューション?ワークフロー*):

	Web ページ: *ユーザーアクションとHTML/JavaScript コードの作動方法*
	ソリューション: *IFrame 内のソリューションにより実行されるアクション*
	ワークフロー: *(MT デザインをベースにした) サーバー上の処理*

HTML ページのロード時:

books という名前のJavaScript 変数が読み込まれます。**books** という名前のJSON オブジェクトが含まれています。

```
var books = {
  "books": [
    {
      "author": "Mary Shelley",
      "title": "Frankenstein; or, The Modern Prometheus",
      "year": 1818,
      "pages": 280
    },
    {
      "author": "Bram Stoker",
      "title": "Dracula",
      "year": 1897,
      "pages": 302
    }
  ]
};
```

Web ページ内でブックリストを表示する

DOMContentLoaded イベントをリッスンするイベントリスナーとJavaScript 関数 (**showbooks**) を使用して**books** 変数のコンテンツがHTML **code** 要素内で表示されます:

```
document.addEventListener('DOMContentLoaded', showbooks);

function showbooks() {
  document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
}

<pre><code></code></pre>
```

HTML ドキュメントがロードされた時、Web ページ内のブックリストを確認することができます。

Web ページをロードする際に自動的に本のリストを自動的にIFrame に送信する

DOMContentLoaded イベントをリッスンするイベントリスナーは、自動的にロードする関数を定義します:

```
document.addEventListener('DOMContentLoaded', function() {
  var embedded = new WebAppIFrame(document.querySelector('iframe'));
  embedded.start('http://localhost:8083/run?d=/public/jsonBooks', books);
});
```

上で定義される関数は**WebAppIFrame.js** を呼び出すことで変数を作成します。JavaScript ファイルへの参照 (**script** 要素を参照してください) は前の JSON サンプルで必要とされていないことに注意してください。

```
<script src="http://localhost:8083/js4.0/WebAppIFrame.js"></script>
```

WebAppIFrame.js (*下のリスティング*) はソリューションをロードして、データを **$MT_EMBEDDEDMESSAGE** に送信します。ソリューションを開始する URL は IFrame の **src** 属性を与えられていませんが、**start** メソッドの最初のパラメータとしてパスされます。

WebAppIFrame.js のコードリスティング:

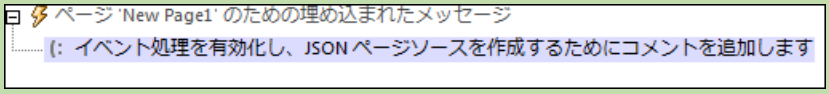
```
'use strict';

function WebAppIFrame(iframe, listener) {
  var _this = this;
```

```
  var _data;
  var _jwt;

  if (listener) {
    window.addEventListener('message', listener, false);
  }

  this.start = function(url, data, jwt) {
    function _start() {
      _this.post({data: _data, jwt: _jwt});
        iframe.removeEventListener('load', _start);
    }

    _data = data;
    _jwt = jwt;
  if (_jwt) {
      url += '&auth';
    }
      iframe.addEventListener('load', _start);
        iframe.src = url + '&embed';
  }

  this.post = function(data) {
    iframe.contentWindow.postMessage(data, '*');
  }
}
```

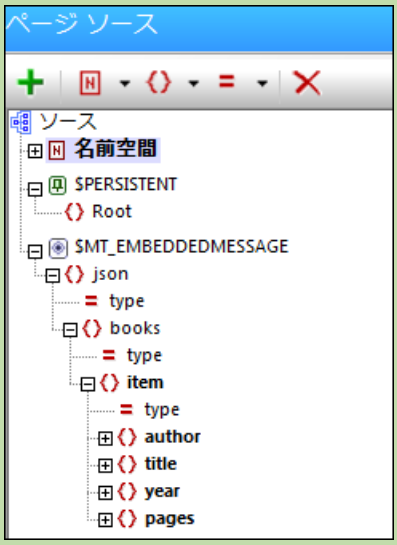
IFrame はソリューション **jsonBooks** をロードし、Web ページからデータを受け取ります。

{books} は(シリアル化された JSON フォームで)自動的にサーバー上のワークフローに送信されます。

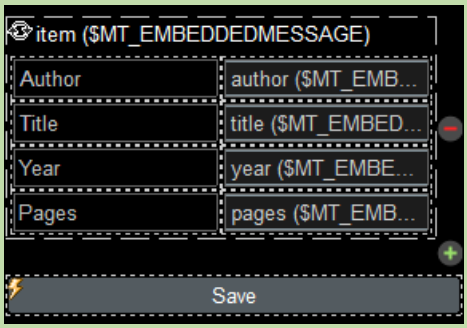
ページイベント **埋め込まれたメッセージ** には ..と定義されているアクションが存在するため、



... ページソース **$MT_EMBEDDEDMESSAGE** は自動的に **{books}** データと共にロードされます。

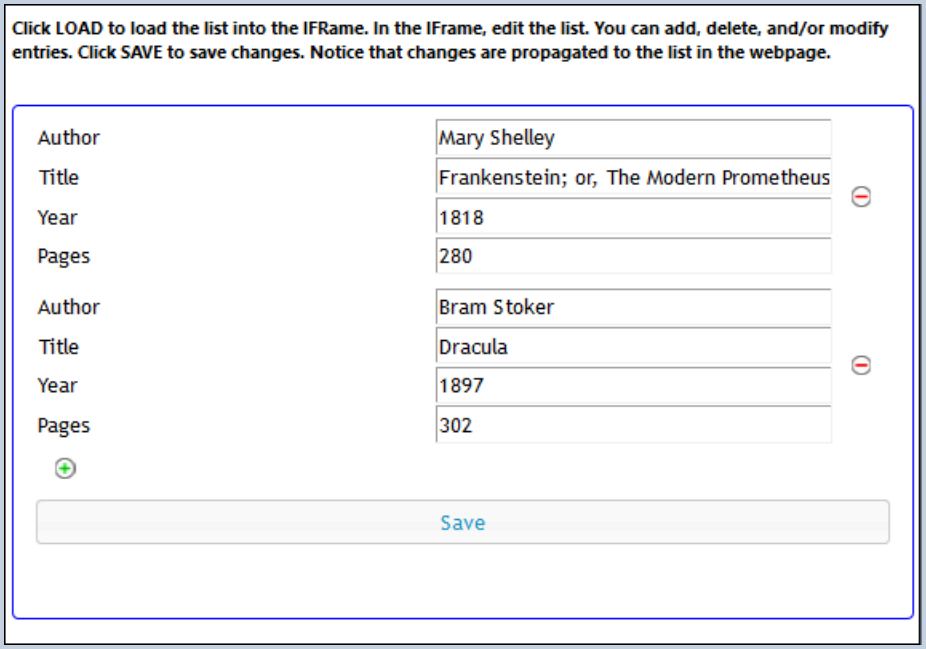


デザインは**item** ノードの繰り返しテーブルを含んでいます。デザインは**item** ノードの繰り返しテーブルを含んでいます。テーブルのセルはそれぞれ**author**、**title**、**year**、および**pages** ページソースノードにリンクされています。



この結果、...

...IFrame 内のソリューションがこのデータと共にロードされます。ワークフローの **$MT_EMBEDDEDMESSAGE** ページソース内のデータと共に繰り返しテーブルは作成されます。



HTML ページを開くとデータを IFrame 内で直接使用することができます。

JWT 認証

このセクションのJWT 認証サンプルは、前のセクション内の設定済みのページソースのWeb ページを編集します。ソリューションへの呼び出しと共に、JWT も送信されます。JWT は文字列として送信される必要があります (すなわち、引用符を用いて送信する必要があります)。下のコード リスティングでは、JWT は青でハイライトされています。

このサンプルで使用されるファイルは **JWT.html** と **jsonBooks.mtd** です。両方のファイルは次の場所にあります: *マイドキュメント*MobileTogether フォルダー: **MobileTogetherDesignerExamples\Tutorials\EmbeddedWebpageSolutions**。ファイルを試すには、MTD ファイルをサーバーにデプロイし、サーバーの設定内でJWT 認証を有効化します (*次のセクションを参照してください*)。**newuser** がサーバーに登録されていない場合、自動的にユーザーとしてインポートされ、ログインが成功します。**jsonBooks.mtd** のコンテナーにアクセスするには、パーミッションを設定をする必要があります。必要な場合は、正確なワークフローをターゲットするためにHTML コードを変更します。

このサンプルファイル内のJWT は **www.altova.com** に設定されたオーディエンスの要求、および (ユーザー名を指定する) **newuser** に設定されたサブジェクトの要求 を使用して作成されました。このJWT を生成するために使用された秘密は **gQkhVQPKkNYts3CraUsmmF6RyEvTCFnt** です。

サーバーの設定

サーバーがWeb ページにより送信されるJWT を非暗号化し検証するために、JWT 認証は次の2 つの設定と共にサーバーの設定内で有効化されている必要があります (*下のスクリーンショット*):

- JWT を生成するために使用される秘密: **gQkhVQPKkNYts3CraUsmmF6RyEvTCFnt**
- JWT を生成するために使用されるオーディエンスの要求の値: **www.altova.com**

メモ 更に、**newuser** が **jsonBooks.mtd** のコンテナーにアクセスできるようにパーミッションを設定することを忘れないでください。

HTML コード リスティング

ファイル **JWT.html** のHTML コード リスティング。JWT は青でハイライトされています。このサンプルで使用されるJavaScript 機能は、全てのブラウザーで使用できない場合があることに注意してください。この場合、ブラウザーに合うよう

にJavaScript を変更してください。

Web ページ コード リスティング

```
<DOCTYPE html>
<html>
  <head>
    <style>
      * {
        font-family:Segoe UI,Tahoma,Arial,Helvetica,sans-serif;
      }
      iframe {
        width:100%;
        height:400px;
        border:2px solid blue;
        border-radius:5px;
        margin:10px 0px;
      }
      code {
        font-size:small;
      }
    </style>
    <script src="http://localhost:8083/js4.0/WebAppIFrame.js"></script>
    <script>
      // The initial book list stored in a variable in JSON format
        var books = {
            "books":[
              {
                "author":"Mary Shelley",
                "title":"Frankenstein; or, The Modern Prometheus",
                "year":1818,
                "pages":280
              },
              {
                "author":"Bram Stoker",
                "title":"Dracula",
                "year":1897,
                "pages":302
              }
            ]
          };

        // Contents of variable books' converted to string and displayed inside HTML element CODE
        function showbooks() {
            document.querySelector('code').innerText = JSON.stringify(books, null, ' ');
          }

        // m = HTML message event; data = container for message from server
        // m.data.json = contents of the 'json' object that was sent from the server
        function receivebooks(m) {
            books = m.data.json;
            showbooks();
          }

        // Handler to show books in webpage on page load
        document.addEventListener('DOMContentLoaded', showbooks);

        // Handler to receive messages from server via solution in IFrame
        window.addEventListener('message', receivebooks);

        // Handler to send data to IFrame on page load
        document.addEventListener('DOMContentLoaded', function() {
          var embedded = new WebAppIFrame(document.querySelector('iframe'));

            embedded.start('http://localhost:8083/run?d=/public/jsonBooks', books,
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJPbmxpbmUgSlbUIEJ1aWxkZXIiLCJpYXQiOjE1MDE4M
zU4MzUsImV4cCI6MTUzMzM3MTgzNSwiYXVkIjoid3d3LmFsdG92YS5jb20iLCJzdWIiOiJuZXd1c2VyIiwiR2lZW
5OYW1IjoiSm9obm55IiwiU3VybmFtZSI6IUvY2tbCIsIkVtYWlsIjoianJvY2tbEBleGFtcGxlLmNvbSIsIUvbGUiOlsiT
```

```
WFuYWdlciIsIByb2plY3QgQWRtaW5pc3RyYXRvciJdfQ.M66SBrP_U30iQeheQWpPTdaBzsJZqK2L7BJQ8gKP-
Lo');
            });
    </script>
  </head>


  <body>
    <h4>An editable list of books in JSON format</h4>
    <h5>The book list, stored as a JSON object in the webpage:</h5>
    <pre><code><!-- The SHOWBOOKS function displays the book list here --></code></pre>
    <h5>The book list is displayed in the Iframe as soon as the HTML page is opened:</h5>
    <iframe frameborder="0"></iframe>
  </body>


</html>
```

チャプター 17

自動化されたテスト

17 自動化されたテスト

自動化されたテストの機能により、デザイン、ページソースデータ、コンポーネントスタイル、または、レイアウト、またはソリューションの環境の差異を検出するために、(基本的には、シミュレーションである)2つのテストランを比較することができます。

このプロセスの仕組みは以下のとおりです:最初に、デザインのベースのテストラン(または、**テストケース**)が記録されます。このテストのケースは、特定のユーザーとデザインのアクションにより作動します。テストケースは、次に異なるパラメーターを用いて再生されます(例えば、異なるページソースデータ、または、デバイスのOS の異なるバージョン)。MobileTogether Designer 内の再生がテストケースから、差異を返すと、記録され、ベースのテストケースと比較されます。記録された再生が、元のテストケースとは異なる、**テストラン**を呼び出します。問題が検出されると、指摘されます。更に、デザインのためのテストケースは、デザインと共に、MobileTogether Server にデプロイされることができます。これにより、テストケースは、再生のために複数のクライアントデバイスにダウンロードすることができます。クライアントデバイス上の再生はサーバに保存され、MobileTogether Designer 内で比較のために取得されます。

一般的な自動化されたテストのシナリオは以下のように進行します:

1. *テストケースの記録*: これは、異なる環境で再生することのできる記録されたテストケースの実行です。
2. *テストケースの再生*: 再生は、テストケースのテストランとして保存されます。再生が、MobileTogether Designer 内の場合、差異を返す再生のみがテストランとして保管されます。テストケースが、MobileTogether Server にデプロイされている場合、これらのクライアントの再生はサーバーに保管されます。
1. *テストランが基のテストケースと比較される場合*: 比較はMobileTogether Designer 内でおこなわれます。異なるレベルを構成することができ、差異を詳細に確認することができます。テストランは、クライアントの再生から返され、(サーバー上に保管され) MobileTogether Designer で比較のために取得されることができます。

ルーチンステップを実行するために自動化されたテストを使用する

特定のルーチンのステップをシミュレーションを実行する都度実行する場合、これらのステップはテストケースとして記録し、後に再生することができます。例えば、デザインは、ユーザーにログインの詳細、または変更しないデータの他のアイテムを入力するようにプロンプトする場合があります。その後、テストケースをルーチンのステップを完了し、追加のテストステップを手動で実行することができます。このように使用することにより、自動化されたテストは、デザイン中に時間を省くことができます。

自動化されたテスト メニューコマンド

プロジェクト メニューに自動化されたテスト機能を実行するコマンドがあります。この機能は、自動化されたテスト ツールバーにもあります(*下のスクリーンショット参照*)。

	新規テストケースの記録: シミュレーター 内で新規テストケースが開始され、ユーザーのアクションを記録します。記録が停止すると、名前をつけて記録を保存するようにプロンプトされます。 詳細に関しては次を参照してください: テストケースの記録 。
	テストケースの再生: 再生のために使用することのできるテストケースオプション コンボボックスで選択されたテストケースを再生します。再生が差異をテストケースから返す場合、再生は保存されます。*次を参照してください*: テストケースの再生。
	クライアント上のテストケースのトライアルラン: 接続されたクライアントで、再生のために使用することのできるテストケースコンボボックス内で選択されているテストケースを再生します。再生が差異を返すと、再生は保存されます。*次を参照してください*: テストケースの再生
	テストケースとテストランの管理: テストケースとテストランの管理 ダイアログを表示します。

*再生のために使用することのできるテストケース*コンボボックス*は、テストケースが記録された後のみ表示されます。ここで選択されているテストケースは、**「テストケースの再生」**または**「クライアント上のテストケースのトライアルラン」**がクリックされると実行されます。

このセクション
このセクションは以下のとおり整理されています:

- テストケースの記録
- テストケースの再生
- テストケースとテストランの管理
- テストケースをサーバーにデプロイする
- テストランの比較

17.1 テストケースの記録

このアクションにより、比較のためのベースのケースとして使用することのできるベースのテストケースが作成されます。

	テストケースの記録: シミュレーター内で新規のテストケースを開始し、テストケースを記録します。記録が停止すると、記録に名前をつけて保存するようにプロンプトされます。詳細に関しては、*テストケースの記録* を確認してください。
	記録の停止: 記録を停止して、記録されたテストケースの名前を指定する、記録されたテストケースの確認ダイアログを開きます。

テストケースの記録は、以下のようにおこないます:

1. アクティブなデザインをテストするためのデザインを作成します。
2. ツールバーアイコン **新規テストケースの記録** (*上のアイコンを参照*) をクリックします。
3. 表示されるシミュレーターウィンドウ内でテスト内に含むステップを実行します。手動でスナップショットを取ることもでき、このオプションは記録オプション内で有効化することができます (*下を参照*)。
4. テストケースを完了すると、(*右下の*) **「閉じる」** をクリックまたはツールバーアイコン **「新規テストケースの記録の停止」** をクリックします。(*上のアイコンを参照*)
5. 表示される記録されたテストケースの確認ダイアログ内にテストケースの名前を入力して、**「保存」** をクリックします。

各デザインのために、記録することのできる(ベースの) テストケースの数に制限はありません。

記録のオプション
テストケースとテストランの管理ダイアログの下の部分で記録および再生のためのオプションを使用することができます。差異を表示する再生は、自動的に記録され、保管されます。ここで設定する記録オプションは再生にも使用することができます。

記録オプション
記録オプションを使用することができます:

- *デザインアクションのログ:* デザインのアクションとは、ユーザーにより明示的にトリガーされないアクションを指します。例としては、ページソースにデータを自動的に保存するなどが挙げられます。このオプションがチェックされていると(デフォルト) デザインアクションがログされます。
- *各ステップの後のスナップショットを自動的に作成:* このオプションは、テストランの再生にではなく、テストケースの記録に適用されます。このオプションをチェックして、テストケースの後に自動的にスナップショットが作成することができます。スナップショットは、スナップショットに含まれる項目の選択オプションで選択されているアイテムが含まれます (*次のポイントを参照してください*)。
- *スナップショットに含まれる項目:* スナップショットに含む項目を選択します。オプション: ページソース、スタイル、および、クライアントビュー (クライアント上のデザインコンポーネントのレイアウトの座標)。これらのスナップオプションの1つを選択し、スナップショットを自動的に作成するオプションを解除すると、シミュレーター内の **「次スナップショットの記録」** ボタンがテストケースの記録中に有効化され、スナップショットを希望するタイミングで取ることができます。

再生オプション
以下の再生オプションを使用することができます:

- テストケースの再生の速度: テストケースの再生の速度を選択することができます。ステップバイステップを選択すると、シミュレーター内の **「次のステップの再生」** ボタンが再生中に有効化され、テストをこのスピードで実行することができます。**「次のステップの再生」** をクリックしてテストを順番を追って実行します。
- 永続データのリセット: 各テストケースには、永続データを元の値にリセットするリセットチェックボックスが存在します。チェックされると、テストケースが再生され、クライアント上の永続データは元の値にリセットされます。

17.2 テストケースの再生

(*テストケースの記録*) テストケースが記録されると、(例えば、異なるページソースデータ または、デバイスの OS の異なるバージョンを使用するなど) 異なるパラメーターを使用してテストケースを再生することができます。再生が差異を返す場合、閉じられる際に再生は自動的にデザインに保存されます。それ以外の場合は、保存されません。保存された再生は、*テストケースとテストランの管理ダイアログ*内で、ベースのテストケースの関連付けられたテストケースとして表示されます (*次を参照: テストケースとテストランの管理*)。

デザイナーとクライアント内での再生

MobileTogether Designer または *クライアントデバイス上*でテストケースを再生することができます。これをおこなうには、ツールバーから対応するアイコンをクリックしてください (*下のスクリーンショット参照*)。これらはそれぞれ、 **「テストケースの再生」** と **「クライアント上のテストケース」** です。

	*テストケースの再生: 再生のために使用することのできるテストケースオプション コンボボックス*で選択されたテストケースを再生します。再生が差異をテストケースから返す場合、再生は保存されます。
	クライアント上のテストケースのトライアルラン: 接続された*クライアント*で、*再生のために使用することのできるテストケースコンボボックス*内で選択されているテストケースを再生します。再生が差異を返すと、再生は保存されます。

再生のオプション

テストケースとテストランの管理ダイアログ の下の部分で記録および再生のためのオプションを使用することができます。差異を表示する再生は、自動的に記録され、保管されます。ここで設定する記録オプションは再生にも使用することができます。

記録オプション
記録オプション を使用することができます:

- *デザインアクションのログ:* *デザインのアクション*とは、ユーザーにより明示的にトリガーされないアクションを指します。例としては、ページソースにデータを自動的に保存するなどが挙げられます。このオプションがチェックされていると (デフォルト) デザインアクションがログされます。
- *各ステップの後のスナップショットを自動的に作成:* このオプションは、テストランの再生にではなく、テストケースの記録に適用されます。このオプションをチェックして、テストケースの後に自動的にスナップショットが作成することができます。スナップショットは、スナップショットに含まれる項目の選択オプションで選択されているアイテムが含まれます (*次のポイントを参照してください*)。
- *スナップショットに含まれる項目:* スナップショットに含む項目を選択します。オプション: ページソース、スタイル、および、クライアントビュー (クライアント上のデザインコンポーネントのレイアウトの座標)。これらのスナップオプションの1つを選択し、スナップショットを自動的に作成するオプションを解除すると、*シミュレーター* 内の **「スナップショットの記録」** ボタンがテストケースの記録中に有効化され、スナップショットを希望するタイミングで取ることができます。

再生オプション
以下の再生オプションを使用することができます:

- テストケースの再生の速度: テストケースの再生の速度を選択することができます。ステップバイステップを選択すると、*シミュレーター* 内の **「次のステップの再生」** ボタンが再生中に有効化され、テストをこのスピードで実行することができます。**「次のステップの再生」** をクリックしてテストを順番を追って実行します。

- 永続データのリセット: 各テストケースには、永続データを元の値にリセットするリセットチェックボックスが存在します。チェックされると、テストケースが再生され、クライアント上の永続データは元の値にリセットされます。

MobileTogether Designer 内でのテストケースの再生

MobileTogether Designer 内でテストケースを再生するには、以下を行います:

1. *再生のための使用することのできるテストケースオプション* コンボボックス内から、(*上のスクリーンショットのツールバー参照*) 再生を希望するテストケースを選択します。ベースの テストの実行 のみが再生されることに注意してください。
2. **テストケースの再生** をクリックします(*上のスクリーンショットのツールバー参照*)。

シミュレーター内で再生が実行されます。再生が差異を返す場合、閉じられる際に再生は自動的に(関連付けられたテストの実行として)保存されます。

クライアントデバイス上でのテストケースの再生

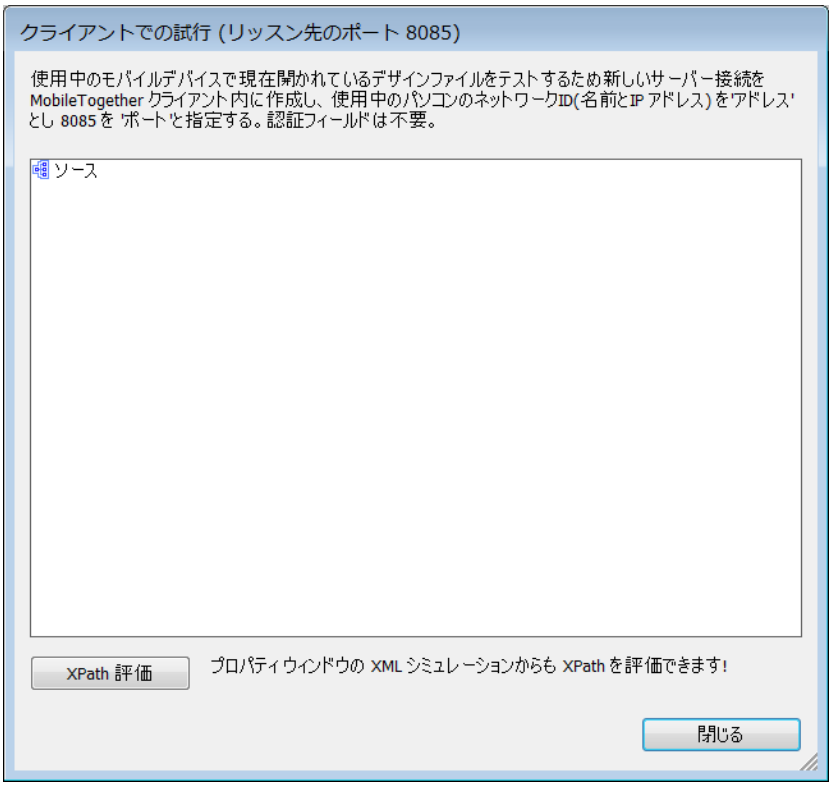
クライアント上でテストの実行の試行をおこなうことにより、特定のクライアントデバイスでテストの実行の機能が正確に作動しているかプレビューし、確認することができます。この場合、 MobileTogether Designer マシンは、MobileTogether Server の役割を果たします。これは、(サーバーとしての役割を果たす) デザイナーマシンがクライアントに接続するための個別のポートを設定する必要があることを意味します。

セットアップは2つの部分から構成されています:

- MobileTogether Designer, 内で、**ツール**| **オプション**| **クライアント上のトライアルラン** に移動して、クライアントデバイスとの通信のために個別のポートをセットアップします。: 例えば、**8089**.
- クライアントデバイス上では、デザイナーマシンを新しいサーバーとして追加します。必要条件: (i) (DOS コマンド **ipconfig** を使用して確認することのできる) マシンの IP アドレス、と(ii) 前のステップ内で構成されているポートの番号

テストケースを再生するには、以下を行います:

1. *再生のために使用することのできるテストケース* コンボボックス (*上のスクリーンショットのツールバー参照*) 内から、再生するテストケースを選択します。テストの実行のみを再生することができることに注意してください。
2. **クライアント上のテストケース** をクリックします。(*上のスクリーンショットのツールバー参照*) 「クライアント上のトライアルラン」ダイアログ が開かれます (*下のスクリーンショット参照*)。



3. *クライアントデバイスで、デザイナーマシンを接続し、ソリューションを更新します。*現在、MobileTogether Designer 内で*アクティブなすべてのソリューション*が表示されます。
4. *クライアントデバイスで、クライアントデバイス上でテストケースを再生するためのソリューションを開始します。*

テストケースは、*クライアント上で*再生されます。 MobileTogether Designer 内で「*クライアント上のトライアルラン*」ダイアログ(*上のスクリーンショット参照*) は、再生が展開されると共にページソースデータと共に更新されます。再生が差異を返す場合、閉じられる際に再生は自動的に(関連付けられたテストランとして)保存されます。差異が存在するかを確認するには、*自動化されたテストツールバー内の***記録の管理**をクリックします。

17.3 テストケースとテストランの管理

テストケースとテストランの管理 ダイアログ (下のスクリーンショット) で以下をおこなうことができます:

- テストケースのために記録のオプションをセットアップします。
- その後のテストランのために記録と再生のオプションをセットアップします。
- 個々のテストケースのテストランは、永続データを元のデータにリセットすることから始まります。
- MobileTogether 記録ファイル(`.mtrecord` ファイル)を再ロードして保存します。デザインファイルごとに1つの `.mtrecord` ファイルが常に存在し、デザインファイルと同じ名前がつけられています(がファイルの拡張子は異なります)。
- テストランを比較します。
- テストケースは関連付けられたテストケースと置き換えられます。
- テストケースをテストランの1つと置き換えます。テストランはテストケースのロールを取ります。他のテストランは削除され、選択されたテストランは、空のグループの新規テストケースとなります。
- MobileTogether Server にベースのテストケースをデプロイし、サーバーからテストケースを取得し、サーバーからテストケースを削除します。

ダイアログは、記録された各テストケース および、関連したテストラン(再生)を関連したデータと共に表示します。データと時刻は、言語などの他の関連したクライアント設定などと同様クライアントデバイスから取得されます。

記録と再生のオプション

テストケースとテストランの管理ダイアログ の下の部分で記録および再生のためのオプションを使用することができます。差異を表示する再生は、自動的に記録され、保管されます。ここで設定する記録オプションは再生にも使用することができます。

記録オプション
記録オプション を使用することができます:

- *デザインアクションのログ:* デザインのアクションとは、ユーザーにより明示的にトリガーされないアクションを指します。例としては、ページソースにデータを自動的に保存するなどが挙げられます。このオプションがチェックされていると(デフォルト) デザインアクションがログされます。
- *各ステップの後のスナップショットを自動的に作成:* このオプションは、テストランの再生にではなく、テストケースの記録に適用されます。このオプションをチェックして、テストケースの後に自動的にスナップショットが作成することができます。スナップショットは、スナップショットに含まれる項目の選択オプションで選択されているアイテムが含まれます(*次のポイントを参照してください*)。

- *スナップショットに含まれる項目:* スナップショットに含む項目を選択します。オプション: ページソース、スタイル、および、クライアントビュー(クライアント上のデザインコンポーネントのレイアウトの座標)。これらのスナップオプションの1つを選択し、スナップショットを自動的に作成するオプションを解除すると、シミュレーター 内の **「スナップショットの記録」**ボタンがテストケースの記録中に有効化され、スナップショットを希望するタイミングで取ることができます。

再生オプション
以下の再生オプションを使用することができます:

- テストケースの再生の速度: テストケースの再生の速度を選択することができます。ステップバイステップを選択すると、シミュレーター 内の **「次のステップの再生」**ボタンが再生中に有効化され、テストをこのスピードで実行することができます。**「次のステップの再生」**をクリックしてテストを順番を追って実行します。
- 永続データのリセット: 各テストケースには、永続データを元の値にリセットするリセットチェックボックスが存在します。チェックされると、テストケースが再生され、クライアント上の永続データはお元の値にリセットされます。

テストケースと関連した テストラン

テストケースの名前は、記録プロセスの最後に定義されます。テストケースとテストランの管理 ダイアログ(*上のスクリーンショット*) を使用し、名前をダブルクリックし名前を編集することができます。各テストケースとテストランは、内部 ID により識別されます。ですから、テストケースとテストランがファイルに保存されると(*次のセクションを参照*)、テストケースは、同様に識別されますが(内部 ID を基本にして)異なる名前がつけられ、名前の変更が(内部 ID をベースにして)識別されます。

テストケースから差異を返す再生は、テストケースと**関連した** テストラン(再生されたテストケース)として保管されます。保管される際に`Test Run` のデフォルトの名前を自動的に与えられます(上のスクリーンショット参照)。これらの名前は、ダブルクリックして編集することにより後に変更することができます。

テストケースとテストランの管理 ダイアログ内では、(*上のスクリーンショット*) テストランはもとのテストケースに依存すると考えられます。関連したテストランを表示、または、非表示するために、テストケースを展開、または、折りたたむこともできます。テストケースが削除されると、すべての関連付けられたテストランも削除されます。

入力のステップをマージする

テストランではなく、テストケースが選択されると、このボタンが有効化され、編集フィールドのみに適用されます。編集フィールドの**入力時** イベントのためにアクションが存在する場合、編集フィールド内でのキーストロークの全てが個別のステップとして記録されます。全てのキーストロークを編集フィールド全体のための単一のステップにマージするには、**入力のステップをマージする**をクリックします。このマージを元に戻すことはできないことに注意してください。

永続データのリセット

テストケースが再生されるとデザインの永続データを元の値にリセットするには、テストケースのリセットチェックボックスを選択します(上のスクリーンショット内の「テストケースとテストランの管理」ダイアログを参照してください)。

テストケースとテストランのロードと保存

すべてのテストケースとそのテストランは、`<mtdesignfilename>.mtrecord` に保存されることができます。各 MobileTogether デザインには、関連した`.mtrecord` ファイルが1つ存在します。例えば、デザイン ファイル `QS01.mtd` には、関連付けされた`QS01.mtrecord` という名前のMobileTogether 記録ファイルが存在します。テストケースとテストランの管理 ダイアログの **「保存」**ボタンをクリックすると、MT 記録ファイルは、デザインファイルと同じファイルに保存されます(*下のスクリーンショット*)。MT 記録ファイルを次に保存すると、既存のMT 記録ファイルを上書きします。

「全ての再生を削除する」をクリックすると、全てのテストケースのテストランを削除することができます。

テストケースとそのテストランをアクティブなデザインのMT 記録ファイルから再ロードする場合は、 **「再ロード」** をクリックします。この場合、ダイアログのテストケースとテストランは、MT 記録ファイル内の既存のファイルと置き換えられます。

テストランのプロパティ
テストランのプロパティを確認するには、選択して、 **「選択項目を比較」** をクリックします。テストケースとテストランを比較ダイアログ内にテストランのプロパティが表示されます。

テストケースの削除と比較

1つ以上のテストケース/ランを削除する場合、チェックボックスを選択して、 **「選択項目を削除」** をクリックします。テストケースを選択すると、すべてのテストランが明示的に選択され、削除されます。しかしながら、警告が表示され、継続するようにプロンプトされます。

テストランがテストケースと異なる場合、比較ペイン内で差異の部分がピンク色のボタンで表示されています (*上のスクリーンショットを参照*)。例えば、テストランのアクションが異なる場合、テストランのアクションセルにはピンク色のボタンが含まれています (*上のスクリーンショットを参照*)。ピンクボタンをクリックする場合、テストランとテストケースが表示されるテストケースとテストランの比較ダイアログが表示されます。このダイアログを開くには、 **「選択項目の比較」** をクリックします。

関連したテストランをテストケースに変更する

テストランをテストケースに変換するには、選択した後、 **「レファレンスを置き換える」** をクリックします。選択されたテストランが、元のテストケースと置き換えられ、テストケースの名前を取ります。置き換えられたテストケースと関連付けられた他のすべてのテストランは、削除されます。アクションが実行される前に、アクションをキャンセルすることのできる、警告メッセージが表示されます。

サーバーへのデプロイ

アクティブなデザインの1つ以上のテストケースをサーバーにデプロイすることができます:

- **「ファイル | MobileTogether Server にデプロイする」** コマンドを使用してデザインとテストケースをデプロイします。
- デザインが既にデプロイ済みの場合、テストケースのみデザイン無しでをデプロイします。テストケースとテストランの管理ダイアログ内の **「デプロイ」** をクリックします (*上のスクリーンショット参照*)。

テストケースがサーバーにデプロイされると、サーバー上でアクティブ化されます。ソリューションがクライアント上で開始される都度、再生することができます。複数のテストケースがアクティブになると、これらのテストケースは、クライアント上で生成されます。 再生の結果はサーバー上に保管されます。テストケースとテストランの管理ダイアログ内の **「取得」** をクリックすることでこれらの再生をMobileTogether Designer で取得することができます (*上のスクリーンショット参照*)。取得された再生は、デプロイされたテストケースの関連付けられたテストケースとして保管され、他の関連付けられたテストケースと同じ方法で扱うことができます。

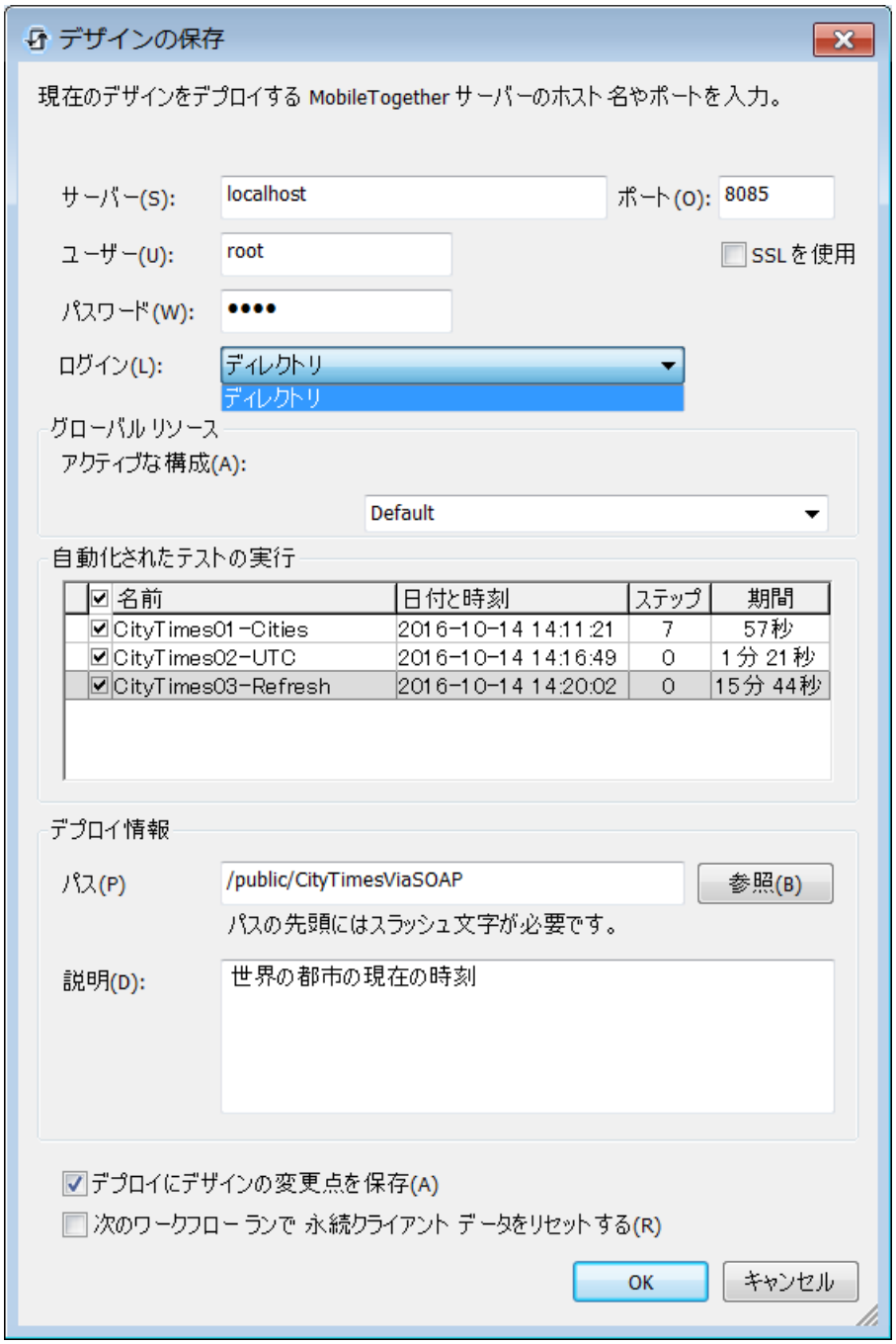
サーバーからデプロイされたテストケースを削除することができます。以下をおこないます: テストケースとテストランの管理ダイアログから(*上のスクリーンショット参照*) 削除するテストケースを選択し、 **「デプロイされた項目を削除する」** をクリックします。

この操作の詳細に関しては、サーバーへのデプロイを参照してください。

17.4 サーバーへテストケースをデプロイする

アクティブなデザインの1つまたは複数のベースのテストケースをサーバーにデプロイすることができます。テストケースがサーバー上でアクティブ化されると、クライアント上でソリューションが開始する都度再生することができるようになります。このように、テストケースは複数のクライアント上で再生することができます。これらの再生はサーバーに保管され、元のテストケースと比較するために、MobileTogether Designer から取得することができます。

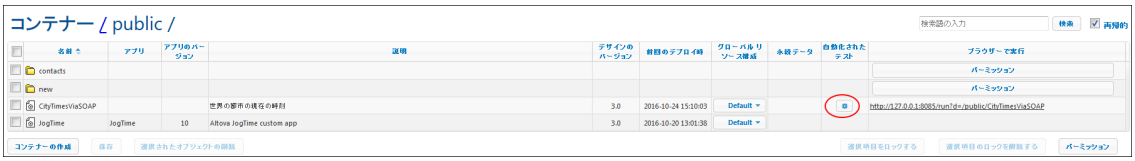
テストケースを MobileTogether Server にデプロイする
テストケースをデザインと共に **「ファイル | MobileTogether Server へデプロイ」** コマンドを使用して MobileTogether Server にデプロイすることができます。このコマンドは次の項目を入力することのできるダイアログを表示します (*下のスクリーンショット*) : (i) サーバーへのアクセスの詳細、(ii) サーバー上のデザインへのパス、および (iii) デザインと共にデプロイするテストケース (デプロイするテストケースを選択してください)。



デザインがすでにサーバーにデプロイされている場合、テストケースとテストランの管理ダイアログ内の **デプロイ** をクリックして、テストケースを更にデプロイすることができます。テストケースは内部 ID により識別され、同じ ID を持つテストケースのみにより上書きされます。

サーバー上でのテストケースの管理

サーバー上のソリューションがテストケースと共にデプロイされていると、ワークフロータブ内で (*下のスクリーンショット*)、ソリューションには自動化されたテストの列内にホイールのアイコンが表示されます (*スクリーンショットを参照*)。



デザインのテストケースを有効化し、クライアント上でテストケースがどのように再生するか定義するには、ソリューションのホイールアイコンをクリックしてください（*上のスクリーンショットに表示されています*）。これは、そのソリューションの自動化されたテストを示すページを表示しています（*下のスクリーンショット*）。

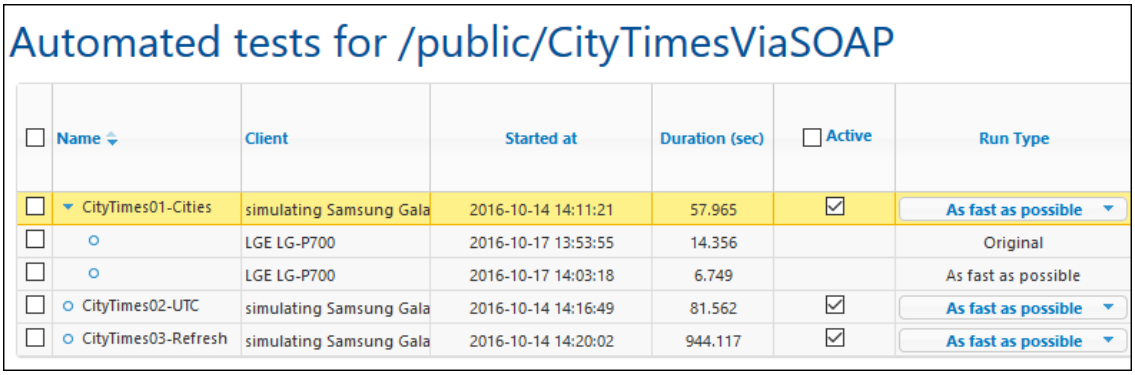
自動化されたテストページは、選択されたソリューションのためにデプロイされたすべてのテストケースを表示しています。クライアントデバイス上での再生のための個々のテストケースを次のようにセットアップすることができます:

1. 「アクティブ」列内から、アクティブ化するテストケースをチェックします。これらのテストケースは、ユーザーがソリューションを開始するとクライアント上で再生されます。複数のテストケースが選択された場合、選択されたテストケースすべてが再生されます。ソリューションのテストケースがアクティブ化されていると、ワークフローのページで、デザインの「自動化されているテスト」の列に赤いホイールのアイコンが表示されます。
2. 再生中のログの詳細を設定します。希望する列のチェックボックスをチェックしておこないます。 テストケースとテストランの管理ダイアログ 内の記録と再生オプションのセクションを参照してください。
3. **「保存」**をクリックして完了します。

テストケースを削除する場合は、左側の列のチェックボックスを選択して、 **「選択項目の削除」**をクリックしてください。サーバー上のテストケースを削除する場合は、(MobileTogether Designer) テストケースとテストランの管理ダイアログ内で削除する項目を選択し、 **「選択項目の削除」**をクリックします。

クライアント上での再生の後

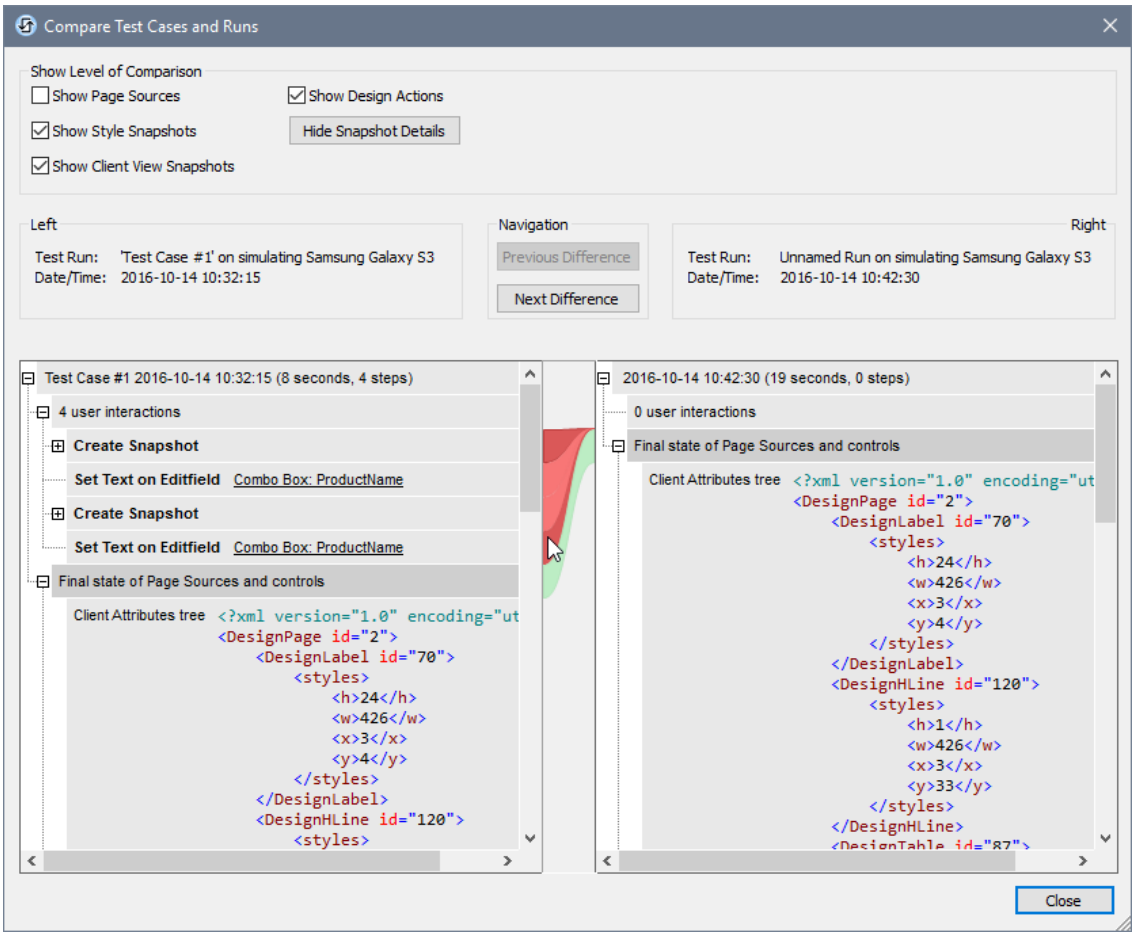
クライアント上でテストケースが再生されると、再生の結果はサーバーにテストケースとして保管され、自動化されたテストページに表示されます（*下のスクリーンショット*）。各再生の結果は、テストケースの子孫として表示されます。例えば、下のスクリーンショットでは、2つの再生がテストケース`CityTimes01-Cities`のために保管されています。再生が発生するクライアントデバイスと再生の回数も表示されていることに注意してください。 テストランを削除するには、左側の列のチェックボックスを選択して、 **「選択項目の削除」**をクリックしてください。



(MobileTogether Designer の) テストケースとテストランの管理ダイアログ 内の **「取得」**をクリックして MobileTogether Designer へテストランを取得することができます。取得はデザインのデプロイされたすべてのテストランに適用され、サーバーからのテストランの削除は含まれません。取得されたテストランは、構造的にMobileTogether Designer 内の元のテストケースに表示され、他の関連したテストランと同様に扱われます。例えば、元のテストケースと抽出されたテストランを比較することができます。

17.5 テストの実行の比較

テストランを元のテストケースと比較して、ステップ、アクション、ページソースデータ、スタイル、およびクライアントのビュー(特定のクライアント内のデザインのレイアウト座標コンポーネント)の差異を確認することができます。テストケースとテストランの管理ダイアログ内から、比較する2つのテストランを選択し、**「チェック項目の比較」**をクリックします。テストケースとテストランの比較ダイアログ(下のスクリーンショット)が表示されます。



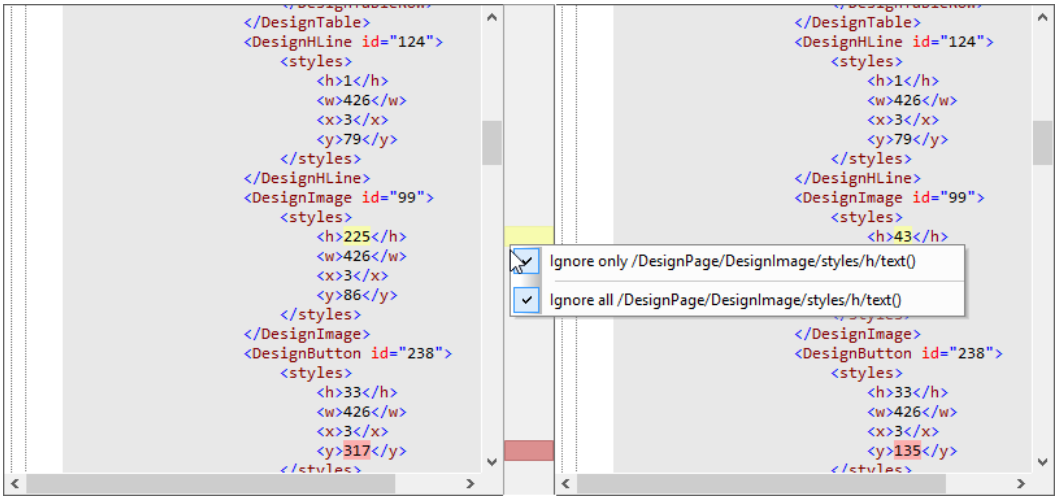
各テストランが異なるペインにツリーとして表示されます(上のスクリーンショット参照)。それぞれの詳細はペインの上に表示されています。ペイン内では、各ツリーは、テストケース(ステップ)とデザインアクションの時系列のシーケンスとして構成されています。スナップショットが、テストケース中に作成された場合、(例えば、ユーザーアクションの後)、スナップショットの詳細はその時点で表示されます。各スナップショットは以下から構成されています: (i) ページソースデータ (ii) スタイルの情報、と (iii) クライアント上のデザインコンポーネントのレイアウト座標。

テストケースとテストランの比較 ダイアログで以下をおこなうことができます:

- 次の比較レベルの表示/非表示: (i) デザインアクション、(ii) ページソースデータ (iii) スタイル、と(iv) クライアントのデザインコンポーネントのレイアウト座標。これらのレベルの1つを表示、または、非表示にするには、それぞれチェックとチェックの解除チェックボックスをチェックしてください。
- 上下のスクロールバーをドラッグし、または、**「前の差異」**と**「次の差異」**ボタンをクリックすることにより、ドキュメント内を移動することができます。
- 他のテストケース内のアイテムの状況をコピーする場合、カーソルでそのアイテムの中央の列をポイントしてください。状況が両方のペインをつなぐ赤いコネクターで表示されます(上のスクリーンショット参照)。
- 両方のツリー内の最後のアイテムは、それぞれの最後の状態を表しています。最後の状態に差異が存在しない場

合、双方の間のコネクタは緑色で表示されます。それ以外の場合は赤で表示されます。

- 差異に関しては、指定されたページソース、または、スナップショットデータノードを無視してください。2つのペインの間の列内の差異コネクタを右クリックし、**<このノードのみを無視** または **<これらのノードすべてを無視** を選択します(*下のスクリーンショット参照*)。無視されるノードは黄色でハイライトされます。これらのノードは、今後の再生で無視する差異として考えられます。これらのノード内の差異を考慮して、**<このノードのみを無視** または **<これらのノードすべてを無視** の選択を解除してください。



- 両方のツリーの最後のアイテムは、最後の状態に相対します。最後の状態に差異がない場合、2つのツリーの間のコネクタは緑で表示され。それ以外の場合は、赤で表示されます。

チャプター 18

AppStore App

18 AppStore App

アプリストアにポストしエンドユーザーがダウンロードするための MobileTogether アプリを作成することができます。MobileTogether Designer 内でカスタム化された **AppStore App** の作成方法は、以下のとおりです:

1. アプリを生成する MobileTogether Designer で、デザインしてテストを行うことができます。他の MobileTogether Designer デザインと同様にプロジェクトを作成します。
2. **[ファイル | AppStore App のためにプログラムコードを生成する]** コマンドを使用してデザインから AppStore App のためのプログラムコードの生成を行います。プログラムコードは、Android、iOS、Windows Phone および Windows のために生成することができます。
3. MobileTogether Server にプロジェクトをデプロイする。他の プロジェクトのデプロイと同じ方法で行います。
4. それぞれのデバイスとオペレーティングシステムのために AppStore App を作成するために生成されたプログラムコードをコンパイルする

プログラムコード生成 とコードのコンパイル の手順はこのセクションのそれぞれのサブセクションで説明されています。

AppStore App とMobileTogether Client 上のソリューションの違い

AppStore App は、MobileTogether ソリューションと異なります。

- AppStore App は、MobileTogether ソリューションと異なります。MobileTogether ソリューションは MobileTogether Server にデプロイされますが、MobileTogether Client を介してもアクセスすることができます。複数のクライアントデバイスは、1 つまたは複数の MobileTogether Server 上の1 つまたは複数のソリューションにアクセスすることができます。
- AppStore App は、実行するために構成する必要のない直接1 つのワークフローにアクセスする、軽量の代替を与えます。 ソリューションとAppStore App 間の違いは下のテーブルにリストされています。

ソリューションとAppStore App の違いは下のテーブルにリストされています。

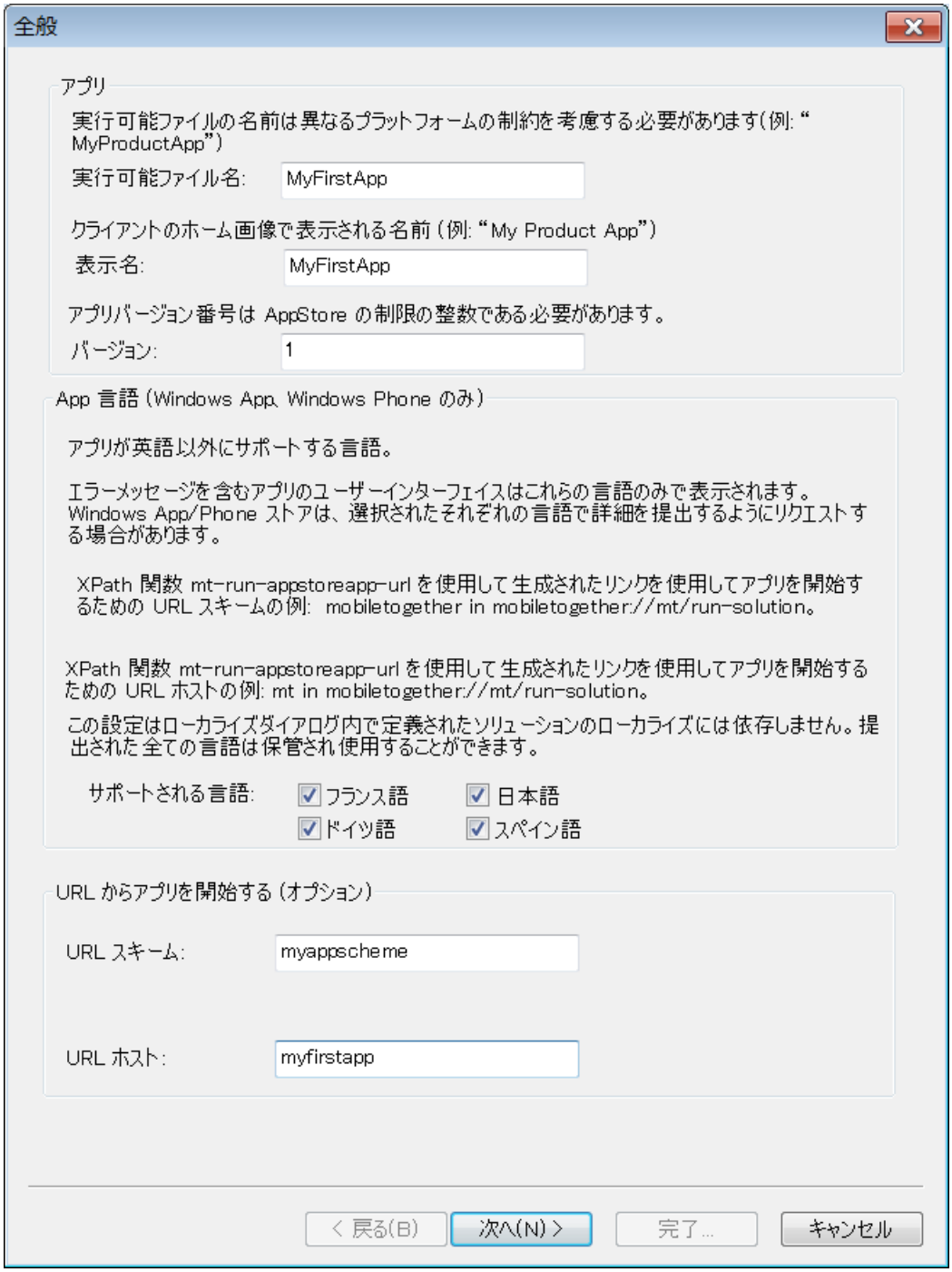
MobileTogether Client 上のソリューション	AppStore App
MobileTogether Client は、アプリストアからダウンロードされます。	AppStore App は、アプリストアからダウンロードされます。
MobileTogether Designer プロジェクトは、MobileTogether Server にソリューションとしてデプロイされます。	MobileTogether Designer プロジェクトは、MobileTogether Server にワークフローとしてデプロイされます。
MobileTogether Client は、上の1 つまたは複数の MobileTogether Server にアクセスします。	AppStore App は、関連したワークフローにアクセスします。1 つのワークフローに専用のアプリです。
エンドユーザーは構成を行い MobileTogether Client を維持する必要があります。複数のサーバーとソリューションにアクセスすることができます。	MobileTogether Client は、必要ありません。AppStore App は、エンドユーザーに簡単かつシンプルな単一のワークフローへのアクセスを提供します。

(C) 2018 Altova GmbH

18.1 プロジェクトからプログラムコードを生成する

Android、iOS、Windows Phone、およびWindows で作動するモバイルデバイスのAppStore App のためにプログラムコードを生成するには、 コマンド **[ファイル| AppStore App のためにコードを生成]** をクリックします。コマンドは、すべてのモバイルフォーマットが選択された場合7つのスクリーンを持つウィザードを開きます。各スクリーンは生成されるコードのための設定を含みます。

- 1: 全般: 名前、バージョン、言語、URL

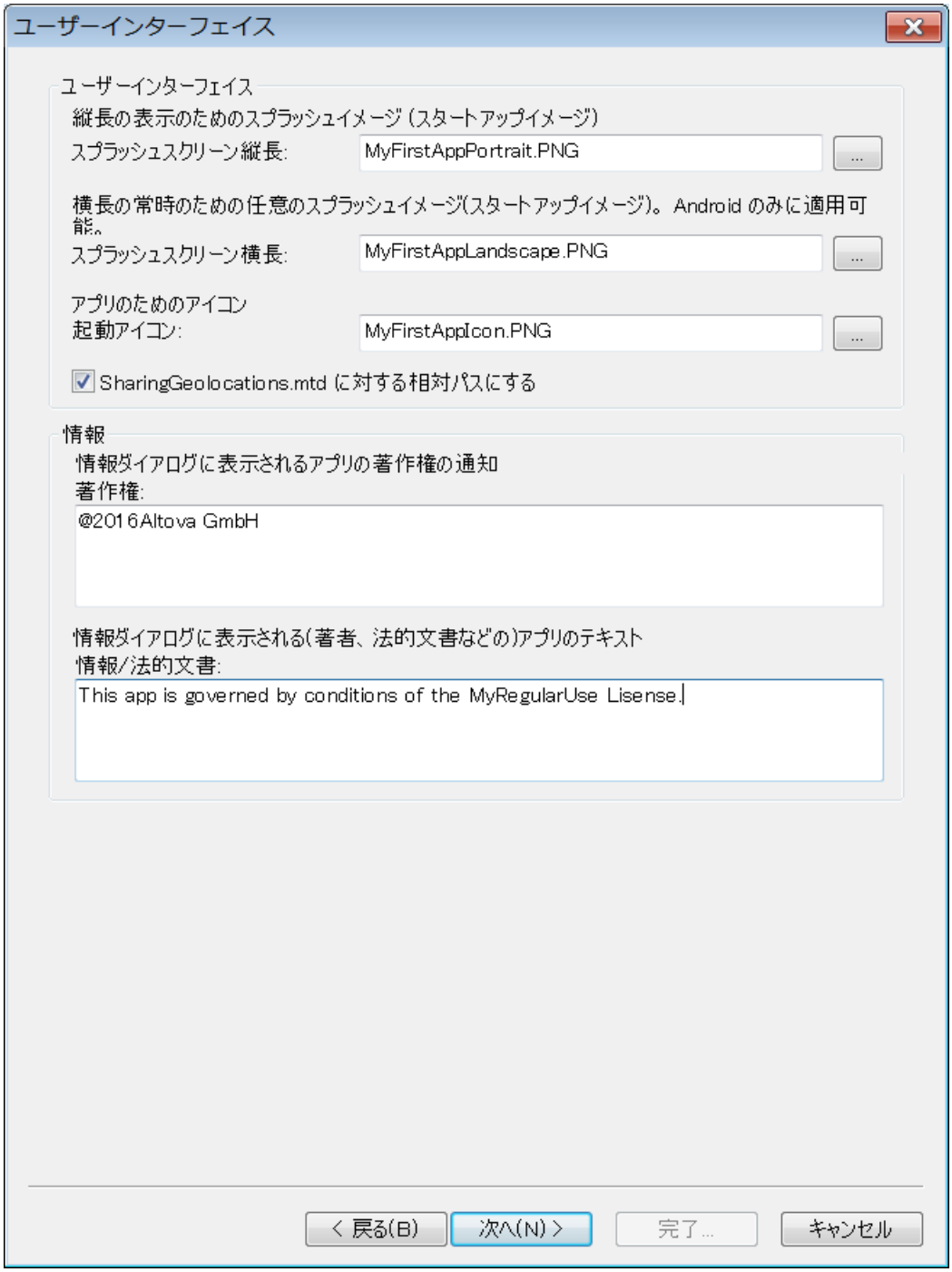


- *実行可能ファイル名:* 内部で使用され、コードにより参照される名前です。空白文字を使用せずに名前を使用してください。
- *表示名:* ユーザーが見ることのできるアプリの名前。
- *バージョン:* アプリのバージョン番号です。整数、または、小数である必要があります。例: `1` または `1.0` または `1.1` または `1.21`。(例えば、アプリがストアにより許可されず、デザインが変更されコードを再生成する場合など) 全てのプラットフォームのためにデプロイしない場合、次をおこなうことが奨励されます。バージョン番号を1つ上げ、プラットフォームのためプログラムコードを生成します。例: **`v1.2`** がiOS 以外の全てのプラットフォームに許可

された場合、(他のストアに提出されていない場合) **v1.3** を iOS のために作成します。**v1.4** は全てのプラットフォームのために作成作成され全てのストアにより許可された場合に使用することができます (非 iOS は **v1.2** から**v1.4** に移動します)。

- *アプリの言語 (Windows App とWindows Phone のみ):* アプリのインターフェイスは**EN**、**ES**、**FR**、**DE**、**JA** で表示することができます。アプリ内に含む言語を選択します。モバイルデバイスの言語が選択された言語の1つの場合、アプリはその言語で表示されます。モバイルデバイスの言語が、選択された言語に含まれていない場合、アプリのデフォルトの言語である英語で表示されます。アプリインターフェイスの言語はソリューション文字列のローカライズには依存しません。
- *URL スキームとホスト:* ハイパーリンクからアプリを開始するURL。ハイパーリンクのターゲットURL は、以下のフォーマットを使用します: **`<url-scheme>://<url-host>`**。一意のURL スキーム と一意のURL ホストを入力します。スキームの情報は、アプリのマニフェストファイル内に保管され、このスキームを使用して開始するURL を開くために使用されるアプリを使用するデバイスを示します。このスキームを持つURL を持つリンクがタップされると、デバイスはURL 、つまりアプリ によりポイントされたリソースにアクセスします。

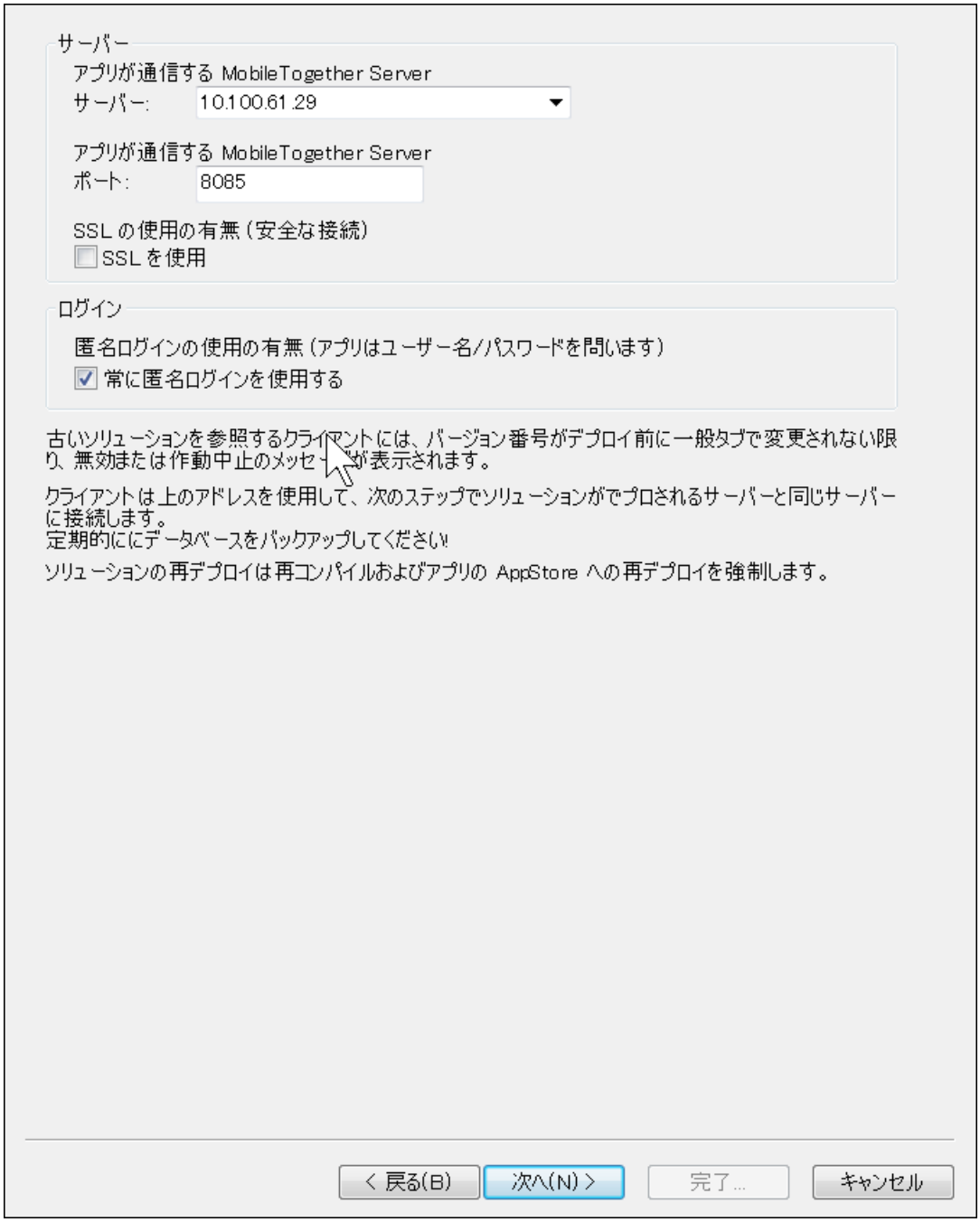
▾ 2: ユーザーインターフェイス: アイコン、著作権、法的



- *スプラッシュスクリーン:* 縦長と横長の向きでのスプラッシュスクリーンを参照することができます。モバイルデバイスのスプラッシュスクリーンはモバイルデバイスの向きにより変更されます。iOS アプリのために、縦長のスプラッシュスクリーンが両方の向きに合わせて描かれます。向き別に、または、デバイス別にスプラッシュスクリーンを希望する場合、異なる次元の異なるスプラッシュスクリーンをここで説明されるとおり生成されたプログラムコードをビルドする前に割り当てます。
- *起動アイコン:* モバイルデバイス内に表示されるアプリを開始するためのアイコン。最大のピクセルのサイズは、`200x200` です。

- *著作権と法的なテキスト:* モバイルデバイス内に表示されるテキスト。

3: サーバー: サーバーとログイン設定



- *サーバー:* ワークフローがデプロイされるサーバーの IP アドレス。
- *ポート:* アプリがアクセスされるサーバーのポート。サーバーのクライアントデバイスアクセスポートは MobileTogether Server 内で設定されます。詳細は MobileTogether Server ユーザーマニュアルを参照してください。

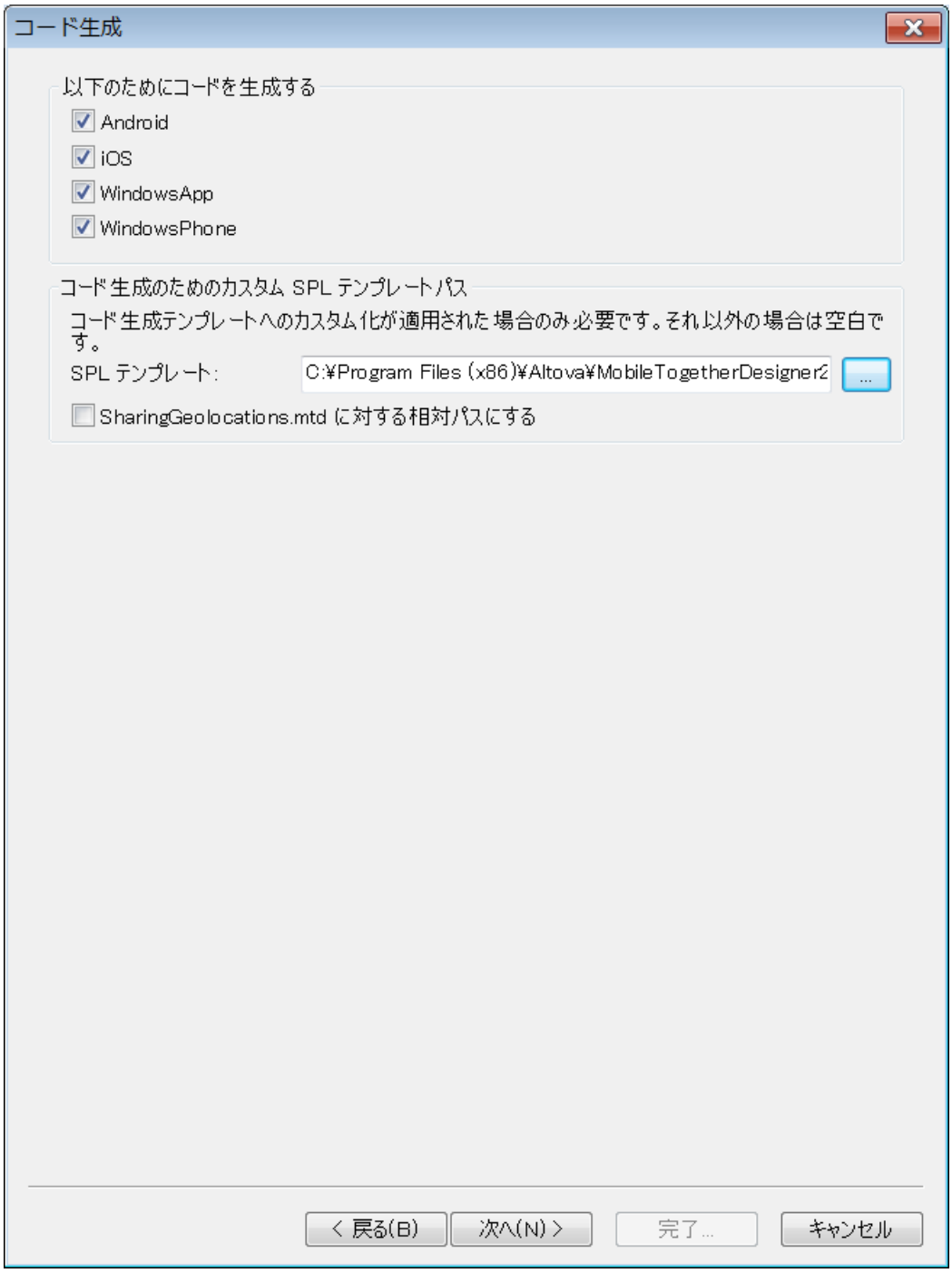
- *SSL:* SSL を使用する場合、MobileTogether Server 内でセットアップしてください。詳細に関しては、MobileTogether Server *ユーザーマニュアル* を参照してください。
- *常に匿名ログインを使用する* ログインの情報無しにユーザーがアプリにアクセスすることを許可する場合このオプションを選択します。それ以外の場合、エンドユーザーはユーザー名とパスワードがログインするために必要です。ログイン資格情報のセットアップに関しては、MobileTogether Server *ユーザーマニュアル* を参照してください。

▾ 4: 特権: ユーザーとアプリの特権

エンドユーザーが永続的データをリセットすることを許可するか設定することができます。アプリの特権は、デバイスのオペレーティングシステムがアプリに与える特権です。ここで選択する特権は、アプリのマニフェストファイル内に保管されています。アプリがインストールされると、デバイスはアプリのマニフェストを確認して、エンドユーザーにアプリがリクエストする特権について通知します。エンドユーザーがこれらの特権を許可すると、アプリはインストールされ、リクエストされた特権がアプリに与えられます。例えば、*SMS の送信* アクションがデザイン何に含まれていると、デフォルトで特権は設定されており、変更することはできません。ロケーションへのアクセスはモバイルデバイスの GPS 場所情報を参照します。上のスクリーンショットでは、デザインは位置情報の機能を使用しており、デバイスの位置情報へのアクセスを必要とします。ロケー

*ションへのアクセス*の資格情報は自動的に選択され、変更することはできません。

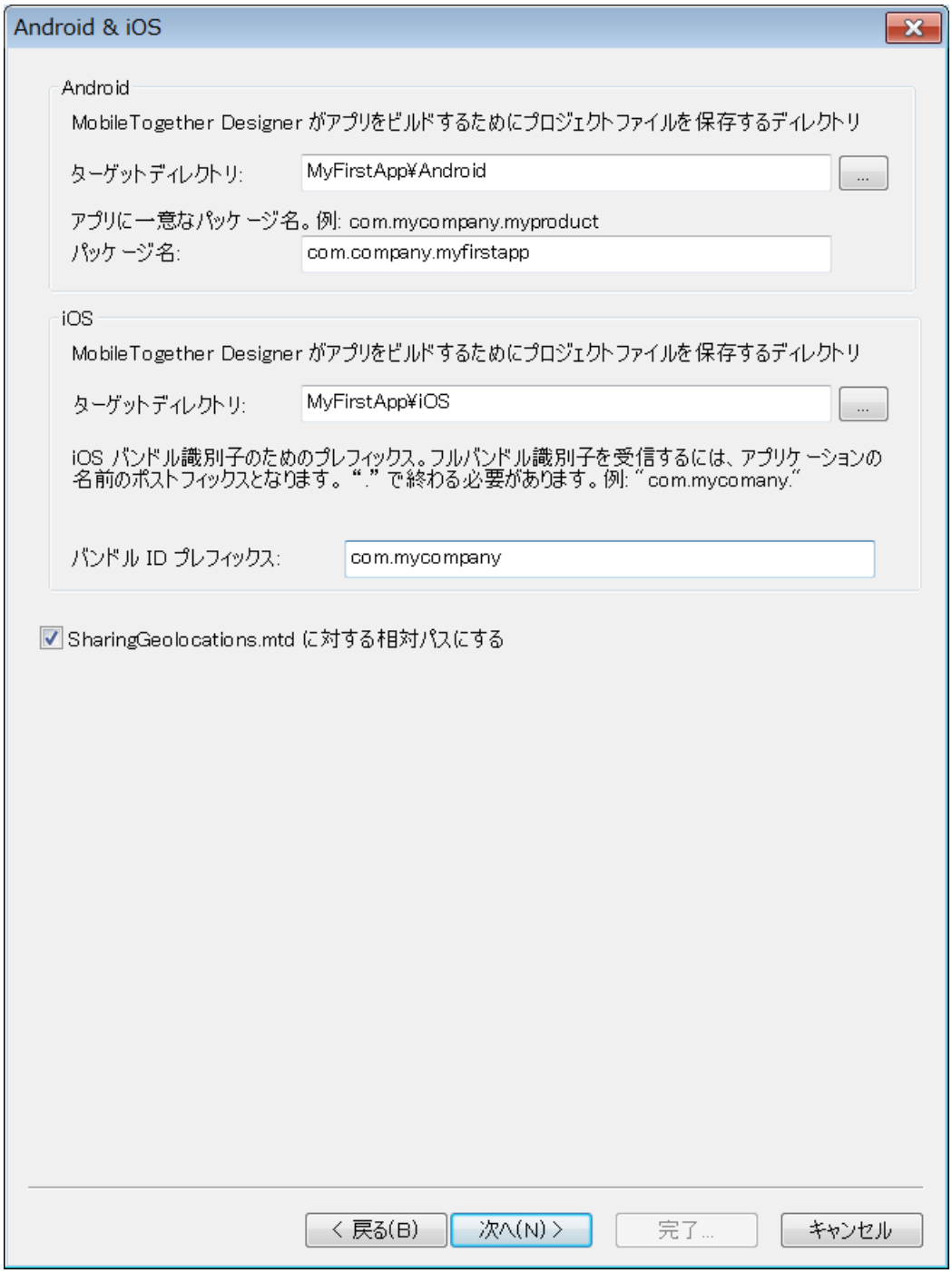
5: コード生成: アプリフォーマットと SPL テンプレートのロケーション



プログラムコードを生成するアプリフォーマットを選択します。異なるアプリフォーマットのための SPL テンプレートがプログラムコードを生成するために必要になります。これらは MobileTogether Designer インストールで提供されており、以下の場所にあります: `C:\Program Files (x86)\Altova\MobileTogetherDesigner4`

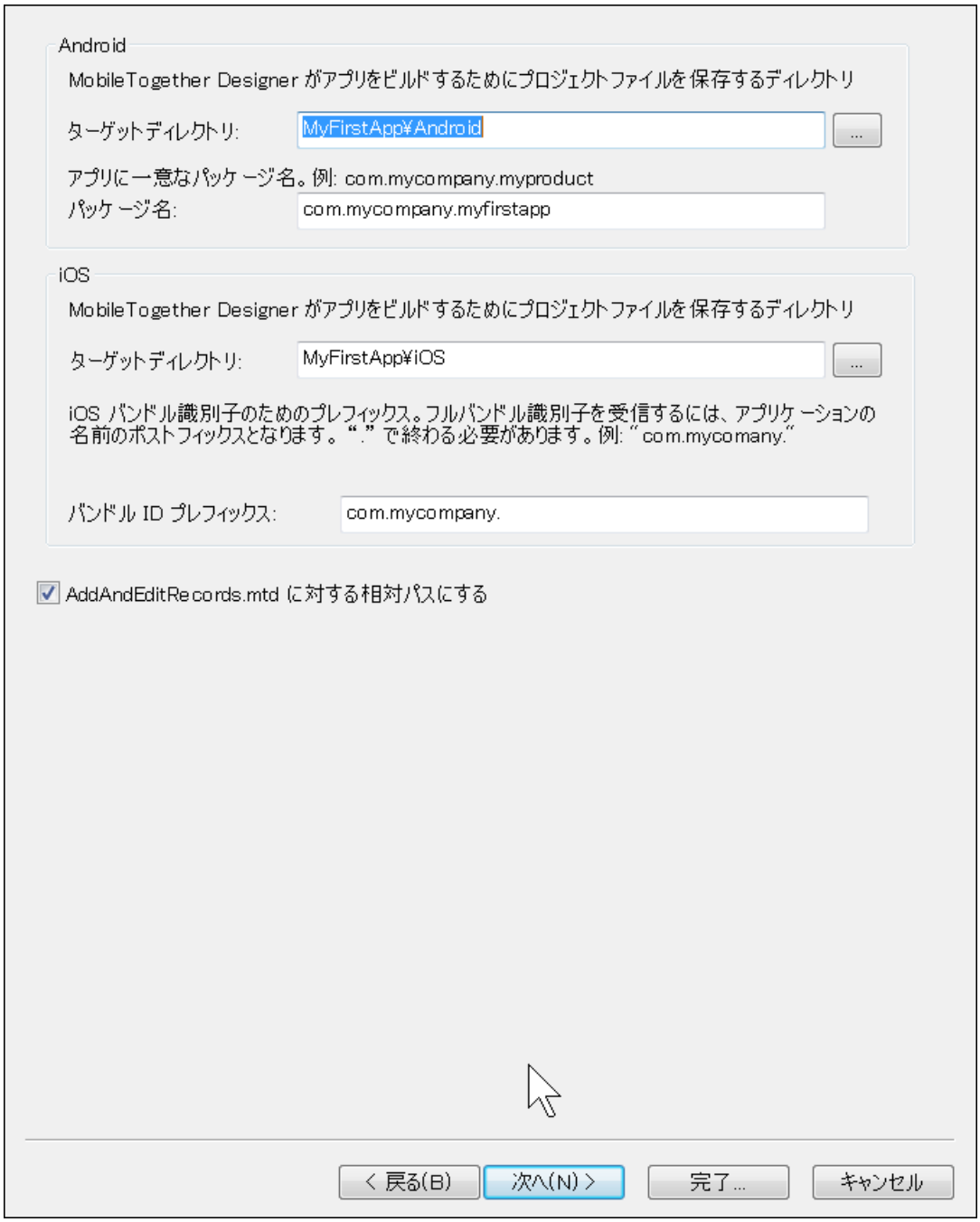
`\MobileTogetherSPL`。1つまたは複数のSPL テンプレートをカスタマイズすると、(カスタム化されたファイルが保存される) SPL テンプレートディレクトリのコピーを作成し、このスクリーン内でこのディレクトリの場所を指定します (次を参照してください *上のスクリーンショット*)。新しいフォルダーはSPLテンプレートディレクトリの構造を再現する必要があります。

▾ 6: プッシュ通知



プッシュの通知の受信時 イベントのためにアクションが定義されている場合、このスクリーンが表示されます。このスクリーン内で、異なるオペレーティングシステムのためにプッシュ通知 (PN) 登録を作成した際に取得された詳細を入力します (次を参照してください: *AppStore App 内のプッシュ通知*)。Android デバイスに送信されたプッシュ通知のために、背景色を選択することもできます。

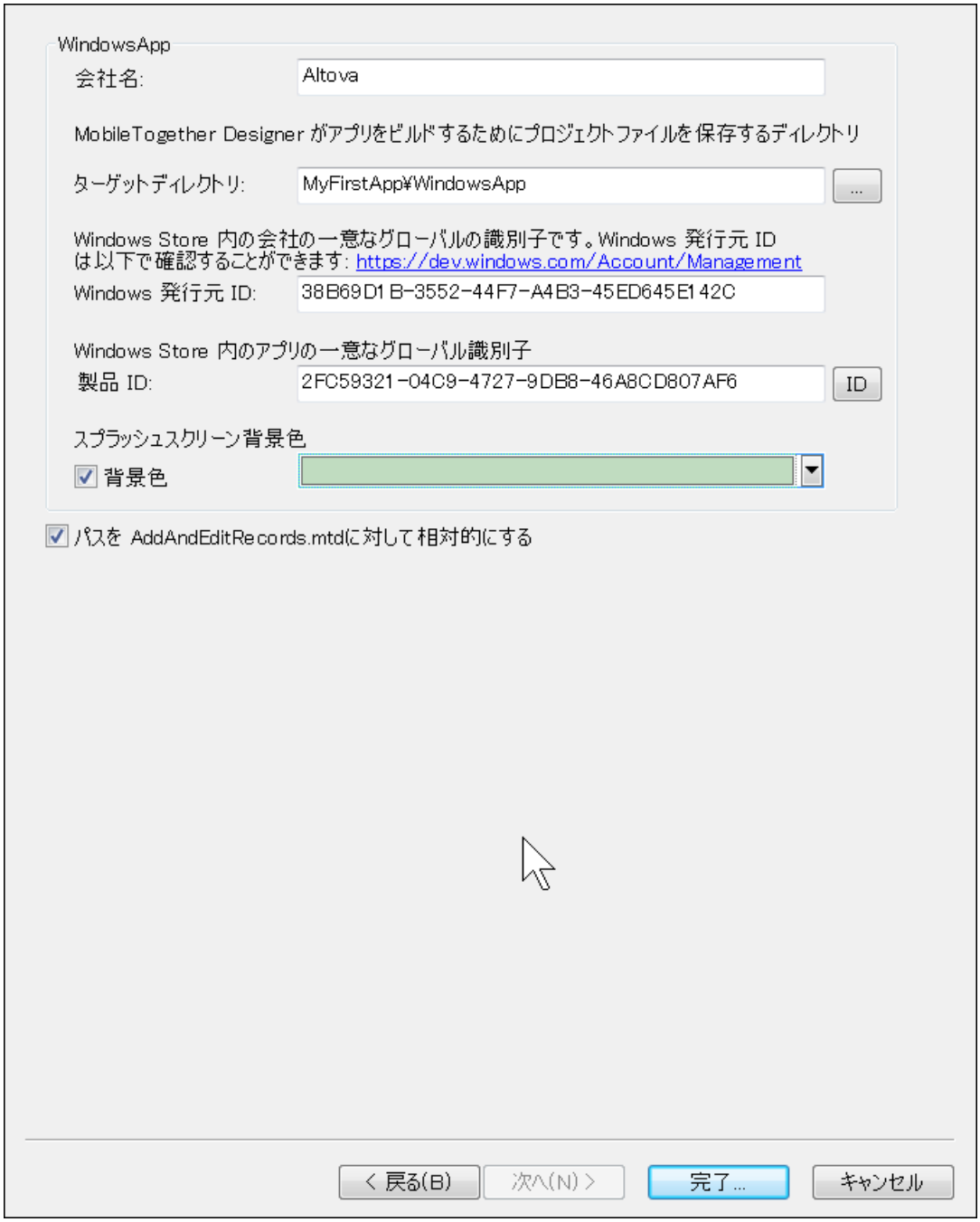
7: Android と iOS



プログラムコードが各フォーマットのために生成されるターゲットディレクトリを設定します。Android パッケージのための

パッケージ名とiOS のためのバンドルID プレフィックスを指定する必要があります。バンドルID プレフィックスが(ドット) 文字 `.` で終わっていることを確認してください。例えば、`com.altova.` は必要条件を満たしています。バンドルID は、スクリーン1 で与えられたアプリ名にバンドルID プレフィックスを加えて構成されています。iCloud アクセスを持つようにアプリを指定している場合、**iCloud コンテナー ID** が自動的に生成され、`<appname>.entitlements` と呼ばれるファイル内に保管されています。プログラムコードのターゲットディレクトリ内にこのファイルは自動的に作成されます。*詳細に関しては、プログラムコードのコンパイル: iOS を参照してください。*

8: WindowsApp

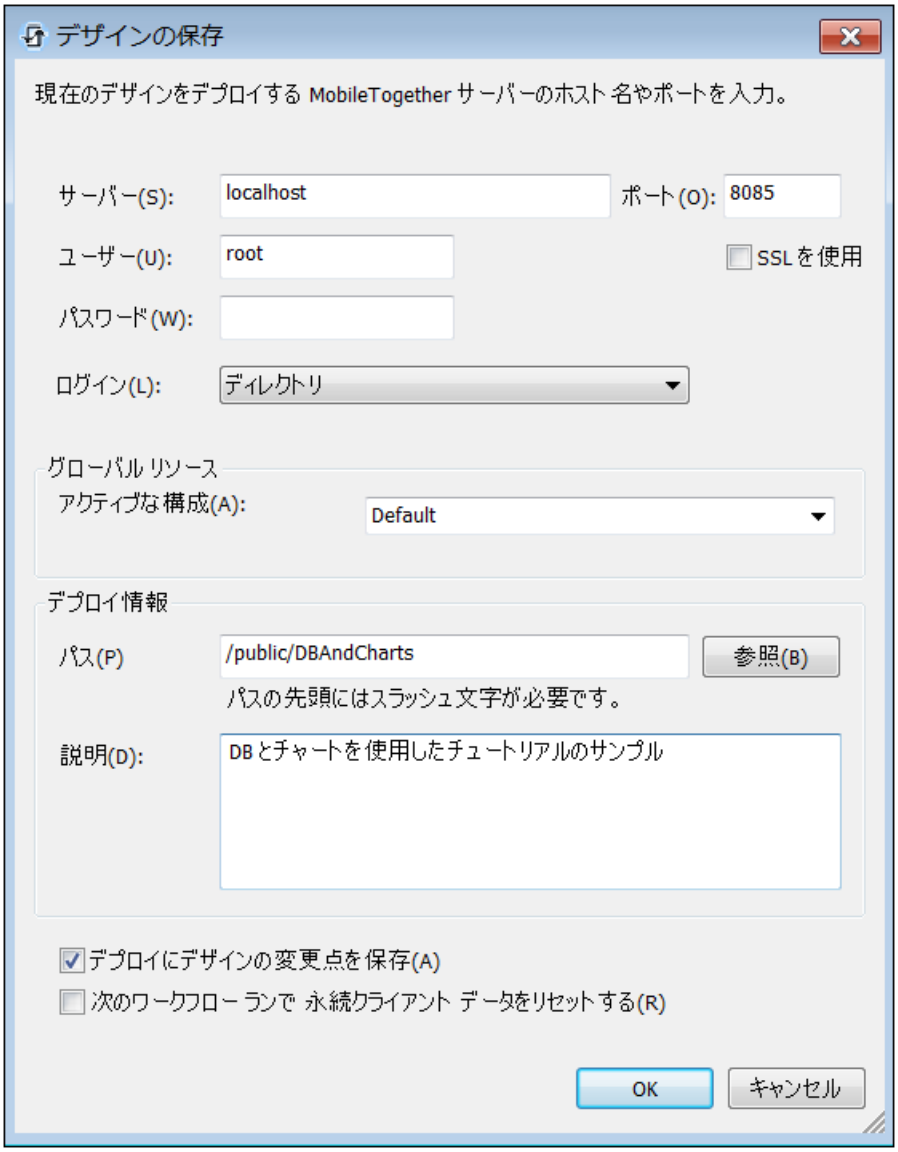


Windows App フォーマットのためにプログラムコードが生成されるターゲットディレクトリを設定します。会社製造元 ID と製品 ID を指定する必要があります。発行元 ID は、デベロッパーアカウントに割り当てられていた発行元 GUID に設定されています。(*Windows App | 必要条件* 参照。発行元 GUID は、Dev Center 内のアカウントサマリーページで見つけることができます)。製品 ID は、アプリを識別するために使用されます。アプリのスプラッシュスクリーンの背景色を選択することもできます。

[プログラムコードの生成] ウィザードに必要な情報を入力し、**[完了]** をクリックすると、[アプリをサーバーにデプロイ] ダイアログが表示されます。

18.2 サーバーにワークフローをデプロイする

[プログラムコードの生成] ウィザード内の**[完了]** をクリックすると、[アプリをMobileTogether Server にデプロイ] ダイアログ (下のスクリーンショット) が表示されます。ダイアログのデータフィールドを**MobileTogether Server にデプロイ**コマンドの説明どおり入力します。



[OK] をクリックすると以下が実行されます:

1. ワークフローMobileTogether Server がにデプロイされます。
2. プログラムコードが指定されたターゲットディレクトリにそれぞれ選択されたフォーマットで生成されます。

生成されたプログラムコードをそれぞれのフォーマット(Android、iOS、Windows App および/または Windows Phone) からAppStore App にコンパイルすることができます。

ワークフローキー

プログラムコードが生成され、ワークフローがサーバーにデプロイされる都度、一意の**ワークフロー キー** がプログラムコードとサーバー上のワークフローに割り当てられます。プログラムコードがコンパイルされると、コンパイルされたフォーマット内にワークフローキーが保管されます。このワークフローキーが特定のワークフローとアプリを関連付ける "握手 "としての役割を果たします。

[プログラムコードの生成] ウィザード を再度破損した場合、プログラムコードは再生成され、ワークフローはサーバーに再デプロイされます。アプリとワークフローは、新しいワークフローキーを与えられます。新しくデプロイされたワークフローが前のワークフローを上書きした場合、前のアプリバージョンはサーバー上の新しいワークフローにアクセスすることはできません。これは、前のアプリのワークフローキーが新しくデプロイされたワークフローのワークフローキーと一致しないからです。新しいバージョンのアプリのみが(同じワークフローキーを持つため)新しくデプロイされたワークフローにアクセスすることができます。

ですから、ソリューションを変更する都度、新しい**バージョン番号** を([プログラムコードの生成ウィザードのスクリーン1内で]) コードのデプロイとワークフローのサーバーへのデプロイ前に割り当てます。この手順であれば、サーバー上の以前のワークフローは、新しいワークフローにより上書きされません。前のバージョンのアプリは以前のワークフローを継続して使用することができ、また新しいバージョンのアプリは新しいワークフローを使用することができます。

メモ デザインを変更していない場合でも、[プログラムコードの生成] ウィザードを実行する都度、新規の一意のワークフローキーが生成されます。

メモ アプリとワークフローの互換性を確認するために、[プログラムコードの生成] ウィザードを定期的にMobileTogether Server をバックアップすることが奨励されます。手順についての詳細は、MobileTogether Server マニュアルを参照してください。

18.3 プログラムコードのコンパイル

[プログラムコードの生成] ウィザードを完了して、[ワークフローをサーバーにデプロイ] ダイアログ内の**[OK]** をクリックすると、アプリのワークフローがサーバーにデプロイされ、選択されたアプリフォーマットのためのプログラムコードが生成されます。プログラムコードは、各アプリフォーマットのために個別に生成され、ウィザード内で指定されたターゲットディレクトリに生成されます。このセクションは、アプリストアにアップロードすることのできるそれぞれのプログラムコードをAppStore App に生成する方法を説明します。

MobileTogether Designer により生成されるプログラムコードは、通常作成する前に変更は必要ありません。しかし、プロジェクトを変更する必要があり、生成されたプロジェクトファイルの代わりにSPL テンプレートファイルを変更することは重要です。[プログラムコードの生成] ウィザードを使用してプロジェクトを後に生成するときに変更が失われないようにするためです。SPL とSPL テンプレートは、SPL テンプレートのセクションで説明されています。

プログラムコードは下にリストされるアプリフォーマットのために生成されます。下のリンクにより対応するサブセクションに移動することができます。各サブセクションはそれぞれのアプリフォーマットのためのプログラムコードのコンパイルの方法について説明しています。

- Android
- iOS
- Windows App

Android

ウィザードを完了すると、MobileTogether Designer は、Android Studio プロジェクトを作成します。Android Studio を使用して、このプロジェクト(**.apk** ファイル)をAndroid アプリに作成することができます。

Android Studio プロジェクト

Android Studio プロジェクトは、([プログラムコードの生成ウィザードのスクリーン6])でプログラムコードの保存場所フォルダーとして選択されたディレクトリに作成されます。

プロジェクト ファイルは、以下の場所に保存されます。これらのファイルは、Android Studio 内でコンパイルされる前に変更される必要はありません。しかし、必要な場合はコンテンツを変更することができます。詳細に関しては、コンテンツのカスタム化を参照してください。

app\src\main\AndroidManifest.xml
これは、Android マニフェストです。[プログラムコードの生成] ウィザード内で入力されたパッケージ名およびアプリバージョンなどアプリに関連した情報を含みます。

app\src\main\java\<package-name>\MainActivity.java
app\src\main\java\<package-name>\<binary-name>.java
<package-name> と*<binary-name>* の値は[プログラムコードの生成] ウィザードのスクリーン6 と1 でそれぞれ入力された値から取得されます。

ディレクトリ**app\src\main\res** は、アプリアイコンのためのリソースファイル、スプラッシュスクリーン、とその他のリソース文字列を含みます。多種のサイズのアプリアイコンは、適切なサブディレクトリで使用することができます。異なるサイズは自動的にMobileTogether Designer により、[プログラムコードの生成ウィザードのスクリーン2] 内で指定されたアイコンファイルから作成されます。

Android アプリの作成

アプリは、Android Studio を使用して作成されます。ここで説明されるステップは、ドキュメントの作成時 (2015 年 10 月) に最新であるAndroid Studio 1.3 RC 4 に関連したものです。

Android Studio 内で(ターゲット ディレクトリ) プロジェクトを開きます。ビルドは自動的に開始され、Gradle Console 内で結果が表示されます。ビルドのプロセス中、以下のメッセージを受信する可能性があります: `AAPT err [...] libpng warning: iCCP: Not recognizing known sRGB profile that has been edited。`これは、無視しても安全なメッセージです。APK ファイルはすぐには作成されず、APK ファイルを作成するためにアプリを実行する必要があります。

アプリの実行とデバッグ

ビルドが完了すると、以下のAndroid Studio コマンドの1つを作成してアプリを実行することができます: **[実行] <app>** または**[デバッグ] <app>。**APK ファイルが作成されます: **app\build\outputs\apk\app-debug.apk。**

Android アプリのリリース

アプリをリリースするには、署名されたAPK を作成する必要があります。以下を行ってください:

1. Android Studio 内で、コマンド**[ビルド| 署名されたAPK の作成]** を発行します。署名されたAPK の生

成ウィザードを開始します。
2. KeyStore パスを入力します。新規のKeyStore を作成する必要がある可能性があります。
3. ウィザードのスクリーンを確認して、必要とされる情報を入力します。
4. **[完了]** をクリックして、署名されたAPK の作成を開始します。

署名されたAPK ファイルは、以下に保管されます: **`app\app-release.apk`**.

メモ **[署名されたAPK の作成]** コマンドは、Android Studio に間違った報告 `lint` エラーをオープンコードファイル内に引き起こす可能性があります。このエラーが発生した場合、[**ファイル | 無効なキャッシュ /再起動**] を発行することにより、エラーは削除されるかもしれません。以下のメッセージを受信した場合: `AAPT err […] libpng warning: iCCP: Not recognizing known sRGB profile that has been edited`, 無視をすることができます。

iOS

iOS アプリのために生成されたプログラムコードは、XCode プロジェクトです。 [プログラムコードの生成ウィザードのスクリーン6]内で指定されたディレクトリに保存されます。XCode を使用して、`.xcodeproj` ファイルを開きiOS アプリを作成します。

必要条件

生成されたプログラムコードから、iOS アプリをビルド、テストおよび発行するには、以下が必要です:

- XCode の最新バージョンは、デベロッパープラットフォームから利用することができます。
- アプリをテストするためには、iOS 9.0 または以降
- Apple デベロッパープログラム内のメンバーシップ。アプリをテストしてアプリストアで発行するために必要です。

アプリのビルドと配布

アプリの作成の大まかな概要は、以下のとおりです。詳細に関しては、Apple アプリの配布のガイドラインを参照してください。

1. XCode 内.`xcodeproj` ファイルのを開きます。
2. チームアプリ ID、プロビジョニングプロファイルがセットアップされていることを確認してください。 手順については配布のためXCode プロジェクトを構成するを参照してください。アドホック配布のためのテストデバイスは対応するプロビジョニングファイルを再生成した後にアクティブになります。
3. 使用されるスプラッシュスクリーンを決定します。デフォルトでは、 [プログラムコードの生成ウィザードのスクリーン2]) 内で指定された縦長のスプラッシュスクリーンが元のアスペクト率と同じにスクリーンに表示されます。個々のスプラッシュスクリーンを異なるデバイスと向きごとにセットアップする場合は、以下を行います: (i) **[サポートするファイル | Info.plist]** に移動して、ローンチスクリーンインターフェイスファイルベース名を削除します、(ii) `images.xcassets` をクリックして、ビュー内を右クリックしてイメージの新規イメージのローンチ、を選択します、(iii) 各解像度と向きのためのイメージを指定します。
4. iOS デバイスがつながれている場合、 **[製品 | デスティネーション]** に移動して、デバイスをチェックします。その他の場合は、**[製品 | デスティネーション]** に移動して、*iOS デバイス*を選択します。コンパイルされたアプリをシミュレーターで実行することはできません。ですから、アプリをテストする場合は、iOS デバイスに接続している必要があります。
5. iOS デバイス上でコンパイルされたアプリをテストするには、(**[製品 | デスティネーション]** でiOS デバイスを選択した後)**[製品 | 実行]** をクリックします。
6. アドホックのApp Store への提出は、**[製品 | アーカイブ]** をクリックして、アプリをエクスポートします。Apple アプリの配布のガイドラインの手順に従ってください。

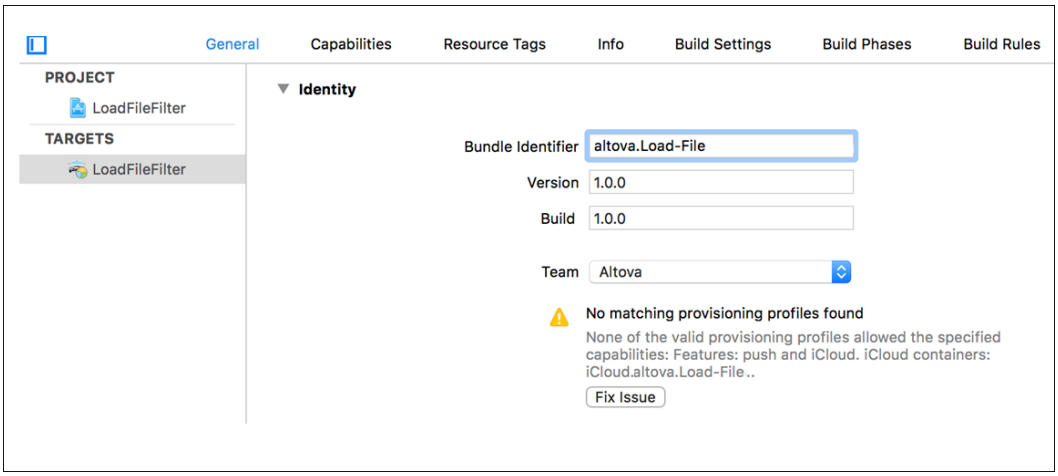
メモ 複数のページのあるiOS アプリでは、アプリの最後のページの**[送信]** ボタンがタップされると、アプリは再起動します。これはiOS アプリは自動的にシャットダウンできないためです。

---

iCloud を使用するための追加ステップ

アプリがiCloud を使用する場合、アプリがファイルがクライアントに保存されることを定義する場合、通常、追加ステップを行うことが必要になります:

1. XCode 内で、一般タブをクリックします (下のスクリーンショット参照)。プロビジョニングファイルが不足していることを通知するメッセージが表示されます。



2. 開発者アカウントの権利がある場合は、**[問題をフィックス]** をクリックします。
3. *機能 タブをクリックします (上のスクリーンショット参照).*
4. *サービスセクションで、iCloud ドキュメントを選択します。*
5. 必要に応じてデフォルトの設定を変更し、**[問題をフィックス]** をクリックします。

上記のステップは、アプリid とiCloud とコンテナーをセットアップするための、開発者アカウントの権利がある場合の、素早い手順を示したものです。

iCloud Container ID は、iCloud がアプリにアクセスするために必要です。プログラムコードの生成中、**iCloud Container ID** は、プログラムコードウィザードのスクリーン6 内で指定されたバンドルID にから自動的に生成されます。生成されたID のフォームは以下のとおりです: `iCloud.<BundleIDPrefix>.<PackageName>`。プログラムコードのターゲットディレクトリ内で自動的に作成される`<appname>.entitlements` と呼ばれるファイルに保管されます。自動的に生成されたID を変更するには、`<appname>.entitlements` 内の名前を変更し、ファイルを保存してください。権利ファイル内のID は、開発者アカウント内のiCloud Container ID に一致する必要があります。

iCloud 使用に関する詳細は、iOS 開発者ドキュメント内の次のトピックを参照してください: 機能の追加 とアプリ内で CloudKit を有効化する。

Windows アプリ

Windows App フォーマットは、(Windows RT で作動するタブレット) とWindows 8.1 または以降で作動する PC などのWindows タッチデバイスのためです。Windows アプリのために生成されたプログラムコードは、C++ です。[プログラムコードの生成ウィザードのスクリーン7] で指定されたディレクトリに生成されます。生成されたC++ プロジェクト (`some-app-name.vcxproj`) をVisual Studio 内でプロジェクトにビルドし、Windows アプリにビルドすることができます。ビルドプロセスの出力は **.XAP** 拡張子を持つファイルを含みます。これは、アプリに必要なすべてのファイルを含む zip ファイルです。このファイルがWindows App Store で発行されるファイルです。

必要条件

生成されたプログラムコードからをWindows アプリをビルドしてテストするには、以下が必要です:

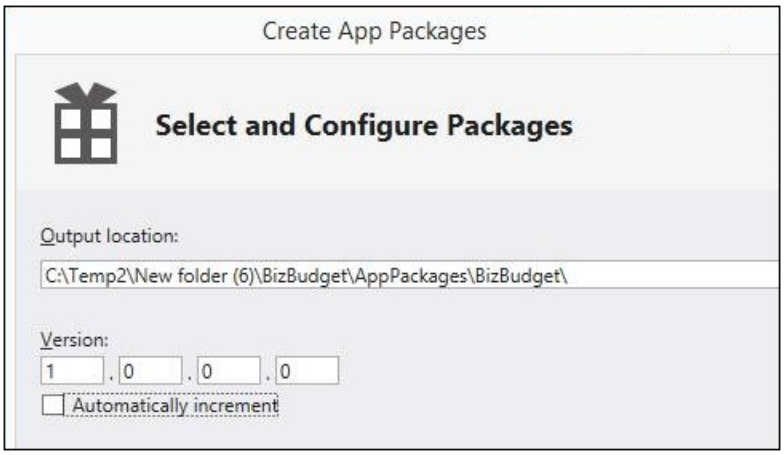
- Visual Studio 2015 Update 3 (*このバージョンのみ*) と *Windows 8.1 (または以降)のためのストアアプリの管理のためのツール。*
- *アプリをテストするためには、*Windows 8.1 または以降
- Windows アプリ開発者アカウント。アプリをWindows Store で発行するために必要です。 Microsoft アカウントを開くために必要な情報を参照してください。

メモ Windows Phone 8.1 はサポートされていません。

アプリの作成

アプリの作成のための広義の概要は、以下のとおりです:

1. Visual Studio 内で生成されたプログラムコードを開きます。[プログラムコードの生成ウィザードのスクリーン7] 。内で指定されたターゲットディレクトリ内の **.vcxproj** ファイルを開きます。
2. Visual Studio 内で、ビルド構成をWindows 32、Windows 64 またはARM に設定します。(*下のスクリーンショット参照*) [プログラムコードの生成ウィザードのスクリーン1] で入力した同じバージョンを入力します。例えば、**10** をウィザードで入力した場合、内に入力する番号は**10.0.0.0**. となります。また、Visual Studio 内では、バージョン設定の自動的振番のオプションを解除してください(*下のスクリーンショット参照*)。このオプションがチェックされていない場合、バージョン番号はアプリを作成する都度更新され、ウィザードに入力する番号と一致しなくなります。



3. **[実行]** ボタンの右のコンボボックスから、**[デバッグ]** を選択して、**[実行]** をクリック または**[F5]** を押す、または **[デバッグ | デバッグの開始]** を選択します
4. Visual Studio がMobileTogether プロジェクト または ワークフローファイルの拡張子の一部の存在を確認でき

ないため例外が報告される場合があります。これらの例外は無視することができます。

5. プログラムコードを変更する場合は、プログラムコードを生成するSPL テンプレートを編集するか、[プログラムコードの生成] ウィザード内のエントリの一部を修正するかを決めます。前者の場合、SPL テンプレートを編集し、プログラムコードを再生成するため[プログラムコードの生成] ウィザードを実行します。
1. リリースするアプリをビルドするためには、**[実行]** ボタンの横のコンボボックスに移動し、**[リリース]** を選択します。**[プロジェクト | ストア | アプリパッケージの作成]** を選択します。これによりウィザードが開かれます。
6. ウィザードのスクリーン内から、リクエストされた情報を提供するオプションを選択します。
7. ウィザードを完了すると、**[作成]** を押します。Visual Studio がアプリの作成を開始します。

選択された各プラットフォーム(Win-32、Win-64、およびARM)、にはAPPXUPLOAD ファイルが作成されます。これは、Windows Store にアップロードするためのファイルです。更に、各プラットフォームのために、それぞれのプラットフォーム上にコンパイルされたアプリをインストールするためのインストーラーを含む(名前内にWindowsStore を持つ)フォルダーが作成されます。これにより、Windows Store で発行する前に、アプリを複数のマシンでテストすることができます。このパッケージからのインストールの方法は次のセクションで説明されています。

アプリを直接インストールする

アプリを直接 Windows 8.1 または以降で作動するPC または タブレットにインストールすることができます。このようなインストール方法は(Windows Store からアプリをダウンロードすることに比べ) とても便利です。例えば、複数のワークステーションでアプリをテストして直接配布する場合など。

1. PC またはタブレット上で、スタートページに移動し、Windows PowerShell を検索します。右クリックをして**[管理者として実行]** を選択します。
2. PowerShell 内に以下を入力します: **`set-executionpolicy unrestricted`**
3. **`[Y]`** で確認し**[Enter]** を押します。(この手順は一度のみ行うだけです)。
4. **`Show-WindowsDeveloperLicenseRegistration`** を入力して、**[Enter]** を押します。
5. Windows Developer License を取得するためにダイアログを完了します。(この手順は一度のみ行うだけです)。
6. PC またはタブレットにコンパイルされたアプリパッケージのファイルをコピーします。
7. PowerShell 内で(**`cd`** コマンドを使用して) ファイルをコピーした場所にナビゲートします。
8. フォルダー内の **`.ps1`** スクリプトを開始するために**`Add-AppDevPackage.ps1`** を入力して、**[Enter]** を押します。(**`ps1`** スクリプトを開始する別の方法は、コンテキストメニューコマンド**[PowerShell 内での実行]** (または類似のコマンド) を使用します。
9. アプリは、スタートページにインストールされ、テストすることができます。

18.4 SPL テンプレート

MobileTogether Designer は、Spy Programming Language (SPL) テンプレートを使用してAndroid、iOS、Windows およびWindows Phone のためのAppStore App プログラムコードを作成するために使用されるファイルを生成します。

- SPL テンプレートは、MobileTogether Designer のインストールの際に与えられ、フォルダー**C:\Program Files (x86)\Altova\MobileTogetherDesigner4\MobileTogetherSPL** 内にあります。このディレクトリを[プログラムコードの生成ウィザードのスクリーン5]内で使用するSPL テンプレートディレクトリとして指定します。
- SPL テンプレートは、[プログラムコードの生成] ウィザード内に入力する情報に関連付けられた変数を使用します。

プログラムコードを変更する必要はなく、生成されるコードを直接変更しては**なりません**。その代わりに、コードを生成するSPL テンプレートを編集します。この方法であれば、テンプレートからプログラムコードを再生成する際に変更が失われることはありません。このセクションはSPL テンプレートの作動および編集方法について説明します。

SPL テンプレートのロケーション

SPL テンプレートはMobileTogether Designer アプリケーションフォルダー内にあります:

C:\Program Files (x86)\Altova\MobileTogetherDesigner4\MobileTogetherSPL

このフォルダーは、各アプリのフォーマット(Android.spl、iOS.spl 、WindowsApp.spl 、と WindowsPhone.spl) のためのエントリポイントSPL テンプレート、共通のライブラリSPL テンプレート(変更されてはならない\CommonLibrary.spl) とそのフォーマットのための追加 SPL を含む各アプリのフォーマットを含みます。それぞれのアプリフォーマット内はZIP ファイルとそのアプリのためのSPL テンプレートファイルです。

各 SPL ファイルは、プロジェクトファイルを生成するテンプレートであり、他のSPL ファイルを呼び出します。Android Studio プロジェクトを作成する場合、例えば、MobileTogether Designer は、(SPL テンプレートディレクトリ内の) Android.spl を処理します。このSPL ファイルは、代わりに、他のプロジェクトファイルを生成し、他のSPL ファイルを呼び出す他のSPL ファイルを呼び出します。これらのSPL テンプレートファイルは、生成されたプログラムコードを変更するために変更することのできるファイルです。これらは、このセクションのサブセクションで説明されているSPL 構文 とSPL 命令を使用して書かれています。

SPL テンプレートを使用してプログラムコードを変更する

プログラムコードのカスタム化の方法は、以下のとおりです:

1. SPL テンプレートディレクトリ全体 (または、変更するテンプレートのみ) を選択したディレクトリにコピーします。この方法で、MobileTogether Designer の再インストールは、カスタム化されたSPL テンプレートに上書きされません。ディレクトリのコピーは元のディレクトリと同じ構造を持つ必要があります。
2. SPL ファイルを必要に応じて変更します。
3. プログラムコードの生成ウィザードを変更されたSPL テンプレートを使用して実行します。[プログラムコードの生成ウィザードのスクリーン5] 内で、カスタム化されたSPL テンプレートを含むディレクトリの場所を指定します。

カスタム化されたテンプレートに従い、プログラムコードが生成されます。

このセクション

このセクションは、以下のとおり構成されています:

- SPL 構文
- SPL 文字列のメカニズム
- $Options のプロパティ
- $Application のプロパティ
- その他のオブジェクト

SPL 構文

SPL テンプレートは生成するプログラムコードのプログラミング言語で構成されています。テンプレートは生成されたプログラムコードにMobileTogether データを統合するためのSPL 命令のスニペットを含みます。SPL 命令は角かっこで囲まれています。例えば、ファイルを生成するためのSPL 命令が黄色でハイライトされた、(XML で書かれた) XML テンプレートがあるとします。

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
        package="[=$Options.androidPackageName]"
        android:versionCode="[=$Options.appVersion]"
        android:versionName="[=$Options.appVersion]">
        <uses-sdk
                android:minSdkVersion="15"
                android:targetSdkVersion="22"/>
</manifest>
```

複数のステートメント

複数のステートメントは、かっこのペア内に含まれることができます。追加ステートメントは前のステートメントから新しいラインまたはコロンで区切られる必要があります。有効なサンプルは、以下のとおりです:

`[$x = 42` `$x = $x + 1]`	`[$x = 42: $x = $x + 1]`

テキスト

角かっこで囲まれていないテキストは出力に直接書き込まれます。角かっこを生成するには、バックスラッシュでエスケープします: `\[` と`\]`。バックスラッシュを生成するには`\\` を使用します。

コメント

命令ブロック内のコメントは常に' 文字から始まり、次のラインまたは終わりの角かっこで終了します。

変数

変数は値を与えることで作成できます。`$` 文字は変数を宣言または使用する場合に使用されます。変数名は大文字と小文字を区別します。変数には以下の型があります:

- 整数: `0` が`false` でその他はすべて`true` の場合、ブール値としても使用されます。
- 文字列: *文字列のメカニズム*を参照
- オブジェクト: MobileTogether により提供されます。例えば、`$Options` オブジェクト

変数の型は最初の割り当てで宣言されます:

`[$x = 0]` は`x` が整数であることを意味します。
`[$x = "teststring"]` は`x` が文字列であることを意味します。

文字列

文字列は、2 重引用符はで囲まれています。 2 重引用符内の**\n** と**\t** は、新しいラインとタブ、**\"**は、リテラルな 2 重引用符、\\ はバックスラッシュとみなされます。文字列の制約は複数のラインでも示すことができます。文字列の連結は **&** 文字を使用します:

```
[$BasePath = $outputpath & "/" & $JavaPackageDir]
```

オブジェクト

オブジェクトは、MobileTogether 構成です。オブジェクトには、**.** operator を使用してアクセスすることのできるプロパティがあります。SPL 内で新しいオブジェクトを作成することはできませんが変数にオブジェクトを割り当てることはできます。

```
class [=$class.Name]
```

このサンプルの出力単語 `class` の後には、スペースと`$class` オブジェクトの`Name` プロパティの値が続きます。

条件子

`ELSE` を伴うまたは伴わないで**IF** ステートメントを以下のように使用します。条件の周りに中かっこを使用しないでください。

```
if condition
        statements
else
        statements
endif
```

サンプル

```
[if $namespace.ContainsPublicClasses and $namespace.Prefix <> ""]
        whatever you want ['inserts whatever you want, in the resulting file]
[endif]
```

反復子

FOREACH ステートメントを使用して以下のように反復します:

```
foreach object in collection
           statements
next
```

サンプル

```
[foreach $class in $classes
       if not $class.IsInternal
              ]      class [=$class.Name];
[      endif
next]
```

文字列のメカニズム

SPL は、以下にリストされる文字列の操作メカニズムを提供します。

---

integer Compare(s)

s (大文字小文字を区別する) に対する文字列の辞書的リレーションを示す値を返します:

<0 = 文字列は s より小さい

0 = 文字列は s と等しい

>0 = 文字列は s より大きい

integer CompareNoCase(s)

s (大文字小文字を区別しない) に対する文字列の辞書的リレーションを示す値を返します:

<0 = 文字列は s より小さい

0 = 文字列は s と等しい

>0 = 文字列は s より大きい

integer Find(s)

サブストリング s の最初のマッチのための文字列を検索します。s の最初の文字のゼロベースのインデックスを、または見つからない場合 -1 を返します。

string Left(n)

文字列の最初の n 文字を返します。

integer Length()

文字列の長さを返します。

string MakeUpper()

大文字に変換された文字列を返します。

string MakeUpper(n)

大文字に変換された最初の n 文字をオプションで含む文字列を返します。

string MakeLower()

小文字に変換された文字列を返します。

string MakeLower(n)

小文字に変換された最初の n 文字をオプションで含む文字列を返します。を返します。

string Mid(n)

ゼロをベースにしたインデックスの位置 n から始まる文字列を返します。

string Mid(n,m)
ゼロをベースにしたインデックスの位置 n から始まり、長さm を持つ文字列を返します。を返します。

string RemoveLeft(s)
Left(s.Length())がサブストリング s に等しい場合、サブストリング s を除く文字列を返します。

string RemoveLeftNoCase(s)
Left(s.Length())がサブストリング s (大文字小文字を区別しない)に等しい場合、サブストリング s を除く文字列を返します。

string RemoveRight(s)
Right(s.Length())がサブストリング s に等しい場合、サブストリング s を除く文字列を返します。

string RemoveRightNoCase(s)
Right(s.Length()) がサブストリング s (大文字小文字を区別しない)に等しい場合、サブストリング s を除く文字列を返します。

string Repeat(s,n)
n 回リピートされたサブストリング s を含む文字列を返します。

string Replace(sOld,sNew)
文字列 sOld を文字列 sNew と置き換えます。

string Right(n)
文字列の最後の n 文字を返します。

文字列プロパティ

以下のプロパティを使用することができます:

- Length: 文字列の長さを返します。 *サンプル:* $Options.deploymentPath.Length は deploymentPath 内に含まれる文字列の長さを返します。
- XMLEncode: XML エンコードされたフォーマット内の文字列の長さを返します。サンプル: $Options.deploymentPath.XMLEncode は、XML-エスケープテキストとして deploymentPath 内に含まれる文字列を返します。

$Options のプロパティ

$Options オブジェクトは、以下にリストされたプロパティを取ることができます。プロパティの大半の値は、[プログラムコードの生成] ウィザード内で提供され、このセクションで説明されています。オブジェクトのプロパティは、. operatorを使用してアクセスすることができます。SPL テンプレートの使用方法のサンプルの一部が下に表示されています。

```
<data
  androidscheme="[=$Options.schemeForRunSolutionUrl]"
  androidhost="[=$Options.hostForRunSolutionUrl]"/>


@Override
public boolean GetServerUsesSsl()
   {
       return [if $Options.isUseSSL = 1]true[else]false[endif];
   }
```

ワークフローに関連したプロパティ

以下のワークフローに関連したプロパティを使用することができます:

- **workflowKey**: ワークフローキーを返します。*サンプル:* `$Options.workflowKey` は、ワークフローキーを返します。プログラムコードの生成 が行われ、関連したワークフローがサーバーにデプロイされると、同じ一意のワークフローキーが割り当てられます。 AppStore App は、ワークフローと同じキーを与えられている場合のみこのワークフローにアクセスすることができます。詳細に関しては、ワークフローをサーバーにデプロイを参照してください。
- **deploymentPath**: デプロイされたワークフローのパスを返します。 *サンプル:* `$Options.deploymentPath` は、MobileTogether Server 上のワークフローパスを返します。ワークフローパスのサンプル: `/Public/DateTime/`。

一般的なプロパティ

以下が [プログラムコード生成ウィザードのスクリーン1] 内で与えられているプロパティです。

appName
visibleAppName
appVersion
hostForRunSolutionUrl
schemeForRunSolutionUrl

ユーザーインターフェイスプロパティ

以下が [プログラムコード生成ウィザードのスクリーン2] 内で与えられているプロパティです。

splashScreenPortraitFilePath
splashScreenLandscapeFilePath
launcherIconFilePath
aboutText
aboutCopyRight

サーバープロパティ

以下が[プログラムコード生成ウィザードのスクリーン3] 内で与えられているプロパティです。

```
serverAddress
serverPort
isServerAccessAlwaysAnonymous
isUseSSL
```

ユーザーとアプリ特権についてのプロパティ

以下が[プログラムコード生成ウィザードのスクリーン4] 内で与えられているプロパティです。

```
mayResetPersistentData
isAllowSMS
isAllowTelephoneCall
isAllowCamera
isAllowLocationAccess
isAllowExternalStorageAccess
isAllowiCloudAccess
isAllowAudioRecording
isAllowPhotoGalleryAccess
```

Android とiOS プロパティ

以下が[プログラムコード生成ウィザードのスクリーン6] 内で与えられているプロパティです。

```
targetDirectoryAndroid
androidPackageName
androidPackageDir
targetDirectoryIOS
iosBundleIdentifierPrefix
```

Windows App とWindows Phone プロパティ

以下が[プログラムコード生成ウィザードのスクリーン7] 内で与えられているプロパティです。

```
windowsAppCompanyName
windowsPhoneCompanyName
windowsCompanyPublisherID
targetDirectoryWindowsApp
windowsAppCompanyProductID
targetDirectoryWindowsPhone
windowsPhoneCompanyProductID
```

$Application のプロパティ

$Application オブジェクトは、以下にリストされるプロパティを取ることができます。オブジェクトのプロパティは. operatorを使用してアクセスすることができます。SPL テンプレートの使用方法のサンプルの一部が下に表示されています。

```
sub CopyFile( byval $sSource, byval $sTarget )
    return $Application.CopyFile( $sSource, $sTarget )
endsub

sub UnzipFile( byval $sZipFile, byval $sTargetDir )
    return $Application.UnzipFile( $sZipFile, $sTargetDir )
endsub
```

プロパティ

CopyFile(sSource, sTarget)
ロケーション sSource にあるファイルをロケーション sTarget にコピーします。

UnzipFile(sZipFile, sTargetDir)
ロケーション sZipFile にある ZIP アーカイブをディレクトリ sTargetDir で展開します。

CopyAndResizeImage(sSource, sTarget, nTargetWidth, nTargetHeight)
ロケーション sSource にあるイメージファイルを ロケーション sTarget にコピーし、コピーされたイメージを nTargetWidth および nTargetHeight により与えられたピクセルのディメンションにサイズ調整します。

その他のオブジェクト

次のオブジェクトは変数としてアクセスすることができます:

`$Host`	MobileTogetherDesigner <*バージョン、例、2.0*>
`$HostShort`	MobileTogetherDesigner
`$HostURL`	`http://www.altova.com/mobiletogether`
`$HostVersion`	メジャー製品バージョン(例えば、1.5 の場合 1)
`$HostRelease`	マイナー製品バージョン(例えば、1.5 の場合 1)
`$RegisteredName`	ライセンスを持つユーザーの名前
`$RegisteredCompany`	ライセンスを持つ企業名
`$CreationDate`	SPL コード生成の日時
`$outputpath`	サーバー上のターゲットワークフローディレクトリ

チャプター 19

サーバーサービス

19 サーバーサービス

サービスとは **MobileTogether Server Advanced Edition** にソリューション(`.mtd` ファイル)としてデプロイされているMobileTogether アクションのセットです。 MobileTogether Server の特定の条件の組合せが満たされるとサービスはサーバー上で実行されます。 MobileTogether Server Advanced Edition の管理インターフェイス内でこれらの条件を定義することができます。

MobileTogether サービスの作成と実行の大まかなステップは、以下のとおりです:

1. MobileTogether Designer 内でのサービスの作成。各サービスは単一の `.mtd` ファイル内に保管されます。
2. MobileTogether Designer からMobileTogether Server Advanced Edition へのサービスのデプロイ。
3. MobileTogether Server 管理者 インターフェース内でサービスをトリガーするために必要な条件を定義します。

各ステップについて、このセクションのサブセクションで説明されています。MobileTogether Server 上のサービスのセットアップに関する詳細は、MobileTogether Server ドキュメントを参照してください。

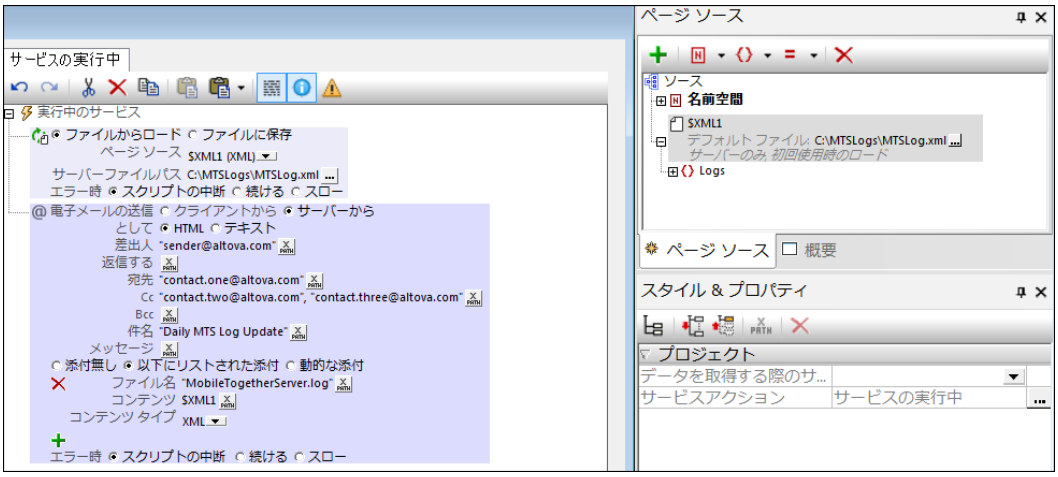
(C) 2018 Altova GmbH

19.1 サービスの作成

サービスはソリューションと同じ方法で作成されます。ツリーからデータを使用するために、ページソースツリーを定義することができます。しかしながら、サービスは(ユーザーインターフェースが使用されず)サーバー上で実行されることが前提であり、すべてのコントロールと新規の(トップ、または、サブ)ページの追加は無効化されています。サーバー上実行することのできるサービスは、サーバーからの電子メールの送信や、ページソース内のノードの更新などのサーバー側のアクションです。このようなアクションのセットはサービスのアクションとして定義することができ、他のアクションは無効化されています。

新規のサービスを作成するには、以下を行います:

1. 「**ファイル**」「**新規のサービス**」をクリックして、サービスのためにデザインファイルを開きます。新規の**サービスデザイン**が作成され、`$MT_SERVICE` ページソースが自動的に作成されます。MobileTogether Designer インターフェイスはソリューションのインターフェイスと同様の外観です。すべてのコントロールが無効化されているため、クライアントインターフェースデザインは不可能なことに気が付きます。代わりに、サービスのために定義するすべてのアクションは、プロジェクトイベントサービスの実行中のタブ内で定義される必要があります。
2. ページソースを使用する場合、`$XML1` という名前のXML ページソースが追加されているページソースペインに追加します(*下のスクリーンショットを参照*)。
3. サービスのアクションダイアログを以下の方法で開くことができます: ページデザインの中央にある**サービスアクションツリー**ボタンをクリックします、または、スタイル & プロパティペイン内で、**サービスアクション** プロパティの追加ダイアログボタンをクリックします。

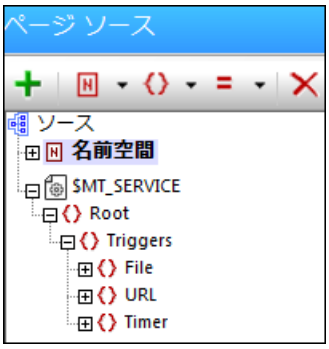


4. ダイアログの左側のペインは、サービスのために使用することのできるすべてのアクションを表示しています(*上のスクリーンショットでは表示されていません*)。使用することのできないアクションは無効化されており、(灰色で表示されています)。サービスの実行中 タブの中にサービスとして実行するアクションをドラッグします。上のスクリーンショットでは、2つのアクションが追加されています: (i) *ファイルからロードする*はMobileTogether Server ログファイルを `$XML1` ページソースにロードします。(ii) *電子メールの送信* は3人の受信者にMobileTogether Server ログの `$XML1` ツリーを添付ファイルとした電子メールを送信します。
5. 「**OK**」をクリックして、サービスのアクションの作成を完了します。
6. `.mtd` のサービスとファイル型のために適した名前を持つファイルを保存 (**Ctrl+S**) します。

メモ 各サービスのために1つのみのアクションのセットを作成することができます。「**ファイル**」「**新規のサービス**」を2番目に選択すると、新規のからのサービスファイルが作成されます。

$MT_SERVICE ページソース

サービスデザインが作成されると`$MT_SERVICE` ページソースは自動的に作成されます。下のスクリーンショットは、ページソースの構造を示しています。



$MT_SERVICES ページソースの構造

```
<Root>
   <Triggers>
      <File name="" filename="" reason=""/>
      <URL name="" url=""/>
      <Timer name=""/>
   </Triggers>
</Root>
```

ランタイムでは、サービスのためにセットされたトリガーに関するデータはページソースにサーバーからパスし、ページソースの適切なノード内に保管されます。例えば、ファイルシステム トリガーを有効化するファイル名は、ページソースの **//File/@filename** ノード内に保管されます。サービスアクションのXPath 式は、これらのノードにアクセスし、これらのノード内に保管されたランタイムの情報はXPath 式により使用されることができます。例えば、(新規のファイルの作成、ファイルの変更、または、ファイルの削除などの)サービスアクションをトリガーするファイルの名前を、トリガーがアクティブ化される理由と共に電子メールの送信 アクション内で送信することができます。これらはランタイムの前に知ることのできない情報です。

ページソースの関連したノードは自動的にランタイムで記入されるため、 **$MT_SERVICE** ページソースの構築、または、ノードを作成する心配は必要ありません。サーバー側 トリガーに関する(追加の) ランタイムの情報のソースとして使用することができます。XPath 式を使用してこの情報にアクセス および、使用することができます: (i) サービスアクションを情報の値を条件にし、(ii) サービスアクション内でデータとしてパスすることができます。

メモ シミュレーションに関しては、シミュレーションのためにのみ使用される**$MT_SERVICE** ページソースにデータを入力することができます。このデータはランタイムに受信されたデータをシミュレートします。シミュレーションのために **$MT_SERVICE** ページソースの作成する方法は、サービストリガーシミュレーション内の説明を参照してください。

サービスプロパティ

スタイル & プロパティペイン内でサービスのためにデータ取得のタイムアウト(秒数)を設定することができます(*上のスクリーンショットを参照*)。タイムアウトの期間の前に、データがサーバーから抽出されない場合、*エラー時* の設定内で設定されたアクションが実行されます(*上のスクリーンショットを参照*)。

サーバーの外部のソース(DB またはURL など) からデータを取得する際のサーバーの待ち時間。値は整数の秒数で、入力またはコンボ ボックスのドロップダウン リストから選択されます。 デフォルトの値は10 秒です。タイムアウトが過ぎると、クライアントにメッセージが表示されます。例外は、ロード アクションの *エラー時*の設定が`Continue` となっている場合です。この場合、`Continue` 設定の*エラー時* ツリー内で定義されているアクションが実行されます。

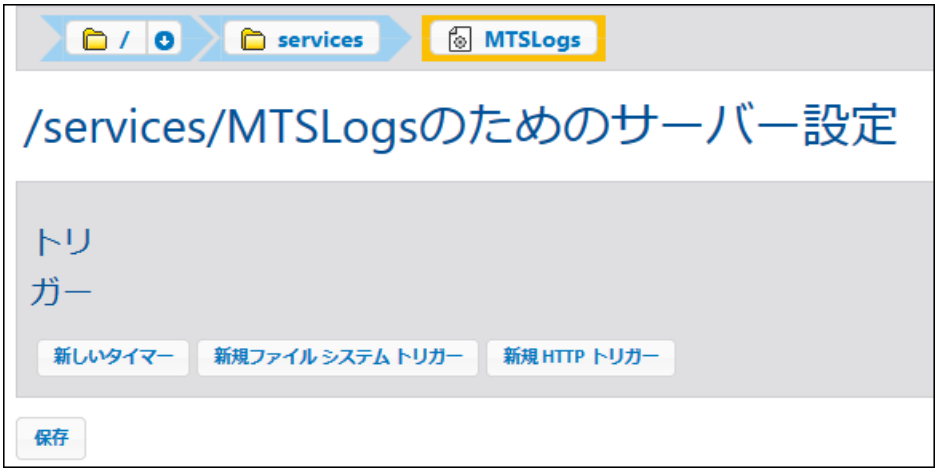
19.2 サービスのデプロイ

サービスファイルが保存されると、MobileTogether Server Advanced Edition にデプロイする必要があります。 **「ファイル | MobileTogether Server へデプロイ」**. を使用して、ソリューションのデプロイと同様に行います。

メモ MTS サービス ファイルを**MobileTogether Server Advanced Edition** のみにデプロイすることができます。スタンダードエディションにデプロイを試みるとエラーが発生します。

19.3 サービスの実行

MobileTogether Server Advanced Edition にサービスがデプロイされると、構成(または設定) インターフェースにアクセスすることのできるフォークフローダブにリストされます。サービスの構成 (または、設定) インターフェースにより、サービスを実行するトリガーを定義し、管理することができます (*下のスクリーンショット参照*)。



次の種類のトリガーを作成することができます:

- 特定の期間内でどの日時にどの頻度でサービスが実行されるか指定することのできる、*タイマー トリガー*。
- サーバー上のファイル、または、ディレクトリ上の変更をチェックして、サービスをトリガーすることのできる、*ファイルシステム トリガー*。
- 指定されているURI ロケーションでリソースへの変更をチェックしてサービスを実行することのできる*HTTP トリガー*。

サービスの設定インターフェイスへのアクセス、および、サービスのためのトリガーの設定方法に関しては、MobileTogether Server ドキュメントを参照してください。

チャプター 20

メニュー コマンド

20 メニュー コマンド

MobileTogether Designer メニュー コマンドは次のメニューに整理されています:

- ファイル
- 編集
- プロジェクト
- ページ
- テーブル
- 表示
- ルーツ
- ウィンドウ
- ヘルプ

(C) 2018 Altova GmbH

20.1 ファイル

[ファイル] メニューは次のコマンドを含みます:

- 新規作成
- 新規のサービス
- 開く
- 再ロード
- 閉じる
- すべて閉じる
- 非アクティブをすべて閉じる
- 保存
- 名前を付けて保存
- コピーに名前を付けて保存
- すべて保存
- MobileTogether Server へデプロイ
- MobileTogether Server から開く
- MobileTogether Server から削除する
- モバイルアプリのためのコード生成
- メールで送信
- 印刷
- 印刷プレビュー
- 印刷設定
- 最近使用したファイル
- 終了

新規作成

新規作成

アイコン

ショートカット

Ctrl+N

説明

新しいドキュメントタブを開き、空のMobileTogether デザインファイルをこのタブにロードします。ドキュメントは一時的にメモリに保管されます。保存する場合は、 `.mtd` 拡張子と共にディスクに保存する必要があります。

新規のサービス

- 新規のサービス
  - *アイコン*

  - *説明*

    メインウィンドウで新規のドキュメントタブを開き、空のMobileTogether Server サービスファイルをこのタブにロードします。このドキュメントは一時的にメモリに保存されます。保管する場合、`.mtd` 拡張子と共にディスクに保存される必要があります。MobileTogether Server サービス機能の詳細に関しては、セクション[サーバーサービス](#)を参照してください。

開く

- 開く
  - *アイコン*

  - *ショートカット*

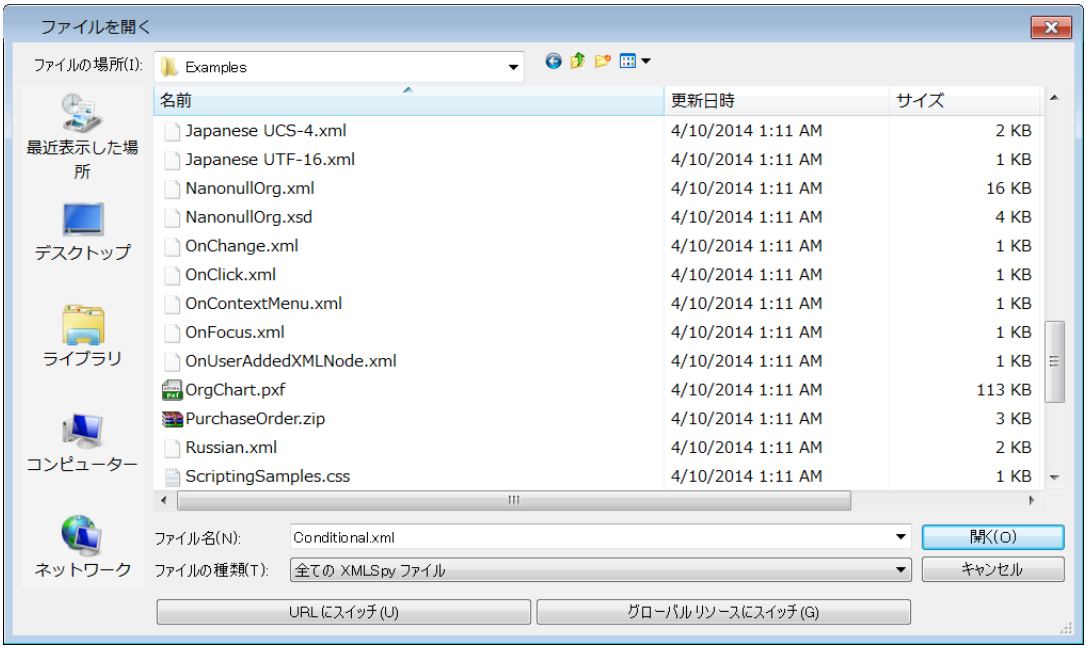
    Ctrl+O

  - *説明*

    開くMobileTogether デザインファイル(.mtd ファイル) を選択することのできる開くダイアログをポップアップします。MTD ファイルがメイン ウィンドウの新しいタブに開かれます。

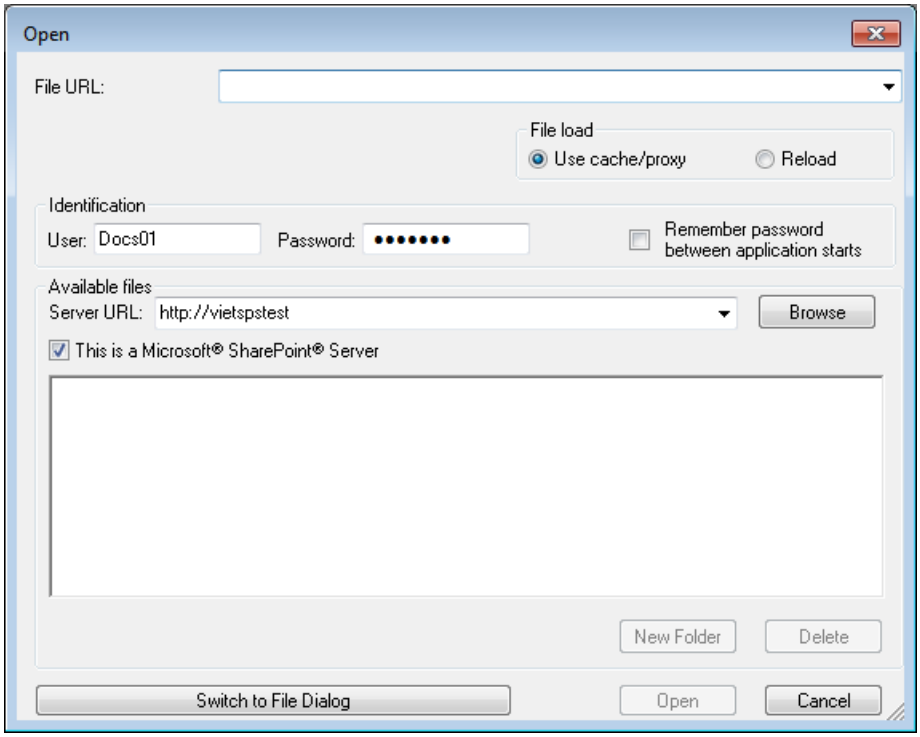
- URL を使用してファイルを開くまたは保存する

  「ファイルを開く」および「ファイルの保存」ダイアログでは、URL を使用して必要なファイルを選択またはファイルを保存することができます。(*下のスクリーンショット参照*)。**「URL に切り替える」**をクリックして選択プロセスに移動します。



URL から(開くまたは保存する) ファイルを選択するには以下を行ってください:

1. **「URL に切り替える」**コマンドをクリックします。これにより、開くまたは保存ダイアログの URL モードにスイッチされます。(下のスクリーンショットは開くダイアログを表示しています)。



2. アクセスするURL をサーバー *URL* フィールドに入力します (上のスクリーンショット)。サーバーが Microsoft® SharePoint® Server の場合、 *Microsoft® SharePoint® Server* チェックボックスをチェックしてください。この種類のサーバー上にあるファイルと作業する場合、下のMicrosoft® SharePoint® Server メモを参照してください。
3. サーバーがパスワードにより保護されている場合、ユーザーID とパスワードをユーザー とパスワードのフィールドに入力してください。
4. **「参照」**をクリックして、サーバーのディレクトリ構造をナビゲートします。
5. フォルダーツリー内でロードするファイルを参照してクリックします。ファイルURL がファイルURL フィールドに表示されます (*上のスクリーンショット参照*)。 **「開く」**または **「保存」**ボタンはこの時点からアクティブになります。
6. **「開く」**をクリックしてファイルをロードするか、 **「保存」**をクリックして保存します。

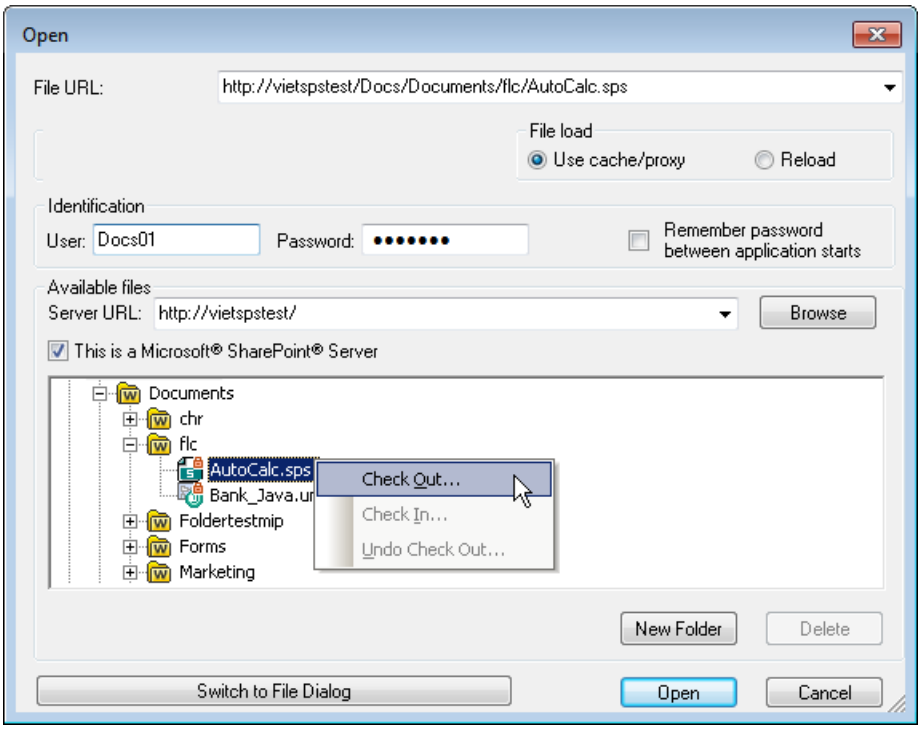
以下の点に注意してください:

- WebDAV をサポートするサーバーとMicrosoft SharePoint Servers のみで参照機能を使用することができます。サポートされるプロトコールはFTP、HTTP およびHTTPS です。
- ファイルを開く際のロード処理を更に管理するには、ローカルのキャッシュまたはプロキシのサーバーからファイルをロードします。(ファイルが事前にロードされることにより更に処理スピードを向上することができます)。また、電子出版またはデータベースシステムなどの作業中のファイルを再ロードして、 **「再ロード」**オプションを選択します。

▾ Microsoft® SharePoint® Server メモ

Microsoft® SharePoint® Servers に関する以下の点に注意してください:

- 「利用可能なファイル」ペインに表示されるディレクトリ構造では、ファイルアイコンにはファイルのチェックイン/チェックアウトの状態を表示するシンボルがあります (*下のスクリーンショット*)。



ファイルを右クリックすると、ファイルで使用することのできるコンテキストメニューがポップアップします(*上のスクリーンショット*)。

- 異なるファイルアイコンが以下に表示されています:

	チェックインされています。チェックアウトできます。
	他のユーザーによりチェックアウトされています。チェックアウトできません。
	ローカルでチェックアウトされています。編集およびチェックインすることができます。

- ファイルをチェックアウトした後、Altova アプリケーションを使用して編集することができ、**「ファイル | 保存 (Ctrl+S)」**を使用して保存することができます。
- 編集されたファイルを「URL を開く」ダイアログのコンテキストメニューからチェックインすることができます(*上のスクリーンショット参照*) または、アプリケーションのメインウィンドウのファイルタブを右クリックすることによりポップアップするコンテキストメニューを使用して (*下のスクリーンショット*).

- ファイルが他のユーザーによりチェックアウトされている場合、チェックアウトすることができません。
- ファイルがローカルでチェックアウトされている場合、コンテキストメニューのチェックアウトを元に戻すコマンドを使用してチェックアウトを元にもどすことができます。これによりファイルを変更することなくサーバーに戻すことができます。
- Altova アプリケーションでファイルをチェックアウトした場合、他の Altova アプリケーションでチェックアウトするこ

とはできません。ファイルは既にチェックアウト済みと見なされます。この時点で使用できるMicrosoft® SharePoint® Server をサポートするAltova アプリケーションのコマンドは、以下の通りです: **「チェックイン」**および **「チェックアウトを元に戻す」**です。

再ロード

- 再ロード
  - *アイコン*

  - *説明*

    MobileTogether Designer 外で変更された開かれているドキュメントを再ロードします。MobileTogether Designer 外でドキュメントが変更されると、変更されたドキュメントを再ロードを問うプロンプトが表示されます。再ロードを選択すると、ファイルが最後に保存された時点からの変更が失われます。

閉じる、すべて閉じる、非アクティブをすべて閉じる

閉じる

説明

アクティブなドキュメントウィンドウを閉じる。ファイルが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

すべて閉じる

説明

すべての開かれているドキュメントウィンドウが閉じられます。ドキュメントが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

非アクティブをすべて閉じる

説明

アクティブなドキュメントウィンドウ以外のすべての開かれているドキュメントウィンドウを閉じます。ドキュメントが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

保存、名前を付けて保存、コピーに名前を付けて保存、すべて保存

- 保存
  - *アイコン*

  - *ショートカット*

    Ctrl+S

  - *説明*

    アクティブなドキュメントのコンテンツをファイルに保存します。

- 名前を付けて保存
  - *説明*

    名前とアクティブなドキュメントを保存するファイルの場所を入力する[名前を付けて保存] ダイアログがポップアップします。新しく保存されたドキュメントは、メイン ウィンドウのアクティブ タブにある元のドキュメントを置き換えます。

- **コピーに名前を付けて保存**
  - *説明*

    アクティブ ドキュメントを保存し名前と場所を入力する[コピーに名前を付けて保存] ダイアログ ボックスがポップアップします。保存されるドキュメントはアクティブ ドキュメントのコピーです。新しく保存されたドキュメントは MobileTogether Designer で開かれません。元のドキュメントがメイン ウィンドウでアクティブなドキュメントとして残ります。

- すべて保存
  - *アイコン*

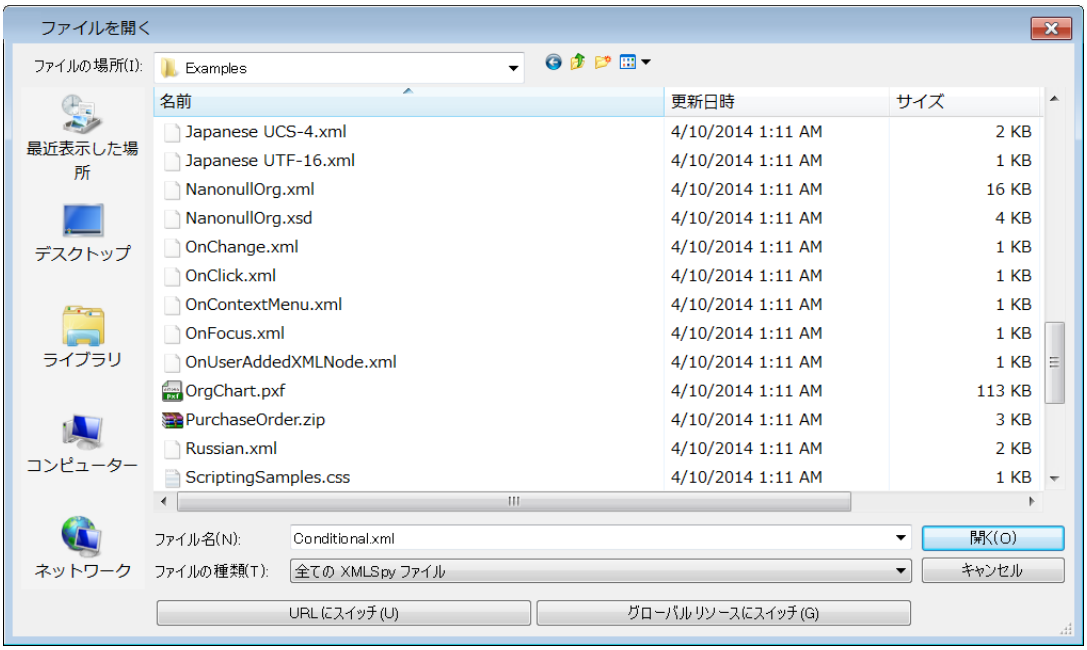
  - *説明*

    開かれているドキュメントに加えられたすべての変更を保存します。コマンドは複数のドキュメントを同時に編集する際に役に立ちます。(例えば、作成されてすぐ) ドキュメントが以前に保存されている場合、 [名前を付けて保存] ダイアログ ボックスがそのドキュメントのために表示されます。

(C) 2018 Altova GmbH

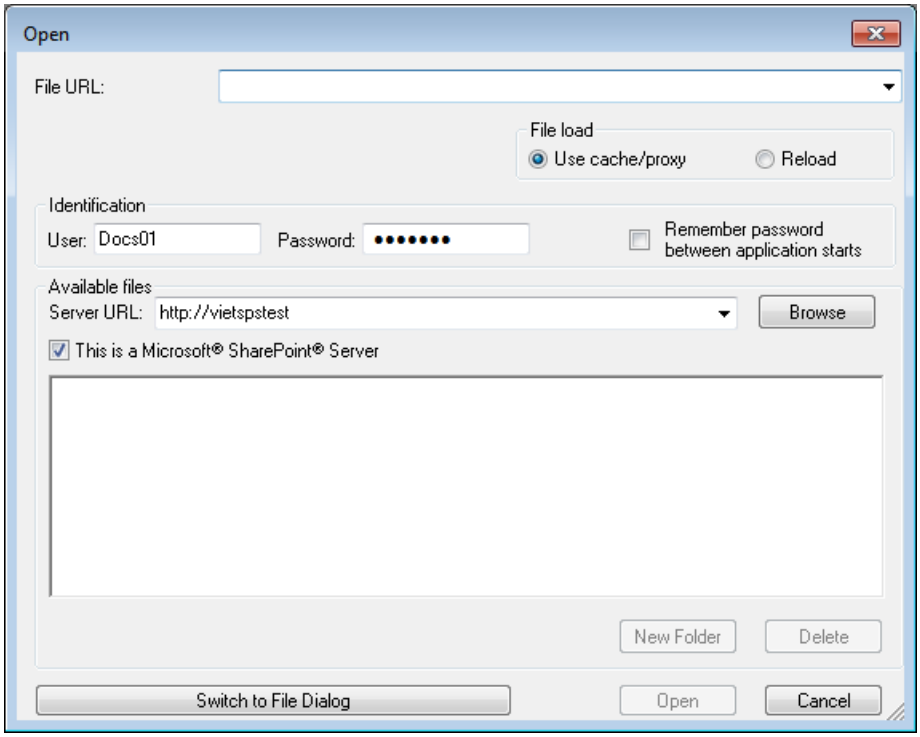
▾ URL を使用してファイルを開くまたは保存する

「ファイルを開く」および「ファイルの保存」ダイアログでは、URL を使用して必要なファイルを選択またはファイルを保存することができます。(*下のスクリーンショット参照*)。 **「URL に切り替える」**をクリックして選択プロセスに移動します。



URL から(開くまたは保存する)ファイルを選択するには以下を行ってください:

1. **「URL に切り替える」**コマンドをクリックします。これにより、開くまたは保存ダイアログの URL モードにスイッチされます。(下のスクリーンショットは開くダイアログを表示しています)。



2. アクセスするURL をサーバー *URL* フィールドに入力します (上のスクリーンショット)。サーバーが Microsoft® SharePoint® Server の場合、 *Microsoft® SharePoint® Server* チェックボックスをチェックしてください。この種類のサーバー上にあるファイルと作業する場合、下のMicrosoft® SharePoint® Server メモを参照してください。
3. サーバーがパスワードにより保護されている場合、ユーザーID とパスワードをユーザー とパスワードのフィールドに入力してください。
4. **「参照」**をクリックして、サーバーのディレクトリ構造をナビゲートします。
5. フォルダーツリー内でロードするファイルを参照してクリックします。ファイルURL がファイルURL フィールドに表示されます (*上のスクリーンショット参照*)。 **「開く」**または **「保存」**ボタンはこの時点からアクティブになります。
6. **「開く」**をクリックしてファイルをロードするか、 **「保存」**をクリックして保存します。

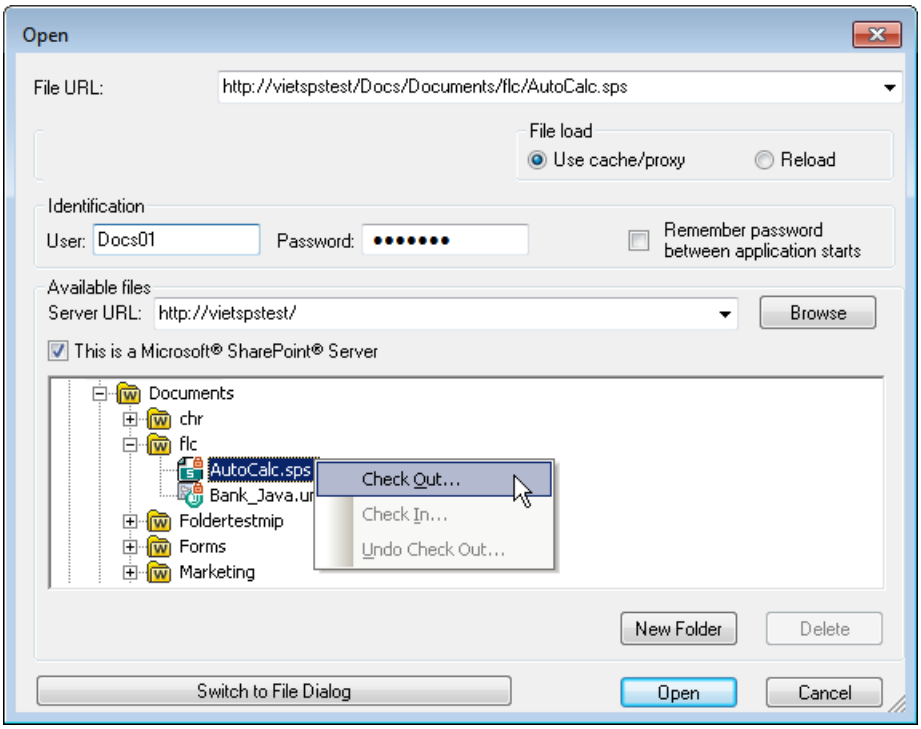
以下の点に注意してください:

- WebDAV をサポートするサーバーとMicrosoft SharePoint Servers のみで参照機能を使用することができます。サポートされるプロトコールはFTP、HTTP およびHTTPS です。
- ファイルを開く際のロード処理を更に管理するには、ローカルのキャッシュまたはプロキシのサーバーからファイルをロードします。(ファイルが事前にロードされることにより更に処理スピードを向上することができます)。また、電子出版またはデータベースシステムなどの作業中のファイルを再ロードして、 **「再ロード」**オプションを選択します。

▾ Microsoft® SharePoint® Server メモ

Microsoft® SharePoint® Servers に関する以下の点に注意してください:

- 「利用可能なファイル」ペインに表示されるディレクトリ構造では、ファイルアイコンにはファイルのチェックイン/チェックアウトの状態を表示するシンボルがあります (*下のスクリーンショット*)。



ファイルを右クリックすると、ファイルで使用することのできるコンテキストメニューがポップアップします(*上のスクリーンショット*)。

- 異なるファイルアイコンが以下に表示されています:

	チェックインされています。チェックアウトできます。
	他のユーザーによりチェックアウトされています。チェックアウトできません。
	ローカルでチェックアウトされています。編集およびチェックインすることができます。

- ファイルをチェックアウトした後、Altova アプリケーションを使用して編集することができ、**「ファイル | 保存 (Ctrl+S)」**を使用して保存することができます。
- 編集されたファイルを「URL を開く」ダイアログのコンテキストメニューからチェックインすることができます(*上のスクリーンショット参照*) または、アプリケーションのメインウィンドウのファイルタブを右クリックすることによりポップアップするコンテキストメニューを使用して (*下のスクリーンショット*).

- ファイルが他のユーザーによりチェックアウトされている場合、チェックアウトすることができません。
- ファイルがローカルでチェックアウトされている場合、コンテキストメニューのチェックアウトを元に戻すコマンドを使用してチェックアウトを元にもどすことができます。これによりファイルを変更することなくサーバーに戻すことができます。
- Altova アプリケーションでファイルをチェックアウトした場合、他の Altova アプリケーションでチェックアウトするこ

とはできません。ファイルは既にチェックアウト済みと見なされます。この時点で使用できるMicrosoft® SharePoint® Server をサポートするAltova アプリケーションのコマンドは、以下の通りです: **「チェックイン」**および **「チェックアウトを元に戻す」**です。

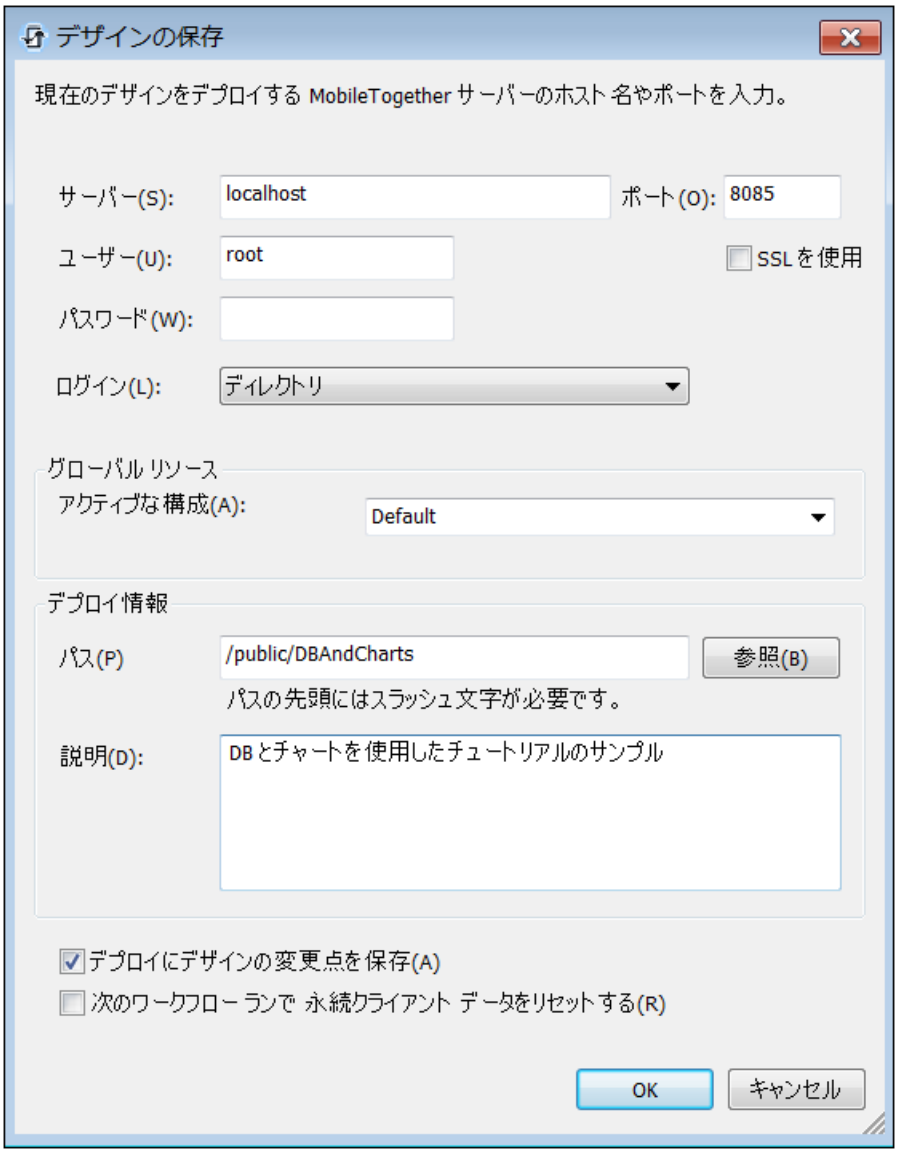
MobileTogether Server へデプロイ

MobileTogether Server へデプロイ

アイコン

説明

(i) サーバーへの接続の詳細と(ii) サーバー上のプロジェクトのデプロイ プロパティを指定することのできる [MobileTogether Server へデプロイ] ダイアログを開きます。 **[OK]** をクリックすると、現在アクティブな MobileTogether デザインファイルが MobileTogether Server へデプロイされます。



デプロイには以下の詳細が必要になります:

- **サーバー名とポート**: ポートはMobileTogether Server 管理者ポートを指しており、MobileTogether Server の*管理者ポートの設定* と一致する必要があります。SSL を使用する場合、MobileTogether Server のセキュリティ保護されたポートを使用してください。
- **ユーザー名とパスワード**: デザイナーからワークフローを保存する特権を与えられているユーザーのユーザー名とパスワード。MobileTogether Server ユーザーとその特権はMobileTogether Server の*ユーザーとロールの設定*で指定されています。
- **ログイン**: ログインが直接かまたはドメインユーザーとしてかを指定してください。(ログインがドメインユーザーとしての場合、ドメイン固有のユーザー名 および パスワードを使用することができます)。コンボボックスのオプションから、直接またはドメインを選択してください。コンボボックスに表示されているドメインのみが、アクティブディレクトリログインのために構成されています。*アクティブディレクトリログインのためのサーバーの構成の詳細に関しては、MobileTogether Server ユーザーマニュアルを参照してください。*
- **グローバルリソースのアクティブな構成**: コンボボックスのドロップダウンリストから使用可能なアクティブな構成を選択してください。使用可能な構成は、グローバルリソース定義ファイルで定義されており、自動的に取得することができます。詳細はAltova グローバルリソースのセクションを参照してください。
- **デプロイパスとソリューションの説明**: サーバーに表示されるデプロイされたソリューションのパスと名前 とソリューションの説明。
- **デプロイにデザインの変更点を保存**: プロジェクトファイルはデプロイ前に保存され、デザインの最新の変更が含まれます。
- **次のワークフローランで永続クライアントデータをリセットする**: ソリューションが次回実行される時に永続クライアントデータをリセットします。

関連項目

プロジェクトファイルのロケーション
[ファイル] ペイン
プロジェクトのデプロイ
サーバー上のデータストレージ

MobileTogether Server から開く

MobileTogether Server から開く

アイコン

説明

MobileTogether Server のロケーションからMobileTogether Server にデプロイされた MobileTogether デザインファイルを開きます。**[参照]** ボタンを使用して開くサーバー上のファイルを選択します。

MobileTogether Serverからデザインを開く際に以下の詳細が必要になります:

- **サーバー名とポート**: ポートはMobileTogether Server 管理者ポートを指しており、MobileTogether Server の*管理者ポートの設定* と一致する必要があります。SSL を使用する場合、MobileTogether Server のセキュリティ保護されたポートを使用してください。
- **ユーザー名とパスワード**: デザイナーからワークフローを保存する特権を与えられているユーザーのユーザー名とパスワード。MobileTogether Server ユーザーとその特権はMobileTogether Server の*ユーザーとロールの設定*で指定されています。
- *ログイン*: 直接またはドメインユーザーとしてのログインを指定します。(ドメインユーザーとしてのログインの場合は、ドメインと特有のユーザー名とパスワードを使用することができます。) コンボボックスのオプションから、直接を選択または使用するドメインを選択してください。コンボボックスに表示されているドメインのみがアクティブディレクトリログインのためにサーバー上で構成されています。*アクティブディレクトリログインのためのサーバー構成の詳細については、「MobileTogether Server ユーザーマニュアル」を参照してください。*
- **パスとソリューションの名前 (デザイン)**: デプロイされたソリューションのパスと名前をクリックして MobileTogether Server にデプロイされたすべてのソリューションを参照します。

MobileTogether Server から削除する

- MobileTogether Server から削除する
  - *アイコン*

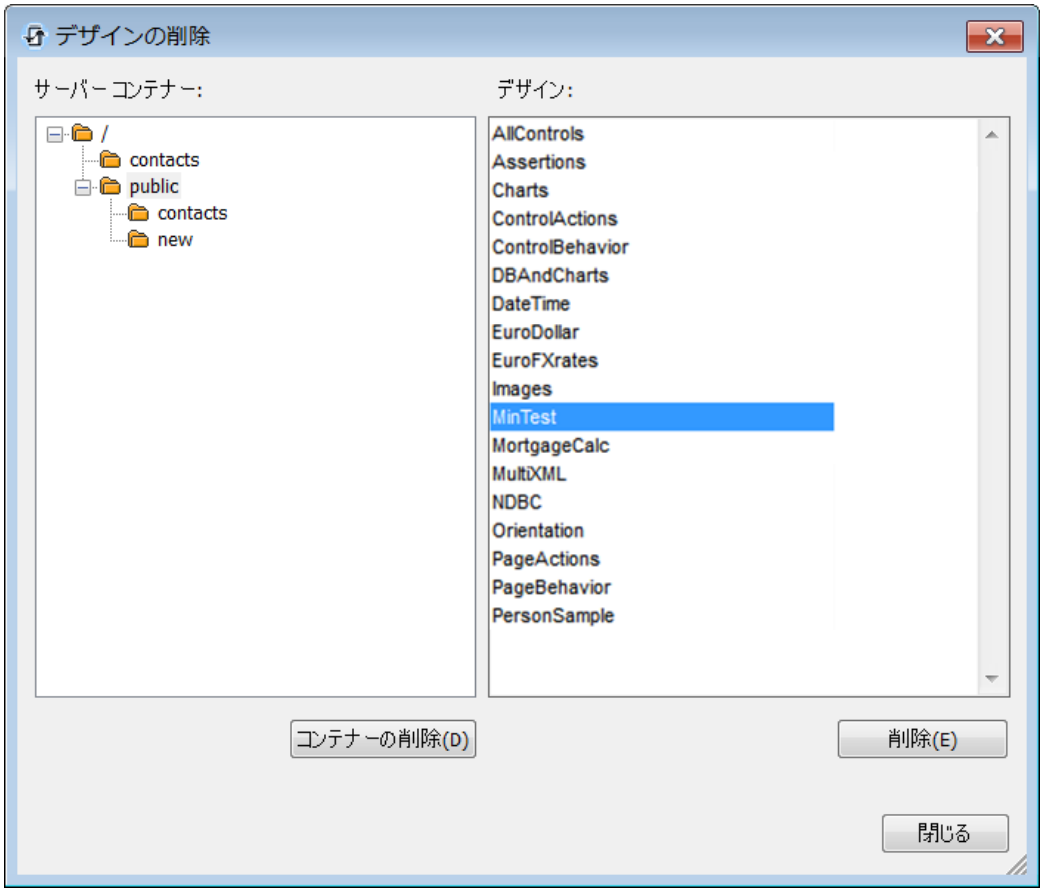
  - *説明*

MobileTogether Server からデプロイされたデザインファイルを削除します。**[続ける]** ボタンを使用して、サーバー上の削除するファイルを選択してください。

サーバーに接続するには以下の詳細が必要になります:

- **サーバー名とポート**: ポートは MobileTogether Server 管理者ポートを指しており、MobileTogether Server の*管理者ポートの設定* と一致する必要があります。SSL を使用する場合、MobileTogether Server のセキュリティ保護されたポートを使用してください。
- **ユーザー名とパスワード**: デザイナーからワークフローを保存する特権を与えられているユーザーのユーザー名とパスワード。MobileTogether Server ユーザーとその特権は MobileTogether Server の*ユーザーとロールの設定*で指定されています。
- *ログイン*: 直接かまたはドメインユーザーとしてのログインを指定します。(ドメインユーザーとしてのログインの場合は、ドメインと特有のユーザー名とパスワードを使用することができます。) コンボボックスのオプションから、直接を選択または使用するドメインを選択してください。コンボボックスに表示されているドメインのみがアクティブディレクトリログインのためにサーバー上で構成されています。*アクティブディレクトリログインのためのサーバー構成の詳細については、「MobileTogether Server ユーザーマニュアル」を参照してください。*

[続ける] をクリックすると、MobileTogether Server フォルダー(コンテナー) とソリューション(デザイン) を表示するウィンドウが表示されます(*下のスクリーンショット*)。削除するコンテナーまたはデザインを参照、選択し、**[コンテナーの削除]** または **[削除]** をクリックします。

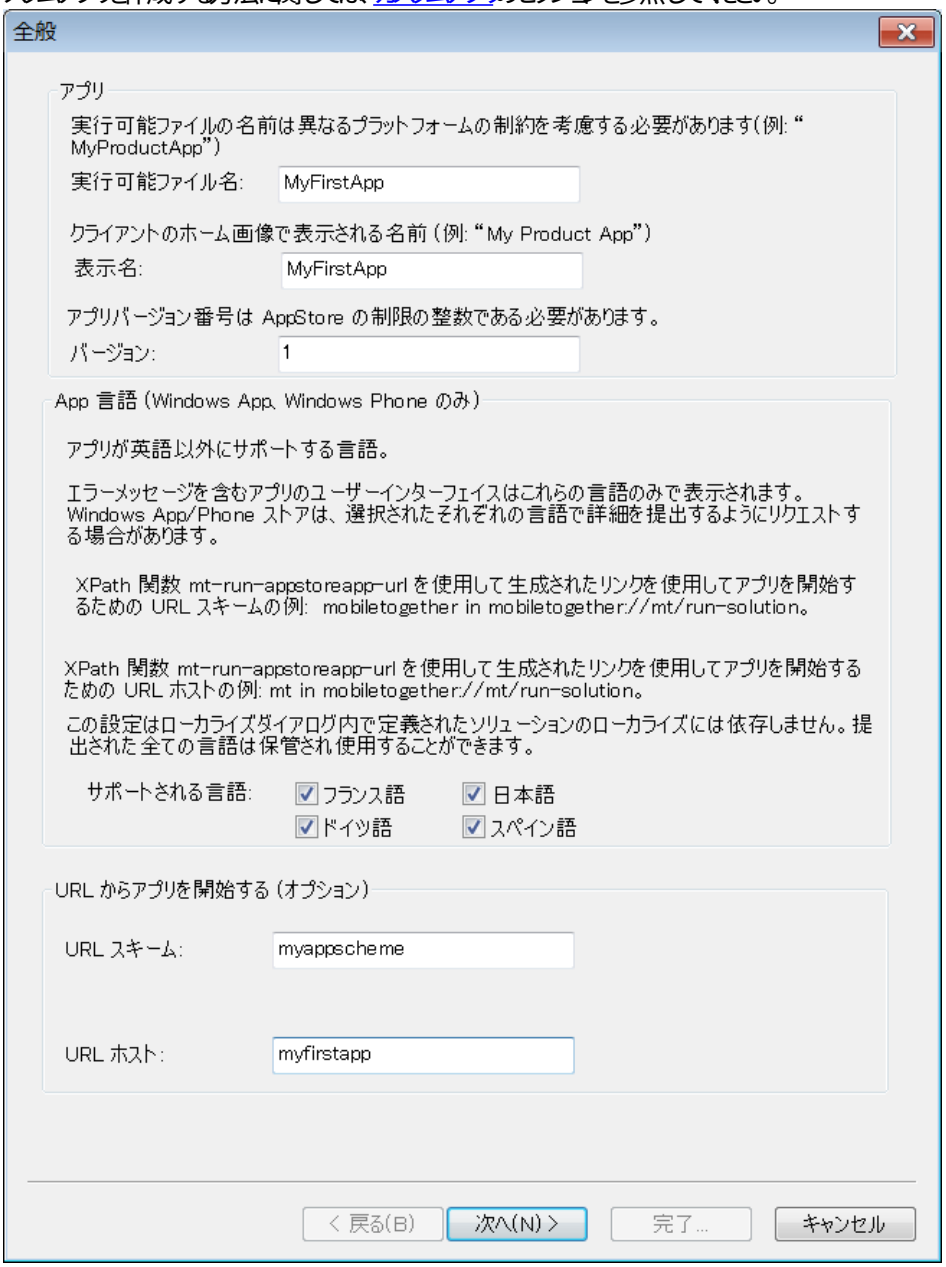

デザインの削除
サーバーコンテナー:
デザイン:
/
contacts
public
contacts
new
AllControls
Assertions
Charts
ControlActions
ControlBehavior
DBAndCharts
DateTime
EuroDollar
EuroFXrates
Images
MinTest
MortgageCalc
MultiXML
NDBC
Orientation
PageActions
PageBehavior
PersonSample
コンテナーの削除(D)
削除(E)
閉じる

モバイルアプリのためのコード生成

モバイルアプリのためのコード生成

説明

カスタムモバイルアプリの作成のためのダイアログを開きます(下のスクリーンショット)。ウィザードを使用しモバイルカスタムアプリを作成する方法に関しては、カスタムアプリのセクションを参照してください。



メールで送信

メールで送信

アイコン

説明

アクティブな MobileTogether デザイン ドキュメント (MTD ドキュメント) を電子メールの添付として送信します。コマンドを選択する際、アクティブな MTD ドキュメントが添付された電子メールが開かれます。電子メールの *宛先:* フィールドに受信者の名前を、*件名:* フィールドに件名の情報を入力して、電子メールアプリケーションの **[送信]** コマンドをクリックします。

印刷

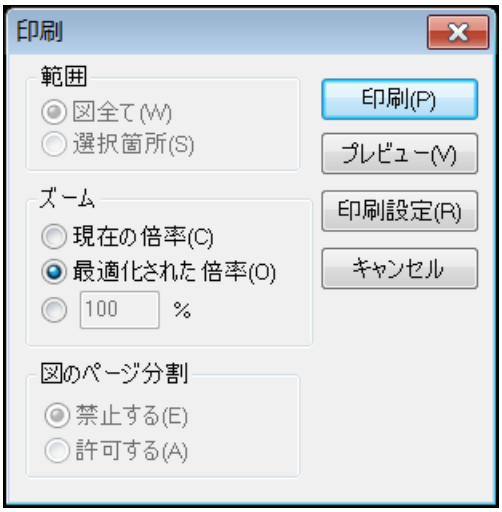
- 印刷
  - *アイコン*

  - *ショートカット*

    Ctrl+P

  - *説明*

    現在アクティブなドキュメントの印刷オプションを選択することのできる印刷ダイアログボックスを開きます (*下のスクリーンショット*)。



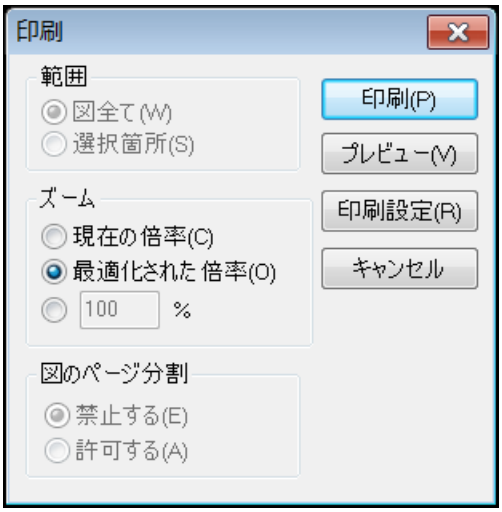
現在印刷中のジョブに適用できるオプションが有効されていると、次のオプションを使用することができます:

- *範囲:* デザイン図全体または現在の選択箇所を印刷するか選択します。
- *ズーム:* 印刷で使用されるズームのレベルを設定します。
- *図のページ分割:* イメージの分割を(許可) または(禁止) するかを選択します。

印刷プレビュー、印刷設定

- 印刷プレビュー
  - *説明*

    [印刷] ダイアログボックスを開きます。(*下のスクリーンショット*)。**[プレビュー]** をクリックして、ダイアログで指定された設定により現在アクティブなドキュメントをプレビューします。



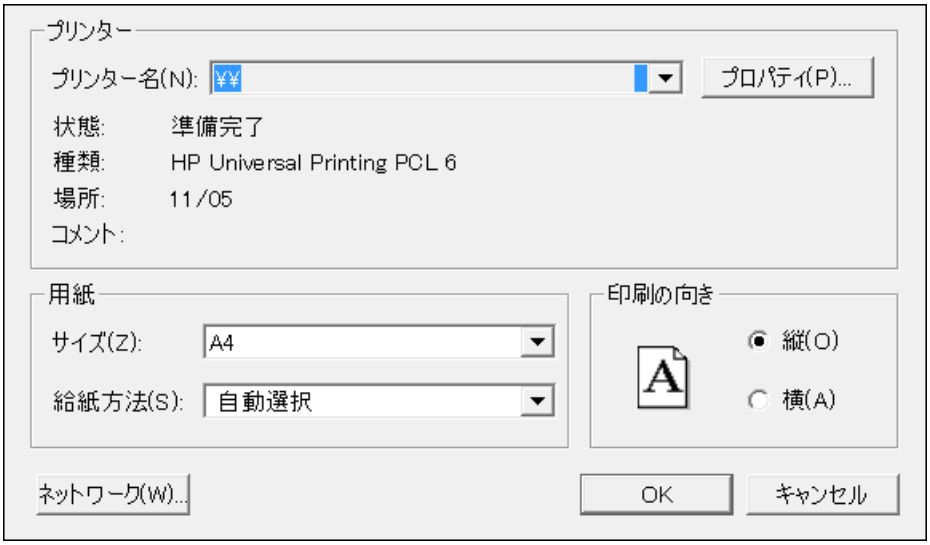
印刷プレビューモードでは、プレビューウィンドウのトップ左の印刷プレビューツールバーにより印刷とプレビューに関連したオプションが与えられています。**[拡大]** と**[縮小]** ボタンを使用してプレビューを拡大または縮小することができます。ページの拡大によりページの長さ全体がプレビューウィンドウにフィットしない場合、**[1 ページ /2 ページ]** ボタンにより1 または2 ページごとに表示することができます。**[次のページ]** と**[前のページ]** ボタンを使用してページ間を移動することができます。ツールバーには全てのページを印刷する、およびプレビューウィンドウを閉じるボタンが含まれています。

メモ 印刷プレビューで背景の色とイメージを有効化するには、次を行います: (i) Internet Explorer の**[ツール]** メニューから**[インターネットのオプション]** をクリックし、詳細設定タブをクリックします; (ii) 設定ボックスの、印刷から背景色とイメージを印刷のチェックボックスをチェックします、そして (iii) **[OK]** クリックします。

印刷設定

- *説明*

  用紙書式やページの向きなどのプリンターの設定を指定する、プリンター固有の [印刷セットアップ] ダイアログボックスを表示します。これらの設定は今後すべての印刷ジョブに適用されます。


プリンター
プリンター名(N):
¥¥
プロパティ(P)...
状態:
準備完了
種類:
HP Universal Printing PCL 6
場所:
11/05
コメント:
用紙
サイズ(Z):
A4
給紙方法(S):
自動選択
印刷の向き
縦(O)
横(A)
ネットワーク(W)...
OK
キャンセル

最近使用したファイル、閉じる

最近使用したファイル

説明

「ファイル」メニューの一番下に最近使用した順に9 つのファイルのリストが表示されます。名前をクリックしてファイルを開くことができます。キーボードを使用してファイルを開く場合、 **「Alt+F」**を押して **「ファイル」**メニューを開き 開く番号を押します。

閉じる

説明

MobileTogether Designer を終了します。保存されていない変更のある開かれているファイルがある場合、これらの変更を保存するようにプロンプトされます。MobileTogether Designer はプログラム設定への変更と最近使用されたファイルの情報も保存します。

20.2 編集

[編集] メニューは次のコマンドを含みます:

- 元に戻す
- やり直し
- 切り取り
- コピー
- 貼り付け
- 削除
- すべて選択

(C) 2018 Altova GmbH

元に戻す、やり直し

- 元に戻す
  - *アイコン*

  - *ショートカット*

    Ctrl+Z

  - *説明*

    制限無く「元に戻す」をサポートします。全てのアクションは元に戻すことができ、1つのコマンドを元に戻してから次のコマンドを元にもどすことができます。「元に戻す」の履歴は **「保存」**コマンドを使用した後に保持され、変更を保存した前の状態のドキュメントに戻ることができます。ます。 **「元に戻す」**と **「やり直し」** コマンドを使用して履歴を前後することができます。(下の **「やり直し」** *コマンドを参照してください*)。

- やり直し
  - *アイコン*

  - *ショートカット*

    Ctrl+Y

  - *説明*

    前に元に戻されたコマンドをやり直すことができます。これにより、完成された作業の完全な履歴を得ることができます。 **「元に戻す」**と **「やり直し」**コマンドを使用して、履歴を前後することができます。

切り取り、コピー、貼り付け、削除

- 切り取り
  - *アイコン*

  - *ショートカット*

    Ctrl+X または **Shift+Del**

  - *説明*

    現在の場所から選択箇所を削除し、選択されたテキストまたはアイテムをクリップボードにコピーします。

- コピー
  - *アイコン*

  - *ショートカット*

    Ctrl+C

  - *説明*

    現在のロケーションから選択範囲を削除することなく、選択されたテキストまたはアイテムをクリップボードにコピーします。このコマンドはMobileTogether Designer 内でデータを複製するために使用することができます。

- 貼り付け
  - *アイコン*

  - *ショートカット*

    Ctrl+V

  - *説明*

    クリップボードの内容を現在カーソルのある場所に挿入します。

- 削除

(C) 2018 Altova GmbH

- *アイコン*

- *ショートカット*

  Del

- *説明*

  現在選択されているテキストまたはアイテムをクリップボードに貼り付けることなく削除します。

すべて選択

- すべて選択
  - *ショートカット*

    Ctrl+A

  - *説明*

    ドキュメント全体のコンテンツを選択します。

(C) 2018 Altova GmbH

20.3 プロジェクト

[プロジェクト] メニューはプロジェクト全体に適用するコマンドを含み、次のコマンドを含みます:

- 検証
- ページソース構造の再ロード
- ワークフローのシミュレーション
- クライアントでの試行
- ワークフローのシミュレーションにサーバーを使用
- 新しいテストケースの記録
- テストケースの再生
- クライアント上でのテストケースのトライアルラン
- テストケースとテストランの管理
- グローバル変数
- すべてのグローバル変数の使用法のリスト
- すべてのページソース変数の使用法のリスト
- XPath/XQuery 関数
- すべてのユーザー定義 XPath/XQuery 関数の使用法のリスト
- アクショングループ
- すべてのアクショングループの使用法のリスト
- スタイルシート
- スタイルシートのすべての使用方法をリストする
- リッチテキストスタイルシート
- キャッシュの概要
- ローカライズ
- シミュレーションする言語
- すべてのファイルとディレクトリの参照のリスト
- すべての外部データ参照のリスト
- 使用されていない関数、ユーザー編集、およびアクショングループのリスト
- OAuth 設定の維持
- OAuth 設定のインポート
- iOS プッシュ通知ボタンセット

検証

検証

アイコン

説明

現在アクティブなプロジェクトを検証します。もしプロジェクトが複数のページを含む場合、すべてのページが検証されます。検証の結果は[メッセージ] ペインにエラーの数と警告としてレポートされます。エラーや警告が検出されるとメッセージが各項目について表示されます。

ページソース構造の再ロード

ページソース構造の再ロード

アイコン

説明

[ページソース] ペインで定義されたページソースデータ構造を再ロードします。このコマンドを使用してページのソースデータをページのデザイン中に更新することができます。

ワークフローのシミュレーション

ワークフローのシミュレーション

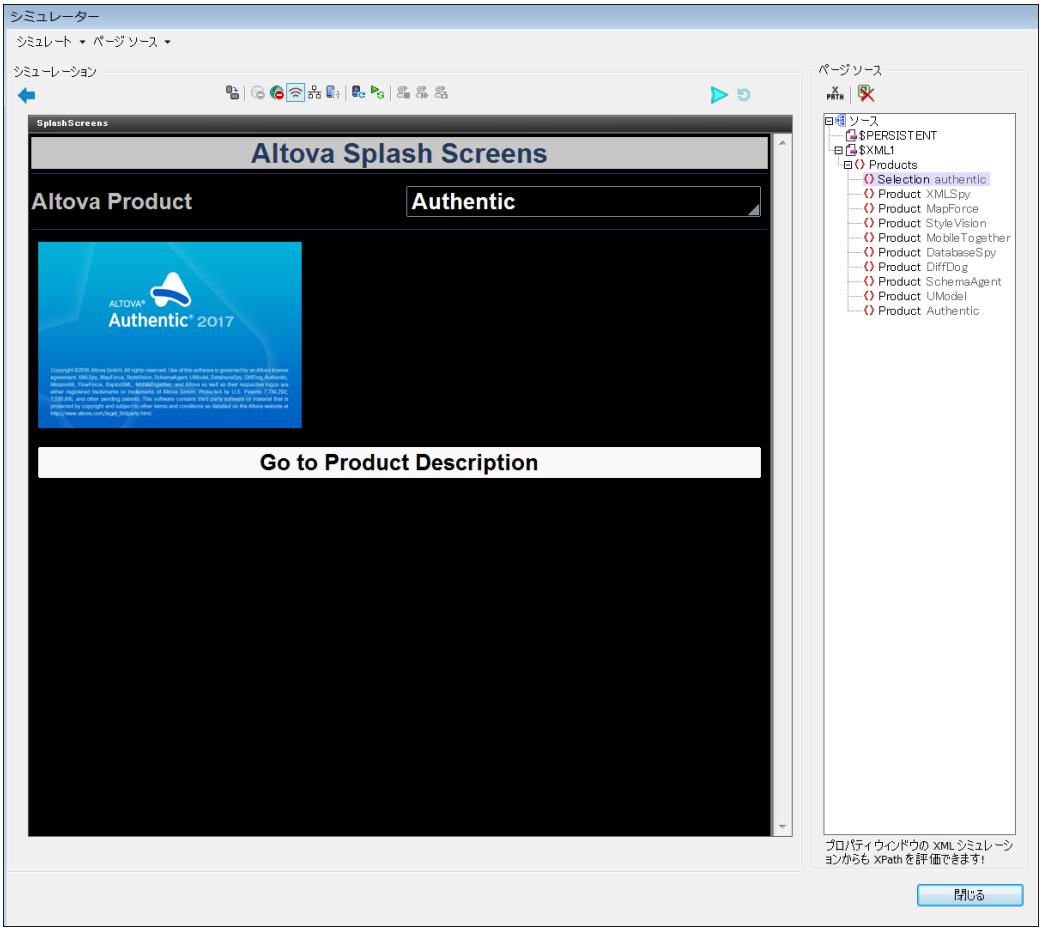
アイコン

ショートカット

F5

説明

テストのために別のウィンドウで、(MobileTogether Designer) のローカルシミュレーターを開始します (*下のスクリーンショット参照*)。シミュレーターは現在アクティブなプロジェクトのワークフローをステップバイステップで実行します。ページ設定ツールバーのデバイスのセレクター で選択されているモバイルクライアントでシミュレーションが実行されます。



クライアントでの試行

クライアント上のトライアルラン

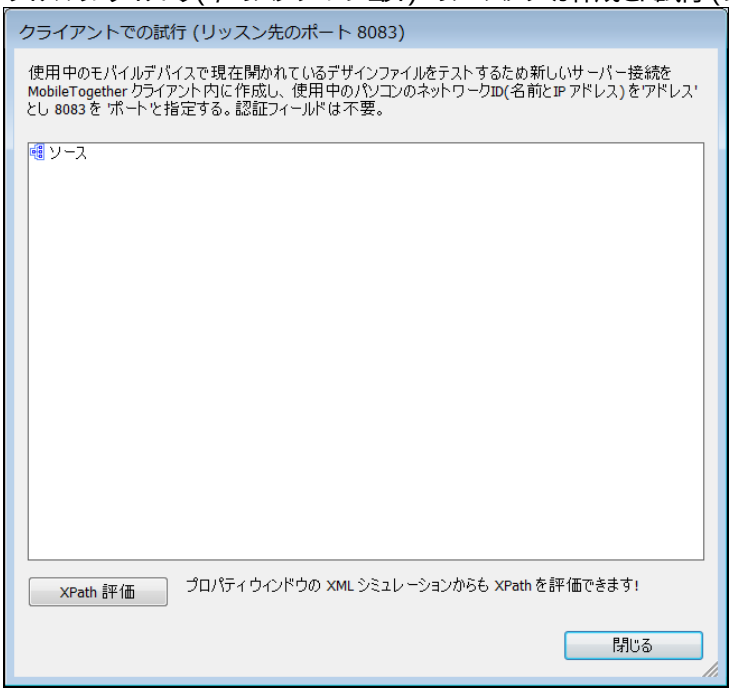
アイコン

ショートカット

Shift+F5

説明

指定されたクライアントでアクティブな MobileTogether デザインファイルをテストします。MobileTogether Designer は MobileTogether Server として機能し、クライアントに直接デザインと関連するデータファイルを処理します。モバイルデバイスの MobileTogether Client アプリは、MobileTogether Designer を実行中のローカル PC にサーバー接続をセットアップしなければなりません。デフォルトで、**8083** がクライアントが接続する必要のあるローカルの PC ポートです。[オプション] ダイアログの クライアントタブでの試行 でこのポートを変更することができます。クライアントと PC の接続が確立され、クライアントでデザインが選択されると、クライアント上のトライアルランダイアログ (下のスクリーンショット) のソースツリーが作成され試行 (シミュレーション) が開始されます。



ワークフローのシミュレーションにサーバーを使用

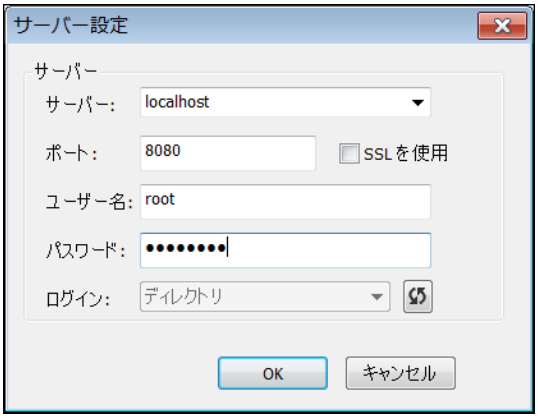
- ワークフローのシミュレーションにサーバーを使用
  - *アイコン*

  - *ショートカット*

    Ctrl+F5

  - *説明*

    [サーバーの設定] ダイアログを表示します (*下のスクリーンショット*)。シミュレーションを実行する MobileTogether Server の接続および認証情報を入力します。**[OK]** をクリックすると、別のウィンドウでシミュレーションが開始します。



新しいテストケースの記録

- 新しいテストケースの記録
  - *アイコン*

  - *説明*

    シミュレーター内で新しいテストケースを開始し、ユーザーアクションを記録します。記録が停止すると、記録に名前を付けて、テストケースとして保存するように問われます。記録オプションは*テストケースとテストランの管理* ダイアログ内で指定することができます。 *詳細に関しては、テストケースの記録を参照してください。*.

テストケースの再生

- テストケースの再生
  - *アイコン*

  - *説明*

    再生のために使用することのできるテストケースコンボボックス内で選択することのできるテストケースを再生します。再生がテストケースから差異を返すと、再生は保存されます。詳細に関しては、*テストケースの再生* を参照してください。*テストケースとテストランの管理* ダイアログ内で再生のためのオプションを指定できることに注意してください。

クライアント上でのテストケースの トライアル ラン

- クライアント上での トライアル テストケース
  - *アイコン*

  - *ショートカット*

    Alt+F5

  - *説明*

    接続されているクライアント上で、「再生のために使用することのできるテストケース」コンボボックス内で選択されているテストケースを再生します。再生が差異を返すと、再生は保存されます。 *詳細に関しては、テストケースの再生を参照してください。*

テストケースとテストランの管理

- テストケースとテストランの管理
  - *アイコン*

  - *説明*

テストケースとテストランの管理 ダイアログを表示します。

テストケースとテストランの管理 ダイアログ (下のスクリーンショット) で以下をおこなうことができます:

- *テストケースのために記録オプションを設定する。*
- *次のテストランのために記録と再生オプションをセットアップする。*
- MobileTogether 記録ファイル(`.mtrecord` ファイル)をロードして保存する。
- *テストランを削除し比較する。*
- *テストケースをテストランの内の一件と代用する。テストランはテストケースの役割を果たします。他のテストランは削除され、選択されたテストランがその(空の) グループの新しいテストケースとなります。*
- MobileTogether Server にテストケースをデプロイし、サーバーからテストランを取得し、テストケースまたはテストランをサーバーから削除します。

詳細に関しては、テストケースとテストランの管理を参照してください。.

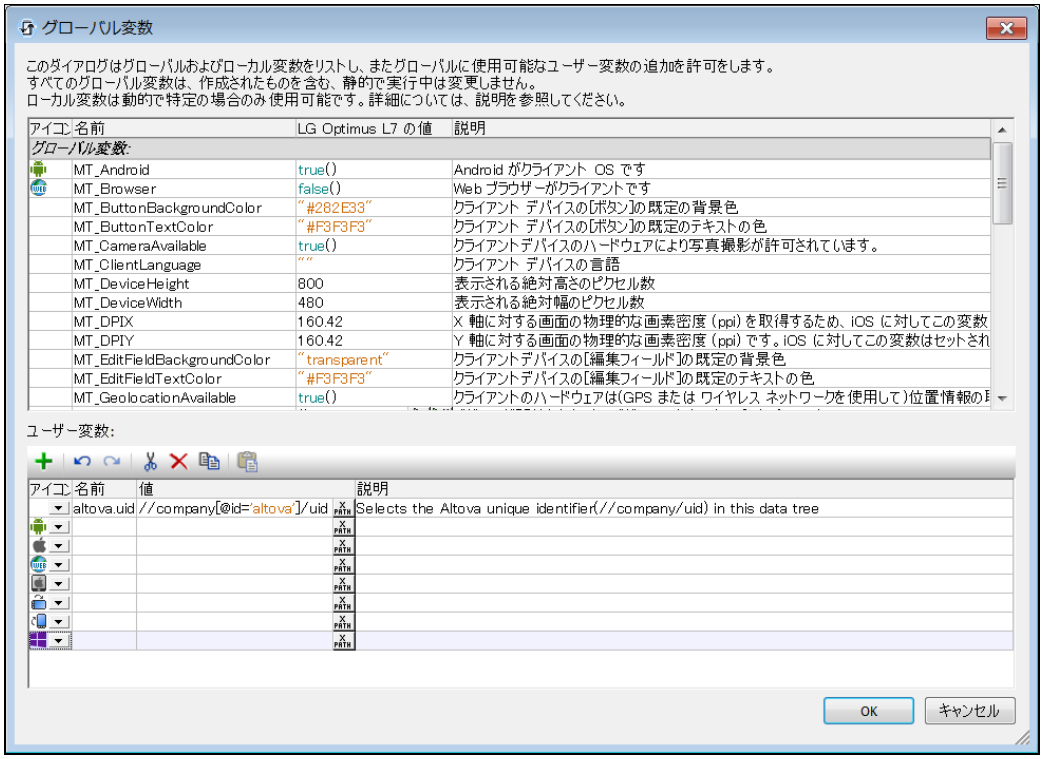
グローバル変数

- グローバル変数
  - *アイコン*

  - *説明*

[グローバル変数] ダイアログを開く(*下のスクリーンショット*)。XPath または XQuery などのデザイナーのプログラミング コンテキストでプロジェクト内でグローバル変数を使用することができます。MobileTogether Designer には、ダイアログの上のペインにリストされるグローバル変数の標準ライブラリがあります。グローバル変数の値は MobileTogether により、シミュレーション中およびクライアントデバイスでアプリが実行される際に割り当てられています。グローバル変数には 3 つの種類があり、ダイアログの変数のリストはこれらの3つの種類別に分類されます:

- 静的な値変数 (*グローバル変数* と呼ばれています): これらの変数はモバイルデバイスの情報を含みます。これらの変数の値はプロジェクトの実行中は変更されることはありません。ペイン上の値列は、現在選択されているモバイルデバイスを指定しています。リストされている値は、すべてその特定のデバイスで使用するためです。シミュレーションでのクライアントデバイスは[デバイスの選択] コンボボックスにより選択されたデバイスです。例えば、変数 **`$MT_Android`** は、使用されるモバイルデバイスが Android の場合、**`true`** の値を持ちます。(デバイスの情報は通常のモバイル通信の手順の一部としてデバイスから送信されます。)
- 動的な値変数 (*ローカル変数* と呼ばれています): これらの変数は、デザインページとそのコントロールに関する変数の情報を含みます。これらの値は実行中に変更することができます。例えば、**`$MT_ControlNode`** 変数は、プロジェクトの実行中のある時点の現在のコントロールのページソースリングがノードであるものにより、異なる値を持ちます。
- *ユーザー変数*: グローバル変数の標準ライブラリに加え、(ダイアログでは*ユーザー変数*と呼ばれてる)自身のグローバル変数を追加することができます。変数には任意の値を与えることができ、この値は、プロジェクトの XPath/XQuery 式で使用することができます。



変数を追加するには、下のペインで以下を行います:

1. (ペインのツールバーにある) **[追加]** または **[挿入]** アイコンをクリックしてリストに行を追加します。
2. 新しい変数の名前を(名前 列に)入力し、変数に説明を与えます(説明列)。*上のスクリーンショット参照。*
3. 値フィールドをクリックして、式 XPath/XQuery ダイアログを表示し、この変数の値を決める XPath 式を入力します。
4. 特定のグループに属する新しい変数を認識するためのアイコンを選択します。
5. **[OK]** をクリックして終了します。変数はグローバル変数として追加され、プログラミング コンテキストで使用することができます。

定義済みグローバル変数の説明はグローバル変数のセクションを参照してください。

すべてのグローバル変数の使用方法のリスト

- すべてのグローバル変数の使用方法のリスト

説明

[メッセージ] ペインにすべてのグローバル変数のリストを返します。各グローバル変数は、変数が使用されている箇所の情報と共にリストされています。このリストは、使用方法を含む定義に直接移動するリンクが含まれており、素早く定義を検索し編集することができます。

すべてのページソース変数の使用方法のリスト

- すべてのページソース変数の使用方法のリスト

説明

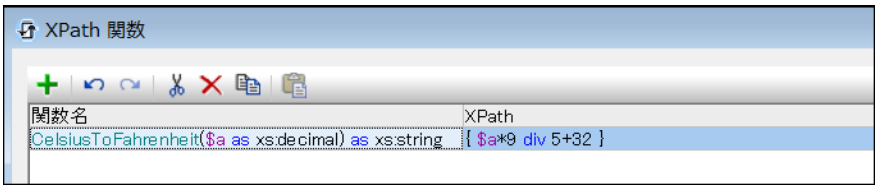
[メッセージ] ペインにすべてのページソース変数のリストを返します。リストの各ページソース変数は、ページソースが使用されている定義へのリンクを含んでおり、その定義を素早く検索および編集することができます。ページソースリンクをクリックすることで、[ページソース] ペイン内のページソースをハイライトします。

XPath/ XQuery 関数

- XPath/ XQuery 関数
  - *アイコン*

  - *説明*

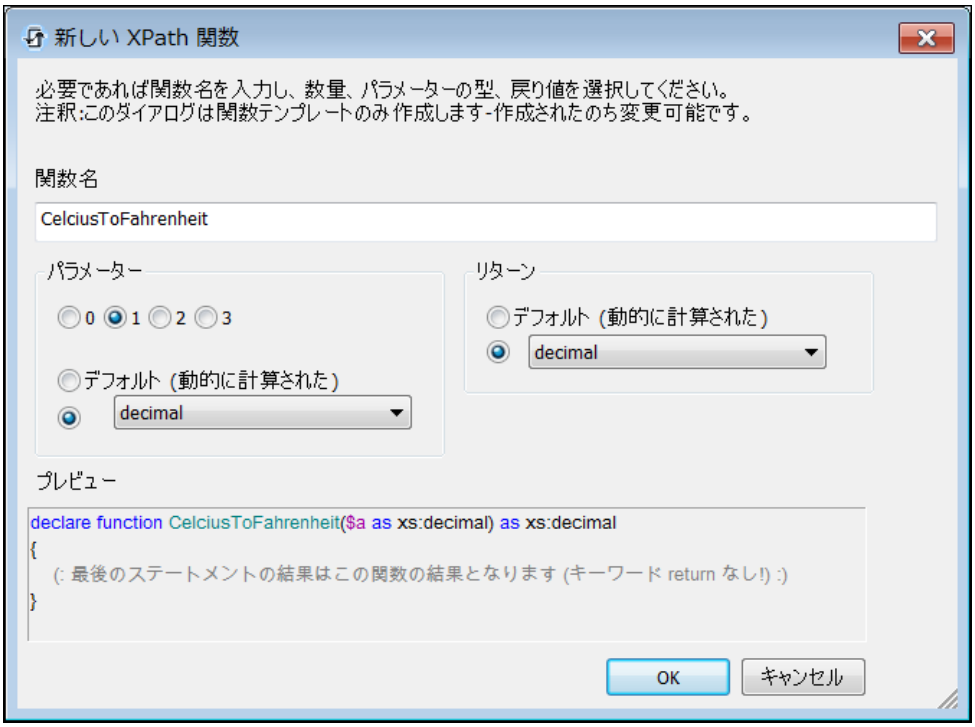
プロジェクト内のすべてのユーザー定義関数をリストする[XPath 関数] ダイアログを開きます。これらのXPath 関数はプロジェクト内のすべてのXPath 関数で使用されることができます。ダイアログのツールバー内の対応するアイコンを使用して関数を追加および削除することができます。関数の定義を編集するには、関数の**[XPath 式の編集]** ボタンをクリックしてください。



ユーザにより定義された新しい XPath 関数の追加

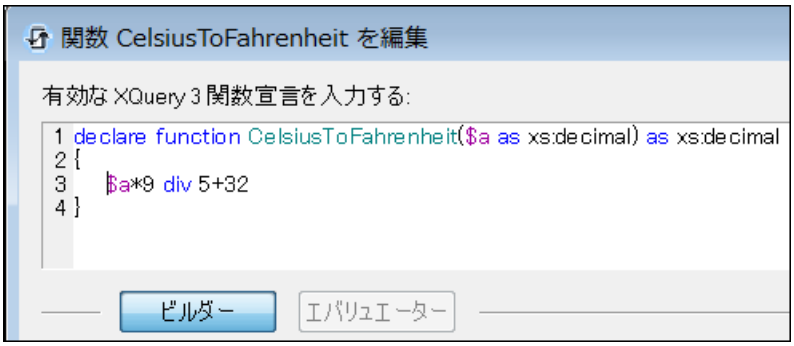
ユーザーにより定義された新しい関数を追加するには次の2 つのステップを踏みます: (i) 関数の宣言と(ii) 関数の定義。

新しい関数を追加するには以下を行います、XPath 関数ダイアログ(*上のスクリーンショット参照*) のツールバーから**[追加]** をクリックします。これにより新しいXPath 関数ダイアログ(*下のスクリーンショット*) が表示されます。



このダイアログで、関数の名前の宣言、関数パラメーター(引数) の数と種類の指定、関数の戻り値の種類を指

定することができます。上のスクリーンショットで摂氏から華氏へ変換する関数の小数点の宣言をおこなっています。関数は摂氏の値を小数とする1 つのパラメーターをもちます。これは、小数、華氏の気温、を出力します。関数の働きに関しては、次のステップで説明されています。関数 (*上のスクリーンショット*) を宣言した後、**[OK]** をクリックします。関数を定義することのできる、新しく宣言された関数のテンプレートを含む関数の編集ダイアログ (*下のスクリーンショット*) が表示されます。



波かっこに関数の定義を入力します。上のスクリーンショットに表示された定義では、**$a** が入力パラメーターです。 **[OK]** をクリックして終了します。関数はXPath 関数ダイアログのユーザー定義関数のリストに追加され、プロジェクトのすべてのXPath 式で使用することができます。

メモ ユーザー定義 XPath 関数は個別の名前空間にある必要はありません。結果、ユーザー定義関数を定義または呼び出す際に名前空間プレフィックスは必要ありません。 XPath デフォルト名前空間 は、拡張関数およびユーザー定義関数を含む、すべてのXPath/XQuery 関数に使用されています。内蔵された関数を含むあいまいさを回避するために、ユーザー定義関数を大文字にすることが奨励されます。

すべてのユーザー定義 XPath/ XQuery 関数のリスト

- すべてのユーザー定義 XPath/ XQuery 関数のリスト
  - *説明*

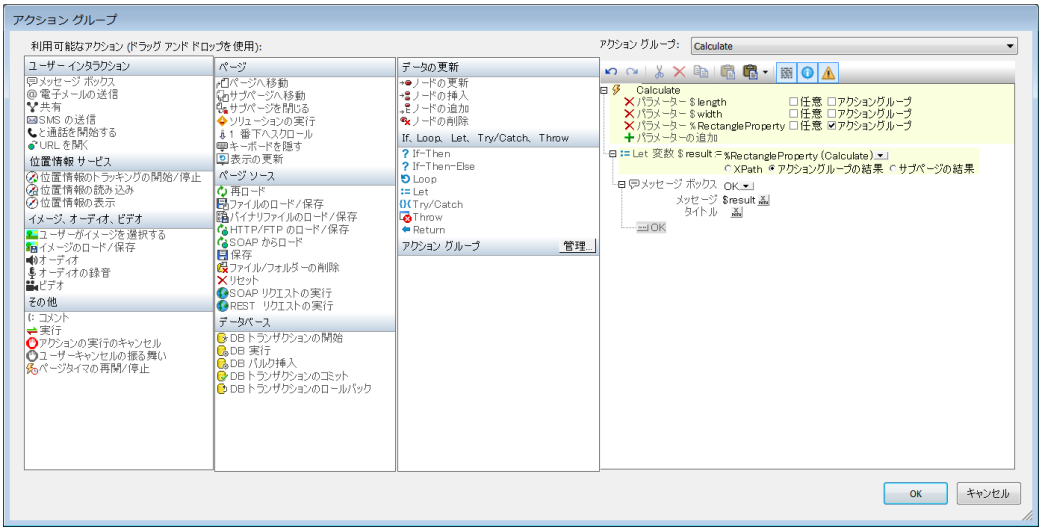
    [メッセージ] ペイン内のプロジェクト内のページで使用されたすべてのユーザー定義 XPath/XQuery 関数のリストを発生するページと共に返します。

アクション グループ

- アクション グループ
  - *アイコン*

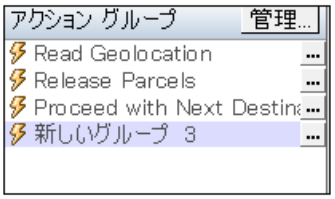
  - *説明*

アクショングループを作成し管理することのできる [アクショングループ] ダイアログを表示します (下のスクリーンショット)。

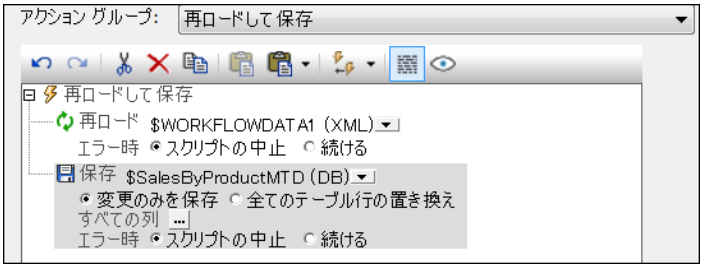


アクショングループを作成するには、以下を行います:

1. [アクショングループ] ペインの**[管理]** をクリックします (*下のスクリーンショット参照*)。



2. [アクショングループの管理] ダイアログが表示されます。ダイアログのツールバーの**[追加]** をクリックします。デフォルトの名前を持つ アクショングループがこのダイアログに追加されます。
3. アクショングループの名前を変更して、**[OK]** をクリックします。[アクショングループの管理] ダイアログが閉じられ、[アクション グループ] ペインのグループのリストに新しいアクション グループが追加されます (*上のスクリーンショット参照*)。
4. [アクション グループ] ペイン(*上のスクリーンショット*) から、編集するアクショングループの**[編集]** アイコンをクリックします。グループのコンテンツはペインの右に表示されます (*下のスクリーンショット参照*)。アクショングループの詳細はアクショングループのコンボボックスのグループを選択することでも表示できます。



5. 右側のペインにアクティブなアクショングループを作成するには、アクショングループダイアログから使用可能なアクションをドラッグ アンド ドロップします。
6. **[OK]** をクリックして終了します。アクショングループはページアクションで使用することができます。

メモ このダイアログの最後のセクションは保存されます。結果、最後のセクションがハイライトされダイアログは再開されます。

すべてのアクション グループの使用法のリスト

- すべてのアクション グループの使用法のリスト
  - *説明*

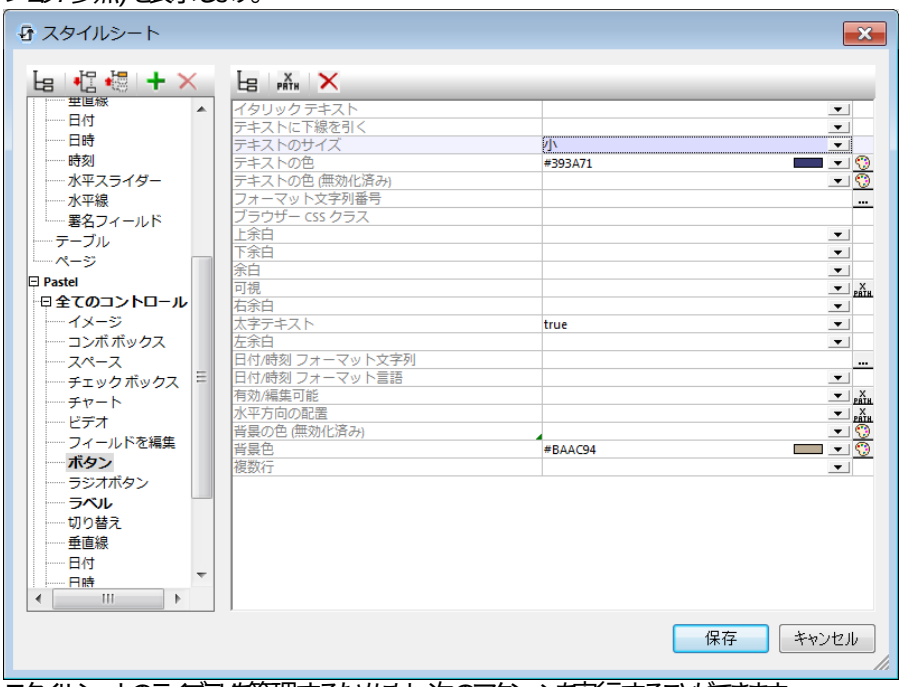
    プロジェクトのページで使用される[メッセージ] ペイン内の全てのアクショングループリストを返します。リストはアクショングループにてオーダーされます。各アクショングループは、直接および間接使用法別に細分類されています。アクショングループを使用するページは、アクショングループのイベントトリガーと共にリストされています。この機能はユーザー定義コンポーネントをレビューして、プロジェクトを整理するときに役に立ちます。リストのリンクをクリックすることで直接それぞれの定義に移動することができます。

スタイルシート

スタイルシート

説明

異なるスタイルシート内でスタイルシートを管理し、スタイルを定義するスタイルシートダイアログ(*下のスクリーンショット参照*)を表示します。



スタイルシートのライブラリを管理するためには、次のアクションを実行することができます:

- ユーザーにより作成されたスタイルシートを追加: **「スタイルシートの追加」** をクリックします。
- ユーザーにより作成されたスタイルシートの名前を変更: 名前をダブルクリックして編集します。
- ユーザーにより作成されたスタイルシートを削除: **「スタイルシートの削除」** をクリックします。

メモ プロジェクトという名前のプロジェクトスタイルシート(*上のスクリーンショット参照*)はデフォルトで存在します。このスタイルシートの名前を変更し、削除することはできません。プロジェクトスタイルシートとユーザーにより作成されたスタイルシートに関する詳細は、*スタイルシートの型とスコープ*を参照してください。

左側のペイン内で、コントロール、テーブル、ページを選択します。コンポーネントのために使用することのできるスタイルは右側のペイン内に表示されています。スタイルの値を設定するには、希望する値を入力、または、スタイルのコンボボックスオプションから値を選択、またはスタイルの値を評価するXPath 式を入力します。詳細に関しては、*スタイルシート プロパティ*を参照してください。定義済みの値を持つスタイルに関しては、右側のペインの **「空白ではないアイテムを表示する」** アイコンをクリックしてください。

完了すると、**「保存」**をクリックします。

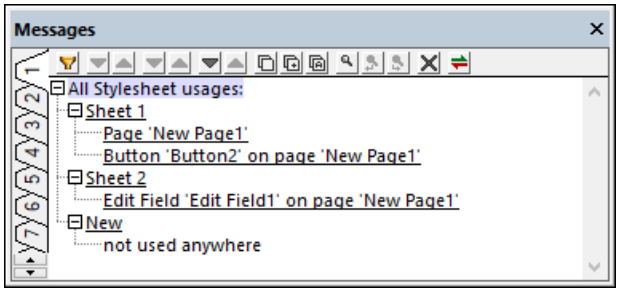
メモ 空ではないアイテムの表示を有効化/無効化するアイコンが各ペインに存在します(すなわち、(少なくとも1つの)値が定義されているアイテム)。空ではないアイテムのみを表示する機能は定義されているスタイルのリストを確認する際に役に立ちます。例えば、現在定義されているスタイルの概要を確認する場合。左側のペインには以下を行うツールバーアイコンが存在します: (i) 全てのアイテムの展開。(ii) 全てのアイテムの折りたたみ。

スタイルシートの作動の概要については、*スタイルシート*を参照してください。

スタイルシートのすべての使用方法をリストする

- スタイルシートのすべての使用方法をリストする
  - *説明*

    ユーザーにより作成されたスタイルシートをページインスタンス、テーブルインスタンス、およびコントロールインスタンスに適用することができます。このコマンドは、メッセージペイン内にプロジェクト内すべてのユーザーにより作成されたスタイルシートリストと使用箇所を表示します (*下のスクリーンショットを参照してください*)。



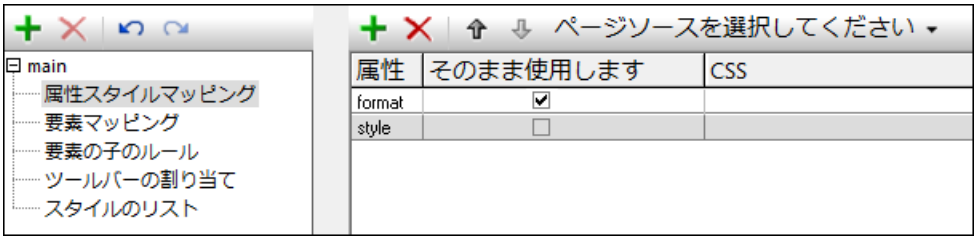
    スタイルシートによりリストは命令することができます。各スタイルシートの下に、スタイルを使用する全てのページ、テーブル及びコントロールインスタンスがリストされます。これらのアイテムをクリックすると対応するページ、テーブル、または、コントロールに移動することができます。

    スタイルシートが使用されていない場合、メッセージペインに報告されます(*上のスクリーンショットを参照*)。使用されていないスタイルシートは**プロジェクト** メニューの **使用されていない関数、ユーザー変数、スタイルシート、アクショングループをリストする** コマンドを使用して報告されます。

リッチテキストスタイルシート

- リッチテキストスタイルシート
  - *説明*

    以下を定義することのできるリッチテキストスタイルシート ダイアログを開きます (*下のスクリーンショット*):
    - ソリューションの表示内のテキストのためのスタイルルール
    - ページソースからソリューションの表示へのスタイルのマッピング



リッチテキストスタイルシート ダイアログの詳細に関しては、リッチテキストのセクションを参照してください。

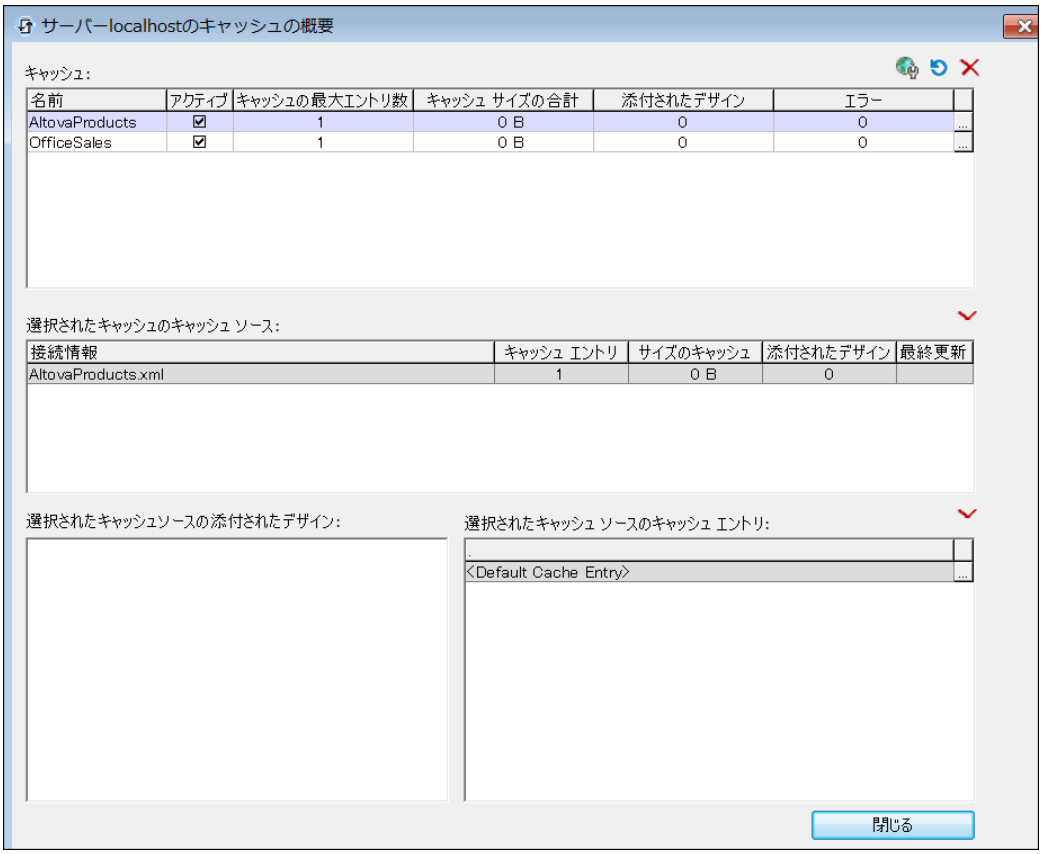
キャッシュの概要

- キャッシュの概要
  - *アイコン*

  - *説明*

MobileTogether Server およびサーバーキャッシュの概要への接続を有効化します: キャッシュ ヒット数、キャッシュのサイズ、消去など。

キャッシュの概要ダイアログ (*下のスクリーンショット*) は、メニューコマンド **[プロジェクト | キャッシュの概要]** からアクセスすることができます。



ダイアログはサーバー上の全てのキャッシュの概要を表示します。また、次を実行することができます:

- キャッシュの有効化 無効化
- キャッシュの削除

キャッシュに関する詳細は キャッシュ のセクションを参照してください。

ローカライズ

- ローカライズ
  - *全般的な説明*

(例えば、ボタンのテキスト) などの異なる **コントロール** または(例えば、コンボボックスの値をリストするドロップダウンなど)他の方法で**コントロール**に関連する文字列をローカライズ(翻訳) することができます。で表示されるローカリゼーション ダイアログを表示します(*下のスクリーンショット*)。コントロールに関連する文字列の他に、**カスタム文字列** をローカライズすることができます。そして、デザインのすべての箇所に`mt-load-string` 関数を使用して、挿入することができます。

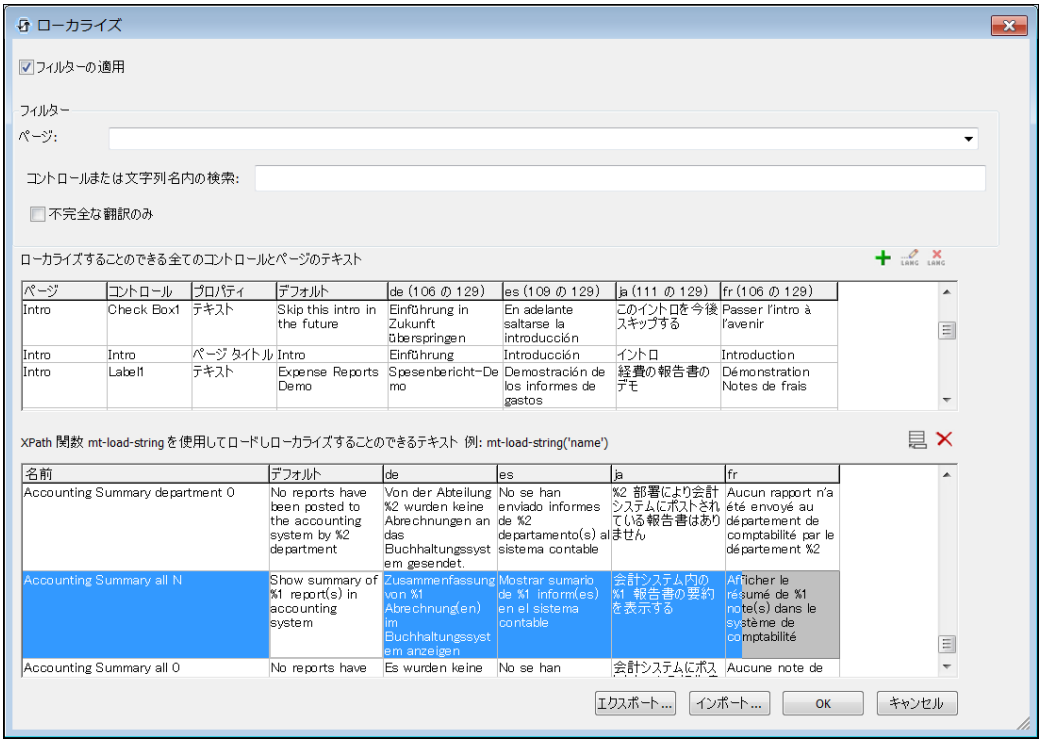
- **コントロールに関連した文字列** は、上のペインに表示されます (*下のスクリーンショットを参照*)。このペインの列は、左から右の順序です: (i) コントロールが表示されるページ; (ii) コントロールの名前; (iii) テキスト文字列を含むコントロール プロパティ; (iv) デフォルトの言語のテキスト文字列; ローカライズされた言語のテキスト内のテキスト文字列を含む追加の列、各言語につき一列ずつ。
- **カスタム文字列** は、のペインに表示されます。ペインの列は、それぞれ左から右の順序です: (i) カスタム文字列の名前; (ii) ローカライズされた言語のテキスト内のテキストカスタム文字列を含む追加の列、各言語につき一列ずつ。

言語の追加 アイコンを使用して、列をローカリゼーション言語に追加します。追加する列数に制限はありません。特定の言語内の コントロールに関連した文字列 または カスタム文字列をローカライズするには、その言語の列に翻訳を入力します。翻訳を保存するには、**[OK]** をクリックします。

ローカリゼーション ダイアログ内で、ローカライズされた(コントロールに関連した)全ての文字列は、ローカライズされたソリューションのバージョンで使用されます。モバイル デバイスの言語設定で、`language-country` バリアント(例えば、**`es-US`** または**`fr-CH`**) が指定されます。下に表示される順序に従い、表示される文字列の言語が選択されます。

1. ソリューションの対応するローカライズされた**`language-country`** (**`es-US`** または**`fr-CH`**) 文字列は存在する場所で使用されます。
2. 文字列のための、`language-country` ローカリゼーション(**`es-US`** または**`fr-CH`**) が存在しない場合、ローカライズされた**`language`** (**`es`** または**`fr`**) 文字列 が存在する場合は使用されます。
3. 文字列のための、`language-country` ローカリゼーション(**`es-US`** または**`fr-CH`**) または `language` ローカリゼーション(**`es`** または**`fr`**) が存在しない場合、デフォルトの言語 が使用されます。

ローカライズされた文字列が定義されている言語でシミュレーションを確認するには、シミュレーション言語 **[プロジェクト | シミュレーション言語]** コマンドを使用して、シミュレーションする言語を設定して、シミュレーションの実行を行います。



- *フィルターの適用*

次のフィルターがローカリゼーションダイアログで使用することができ、組み合わせも行うことができます:

- **ページ**: そのページのみの文字列を確認するためにドロップダウンリストからページを選択します。すべてのページを選択するには、オプションを空にするか、または空のエントリを選択します。これにより、コントロールに関連した文字列 (上のペイン) のみにフィルターを適用します。
- **コントロールに関連した文字列 および カスタム文字列**: 検索するテキストを入力します。すべてのコントロールの名前 (2番目の列の上のペイン) および検索テキストを含むカスタム文字列名 (下のペインの最初の列) が表示されます。これは、両方のペインへ適用されるフィルターです。
- **不完全な翻訳 のみ**: このオプションをクリックして、不完全な翻訳のみを表示します。不完全な翻訳とは、1つ以上のローカリゼーションが不足している文字列をさします。このフィルターは、両方のペインに表示されます。このオプションが選択されている間に翻訳が入力されるとフィルターをアップデートするために表示を更新する必要があるかもしれません。オプションの選択を解除、再び選択することで更新を行うことができます。

- *言語列の追加、名前変更、および削除*

これらのアイコンは、右側の上のペインの上にあります。

+	**言語の追加**	言語名を入力または選択することのできる言語の追加 ダイアログです。**[OK]** をクリックすると、その言語名を持つ列が(上のおよび下の)両方のペインに追加されます(*関連した* `mt-available-languages` *XPath 関数も参照してください*)。
LANG	**言語の名前変更**	名前変更する言語列内(ペイン内の行)にカーソルをポイントすると、**[言語の名前変更]** をクリックします。言語の変更 ダイアログが、編集フィールド内で選択済みの言語と共に表示されます。言語名を変更して、**[OK]** をクリック

		します。両方のペイン内の変更される列名をクリックします。
	言語の削除	削除する言語列内(ペイン内の行)にカーソルをポイントすると、**[言語の削除]** をクリックします。列を削除するかを問われると、**[はい]** をクリックして、列およびエンティティを削除します。**[いいえ]** をクリックするとキャンセルされます。

カスタム文字列 (下のペイン)

カスタム文字列は、デザイン内で使用されるすべてのテキストであることができます。ペインの右上にあるアイコンを使用して下のペインで管理されます。

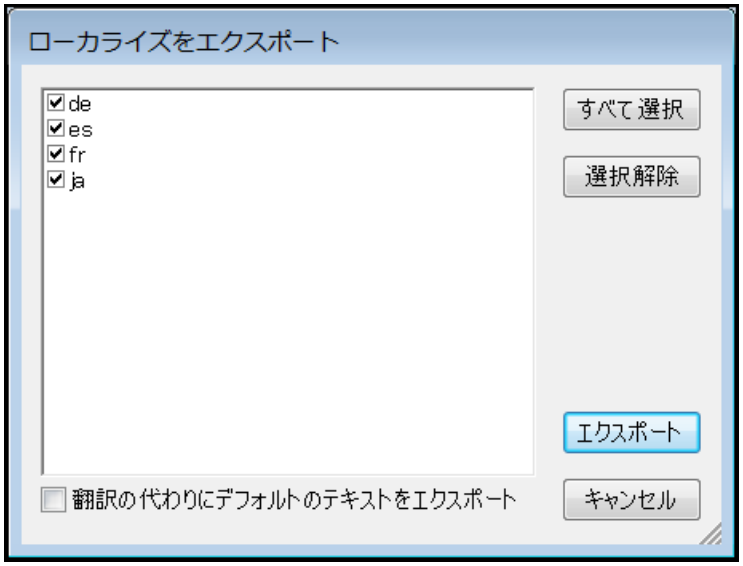
	ローカライズされた文字列の追加	カスタム文字列に行を追加します。新規に追加されたカスタム文字列は、編集することのできる、デフォルトの名前で作成されます。名前をクリックして編集します。 XPath 式からのカスタム文字列を参照するために名前は使用されます。下を参照してください。
	行の複製	下のペインで選択されているカスタム文字列(名前とローカライズされた内容)を複製します。これにより、既存の文字列に類似した文字列を追加する場合、時間を節約することができます。2つの文字列の名前が同じの場合、両方の名前が赤で表示されます。複製されて文字列を必要に応じて編集します。
	ローカライズされた文字列の削除	現在選択されているカスタム文字列を削除します。

デザイン内でカスタム文字列を使用するには、XPath 式内の`mt-load-string` 関数を使用します。例: `mt-load-string('NameOfString')`。

ローカリゼーションのエクスポートおよびインポート

ローカライズされた文字列は、XML ファイルにエクスポートすることができます。1つまたは複数の言語を単一のファイルにエクスポートすることができます (*下のスクリーンショットを参照*)。利点は、翻訳者が個別の言語ファイルで作業できることです。翻訳された、 XML ファイル内の文字列は、プロジェクトにインポートすることができ、ローカリゼーションダイアログ内の対応するローカリゼーション言語に入力されます。

ローカリゼーションダイアログの **[エクスポート]** をクリックすると、(このトピックの最初のスクリーンショット参照)、ローカリゼーションダイアログのエクスポートが表示されます (*下のスクリーンショット*) 。ダイアログに表示される言語は、プロジェクトのローカリゼーション言語として設定された言語です。1つまたは複数の言語にエクスポートを選択することができます。



*翻訳の代わりに デフォルト テキストをエクスポートする オプション*が選択されている場合、エクスポートされた ファイルは、デフォルトの言語テキストを、既に完了している翻訳の代りにすべてのローカリゼーション言語 属性 の値として、デフォルト 言語 テキストを含みます。(*下のXML ファイル リスト参照*)。プロジェクトは、しかし、翻訳を保管します。デフォルト 言語 テキストをエクスポートするには、翻訳メモリと作業することのできる翻訳ツールが役に立ちます。 これは、XML ファイルが翻訳 ツールにインポートされると、翻訳 メモリが自動的にすべてのインポートされた文字列を、メモリ内にあるものとして翻訳するからです。ですから、新しくインポートされた デフォルト テキスト 文字列が、以前に翻訳されていると、再度翻訳メモリを使用して、自動的に翻訳されるのです。 残りの語句は、手動で翻訳されます。プロジェクトに翻訳されたXML ファイルがインポートされると、ローカリゼーションダイアログの対応するローカリゼーション言語列に入力されます。

[ローカリゼーションのエクスポート] ダイアログ内の**[エクスポート]**をクリックすると、名前およびXML ファイルの場所を指定することのできる保存ダイアログが表示されます。

単一言語のXML リストの エクスポート

```
<Localizations version="1">
  <Controls>
    <Control default="SelectEmployee" property="PageTitle" name="SelectEmployee"
id="2">
      <Languages de="Mitarbeiter auswählen"/>
    </Control>
    ...
  </Controls>
  <Strings>
    <String default="Admin" name="Role-A Name">
      <Languages de="Administration"/>
    </String>
    ...
  </Strings>
</Localizations>
```

複数言語の XML リストのエクスポート

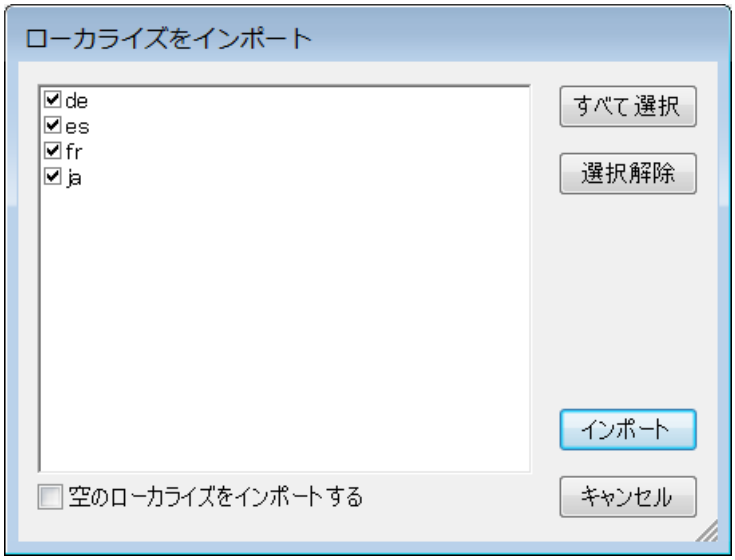
```
<Localizations version="1">
  <Controls>
    <Control default="SelectEmployee" property="PageTitle" name="SelectEmployee"
```

```
id="2">
    <Languages de="Mitarbeiter auswählen" es="Seleccionar empleado" fr="""/>
  </Control>
  ...
 </Controls>
 <Strings>
  <String default="Admin" name="Role-A Name">
    <Languages de="Admin" es="Administración" fr="""/>
  </String>
  ...
 </Strings>
</Localizations>
```

メモ ((翻訳の代わりに)エクスポート デフォルト言語 テキストのエクスポートが有効化されていない場合)ローカリゼーション言語に翻訳されていない文字列は、空の文字列 としてエクスポートされます。

翻訳された XML ファイルのインポート

翻訳されたXML ファイルをインポートするには、ローカリゼーションダイアログ内の**[インポート]** をクリックします (このトピックの最初のスクリーンショットを参照してください)。これは、インポートするXML ファイルを選択することのできる、[開く] ダイアログを表示します。**[開く]** をクリックすると、ローカリゼーションのインポートダイアログが表示されます (下のスクリーンショット)。ローカリゼーション内の対応する列をもつXML ファイル内で見つけることのできる言語が表示されている言語です。ダイアログXML ファイル内にローカリゼーション言語が存在し、対応するローカリゼーション言語名がプロジェクト内に存在しない場合、その言語は、表示されません。



インポートする言語を選択します。空のローカリゼーションオプションを選択した場合、XML ファイル内の空のローカリゼーション文字列がすべての既存のローカリゼーション文字列を空でない場合でも上書きします。**[インポート]** をクリックして終了します。

翻訳された言語 文字列 は、プロジェクトにインポートされ、ローカリゼーションダイアログの対応するローカリゼーション言語 列に入力されます。インポートされたXML ファイルの構造 および コンテンツは、MobileTogether Designer が正確に処理できるXML ファイルである必要があります。ローカリゼーション言語 属性以外の他の属性の値を変更しないことが重要です。

メモ XML ファイル内では、翻訳されている文字列は、デフォルトの言語文字列と同じの場合、翻訳された

文字列は対応するエントリにインポート**されず**、ローカライズ言語のカラムは空のままです。ソリューションが実行されると、ローカライズ言語エントリが存在しないため、デフォルトの言語の文字列が使用されます。このように、言語間の文字列の重複は回避されます。

シミュレーションする言語

- シミュレーションする言語
  - *説明*

    MobileTogether Designer がローカライズされている場合、このコマンドによりシミュレーターで使用される言語を選択することができます。コマンドにはメニューアイテムとしてローカライズする言語を含むサブメニューがあります。

    プロジェクトが 1 つ以上の言語にローカライズされている場合、これらの言語は **[シミュレーションする言語]** コマンドのサブメニューで使用することができます。このサブメニューで、プロジェクトのシミュレーションで使用される言語を選択することができます。シミュレーションする言語が変更されるまで、選択された言語のローカライズされた文字列がプロジェクトのシミュレーションで使用されます。

すべての外部デー参照のリスト

- すべての外部デー参照のリスト
  - *説明*

    [メッセージ] ペインにプロジェクト内で参照されているすべての外部データソースのリストを返します。これらは、HTTP、REST、およびSOAPによりアクセスされたデータソースを含みます。リスト内のリンクをクリックすることにより、選択されたリソースを参照するデザイン定義に直接移動することができます。

使用されていない関数、ユーザー変数およびアクション グループのリスト

- 使用されていない関数、ユーザー変数およびアクション グループのリスト

説明

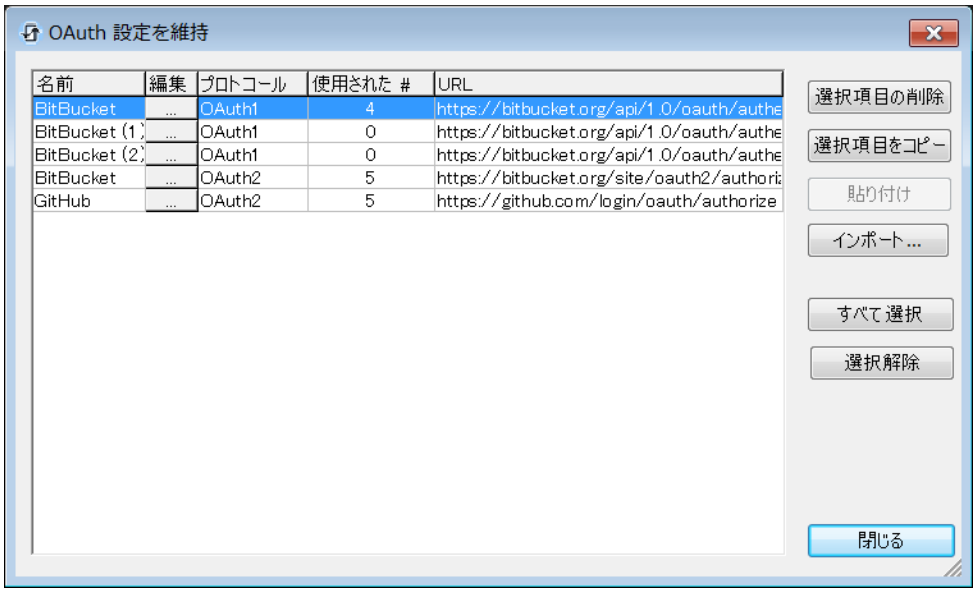
[メッセージ] ペインにすべての**使用されていない** XPath/XQuery 関数、ユーザー関数、スタイルシート、および アクショングループ のリストを返します。これは、ユーザー定義されたコンポーネントをレビューし、プロジェクトを整理する時に役に立ちます。リストのリンクをクリックすると、対応する定義に直接移動することができます。

OAuth 設定の維持

- OAuth 設定の維持
  - *説明*

MobileTogether Designer 内で作成されるREST リクエストはOAuth を使用して認証することができます。手順については、REST リクエスト設定のセクションを参照してください。

MobileTogether Designer プロジェクト内で複数のOAuth 設定定義を作成することができます。これらの定義はプールに保管され、認証 REST リクエストのためのプールから定義をドキュメント内で使用することができます。[OAuth 設定の維持] ダイアログ (*下のスクリーンショット*) のダイアログによりアクティブなプロジェクトの OAuth 定義 を管理することができます。ダイアログは、アクティブなプロジェクトの定義のプール内のOAuth 設定定義を表示します。プールから定義を削除、他のMobileTogether Designer プロジェクトから定義をインポート、クリップボードから定義をコピー、クリップボードに定義をコピーすることができます。



ダイアログには、以下の列があります:

- *名前*: 作成時に設定定義に割り当てられた名前。編集することはできません。
- *編集*: 選択された定義の設定を編集する[OAuth 設定] ダイアログを開くボタン。
- *プロトコール*: OAuth1 またはOAuth2 を使用するかを決める定義
- *使用された*: 現在のプロジェクト (デザイン) 内での定義の使用回数。
- *URL*: 定義のエンドポイントのURL の一番長い共通部分

このダイアログ内で次のアクションを実行することができますダイアログ:

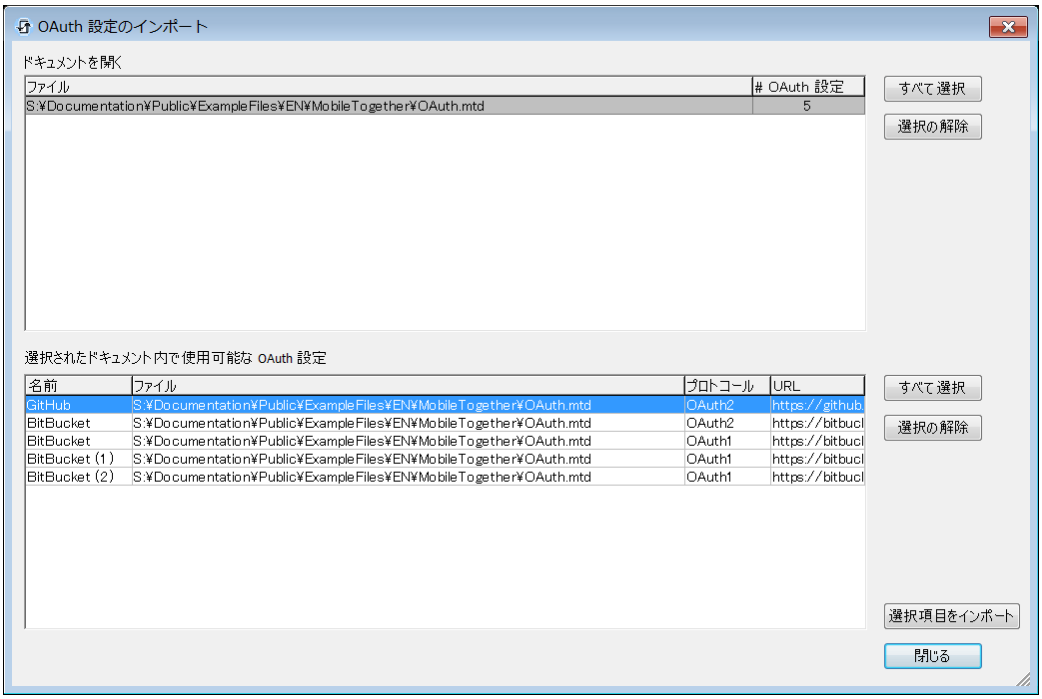
- *選択項目の削除*: 1つ以上の定義を定義のために選択することができます。
- *選択項目のコピー*: クリップボードに1つ以上の定義をグループとしてコピーすることができます。
- *貼り付け*: このボタンは、1つ以上のOAuth 設定定義が現在クリップボードにある場合有効化されます。クリップボードのコンテンツを、現在のプロジェクトの定義のプールに貼り付けます。
- *インポート*: 他の開かれているMobileTogether Designer ファイルから1つ以上のOAuth 定義をインポートすることのできる**[OAuth 設定のインポート]** ダイアログを開きます。詳細に関しては、**[OAuth 設定のインポート]** ダイアログを参照してください。
- *すべて選択*: すべての定義を選択します。
- *選択解除*: 定義を選択しません。

OAuth 設定のインポート

- OAuth 設定のインポート
  - *説明*

MobileTogether Designer 内で作成されるREST リクエストは、OAuth により認証することができます。認証方法に関しては REST リクエスト設定 のセクションを参照してください。

MobileTogether Designer プロジェクト内で複数のOAuth 設定 定義を作成することができます。これらの定義はプールに保管され、REST リクエストの認証のためにプールから定義を使用することができます。[OAuth 設定のインポート] ダイアログ (*下のスクリーンショット*) により他の開かれているMobileTogether Designer プロジェクトから定義を現在のプロジェクトにインポートすることができます。



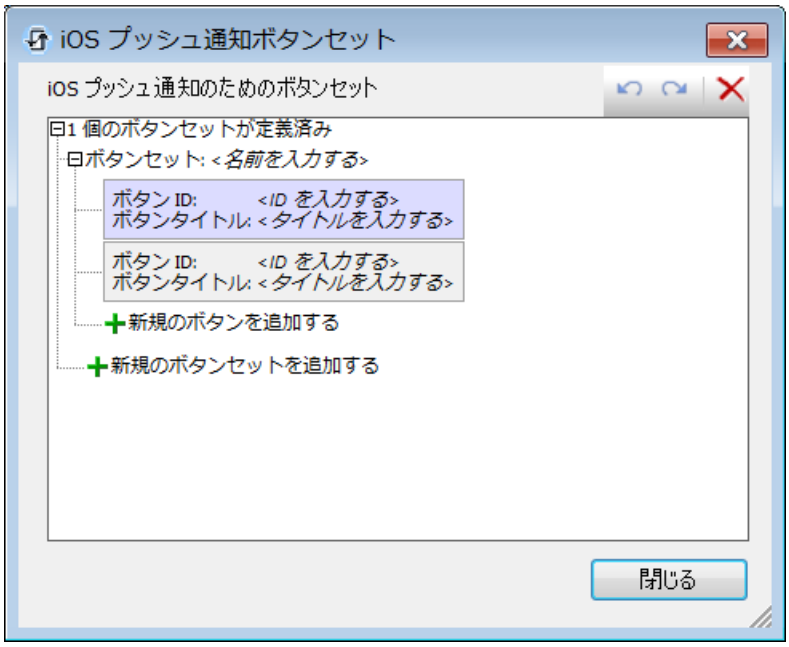
[ドキュメントを開く] ペイン(*上のスクリーンショット参照*) は、現在 MobileTogether Designer 内で開かれている他のすべてのドキュメントを表示します。下のペイン内で表示されるOAuth 設定 定義 の1つ以上のドキュメントを選択します。下のペインから現在のMobileTogether Designer プロジェクトにインポートする1つ以上の定義を選択し、**[選択項目をインポート]** をクリックします。定義はインポートされ、**[OAuth 設定の維持]** ダイアログ内で確認することができます。

iOS プッシュ通知ボタンセット

- iOS プッシュ通知ボタンセット
  - *説明*

    iOS デバイスで受信される プッシュ通知 (プッシュ通知) 内で表示されるボタンを作成するためにこのコマンドは使用されます。iOS プッシュ通知ボタンは AppStore App 内で使用することができますが、MobileTogether アプリでは使用することができません。プッシュ通知ボタンはソリューションの受信内で作成されます。ここで定義されるボタンセットの名前は、ソリューションの受信内でプッシュ通知が受信される際に、プッシュ通知内でどのボタンセットがソリューションの送信を表示するかを示すために使用されます。ソリューションの送信と受信が同じ場合、ソリューション内で定義されるプッシュ通知ボタンセットはプッシュ通知ボタンセットの選択のためのコンボボックス内で使用できるようになります。

    このコマンドを選択すると、iOS プッシュ通知ボタンセット ダイアログが表示されます (*下のスクリーンショット*).



    ボタンは、1つ、2つ、3つのボタンセットとして作成されます。作成するボタンセットの数量に制限はありません。ボタンセットは最多で3つのボタンを持つことが出来ます。

    ボタンセットを作成するには、「新規ボタンセットを追加」をクリックします。ボタンセット、または、ボタンを削除するには、アイテムを選択し、ダイアログの右上にある **「削除」** アイコンをクリックします。ダイアログの右上のアイコンを使用しダイアログの編集アクションを元に戻す、または、やり直すことが出来ます。

    メモ プッシュ通知ボタンはボタンコントロールに関連せず、区別してください。

    プッシュ通知に関する概要と詳細は、プッシュ通知 を参照してください。

20.4 ページ

[ページ] メニューにはプロジェクトの現在アクティブなページに適用されるコマンドが含まれています:

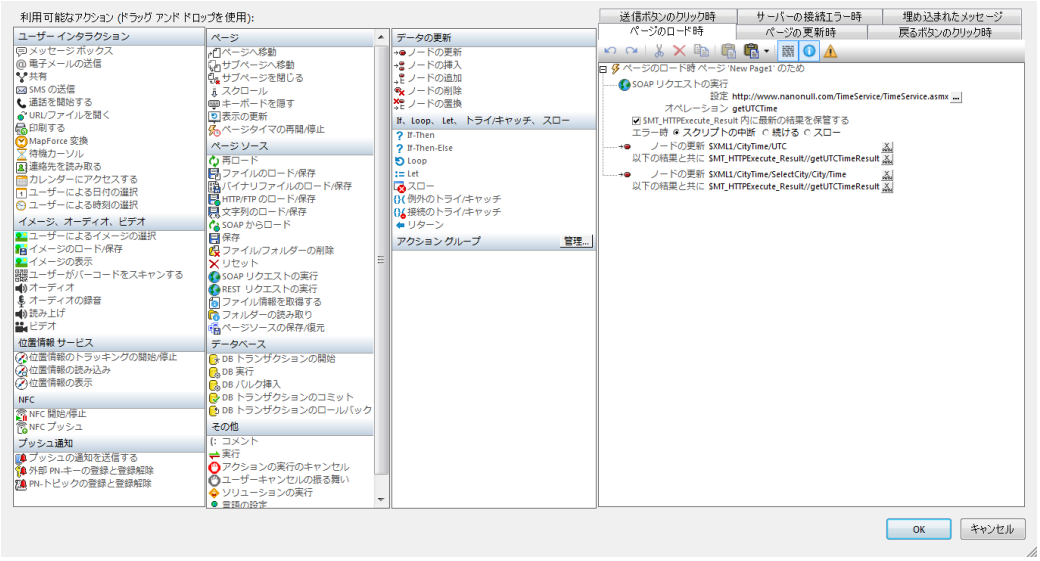
- ページ アクション
- アクションの概要
- コントロールへジャンプ
- タブ順序の表示 定義

メモ **[ページ]** メニューコマンドはページデザインビューでのみ使用することができます。

ページ アクション

- ページ アクション
  - *説明*

現在アクティブなページの [アクション] ダイアログを表示します (*下のスクリーンショット参照*)。ダイアログの左側のペインは、機能別に分類された使用することのできるアクションを表示しています。右側のペインはそのページで使用することのできるイベントのタブを含みます。使用することの可能なイベントはプロジェクトのワークフローロールにより決まります。例えば、**[戻る]** ボタンを押すことにより、戻ることのできないページにはボタンのクリック時 はありません。

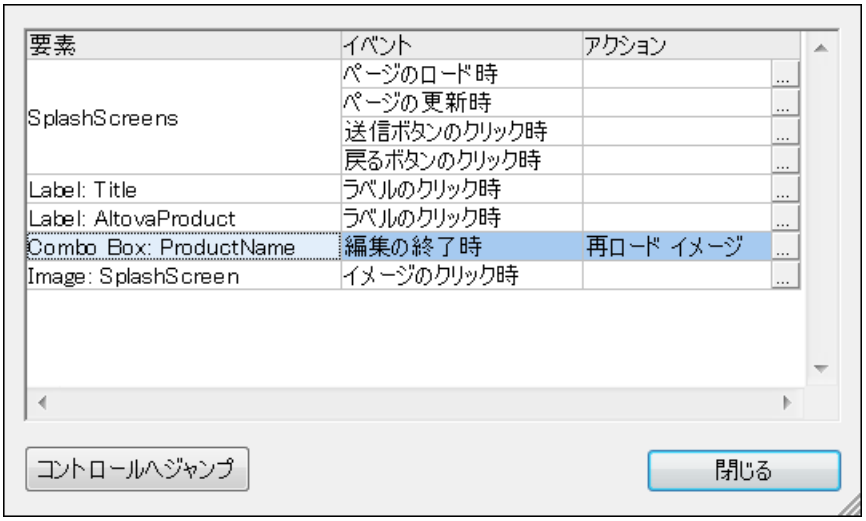


使用することのできるイベントが発生する際に特定のアクションを実行する場合、左側のペインのアクションを右側のペインのアクションにドラッグします。必要に応じてアクションの追加プロパティを指定します。(個別のページアクションについての詳細はの[ページアクション]セクションを参照してください。) 任意で [プロジェクトのために定義されたアクショングループ] を追加することができる点にも注意してください。**[OK]** をクリックして終了する。

アクションの概要

- アクションの概要
  - *説明*

    現在アクティブなページのアクションの概要ダイアログを表示します (*下のスクリーンショット参照*)。ダイアログはコントロール アクションを表示しています。ページデザイン内の各コントロールは、イベントと対応するアクションと共にリストされています。これが コントロール アクション です。ダイアログはアクティブなページで使用することのできるイベントも表示しています。これがページアクションです。例えば、下のスクリーンショットでは、`Income` アイテムはページアイテムです: ページアクション を持つページイベントがあります。他のアイテムすべてはコントロールです: コントロール アクション を持つコントロールイベントがあります。



- イベントのアクションを作成または既存のアクションを編集するには、イベントの **[編集]** アイコンをクリックします (*上のスクリーンショット参照*)。これにより[コントロール アクション] ダイアログまたは、[ページ アクション] ダイアログへ移動することができます。
- (ページではなく) コントロールが選択された場合、**[コントロールへジャンプ]** ボタンが有効化されます。クリックすることで、デザイン内の各当するコントロールまで移動することができます。

メモ このダイアログの最後のセクションは保存されます。結果、最後のセクションがハイライトされダイアログは再開されます。

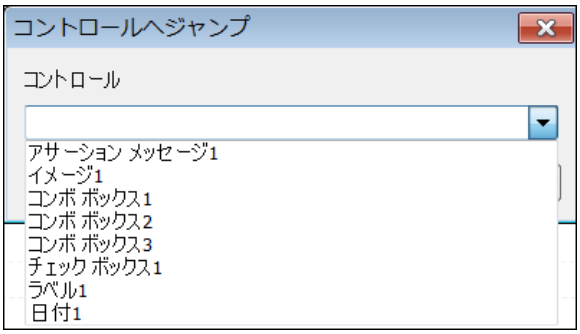
コントロールへジャンプ

- コントロールへジャンプ
  - *ショートカット*

    Ctrl+J

  - *説明*

    [コントロールへジャンプ] ダイアログを表示します(*下のスクリーンショット*)。ダイアログには、現在アクティブなページデザインを表示するドロップダウンリスト付きのコンボボックスがあります。ページコントロールは`Name` プロパティの値をアルファベット順にしてリストされています。



ドロップダウンリストからページコントロールを選択するか、またはコントロールの名前(自動補完を使用することが可能)を入力します。**[OK]** をクリックして終了します。ダイアログボックス内で選択されたページコントロールは、デザイン内で選択されます。コントロールがデータノードと関連付けられた場合、[ページ ソース] ペイン内のそのノードに焦点が当てられます。

タブの順序の表示 /定義

- タブの順序の表示 /定義
  - *ショートカット*

    Ctrl+T

  - *説明*

    タブの順序の機能はページ上のコントロールのためにタブの順序を定義します。(Web とWindows クライアントのみ)定義されると、デバイスのユーザーが**Tab** キーをタップすると、タブの順序のシーケンス内の次のコントロールにフォーカスが移動されます。

    選択されると、このコマンドは以下を行います: (i) タブの順序オプションダイアログを表示します (*下のスクリーンショット参照*)、(ii)デザイン内のすべてのコントロール上でタブの順序のシーケンス内で番号を与えられる可能性のある青い色の番号アイコンを示します (*下のスクリーンショット参照*)、(iii) コントロール プロパティ入力 /エスケープ が設定済みのコントロール上で **入力** と**エスケープ** キー ショートカットを示すアイコン(緑とピンクのアイコン)を表示します (*下のスクリーンショット参照*)。

    タブの順序のシーケンスを設定するには、以下を行います:

    1. タブの順序オプションダイアログを開きます。*次のタブ番号* フィールドは、**1** を表示します。タブの順序のシーケンスに割り当てられているすべてのコントロールが疑問符を含む青い丸と共に表示されます。
    2. タブの順序のシーケンス内で番号 **1** を割り当てるコントロールをクリックします。コントロールのアイコンは番号 **1** を含み、*次のタブ番号* フィールドは **2** に変更されます。
    3. タブの順序のシーケンスの2番目にするコントロールをクリックします。.コントロールのアイコンは、番号 **2** を含み、はに変更されます。*次のタブ番号* フィールドは **3** に変更されます。 .

    以下の点に注意してください:

    - *次のタブ番号* フィールドは常に次の選択に割り当てられます。この番号は、手動、または、自動的に増加することができます。
    - *自動インクリメント* オプションにより、次の番号を自動的に1つずつ増やすことができます。このオプションが選択されていない場合、は増分せず、選択するコントロールの番号は*番号を一意に保つ* オプションにより決定されます。
    - *次のタブ番号* フィールドを設定し、コントロールをクリックすることにより、個別のコントロールの番号を変更することができます。
    - *番号を一意に保つ* オプションにより、次の選択が前の選択番号とは異なることを保証することができます。
    - **割り当て済みの番号を全てクリアする**をクリックして、デザイン内の割り当て済みのすべての番号を削除することができます。

    メモ タブの順序の機能はWeb とWindows クライアント上でのみ使用することができます。

メモ コントロールのためのシーケンス内の番号にコントロールと**タブの順序**プロパティを設定し、タブの順序の機能は個別のコントロールのために設定することができます。

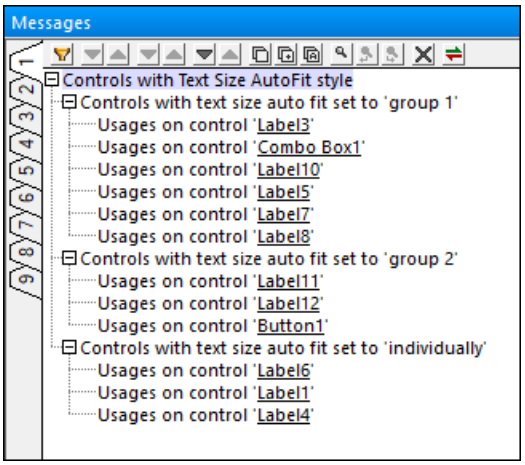
タブシーケンスの表示、入力 /エスケープショートカットに割り当てられているコントロールの表示
タブの順序のオプションダイアログが開かれると、デザイン内で次のインディケーターが表示されます:

- タブの順序のシーケンスに割り当てられているすべてのコントロールは、シーケンス内の場所を示す番号を含む青い丸と共に表示されています。タブの順序のシーケンスにコントロールが割り当てられていない場合、疑問符を含む青い丸が表示されます。
- **入力** または **エスケープ** キー ショートカットに割り当てられているコントロールは、対応するショートカットインディケーターと共に表示されています。(*上のスクリーンショットを参照*)。1つ以上のコントロールが同じ(**入力** または **エスケープ**) ショートカットに割り当てられている場合、ショートカットは最初に表示され、有効化されているコントロールに適用されます。

テキストサイズの自動調整グループのリスト

- テキストサイズの自動調整グループのリスト
  - *説明*

    メッセージペイン内でテキストサイズを自動的に調整するプロパティが設定され、グループ別に自動的に調整されたページ上の全てのコントロールをリストします（*下のスクリーンショットを参照してください*）。テキストサイズの自動調整プロパティの詳細に関しては、ラベルまたはボタンコントロールなどの、使用することのできるプロパティのために使用することのできるコントロールの詳細を参照してください。



20.5 テーブル

テーブル メニューコマンドはテーブル、が拡張子、テーブル行、テーブル列により選択されていると使用することができます。以下のコマンドが含まれています:

- テーブルの挿入
- テーブルの削除
- 行の挿入
- 行の追加
- 行の削除
- 列の挿入
- 列の追加
- 列の削除
- 左側のセルをジョイン
- 右側のセルをジョイン
- 上のセルをジョイン
- 下のセルをジョイン
- セルを水平に並べて分割
- セルを垂直に並べて分割
- 追加/削除ボタンの表示
- テーブルヘッダーの挿入
- テーブルフッターの追加
- テーブル先行列の挿入
- テーブル末尾列の追加
- テーブルヘッダーの削除
- テーブルフッターの削除
- テーブル先行列の削除
- テーブル末尾列の削除
- 行を繰り返し行に変換する
- 行を静的な行に変換する
- 繰り返しテーブルへの変換
- 非繰り返しテーブルへの変換
- 列を繰り返し列に変換する
- 列を静的な列に変換する

テーブルに関する詳細は、「オブジェクト機能のデザイン」「テーブル」のセクションを参照してください。

テーブルの挿入 /削除

テーブルの挿入 と**テーブルの削除** コマンドは、テーブル(静的、繰り返しまたは動的な行 および/または 動的な列を持つテーブル) 内のセルが選択されている場合使用することができます。

- テーブルの挿入
  - *アイコン*

  - *説明*

    テーブルの挿入 コマンドは、選択されたセルの左側に2x2 のディメンションの静的なテーブルを挿入します。

- テーブルの削除
  - *アイコン*

  - *説明*

    テーブルの削除 コマンドは現在選択されているテーブルを削除します。

行 /列の挿入 /追加 /削除

テーブルに表示される2つの 挿入」、2つの 追加」、および2つの 削除」コマンドは、テーブル(静的、繰り返しまたは動的な行 および/または動的な列を持つテーブル) 内の列または行が選択されている場合使用することができます。

行の挿入	**列の挿入**
行の追加	**列の追加**
行の削除	**列の削除**

挿入」と **追加」**コマンドにより現在選択されているセルに対しそれぞれ行 列を挿入 追加することができます。このように追加された行と列は静的です。これは、例えば、デザイン内に静的な行の1つが追加されると、出力内の静的な行として表示されることを意味します。もちろん、繰り返し構造内に列が追加されると、静的な行は構造の反復と共に繰り返されます。

行の削除 と**列の削除** コマンドは現在選択されている行 列をそれぞれ削除します。

テーブルセルが選択されると、これらのコマンドはコンテキストメニューコマンド として使用することができます。

セルのジョイン /分割

(下のテーブルに表示されている) 4 つの **「ジョイン」** コマンドと2 つの **「分割」** コマンドにより現在選択されているセルを隣接したセルにジョイン、または、現在選択されているセルを分割することができます。これらのコマンドはテーブルのセルのために使用することができます(静的、繰り返しまたは動的な行 および/または 動的な列を持つテーブル)。

左側のセルをジョイン	**セルを水平に並べて分割**
右側のセルをジョイン	**セルを垂直に並べて分割**
上のセルをジョイン	
下のセルをジョイン	

セルのジョイン
セルを近隣したセルに水平に、または、垂直にジョインすることができます。セルのジョインは、近隣するセルが存在する限り複数回おこなうことができます。セルのジョインは、セルを水平、または、垂直に2つまたは複数の列と行に対して行うことができます。詳細に関しては次を参照してください: *行 列のジョインと分割*。

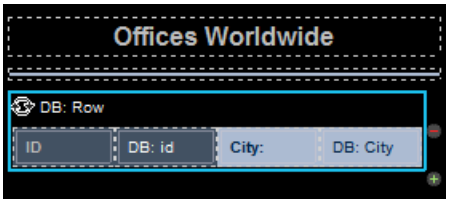
セルの分割
セルを水平、または、垂直に2つのセルに分割することができます。セルが水平に分割されると、元のセル内のコンテンツ、または、書式が新しいペアの左側のセルに保管されます。右側のセルは空です。セルが垂直に分割されると、新しいペアの上のセルに元のセル内のコンテンツ、または、書式が保管されます。下のセルは空です。

追加 /削除ボタンの表示

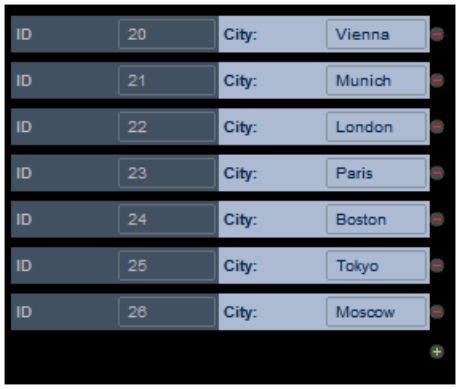
追加 /削除ボタンの表示

説明

追加 /削除ボタンの表示 コマンドは繰り返しテーブル、または、動的な行 を持つテーブルの繰り返し行に対して使用することができます。選択された行のために追加 /削除 ボタンを作成します。下のスクリーンショットが有効化された追加 /削除 ボタンを持つ動的な行が存在するテーブルのデザインを表示しています (テーブルの右下にある 2 つのアイコンにより示されています)。



上のスクリーンショットはデザイン内のテーブルを表示していますが、下のスクリーンショットはこのようなテーブルがクライアントデバイス上でどのように表示されるかを表示しています。 **「削除」** ボタンを押すことにより行を削除することができます (*下のスクリーンショットを参照してください*)。これにより、対応する行データが元のデータソースから削除されます。 **「追加」** ボタンをタップして新規の行を追加することができます。



ヘッダー /フッター、先行 /末尾列の追加

これら4つのコマンドはそれぞれ、テーブルのヘッダー、フッター、先行列、または、末尾列を追加します。これらは動的な(繰り返し) 行 または動的な(繰り返し) 列を持つテーブルに適用されます。

- テーブルヘッダーの挿入
  - *アイコン*

  - *説明*

    ヘッダーが無い動的な行を持つテーブルに空のヘッダー行を挿入します。テーブルにヘッダーが既存の場合、このコマンドは無効化になります。

- テーブルフッターの挿入
  - *アイコン*

  - *説明*

    フッターが無い動的な行を持つテーブルに空のフッター行を挿入します。テーブルにフッターが既存の場合、このコマンドは無効化になります。

- テーブル先行列の挿入
  - *アイコン*

  - *説明*

    先行列を持たない動的な列が存在するテーブル内に空の先行列を追加します。テーブルに先行列が既に存在する場合、このコマンドは無効化になります。*(メモ テーブルの左側の列が垂直ヘッダー列としての役割を果たす場合、テーブルの先行列として参照されます。)*

- テーブル末尾列の追加
  - *アイコン*

  - *説明*

    末尾列を持たない動的な列が存在するテーブル内に空の末尾列を追加します。テーブルに末尾列が既に存在

する場合、このコマンドは無効化になります。*(メモ: テーブルの右側の列が垂直フッター列としての役割を果たす場合、、テーブルの末尾列として参照されます。)*

ヘッダー /フッター、先行 /末尾列を削除する

これら4つのコマンドはそれぞれ、テーブルのヘッダー、フッター、先行列、または、末尾列を削除します。これらは動的な(繰り返し) 行 または動的な(繰り返し) 列を持つテーブルに適用されます。

- テーブルヘッダーの削除
  - *アイコン*

  - *説明*

    動的な行を持つテーブルのヘッダーを削除します。テーブルにヘッダーが存在しない場合、このコマンドは無効化になります。

- テーブルフッターの削除
  - *アイコン*

  - *説明*

    動的な行を持つテーブルのフッターを削除します。テーブルにフッターが存在しない場合、このコマンドは無効化になります。

- テーブル先行列の削除
  - *アイコン*

  - *説明*

    動的な列が存在するテーブルの先行列を削除します。テーブルに先行列が存在しない場合、このコマンドは無効化になります。*(メモ: テーブルの左側の列が垂直ヘッダー列としての役割を果たす場合、テーブルの先行列として参照されます。)*

- テーブル末尾列の削除
  - *アイコン*

  - *説明*

    動的な列が存在するテーブルの末尾列を削除します。テーブルに末尾列が存在しない場合、このコマンドは無

効化になります。*(メモ: テーブルの右側の列が垂直フッター列としての役割を果たす場合、、テーブルの末尾列として参照されます。)*

行を繰り返し/静的な行に変換する

- この行を繰り返し行に変換する

  - *説明*

    選択された行が動的な (繰り返し) 行では無い場合、このコマンドは繰り返し行に変換します。選択された行が繰り返し行である場合、このコマンドは無効化になります。

- この行を静的な行に変換する

  - *説明*

    選択された行が動的な (繰り返し) 行である場合、このコマンドは静的な行に変換します。選択された行が繰り返し行では無い場合、このコマンドは無効化になります。

繰り返し/非繰り返しテーブルへの変換

- 繰り返しテーブルに変換する

  - *説明*

    選択されたテーブルが繰り返しテーブルでは無い場合、このコマンドは繰り返しテーブルに変換します。選択されたテーブルが繰り返しテーブルである場合、このコマンドは無効化になります。

- 非繰り返しテーブルに変換する

  - *説明*

    選択されたテーブルが繰り返しテーブルである場合、このコマンドは非繰り返しテーブルに変換します。選択されたテーブルが繰り返しテーブルでは無い場合、このコマンドは無効化になります。

(C) 2018 Altova GmbH

列を繰り返し/非繰り返し列へ変換する

- この列を繰り返し列に変換する

  - *説明*

    選択された列が動的 (繰り返し) 列では無い場合、このコマンドは列を繰り返し列に変換します。選択された列が繰り返し列の場合、このコマンドは無効化になります。

- この列を静的な列に変換する

  - *説明*

    選択された列が動的 (繰り返し) 列の場合、このコマンドは列を静的な列に変換します。選択された列が繰り返し列では無い場合、このコマンドは無効化になります。

20.6 表示

[表示] メニューには次のコマンドが含まれます:

- ステータスバー
- ページ
- ファイル
- コントロール
- ページ ソース
- 概要
- スタイル & プロパティ
- メッセージ
- 全てをオン/オフ
- ズーム
- 拡大
- 縮小
- ズームを100% にセット
- 選択合わせてズームする
- ウィンドウのサイズに合わせて大きさを変更

(C) 2018 Altova GmbH

ステータス バー とペイン

表示メニューのコマンドをクリックすることにより、ステータスバーと各種ペインをオン/オフに切り替えることができます:

- ステータスバー
- [ページ] ペイン
- [ファイル] ペイン
- [コントロール] ペイン
- [ページ ソース] ペイン
- [概要] ペイン
- [スタイル & プロパティ] ペイン
- [メッセージ] ペイン

[全てをオン/オフ] コマンドによりすべてのペインおよびステータスバーを表示または非表示状態で切り替えることができます。

ズームのレベル

ズームコマンドにより、ページデザインの拡大率を(10% と100% の間で) およびワークフローダイアグラム(10% と200% の間で) を柔軟に変更することができます。ズームコマンドはページデザインビューで有効化することができます。

- ズーム
  - *説明*

    ページデザインビュー内で10% から100% へと拡大率を変更することのできる、スライドルールを含むズームダイアログを開きます。

- 拡大
  - *アイコン*

  - *説明*

    ダイアグラムを10 パーセント ポイント(`100`, `110`, `120...`) ごとに拡大します。ページデザインビュー内では、テンプレートサイズの制限により既に上限に達している場合があります。

- 縮小
  - *アイコン*

  - *説明*

    コマンドが実行されるたびに、ダイアグラムを10 パーセントポイント(`100`、`90`、`80...`) ごとに縮小します。

- ズームを 100%にリセットする
  - *説明*

    ズーム要素を100% にリセットします。これは元のサイズに戻すための手っ取り早い方法です。

20.7 ツール

[ツール] メニューは次のコマンドを含みます:

- グローバル リソース
- アクティブな構成
- ユーザー定義ツール
- カスタマイズ
- ツールバーとウィンドウの復元
- オプション

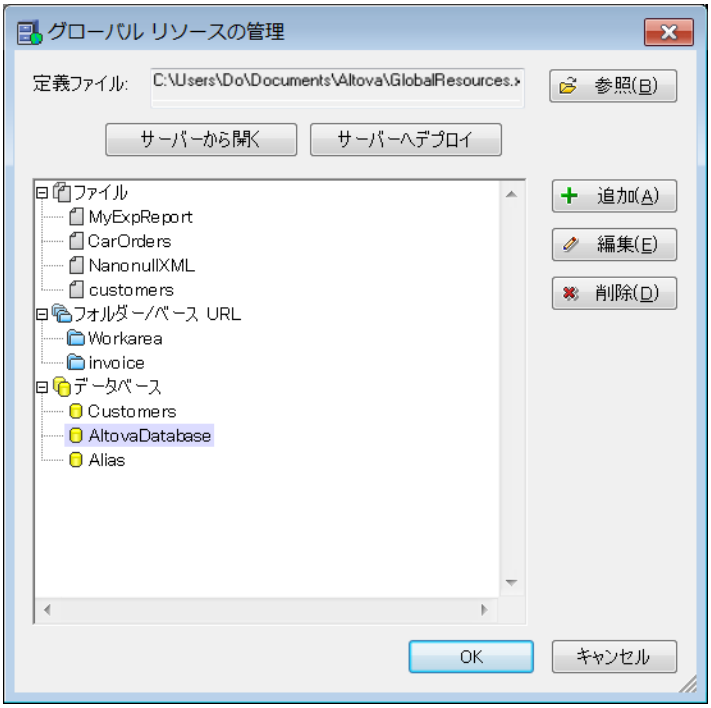
グローバル リソース

- グローバル リソース
  - *アイコン*

  - *説明*

以下を行うことのできる、グローバルリソースダイアログを表示します (*下のスクリーンショット*):

- グローバルリソースで使用するグローバルリソースXML ファイルを指定します (*定義ファイル*)。
- ファイル、フォルダー、データベースグローバルリソース(またはエイリアス) を追加します。
- 各グローバルリソース(エイリアス) のための各種の構成を指定します (これを行うにはグローバルリソースを編集します。)
- **[サーバーから開く]** と**[サーバーへデプロイ]** により、グローバルリソースをサーバーから開き、またグローバルリソースをMobileTogether Server へデプロイすることができます。

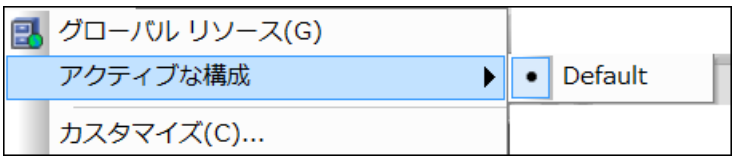


グローバルリソースの定義方法の詳細はグローバルリソースの定義のセクションで説明されています。

アクティブな構成

- アクティブな構成
  - *説明*

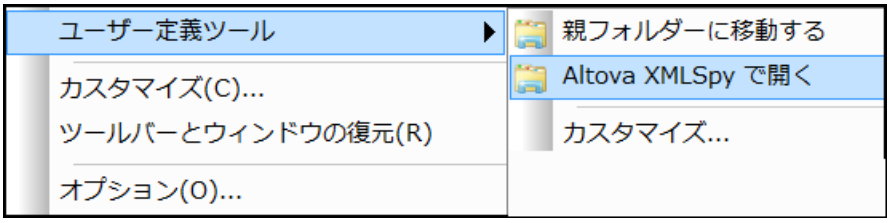
    コマンドロールにマウスをかざすと、現在アクティブなグローバルリソースXML ファイルで定義されているすべての構成を含むサブメニューが展開されます。



現在アクティブな構成はドットで表示されています。上のスクリーンショットでは、現在アクティブな構成は `Default` です。アクティブな構成を変更するには、アクティブにする構成を選択してください。

ユーザー定義ツール

ユーザー定義ツール コマンドにカーソルを置くと、外部アプリケーションを使用するカスタム化されたコマンドを含むサブメニューが表示されます。下のスクリーンショットでは、2つのカスタムコマンドが表示されています: (i) Windows Explorer 内で親フォルダーを開く、(ii) Altova XMLSpy でアクティブなファイルを開く。カスタマイズダイアログのツールタブ内でこれらのコマンドを作成することができます。これらのカスタムコマンドの1つをクリックすると、このコマンドに関連するアクションが実行されます。



作成されたカスタムコマンドのリストの下に、**「カスタマイズ」** コマンド(*スクリーンショットを参照*)があります。これによりカスタマイズダイアログのツールタブが開かれ、既存のカスタムコマンドを編集、または、新規のカスタムコマンドを作成することができます。

カスタマイズ

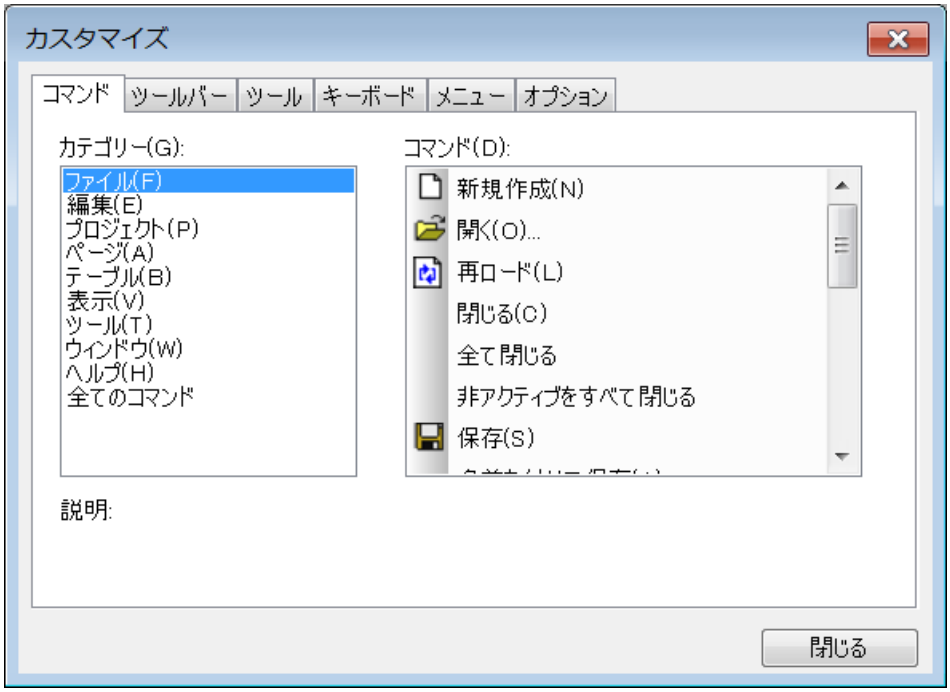
個人のニーズに応えるため、カスタマイズコマンドを使用してMobileTogether Designer をカスタマイズすることができます。

コマンド
ツールバー
ツール
キーボード
メニュー
オプション

コマンド

▾ コマンド

コマンドタブにより、優先順位に従いアプリケーションコマンドをメニューとツールバーに追加することができます。新しいアプリケーションまたはメニューを作成することができない点に注意してください。



ツールバーまたはメニューにコマンドを追加するには、以下を行ってください:

1. **[全てのコマンド]** カテゴリをカテゴリリスト ボックスから選択します。使用することのできるコマンドはコマンド リスト ボックスに表示されます。
2. コマンド リストボックスのコマンドをクリックして、既存のメニューまたはルールバーにドラッグします。コマンドをドロップする妥当なポジションにカーソルをかざすとI-ビームが表示されます。
3. コマンドを挿入するために、マウスボタンをリリースします。

   以下の点に注意してください:

   - コマンドをドラッグすると、マウスポインターの先に小さなボタンが表示されます: これによりコマンドが現在ドラッグされていることを示します。
   - ポインターの下の "x" は、コマンドが現在のカーソルの位置にドロップできないことを示しています。
   - コマンドがドロップできる場所 (ツールバーまたはメニュー) にカーソルが移動されると、"x" が消えI-ビームが有効な場所を示します。
   - コマンドはメニューまたはツールバーに置くことができます。自身のツールバーを作成する場合、ツールバーを作成する、このカスタマイズメカニズムを使用します。
   - 閉じられたメニューにカーソルを移動すると、そのメニューを開くことができ、メニューのどの場所にでもコマンドを挿入することができます。

コマンドをテキストメニューに追加するには、以下を行ってください:

1. [カスタマイズ] ダイアログの**メニュー**タブをクリックします。
2. [コンテキストメニュー] ペインで、コンボボックスからコンテキストメニューを選択します。選択されたコンテキストメニューが表示されます。
3. [カスタマイズ] ダイアログで、コマンドタブに切り替えます。
4. コマンドリストボックスから作成するコマンドをドラッグしてコンテキストメニュー内の希望する場所にドロップします。

メニュー、コンテキストメニューまたはツールバーからコマンドを削除する、またはメニュー全体を削除するには、以下を行ってください。

1. カスタマイズダイアログが開いた状態で(任意のタブを選択して) メニューまたはメニューコマンドを右クリックします。
2. **[削除]** を表示されるコンテキストメニューから選択します。または、メニューまたはメニューコマンドを "x" アイコンがマウスポインターの下に表示されるまでドラッグし、メニューまたはメニューコマンドにドロップします。

削除されたメニューコマンドを回復するには、このセクションで説明されているメカニズムを使用します。削除されたメニューを回復するには **[ツール | カスタマイズ | メニュー]** に移動して、[アプリケーション フレーム メニュー] ペインの **[リセット]** ボタンをクリックします。または、 **[ツール | カスタマイズ | ツールバー]** に移動して、メニューバーを選択し、 **[リセット]** ボタンをクリックします。

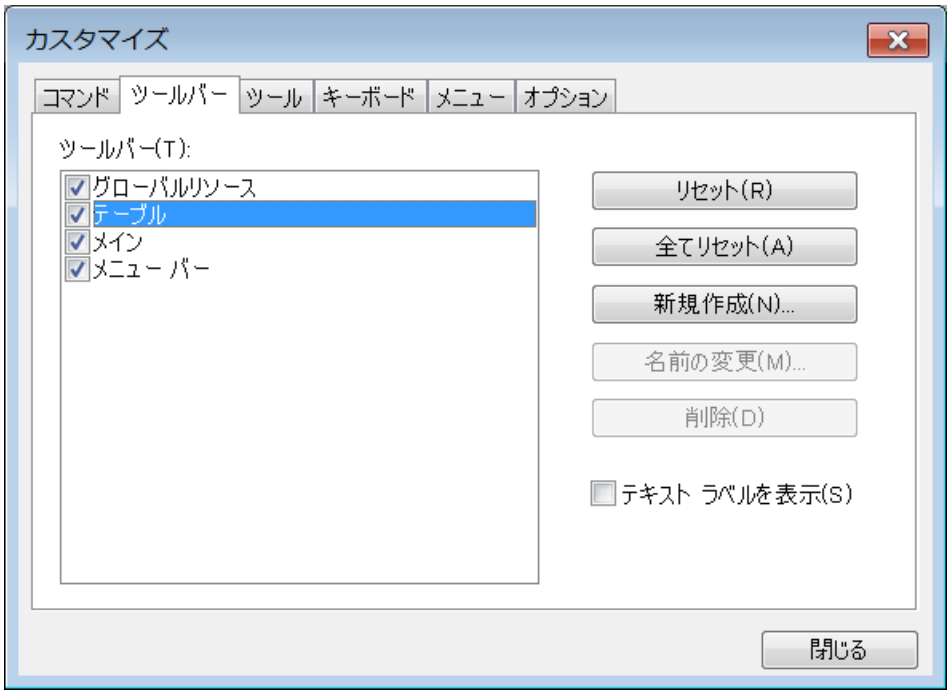
ツールバー

ツールバー

アプリケーションツールバーには最も頻繁に使用されるメニュー コマンドのアイコンがあります。各アイコンの情報は、カーソルがアイコンの上にかざされると、ヒントおよびステータスバーに表示されます。ツールバーをフロートするウィンドウとして表示される画面の任意の場所にドラッグすることができます。

ツールバータブにより以下を行うことができます:

- 特定のツールバーをアクティブ化または非アクティブ化する(つまり、インターフェイスに表示されるツールバーを決定する)
- 各ツールバーに表示されるアイコンを設定する
- 特定のツールバーを作成する



次の機能を使用することができます:

- **ツールバーのアクティブ化または非アクティブ化**: ツールバーリストボックスのチェックボックスをクリックする。
- **新しいツールバーを追加する [新規作成]** ボタンをクリックして、表示される[ツールバー名前] ダイアログ内のツールバーに名前を付ける。 コマンド タブからコマンドを新しいツールバーにドラッグします。
- **追加されたツールバーの名前を変更する** ツールバーペインから追加されたツールバーを選択します。**[名前の変更]** ボタンをクリックして、表示される[ツールバー名前] ダイアログの名前を編集します。
- **メニューバーをリセットする** [ツールバー] ペインからメニューバーアイテムを選択して、**[リセット]** をクリックします。これによりメニューバーをアプリケーションが最初にインストールされた時の状態にリセットします。
- **全てのツールバーとメニューコマンドをリセットする [全てリセット]** ボタンをクリックします。これにより、すべてのツールバーとメニューをアプリケーションが最初にインストールされた時の状態にリセットします。
- **ツールバーを削除する** ツールバーペイン内で削除するツールバーを選択して、**[削除]** をクリックします。
- **特定のツールバーのコマンドのテキストラベルを表示する** ツールバーを選択して、テキストラベルチェックボックスの表示をクリックします。各ツールバー別にテキストラベルがアクティブ化されなければならない点に注意してください。

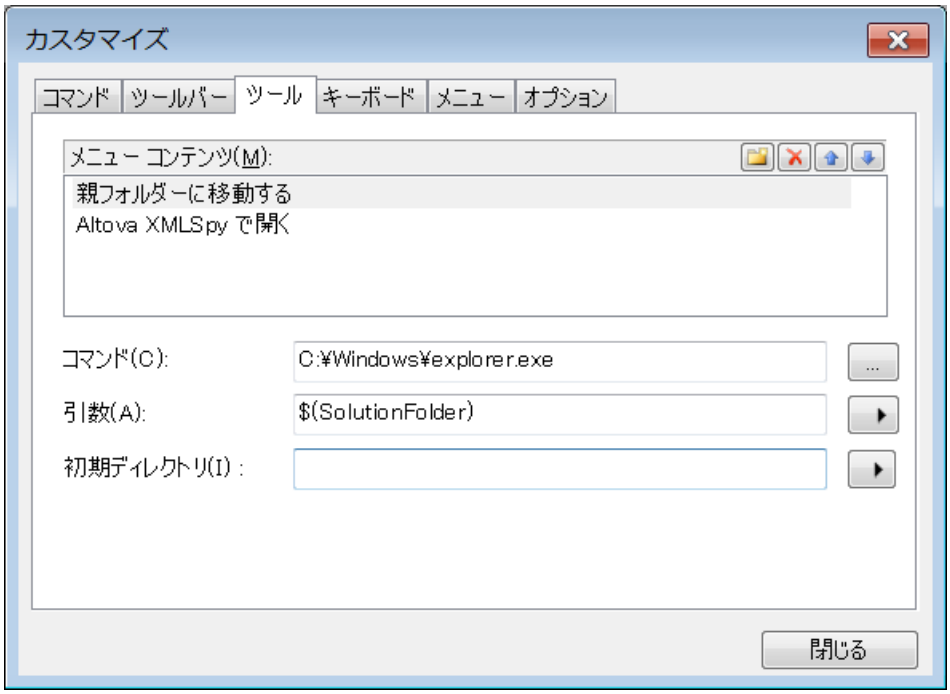
メモ ツールバーにコマンドを追加するには、コマンドタブのコマンドリストボックスからコマンドをツールバーにドラッグしま

す。ツールバーからコマンドを削除するには、[カスタマイズ] ダイアログを開き、任意のタブを選択して、コマンドをツールバーにドラッグします (詳細は コマンド を参照してください)。

ツール

▾ ツール

「ツール」 タブによりMobileTogether Designer 内から外部アプリケーションを使用するためのコマンドをセットアップすることができます。これらのコマンドは **「ツール」 ユーザー定義ツール」** メニューに追加されます。例えば、MobileTogether Designer 内のアクティブなファイルを(ツール) タブ内に作成した **「ツール | ユーザー定義ツール」** メニューをクリックして XMLSpy などの外部アプリケーションで開くことができます。



外部アプリケーションを使用するコマンドのセットアップは、以下のようにおこないます:

1. *メニュー コンテンツ*ペイン内で(*上のスクリーンショット参照*)、ペインのタイトルバー内の **「新規作成」** アイコンをクリックして、作成されたアイテムライン内にメニューコマンドの名前を入力します。 上のスクリーンショット内では、メニュー コマンド **「親フォルダーに移動する」** が入力されています。Windows Explorer 内でアクティブなドキュメントの親フォルダーを開くこのコマンドを使用する予定です。 **「新規作成」** アイコンをクリックしてコマンドリストにコマンドを追加することができます。上のスクリーンショット内では、例えば、Altova XMLSpy プログラム内でアクティブなドキュメントを開くためにコマンドを作成することができます。 **「アイテムを上に移動」** と **「アイテムを下に移動」** アイコンを使用して他のコマンドに対してリスト内を上または下に移動することができます。 **「削除」** アイコンをクリックしてコマンドを削除します。
2. 外部アプリケーションをコマンドと関連付けるには、*メニュー コンテンツ*ペイン内でコマンドを選択します。*コマンド*フィールド内で、外部アプリケーションの実行可能ファイルへのパスを入力、または参照します。上のスクリーンショット内では、Windows Explorer 実行可能ファイルへのパスが*コマンド*フィールド内に入力されています。
3. *引数* フィールド内で外部アプリケーションにパスされる引数は選択されます (*上のスクリーンショット参照*)。使用することのできる引数は、*引数* フィールドのボタンをクリックすると表示され、下に説明が表示されます。引数を選択すると、引数フィールドにコード文字列が表示されます。例えば、上のスクリーンショット内では、Windows Explorer でソリューションデザインであるべきアクティブなドキュメント内の開かれるフォルダーとしてパスされます。
4. 現在の作業ディレクトリ(任意)を指定する場合、*最初のディレクトリ*フィールドに入力します。
5. **「閉じる」** をクリックして完了します。

「ツール」 ユーザー定義ツール」 メニュー内に作成されたユーザー定義コマンドが表示されます。

(「ツール」 ユーザー定義ツール」 メニュー内の)作成されたユーザー定義コマンドをクリックすると、コマンドに関連付けられたアクションが実行されます。スクリーンショット内に表示されるコマンドのサンプルは以下をおこないます: Windows Explorer 内でアクティブなソリューションがロケートされるフォルダーが開かれます。Altova XMLSpy コマンド内の開くコマンドは、XMLSpy 内でアクティブなドキュメントを開きます。XMLSpy にパスされる引数は親フォルダーへのパスではなく、アクティブなドキュメントへのパスです。

引数

引数 フィールドは外部アプリケーションコマンドにパスされる引数を指定します。

- *ソリューション ファイル名:* アクティブなソリューション デザインファイルの名前です。
- *ソリューション ファイル パス:* アクティブなファイルの名前を含むアクティブなソリューション デザインファイルへのパスです。
- *ソリューション フォルダー:* アクティブなソリューション デザインファイルの親フォルダーです。
- *一時 フォルダー:* 一時ファイルを保管するためのWindows システムフォルダーへのパス。

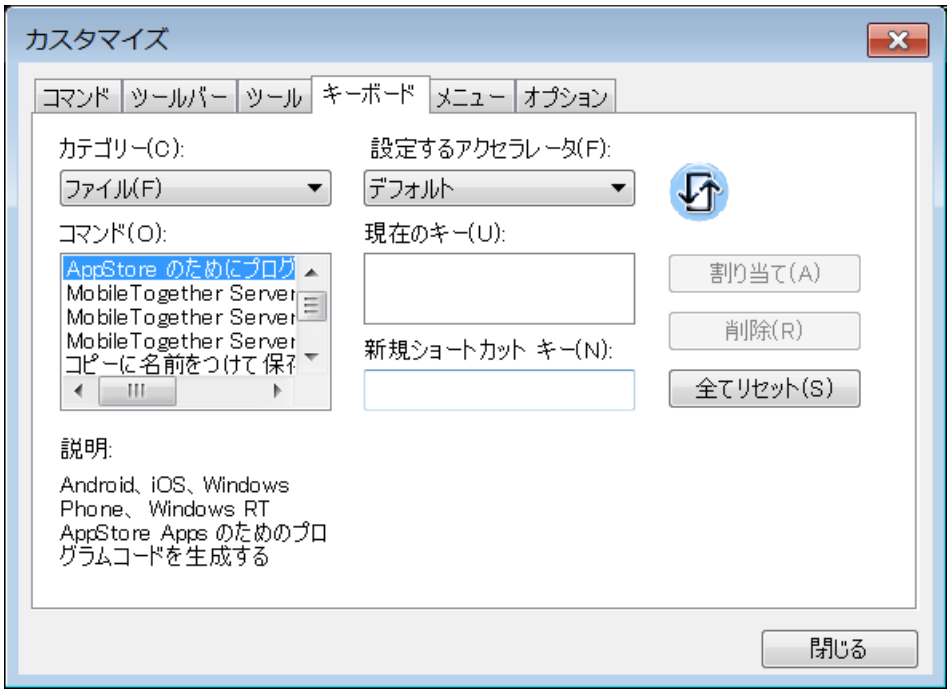
最初のディレクトリ

*最初のディレクトリ*エントリは任意です。 コマンドのために作成された最初のディレクトリを設定します。

キーボード

キーボード

キーボードタブにより、コマンドのための新しいキーボードショートカットを作成することができます。



ショートカットの割り当て

コマンドにショートカットをわりあてるには、次を行ってください。

1. カテゴリーコンボボックスの全てのコマンドカテゴリーを選択します。
2. コマンドリストボックスから、新しいショートカットを割り当てるコマンドを選択、または変更するショートカットがあるコマンドを選択します。
3. 新しいショートカットキーテキストボックスをクリックして、ショートカットを割り当てるコマンドを押します。コマンドにショートカットが割り当てられていない場合、**[割り当て]** ボタンを有効化されています。コマンドにショートカットが既に割り当てられている場合、テキストボックスの下にそのコマンドは表示され、**[割り当て]** ボタンは非有効化されています。(新しいショートカットキーテキストボックスをクリアするには以下のコントロールキーを押します。**Ctrl**、**Alt** または **Shift**)。
4. **[割り当て]** ボタンをクリックしてショートカットを割り当てます。現在のキーリストボックスにショートカットは表示されます。単一のコマンドに複数のショートカットを割り当てることができます。
5. **[閉じる]** ボタンをクリックして確認します。

ショートカットの削除

ショートカットは複数のコマンドに割り当てることができません。ショートカットを削除する場合、現在のキーリストボックスのショートカットをクリックして **[削除]** ボタンをクリックします。**[閉じる]** をクリックします。

アクセラレータの設定

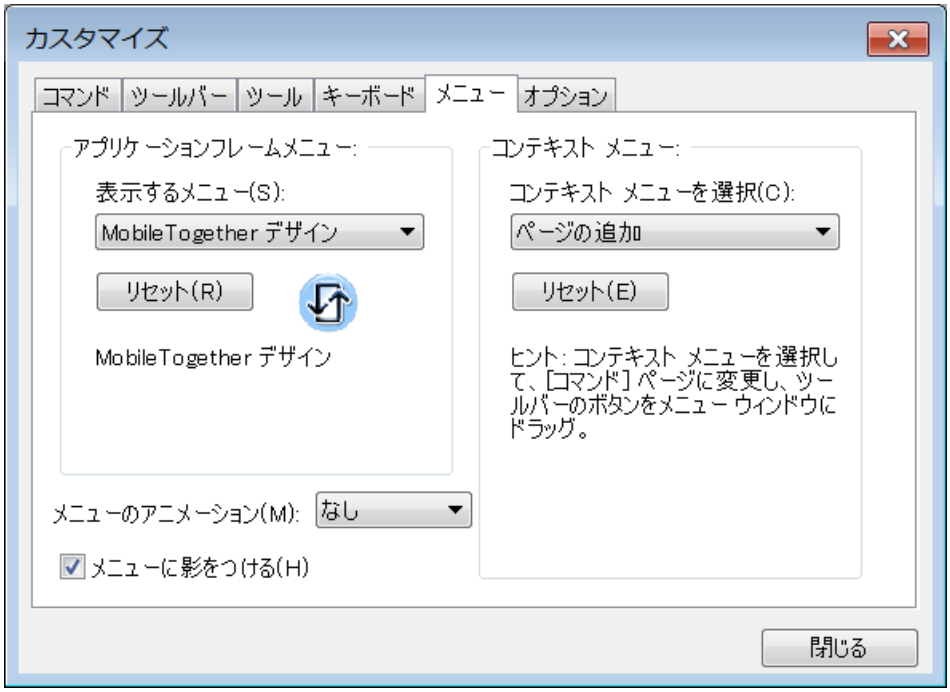
現在使用できる機能はありません。

(C) 2018 Altova GmbH

メニュー

▾ メニュー

メニュータブにより2 つのメインメニューバー(デフォルトとアプリケーション メニュー バー) およびコンテキストメニューをカスタマイズすることができます。



デフォルトのメニューバーおよびアプリケーションのメニューバーをカスタマイズする

デフォルトのメニューバーはメインウィンドウでドキュメントが開かれていない時に表示されているメニューバーです。アプリケーションのメニューバーは1 つ以上のドキュメントがメインウィンドウで開かれている時に表示されているメニューバーです。各メニューバーは個別にカスタマイズすることができ、1 つのメニューバーに対して行われカスタマイズの変更は互いに影響しません。メニューバーをカスタマイズするにや、メニューの表示コンボボックスから選択します (*上のスクリーンショット参照*)。[カスタマイズ] ダイアログのコマンドタブに切り替えて、コマンドリストボックスからコマンドをドラッグしてメニューバーまたはメニューにドロップします。

メニューからコマンドを削除してメニューバーをリセットする

メニュー全体またはメニュー内のコマンドを削除するには、メニューまたはメニューコマンドを選択して、(i) 右クリックして **[削除]** を選択する。または(ii) メニューバーまたはメニューからドラッグします。2 つのメニューバー(デフォルトとアプリケーションメニューバー) をインストールした初期の状態に、メニューの表示コンボボックスからメニューを選択して、コンボボックス下の **[リセット]** ボタンをクリックして、リセットすることができます。

コンテキストメニューのアプリケーションをカスタマイズする

コンテキストメニューはアプリケーションのインターフェイスの特定のオブジェクトを右クリックすると表示されるメニューです。これらのコンテキストメニューは次の方法でカスタマイズすることができます:

1. テキストメニューの選択コンボボックスからコンテキストメニューを選択します。これによりコンテキストメニューがポップアップします。
2. [カスタマイズ] ダイアログのコマンドタブに切り替えます。
3. コマンドリストボックスからコンテキストメニューへコマンドをドラッグします。
4. コンテキストメニューからコマンドを削除する場合は、コンテキストメニューのコマンドを右クリックして、**[削除]** をクリックします。または、削除するコマンドをコンテキストメニュー外へドラッグします。

コンテキストメニューをインストールした初期の状態に、メニューの表示コンボボックスからメニューを選択して、コンボボックス下の**[リセット]** ボタンをクリックして、リセットすることができます。

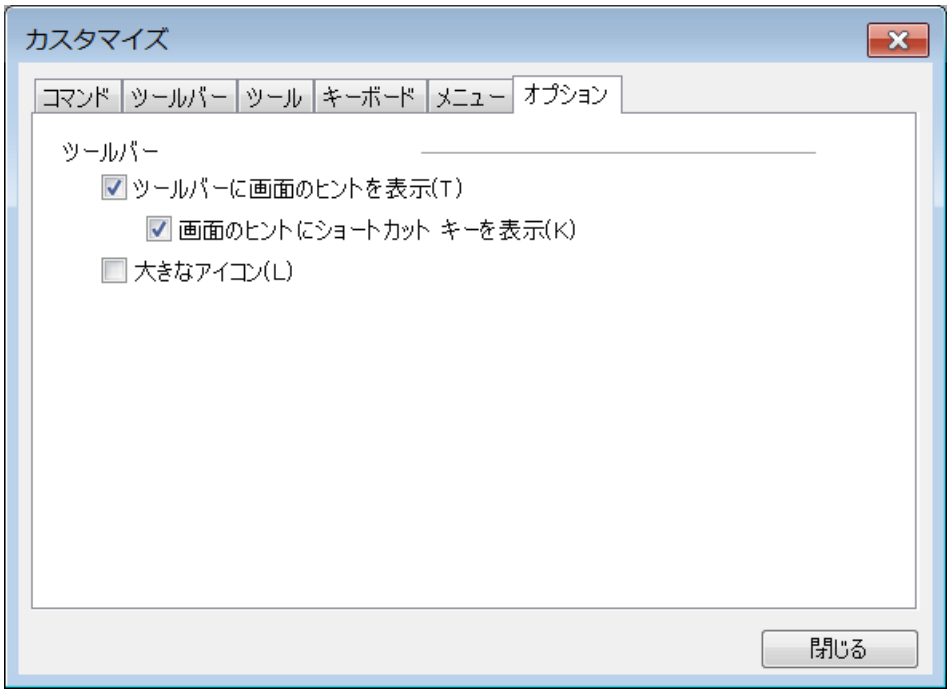
メニュー に影をつける

[メニューに影を付ける] チェックボックスを選択して、すべてのメニューに影をつける。

オプション

▾ オプション

オプションタブにより一般環境設定の定義を行うことができます。



次のオプションの切り替えを行うためにチェックボックスを選択します:

- **ツールバーに画面のヒントを表示**: ツールバーのアイコンにマウスをかざすとポップアップが表示されます。ポップアップはアイコンの機能の短い説明と **ショートカットキーを表示** オプションがチェックされており、ショートカットが割り当てられている場合、関連したキーボードショートカットが表示されます。
- **画面のヒントにショートカットキーを表示**: 画面のヒントにショートカットの情報が表示されるかを定義します。
- **大きなアイコン**: ツールバーのアイコンの大きさを標準と大きなアイコン間で切り替えることができます。

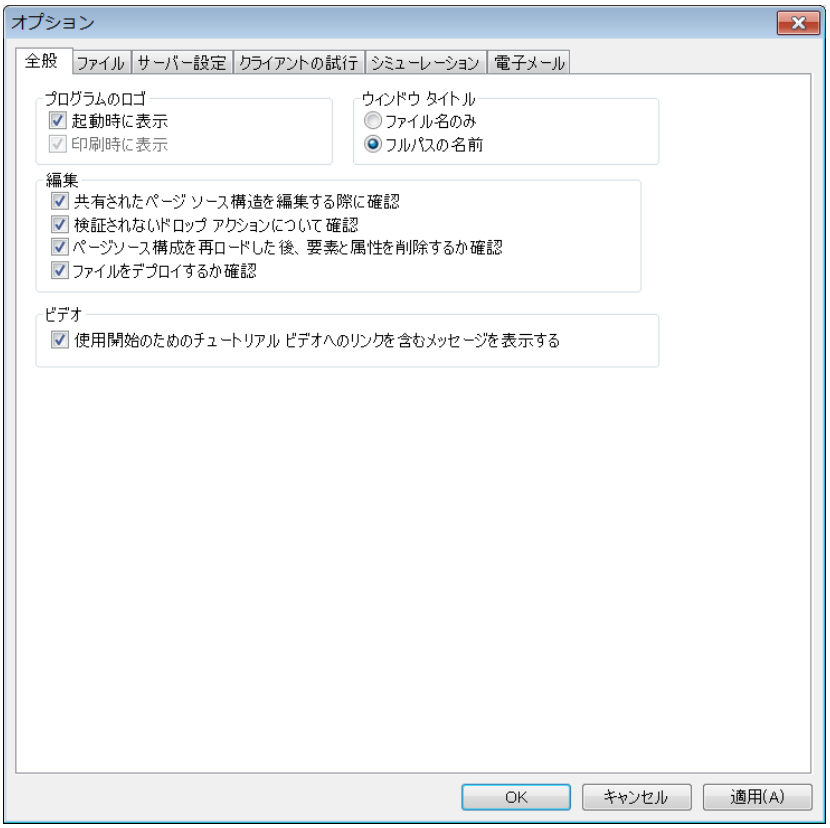
ツールバーとウィンドウの復元

- ツールバーとウィンドウの復元
  - *説明*

    MobileTogether Designer をシャットダウンし、すべてのツールバーとウィンドウをインストール時の元の状態にリセットして再起動します。

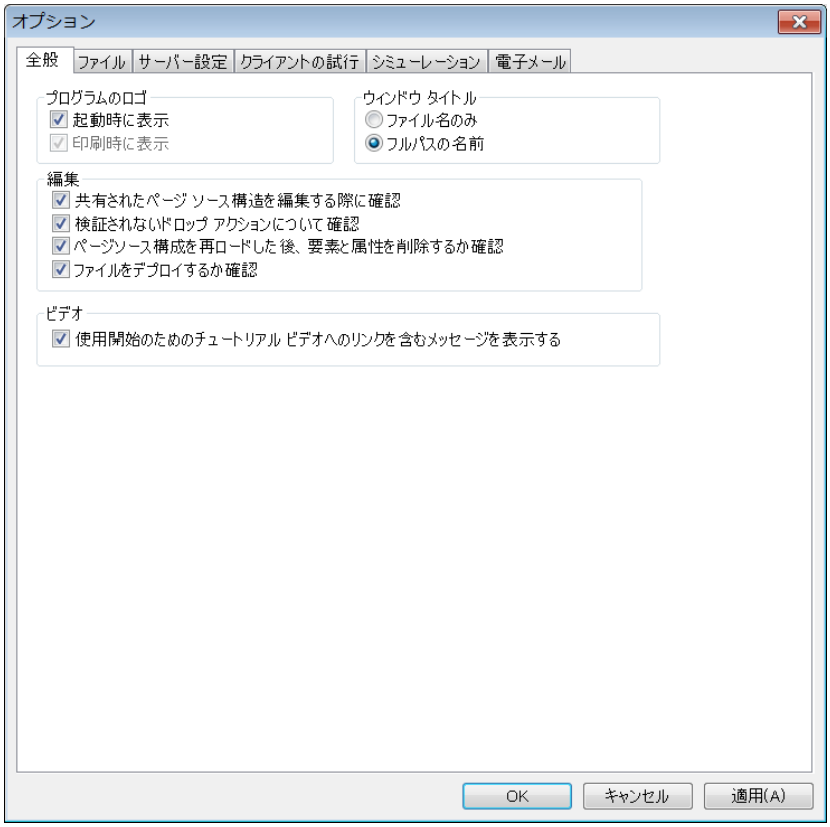
オプション

[オプション] コマンドはオプションダイアログを表示します。(*下のスクリーンショット*)。設定は下で説明されている複数のタブで使用することができます。



全般

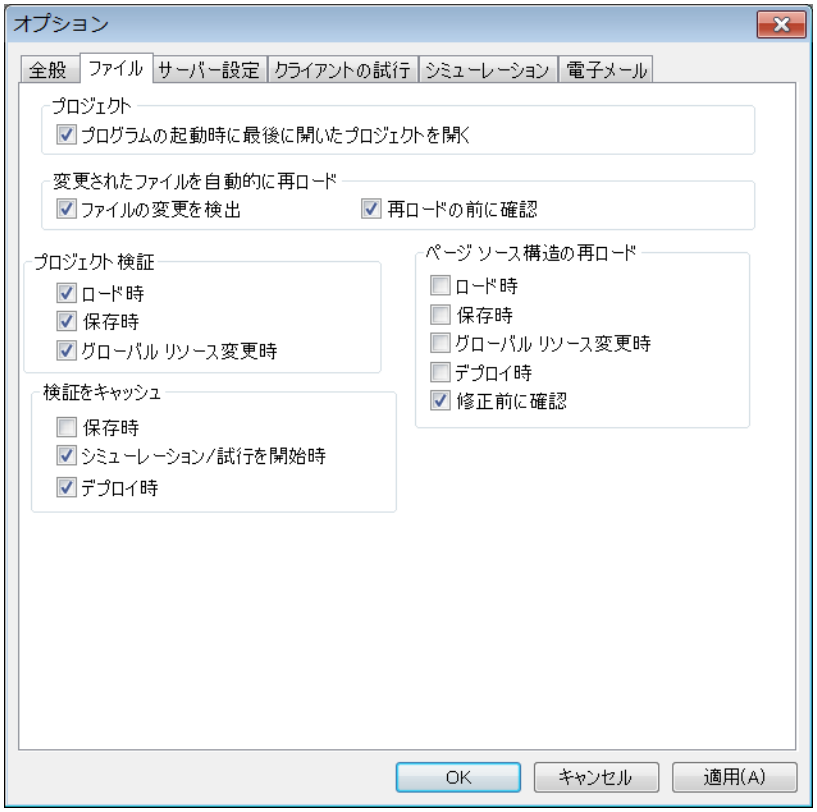
全般タブでは(*下のスクリーンショット*) スクリーンショットに表示されている通り設定を定義することができます。



- *プログラムのロゴ*: プログラムの開始または印刷の出力で表示することができます。
- *ウィンドウのタイトル*: アプリケーションウィンドウはファイル名またはフルファイルパスとファイル名を表示することができます。
- *編集 | 確認*: デザイナー入力が必要な場合、アクションを使用することができるか確認されます。例えば、共有されるページリソースが変更された場合、リソースを共有するすべてのページで変更を使用することができるか、また変更は現在のページのみに適用されるか確認されます。
- *ビデオ*: MobileTogether Designer がデザイン無しで開かれた場合、MobileTogether Designer デモビデオについてのメッセージを表示します。(MobileTogether Designer をデザイン無しで開始するには、すべてのデザインを閉じて、MobileTogether Designer を閉じます。) メッセージは、Altova Web サイト上の デモビデオのページ へのリンクを含んでいます。このページのビデオでは MobileTogether Designer の機能を簡単に紹介しています。
- *デザイン ビュー: 列の幅の最小値*: デザインビュー内のテーブル最小の幅を設定します。スライダーは、**0** から**7** へのスケールのオプションを提供します。スライダーをドラッグして、**左に移動]** と **[右に移動]** キーを使用してください。選択された値は、クライアントデバイス上のテーブル列の幅に影響を与えません。

ファイル

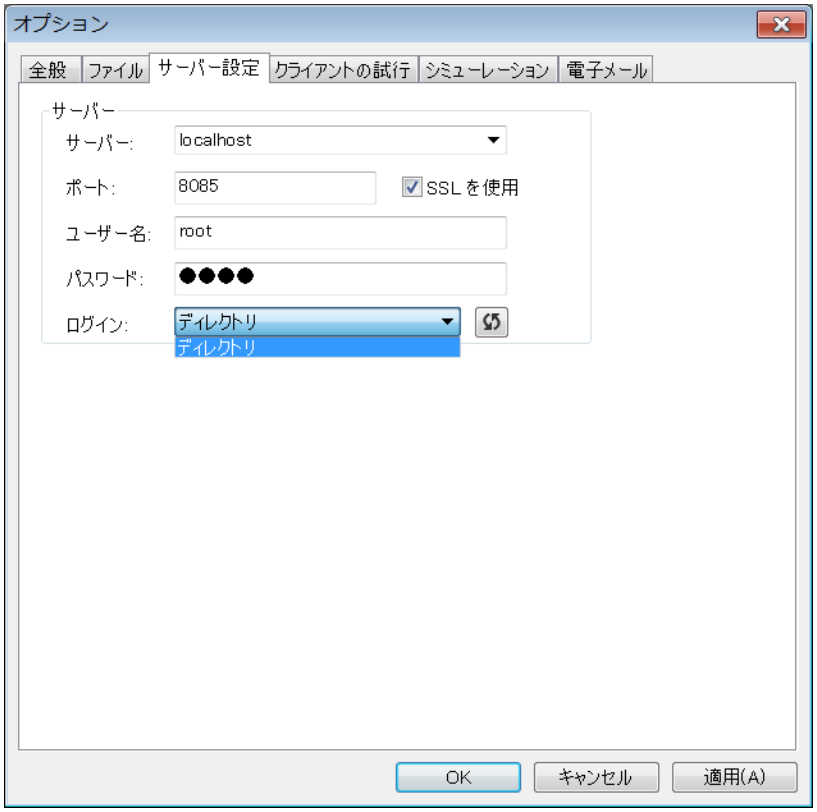
ファイルタブでは (*下のスクリーンショット*) スクリーンショットに表示されている通り設定を定義することができます。



- *プロジェクト*: 選択された場合、プログラムの開始時、プログラムの前回終了時開かれていたすべてのプロジェクトが開かれます。
- *変更されたファイルの再ロード*: 他のユーザにより加えられた変更を確認し、再ロードするか確認するオプション。ファイルが再ロードされた場合、最後の保存時からの変更が失われます。
- *プロジェクトの検証*: プロジェクトの検証の実行。オプションを選択してください。
- *ソース構造の再ロード*: ページソース構造の再ロード。オプションを選択してください。*修正前に確認* オプションにより、1つまたは複数の他のページにより共有されたページソース構造の修正の前に確認することができます。例えば、このオプションが選択されており、共有されているページソース構造に修正が加えられると、ユーザーは次のオプションから選択するように問われます：(i) 共有された構造がすべての発生の際に(すなわち、発生するすべてのページで)修正される(ii) データ構造のデータのコピーがこのページのために異なる名前で保存されるか、このデータ構造が他のページ内のデータ構造に影響を与えることなくデータ構造を編集する(iii) 編集をキャンセルする。
- *キャッシュの検証*: キャッシュの検証の指定。オプションを選択してください。

サーバー設定

サーバー設定タブ(*下のスクリーンショット*)により、MobileTogether Designer に接続するための MobileTogether Server の接続と認証の設定を定義することができます。これらの設定はソリューションがサーバーにデプロイされる時 および、サーバーがワークフローシミュレーションに使用される時に使用されます。ユーザーは対応する MobileTogether Server の権限を有する必要があります: *デザイナーからワークフローを保存する* および *サーバー シミュレーションの実行。* ユーザーの MobileTogether Server へのアクセス権限は MobileTogether Server の Web UI で定義されています。詳細は 「MobileTogether Server ユーザー マニュアル」 を参照してください。

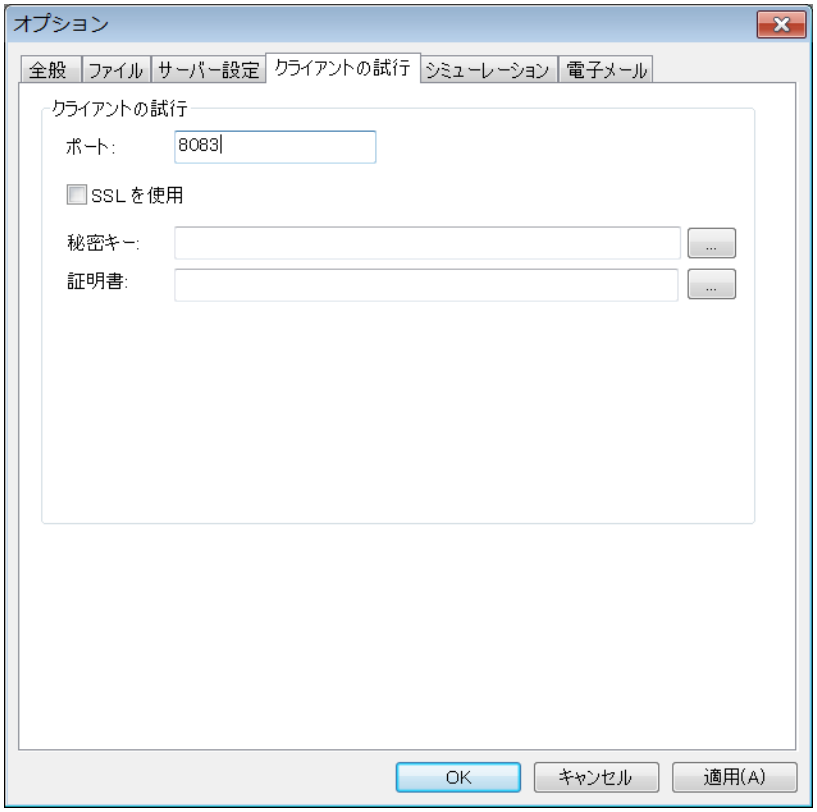


MobileTogether Server のアクティブ ディレクトリログインの機能 がドメインユーザーに対して有効化されている場合、MobileTogether Designer からMobileTogether Server へ接続するログイン情報は、ドメインの資格情報です。例えば、オフィスネットワークのWindows ユーザー名とパスワードが、MobileTogether Server 認証の使用を有効化している場合、ドメイン固有のユーザー名およびパスワードを入力することができます。

MobileTogether Server に直接指定されているユーザー資格情報を選択するか、または、ドメイン固有のユーザー資格情報を使用するかは、ログインコンボボックスから適切なオプションを選択してください(*上のスクリーンショット*)。コンボボックスの横のボタンはMobileTogether Server との接続を更新するためのボタンです。

クライアントの トライアル ラン

クライアントのトライアルランタブで(*下のスクリーンショット*)、MobileTogether Designer がクライアントと接続するローカルポートを定義することができます。クライアントのトライアルラン機能では、MobileTogether Designer は MobileTogether Server として機能し、デザインと関連したデータファイルを直接クライアントにサーブします。



- *SSL:* SSL が使用されているかどうか。
- *秘密キー*、*証明書*: (SSL が使用されている場合) SSL 秘密キーと証明書を参照します。

シミュレーション

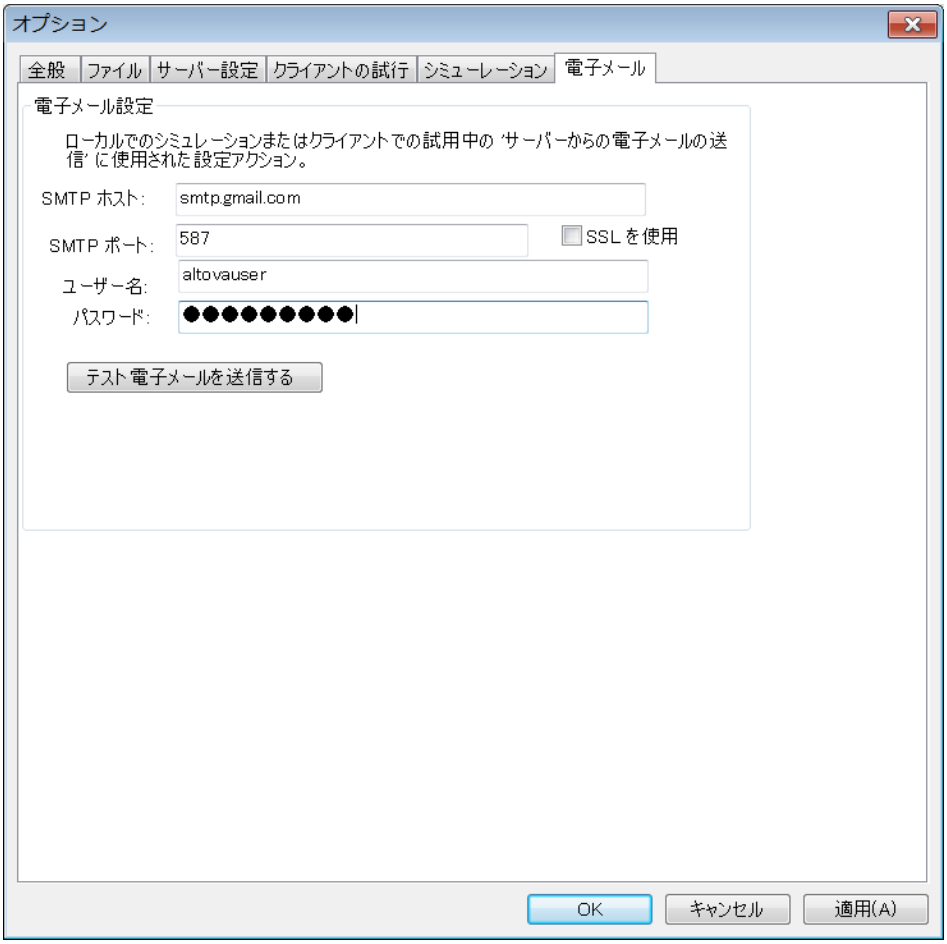
シミュレーションタブで (*下のスクリーンショット*)、シミュレーションに使用されるユーザー名とロール、および再帰的なサブページで警告が発行されるか、また、コメントと処理の指示がシミュレーターの XML データツリーで表示するかを指定できます。

- *シミュレーションのチェック| 再帰的サブページ:* 再帰的サブページに関しての警告が発行されるか指定します。
- *サービストリガーのためのダイアログの表示:* サービスシミュレーションの開始時.ダイアログの表示をオンとオフに切り替えます。
- *シミュレーションする言語:* ローカライズのオプションに対するヒントです。プロジェクトはローカライズされることができ、プロジェクト内のテキスト文字列をローカライズダイアログ内で翻訳することができます ([**プロジェクト| ローカライズ**])。プロジェクトが複数の言語にローカライズされている場合、シミュレーションする言語を[**プロジェクト| シミュレーションする言語**] コマンドのサブメニューから選択することができます。

- *XML データツリー:* コメントおよび処理命令がシミュレーターのXML データツリー内で表示されるかを指定します。
- *クライアントデバイス上のファイル:* デザインがクライアントデバイス上のファイルを参照する場合、これらのファイルはシミュレーション中にアクセスすることはできません。シミュレーション中は、このオプション内で指定されたオプションがクライアント側のファイルのために検索されます。クライアント側のファイルが、デザイン内で参照されたときと同じ名前でここに保存されている場合、シミュレーション中に正確にアクセスする事ができます。ます。
- *オプション:* ここで指定される設定はシミュレーターのデフォルトの設定を決定します。シミュレーターが開始した後、これらの設定をシミュレータのツールバー内、または、**[シミュレーション]** メニューで変更する事ができます。WiFi、LAN、またはモバイルネットワーク接続によりシミュレーションを開始するかを選択することができます。デフォルトでは、他の2つの設定はオフですが、**[*WiFi* オプションがオンの場合をシミュレート]** はオンに設定されています。(シミュレーター内では、オプションを随時変更することができます。) **[デフォルトの設定]** ボタンは、デフォルトの設定にオプションをリセットします。

電子メール

電子メール タブ内の設定は、電子メールサービスプロバイダー(通常使用中のISP) のSMTP サーバーにアクセスするために、のために、ローカルシミュレーション中に使用されます。サーバーを介して、エンドユーザーからの電子メールの送信を有効化する (サーバーからの) 電子メールの送信 アクションにより使用されます。リアルタイムのシナリオでは、SMTP サーバーへのアクセスは、MobileTogether Server 内で構成されます。しかし、ローカルシミュレーション中には、SMTP サーバー情報は、使用することができません(これは、ローカルシミュレーション中 MobileTogether Server にアクセスされないからです)。ローカルシミュレーションのための、SMTP サーバー設定はですからこのタブに入力されます (*下のスクリーンショット*)。



- *SMTP ホスト および SMTP ポート:* これらは、SMTP ホスト名 および ISP の SMTP サーバーの SMTP ポートです。これらの詳細は、使用中の ISP により提供されます。
- *ユーザー名 および パスワード:* 電子メールアカウントの、ユーザー名 および パスワードは、電子メールのサービスプロバイダーに登録されています。

詳細を入力した後、[**OK**] をクリックします。電子メールを送信して、設定が正確に作動しているか確認することができます。

20.8 ウィンドウ

[ウィンドウ] メニューは次のコマンドを含みます:

- 重ねて表示
- 上下に並べて表示
- 左右に並べて表示
- 閉じる
- すべて閉じる
- 非アクティブを全て閉じる
- 現在開かれているウィンドウのリスト

(C) 2018 Altova GmbH

重ねて表示と並べて表示

「ウィンドウ」 メニューには、MobileTogether Designer ウィンドウがどのようにGUI で表示 (重ねて表示、並べて表示、または最大化して表示) されるかを指定するコマンドがあります。ウィンドウを最大化するには、そのウィンドウの **「最大化」** ボタンをクリックします。

閉じる、全て閉じる、非アクティブを全て閉じる

閉じる

説明

アクティブなドキュメントウィンドウを閉じる。ファイルが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

すべて閉じる

説明

すべての開かれているドキュメントウィンドウが閉じられます。ドキュメントが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

非アクティブをすべて閉じる

説明

アクティブなドキュメントウィンドウ以外のすべての開かれているドキュメントウィンドウを閉じます。ドキュメントが変更されると、(タイトルバーのファイル名の後にアスタリスク * が表示され) ファイルを最初に保存するか問われます。

(C) 2018 Altova GmbH

現在開かれているウィンドウのリスト

現在開かれているすべてのドキュメントウィンドウは、**「ウィンドウ」** メニューの下に、アクティブなウィンドウにチェック付きで、ドキュメント名別にリストされています。他のウィンドウをアクティブにするには、アクティブにするウィンドウの名前をクリックしてください。このリストは現在開かれているすべてのウィンドウを表示し、ウィンドウ間で素早く切り替えることができます。また、**CTRL+TAB** または **CTRL+F6** キーボードショートカットを使用して、開かれているウィンドウ間を循環することができます。

ウィンドウ ダイアログ

開かれているウィンドウの下のリストが **「ウィンドウ」** ダイアログのエントリです。このエントリをクリックすることで、すべての開かれているウィンドウのリストと選択されたウィンドウへ適用することのできるコマンドを与える「ウィンドウ」ダイアログが開かれます。(名前をクリックすることでウィンドウを選択することができます。)

警告: 「ウィンドウ」ダイアログを終了するには、**「OK」** をクリックします。**「ウィンドウを閉じる」** ボタンをクリック**しない**でください。**「ウィンドウを閉じる」** ボタンはウィンドウダイアログで現在選択されているウィンドウを閉じます。

20.9 ヘルプ

[ヘルプ] メニューには、ヘルプの使用または、MobileTogether Designer についての詳細を得るためのコマンド、および情報とAltova Web サーバー上のサポートページへのリンクがあります。**[ヘルプ]** メニューにはまた、製品を購入するとライセンスキーコードを入力することのできる[登録] ダイアログがあります。

[ヘルプ] メニューは次のコマンドを含みます:

- 目次
- インデックス
- 検索
- ソフトウェアのアクティベーション
- オーダーフォーム
- 登録
- 最新情報のチェック
- サポート センター
- デモビデオを観る
- インターネット上のMobileTogether Designer
- MobileTogether Designer について

目次、インデックス、検索

- 目次
  - *説明*

    ヘルプウィンドウの左側のペインに目次を表示した、MobileTogether Designer の画面上のヘルプマニュアルを開きます。目次はヘルプドキュメント全体の概要を表示しています。目次のエントリをクリックしてトピックに移動することができます。

- インデックス
  - *説明*

    ヘルプウィンドウの左側のペインにキーワード インデックスを表示したMobileTogether Designer の画面上のヘルプマニュアルを開きます。目次はヘルプドキュメント全体の概要を表示しています。インデックスはキーワードをリストし、キーワードをダブルクリックすることでトピックへ移動することができます。キーワードが1つ以上のトピックにリンクされている場合は、トピックのリストが表示されます。

- 検索
  - *説明*

    ヘルプウィンドウの左側のペインに検索ダイアログを表示したMobileTogether Designer の画面上のヘルプマニュアルを開きます。単語を検索するには、入力フィールドに検索対象を入力して、「**Return**」を押します。ヘルプシステムは、ヘルプドキュメント全体で全文検索を行いヒットしたリストを返します。アイテムを表示するためにはアイテムをダブルクリックします。

アクティベーション、注文フォーム、登録、最新情報のチェック

- ライセンス登録
  - *説明*

    Altova 製品ソフトウェアをダウンロードすると、無料評価キーまたは購入されたライセンスキーを使用して、製品のアクティベーションを行うことができます。

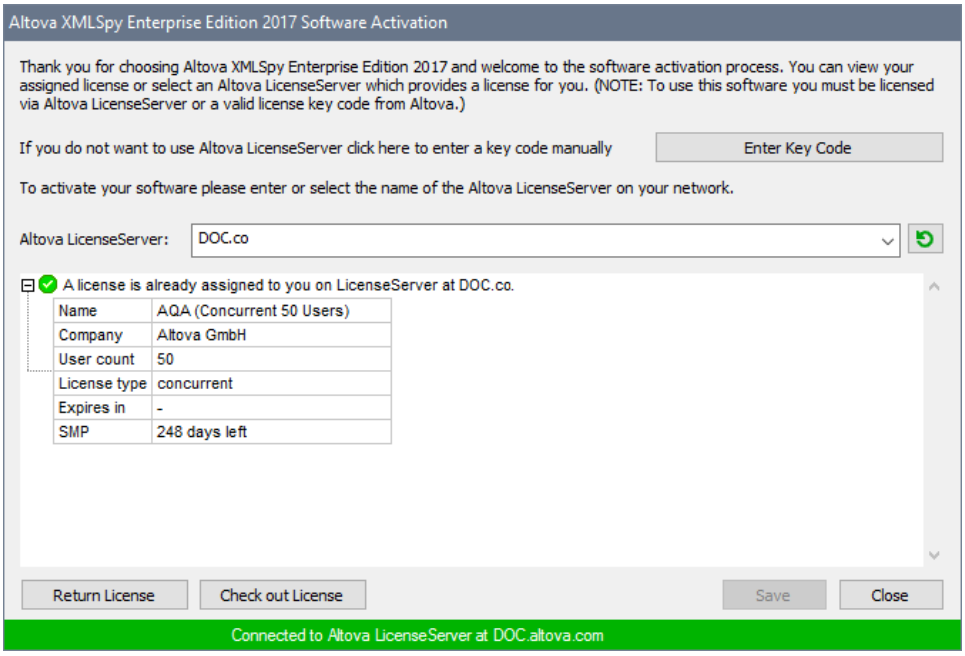
    - ***無料評価キー：***初めて製品のダウンロードとインストールを行うと、ソフトウェアアクティベーションダイアログが表示されます。ダイアログでは無料評価キーコードをリクエストすることができます。ユーザーの名前、所属会社名、そして電子メールアドレスを入力して、無料評価キーを取得するボタンをクリックしてください。入力された電子メールアドレスに評価キーが送信されます。ソフトウェアアクティベーションのダイアログボックスにて送信されたキーコードを入力し、**「OK」**をクリックすることで、Altova 製品を使用することができるようになります。ソフトウェアは30日の間アンロックされます。
    - ***購入されたライセンスキー：***ソフトウェアアクティベーションダイアログにはライセンスキーのキーコードを購入するボタンが含まれています。このボタンをクリックするとAltova のオンラインショップが表示され、製品に対応したライセンスキーを購入することができます。ライセンスにはシングルユーザーとマルチユーザーという二種類のライセンスがあります。両種類のキーコードが電子メールにより送信されます。シングルユーザーライセンスにはライセンスデータや組織名、そしてキーコードが含まれています。ライセンス契約では、ライセンスされている数を超えて組織内のコンピューターにAltova 製品をインストールできないということに注意してください。電子メール内に含まれているライセンス情報を、登録ダイアログに正確に入力するように注意してください。

    メモ ソフトウェアアクティベーションダイアログにてライセンス情報を入力する際には、ライセンス電子メールに記された情報を正確に入力するようにしてください。マルチユーザーライセンスの場合、Name フィールドに、各ユーザーの名前を入力してください。

    ソフトウェアアクティベーションダイアログ（*下のスクリーンショット*）は、**「ヘルプ | ソフトウェア アクティベーション」**をクリックすることにより常にアクセスすることができます。

    以下の方法によりソフトウェアをアクティブ化することができます:

    - （**「新しいキーコードを入力する」**をクリックして）ライセンスキーの情報を入力します。または、
    - （**「Altova LicenseServer を使用」**をクリックして）ネットワーク上のAltova LicenseServer からライセンスを取得します。Altova LicenseServer は、使用中のAltova 製品のためのライセンスをライセンスプール上に有する必要があります。ライセンスがLicenseServer プール内に存在する場合、Software Activation ダイアログ内に表示されます（*下のスクリーンショット*）。**「保存」**をクリックしてライセンスを取得します。



ライセンスが1度取得されると、7日間は、LicenseServer に戻すことができません。7日過ぎると、(**ライセンスを戻す**」をクリックして) ライセンスを戻すことができ、ライセンスは、他のクライアントから取得することができます。LicenseServer 管理者は、LicenseServer のWeb UI を使用して、取得されたライセンスの割り当てを解除することができます。

ライセンスのチェックアウト

ライセンスが製品マシン上に保管されるように、ライセンスをライセンスプールから30 日間チェックアウトすることができます。これにより、オフラインで作業することが可能になります。これはとても役に立ちます。Altova LicenseServer にアクセスできない環境(例えば、旅行中にAltova 製品がインストールされたラップトップコンピューターで作業する)場合などが挙げられます。ライセンスはチェックアウトされていますが、LicenseServer は、ライセンスが使用中と表示し、ライセンスは他のマシンで使用することができません。ライセンスはチェックアウトの期間が終わると自動的にチェックインされた状態を戻します。 または、チェックアウトされたライセンスはソフトウェアのライセンスの認証ダイアログのボタンを使用して**チェックイン**することができます。

ライセンスをチェックアウトするには以下をおこないます: (i) ソフトウェアのライセンスの認証ダイアログで **ライセンスのチェックアウト** をクリックします (*上のスクリーンショット参照*)。(ii) ライセンスのチェックアウトダイアログ内から、チェックアウトの期間を選択し、**チェックアウト**をクリックします。ライセンスがチェックアウトされます。ソフトウェアのライセンスの認証ダイアログは、チェックアウト期間の終わりを含むチェックアウトの情報を表示します。ダイアログ内の**ライセンスのチェックアウト** ボタンは、**チェックイン**ボタンに変わります。**チェックイン**ボタンをクリックして、ライセンスをチェックインすることができます。ライセンスは自動的にチェックイン状態に戻されるため、選択したチェックアウトの期間がオフラインで作業する期間をカバーするように確認してください。

Altova LicenseServer を使用することにより、IT 管理者は、リアルタイムでネットワーク上の全てのライセンスの概要、および、クライアントの割り当てと、クライアントのライセンスの使用状況を確認することができます。LicenseServer を使用する利点は、ですから、多数のAltova ライセンスを管理することのできる管理機能です。 Altova LicenseServer は、Altova Web サイトで無料で提供されています。Altova LicenseServer およびAltova LicenseServer を使用したライセンスの供与に関する詳細は、Altova LicenseServer ドキュメントを参照してください。

- 注文フォーム
  - *説明*

    ソフトウェア製品のライセンスを購入する準備が整った場合、ソフトウェアアクティベーションダイアログにて **「キーコードを購入する」** ボタンを選択するか、**「ヘルプ | 注文フォーム」** コマンドを選択して、Altova オンラインショップへアクセスすることができます。

- 登録
  - *説明*

    Altova 製品登録ページをブラウザーのタブに表示します。Altova ソフトウェアを登録することにより、最新の製品の情報が得られます。

- 最新情報のチェック
  - *説明*

    Altova サーバーに接続して、より新しいバージョンの製品が利用可能かどうかチェックし、その結果を表示します。

他のコマンド

- サポートセンター
  - *説明*

    インターネット上にあるAltova サポートセンターへのリンクとなっています。サポートセンターにはFAQ やディスカッションフォーラムが含まれており、問題の解決方法を探ったり、Altova の技術サポートスタッフへアクセスすることができます（現在英語のみの提供となります）。

- デモビデオの表示
  - *説明*

    Altova Web サイト上のMobileTogether Designer デモビデオのページにリンクされます。このページのビデオでは、MobileTogether Designer の使用方法および使用開始について説明されています。

- インターネット上の MobileTogether Designer
  - *説明*

    インターネット上にあるAltova Web サイトへのリンクとなっています。Altova Web サイトでは、MobileTogether Designer や関連するテクノロジーについて確認することができます。

- MobileTogether Designer について
  - *説明*

    スプラッシュ画面と製品のバージョン番号が表示されます。

チャプター 21

よくある質問

21 よくある質問

プロジェクトで ODBC を介して MySQL を使用しています。MobileTogether Server にソリューションをデプロイして、実行すると、次のエラーが表示されます: データベース: [Microsoft][ODBC ドライバーマネジャー] データソース名が見つかりませんまたデフォルトのドライバーが指定されていません ...データベースからの取得にエラーが発生しました。

MobileTogether Server は、Windows サービスにインストールされていると考えます。このため、デフォルトは Windows アカウントでは実行されません。この問題を解決するには次のオプションがあります:

- データソース名 (DSN) がユーザー DSN として定義されています。Windows コントロールパネルを使用して、システム DSN を移動/再作成します。必要であれば、MobileTogether Designer 内のやり直し、再度デプロイします。
- MobileTogether Server サービスを Windows ユーザーアカウントに割り当てます。 MobileTogether Server のインストール後、Windows サービスアプリを使用して、ユーザーアカウントを割り当てます。
- MobileTogether のグローバルリソースメカニズムは異なる接続を使用します。または、異なるデータベースを MobileTogether Server と MobileTogether Designer で使用します。

Sybase DB (Sybase ASE ODBC Driver 4.10.00.00) に対して、ODBC 接続を使用しています。レジストリまたは ファイル DSN にユーザー名およびパスワードが保管されていないようですが、なぜでしょう?

IODBC-管理者を使用してデータソースを作成した場合、ユーザー名とパスワードはレジストリまたはファイル DSN に保管されません。 MobileTogether Server へ接続するためには、ユーザー名とパスワードを入力するポップアップは表示されず、この情報を手動で入力する必要があります。この手順は以下のために行われる必要があります: システム DSN、(REG_SZ としての) User DSN および File DSN。

- `Name: UID  Value: <Your UserID>`
- `Name: PWD  Value: <Your Password>`

Web ブラウザーをクライアントとして使用した場合、永続的データはどこに保管されるのですか?

Web クライアントに保存されるデータは、ブラウザーの (ストレージとして知られる) ローカル ストレージに保存されます。HTML 5.0 ローカルストレージは以下のブラウザーでサポートされます:

IE 8.0 +	Firefox 3.5+	Safari 4.0+	Chrome 4.0+	Opera 10.5+	iPhone 2.0+	Android 2.0+

チャプター 22

参照

22 参照

このセクションに含まれる情報:

- XSLT とXPath/XQuery 関数
- ライセンス情報

(C) 2018 Altova GmbH

22.1 XSLT と XPath/ XQuery 関数

このセクションでは、XPath および/または XQuery 式で使用することのできる、Altova 拡張関数と他の拡張関数をリストします。Altova 拡張関数は Altova の XSLT および XQuery エンジンで使用することができ、W3C 標準で定義された関数ライブラリで使用することのできる機能に追加して機能を提供します。

一般的な情報

以下の一般的な情報に注意してください:

- W3C 仕様により定義されているコア関数ライブラリの関数は、関数の呼び出しにプレフィックスは必要ありません。これは、XSLT および XQuery エンジンが、XPath/XQuery 関数仕様で指定されている `http://www.w3.org/2005/xpath-functions` プレフィックス無しの関数をデフォルト関数の名前空間に属するものとして読み込むためです。
- 関数において、各アイテムが引数となるようなシーケンスが期待されており、2つ以上のアイテムがシーケンスにより呼び出された場合、エラーが返されます。
- 全ての比較は Unicode コードポイントコレクションを使用することで行われます。
- QName の結果は `[prefix:]localname` という形式でシリアル化されます。

xs:decimal の精度

精度とは、数値内にある桁数のことで、仕様では少なくとも18桁が求められます。`xs:decimal` 型に結果が収められる除算の場合、端数処理を行うことなく精度は小数点以下の19桁になります。

黙示的なタイムゾーン

2つの `date`、`time`、または `dateTime` 値を比較する場合、比較する値のタイムゾーンを明らかにする必要があります。値の中にタイムゾーンが明示的に与えられていない場合、黙示的なタイムゾーンが使用されます。黙示的なタイムゾーンはシステムクロックから取得され `implicit-timezone()` 関数によりその値をチェックすることができます。

照合順序

デフォルトの照合順序は、Unicode コードポイントをベースに文字列を比較する Unicode コードポイント照合順序です。エンジンは Unicode 照合アルゴリズムを使用しています。他のサポートされる照合順序は下にリストされる ICU 照合順序 です。使用するには、サポートされる照合順序のリストの URI を提供してください(下のテーブル)。`max` と `min` 関数を含む、文字列の比較は、指定された照合順序に沿って行われます。照合順序オプションが指定されていない場合、デフォルトの Unicode コードポイント照合順序が使用されます。

言語	URI
da: デンマーク語	da_DK
de: ドイツ語	de_AT, de_BE, de_CH, de_DE, de_LI, de_LU
en: 英語	en_AS, en_AU, en_BB, en_BE, en_BM, en_BW, en_BZ, en_CA, en_GB, en_GU, en_HK, en_IE, en_IN, en_JM, en_MH, en_MP, en_MT, en_MU, en_NA, en_NZ, en_PH, en_PK, en_SG, en_TT, en_UM, en_US, en_VI, en_ZA, en_ZW
es: スペイン語	es_419, es_AR, es_BO, es_CL, es_CO, es_CR, es_DO, es_EC, es_ES, es_GQ, es_GT, es_HN, es_MX, es_NI, es_PA, es_PE, es_PR, es_PY, es_SV, es_US, es_UY, es_VE
fr: フランス語	fr_BE, fr_BF, fr_BI, fr_BJ, fr_BL, fr_CA, fr_CD,

	fr_CF, fr_CG, fr_CH, fr_CI, fr_CM, fr_DJ, fr_FR, fr_GA, fr_GN, fr_GP, fr_GQ, fr_KM, fr_LU, fr_MC, fr_MF, fr_MG, fr_ML, fr_MQ, fr_NE, fr_RE, fr_RW, fr_SN, fr_TD, fr_TG
it: イタリア語	it_CH, it_IT
ja: 日本語	ja_JP
nb: ノルウェー語 (ブークモール)	nb_NO
nl: オランダ語	nl_AW, nl_BE, nl_NL
nn: ノルウェー語 (ニーノシュク)	nn_NO
pt: ポルトガル語	pt_AO, pt_BR, pt_GW, pt_MZ, pt_PT, pt_ST
ru: ロシア語	ru_MD, ru_RU, ru_UA
sv: スウェーデン語	sv_FI, sv_SE

名前空間軸
名前空間軸は XPath 2.0 にて廃止されましたが、名前空間軸の使用はサポートされています。XPath 2.0 メカニズムにより名前空間情報へアクセスするには、in-scope-prefixes()、namespace-uri()、namespace-uri-for-prefix(関数を使用してください。

Altova 拡張関数

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]** を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)。**

XPath 関数 (XSLT 内のXPath 式で使用):	XP1 XP2 XP3.1
XSLT 関数 (XSLT 内のXPath 式で使用):	XSLT1 XSLT2 XSLT3
XQuery 関数 (XQuery 内のXQuery 式で使用):	XQ1 XQ3.1

XPath/XQuery 関数

XPath/XQuery 関数は、XQuery 関数のXPath 式で使用することができます:

- 日付 時刻
- 位置情報
- イメージに関連した
- 数値
- シーケンス
- 文字列
- その他

XPath/ XQuery 関数: 日付と時刻

Altova の日付/時刻拡張関数は XPath とXQuery 式で使用することができ、XML スキーマの異なる日付および時刻データ型で保存されているデータを処理するための追加機能を提供します。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)。**

XPath 関数 (XSLT 内のXPath 式で使用):	XP1 XP2 XP3.1
XSLT 関数 (XSLT 内のXPath 式で使用):	XSLT1 XSLT2 XSLT3
XQuery 関数 (XQuery 内のXQuery 式で使用):	XQ1 XQ3.1

機能によりグループ化

- xs:dateTime に期間を追加して、xs:dateTimeを返す
- xs:date に期間を追加して、xs:date を返す
- xs:time に期間を追加して、return xs:time を返す
- フォーマットと期間の取得
- 現在の日付/時刻を生成する関数からタイムゾーンを削除する
- 日付から整数を週の曜日として返す
- 日付から周数を整数として返す
- 各型の構文コンポーネントから日付、時刻、期間 の型を構築する
- 文字列入力から日付、日付時刻 または時刻 を構築する
- 年齢に関連した関数

アルファベット順にグループ化

altova:add-days-to-date
altova:add-days-to-dateTime
altova:add-hours-to-dateTime
altova:add-hours-to-time
altova:add-minutes-to-dateTime
altova:add-minutes-to-time
altova:add-months-to-date
altova:add-months-to-dateTime
altova:add-seconds-to-dateTime
altova:add-seconds-to-time
altova:add-years-to-date
altova:add-years-to-dateTime
altova:age
altova:age-details
altova:build-date
altova:build-duration
altova:build-time
altova:current-dateTime-no-TZ
altova:current-date-no-TZ
altova:current-time-no-TZ
altova:format-duration
altova:parse-date
altova:parse-dateTime

altova:parse-duration
altova:parse-time
altova:weekday-from-date
altova:weekday-from-dateTime
altova:weeknumber-from-date
altova:weeknumber-from-dateTime

[トップ]

---

xs:dateTime に期間を追加する XP3.1 XQ3.1

これらの関数は**xs:dateTime** に期間を追加し、**xs:dateTime** を返します。xs:dateTime 型はCCYY-MM-DDThh:mm:ss.sss のフォーマットです。これはxs:date とxs:time フォーマットの連結で、T により区切られています。タイムゾーンサフィックス+01:00 (for example)は任意です。

add-years-to-dateTime [altova:]

add-years-to-dateTime(DateTime *as xs:dateTime*, **Years** を *xs:integer*) as **xs:dateTime** とする XP3.1 XQ3.1

日付までの期間を年数で表示します。第 2 の引数は第 1 の引数として与えられたxs:date に追加される年数です。結果はxs:date 型です。

サンプル

- **add-years-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), 10) returns 2024-01-15T14:00:00
- **add-years-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), -4) returns 2010-01-15T14:00:00

add-months-to-dateTime [altova:]

add-months-to-dateTime(DateTime *as xs:dateTime*, **Months** を *xs:integer*) as **xs:dateTime** とする XP3.1 XQ3.1

xs:dateTime に月数での期間を追加します (*下のサンプル参照*)。第 2 の引数は、第 1 の引数として与えられたxs:dateTime に追加される月数です。結果はxs:dateTime 型です。

サンプル

- **add-months-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), 10) 2014-11-15T14:00:00 を返します。
- **add-months-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), -2) 2013-11-15T14:00:00 を返します。

add-days-to-dateTime [altova:]

add-days-to-dateTime(DateTime *as xs:dateTime*, **Days** *as xs:integer*) を **xs:dateTime** とする XP3.1 XQ3.1

xs:dateTime に日数での期間を追加します (*下のサンプル参照*)。第 2 の引数は、第 1 の引数として与えられたxs:dateTime に追加される日数です。結果はxs:dateTime 型です。

サンプル

- **add-days-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), 10)は2014-01-25T14:00:00 を返します。
- **add-days-to-dateTime**(xs:dateTime("2014-01-15T14:00:00"), -8)は2014-01-

25T14:00:00 を返します。

add-hours-to-dateTime [altova:]

add-hours-to-dateTime(DateTime *as xs:dateTime*, **Hours** *as xs:integer*) を **xs:dateTime** とする **XP3.1 XQ3.1**

xs:dateTime に時間数での期間を追加します (*下のサンプル参照*)。第 2 の引数は、第 1 の引数として与えられた xs:dateTime に追加される時間数です。結果は xs:dateTime 型です。

サンプル

- **add-hours-to-dateTime**(xs:dateTime("2014-01-15T13:00:00"), 10) は 2014-01-15T23:00:00 を返します。
- **add-hours-to-dateTime**(xs:dateTime("2014-01-15T13:00:00"), -8) は 2014-01-15T05:00:00 を返します。

add-minutes-to-dateTime [altova:]

add-minutes-to-dateTime(DateTime *as xs:dateTime*, **Minutes** *as xs:integer*) を **xs:dateTime** とする **XP3.1 XQ3.1**

xs:dateTime に分数での期間を追加します (*下のサンプル参照*)。第 2 の引数は、第 1 の引数として与えられた xs:dateTime に追加される分数です。結果は xs:dateTime 型です。

サンプル

- **add-minutes-to-dateTime**(xs:dateTime("2014-01-15T14:10:00"), 45) 2014-01-15T14:55:00 を返します。
- **add-minutes-to-dateTime**(xs:dateTime("2014-01-15T14:10:00"), -5) 2014-01-15T14:05:00 を返します。

add-seconds-to-dateTime [altova:]

add-seconds-to-dateTime(DateTime *as xs:dateTime*, **Seconds** を *xs:integer*) as **xs:dateTime** とする **XP3.1 XQ3.1**

xs:dateTime に秒数での期間を追加します (*下のサンプル参照*)。第 2 の引数は、第 1 の引数として与えられた xs:dateTime に追加される秒数です。結果は xs:dateTime 型です。

サンプル

- **add-seconds-to-dateTime**(xs:dateTime("2014-01-15T14:00:10"), 20) 2014-01-15T14:00:30 を返します。
- **add-seconds-to-dateTime**(xs:dateTime("2014-01-15T14:00:10"), -5) 2014-01-15T14:00:05 を返します。

(C) 2018 Altova GmbH

[トップ]

---

xs:date に期間を追加する XP3.1 XQ3.1

これらの関数は**xs:date** に期間を追加し、**xs:date** を返します。xs:date 型はCCYY-MM-DD フォーマットです。

add-years-to-date [altova:]

add-years-to-date(Date *as xs:date*, **Years** *as xs:integer*) を**xs:date** とする
XP3.1 XQ3.1
日付までの期間を年数で表示します。第 2 の引数は第 1 の引数として与えられたxs:date に追加される年数です。結果はxs:date 型です。

サンプル

- **add-years-to-date**(xs:date("2014-01-15"), 10) は2024-01-15 を返します。
- **add-years-to-date**(xs:date("2014-01-15"), -4) は2010-01-15 を返します。

add-months-to-date [altova:]

add-months-to-date(Date *as xs:date*, **Months** *as xs:integer*) を**xs:date** とする
XP3.1 XQ3.1
日付までの期間を月数で表示します。第 2 の引数は第 1 の引数として与えられたxs:date に追加される月数です。結果はxs:date 型です。

サンプル

- **add-months-to-date**(xs:date("2014-01-15"), 10) 2014-11-15 を返します。
- **add-months-to-date**(xs:date("2014-01-15"), -2) 2013-11-15 を返します。

add-days-to-date [altova:]

add-days-to-date(Date *as xs:date*, **Days** *as xs:integer*) を**xs:date** とする
XP3.1 XQ3.1
日付までの期間を日数で表示します。第 2 の引数は第 1 の引数として与えられたxs:date に追加される日数です。結果はxs:date 型です。

サンプル

- **add-days-to-date**(xs:date("2014-01-15"), 10)は 2014-01-25 を返します。
- **add-days-to-date**(xs:date("2014-01-15"), -8) は2014-01-07 を返します。

[トップ]

フォーマットと期間の取得 XP3.1 XQ3.1

これらの関数は**xs:date** に期間を追加し、**xs:date** を返します。xs:date 型はCCYY-MM-DD フォーマットです。

format-duration [altova:]

format-duration(Duration *as xs:duration*, **Picture** *as xs:string*) as **xs:string** とする XP3.1 XQ3.1

第 1 の引数として提出された期間を、第 2 の引数として提出された文字列によりをフォーマットします。出力は、文字列によりフォーマットされたテキスト文字列です。

サンプル

- **format-duration**(xs:duration("P2DT2H53M11.7S"), "Days:[D01] Hours:[H01] Minutes:[m01] Seconds:[s01] Fractions:[f0]") は"Days:02 Hours:02 Minutes:53 Seconds:11 Fractions:7" を返します。
- **format-duration**(xs:duration("P3M2DT2H53M11.7S"), "Months:[M01] Days:[D01] Hours:[H01] Minutes:[m01]") は"Months:03 Days:02 Hours:02 Minutes:53" を返します。

parse-duration [altova:]

parse-duration(InputString *as xs:string*, **Picture** *as xs:string*) を**xs:duration** とする XP3.1 XQ3.1

パターン化された文字列を最初の引数として、文字を第 2 の引数とします。入力文字列は文字をベースに解析され、xs:duration が返されます。

サンプル

- **parse-duration**("Days:02 Hours:02 Minutes:53 Seconds:11 Fractions:7", "Days:[D01] Hours:[H01] Minutes:[m01] Seconds:[s01] Fractions:[f0]") は "P2DT2H53M11.7S" を返します。
- **parse-duration**("Months:03 Days:02 Hours:02 Minutes:53 Seconds:11 Fractions:7", "Months:[M01] Days:[D01] Hours:[H01] Minutes:[m01]") は "P3M2DT2H53M" を返します。

[Top]

xs:time に期間を追加する XP3.1 XQ3.1

これらの関数は**xs:time** に期間を追加し、 **xs:time** を返します。 **xs:time** 型はhh:mm:ss.sss 構文フォームです。

文字 Z は協定世界時 (UTC) を表します。他のタイムゾーンは、UTC との差異を+hh:mm または -hh:mm のフォーマットで表示しています。タイムゾーンの値が無い場合は、UTC ではない未知のタイムゾーンとして見なされます。

add-hours-to-time [altova:]

add-hours-to-time(Time *as xs:time*, **Hours** *as xs:integer*) を**xs:time** とする XP3.1 XQ3.1

日付までの期間を時間数で表示します。第 2 の引数は第 1 の引数として与えられたxs:time に追加される時間数です。結果はxs:time 型です。

サンプル

- **add-hours-to-time**(xs:time("11:00:00"), 10) は21:00:00 を返します。
- **add-hours-to-time**(xs:time("11:00:00"), -7) は04:00:00 を返します。

add-minutes-to-time [altova:]

add-minutes-to-time(Time *as xs:time*, **Minutes** *as xs:integer*) を**xs:time** とする XP3.1 XQ3.1

日付までの期間を分数で表示します。第 2 の引数は第 1 の引数として与えられたxs:date に追加される分数です。結果はxs:date 型です。

サンプル

- **add-minutes-to-time**(xs:time("14:10:00"), 45) 14:55:00 を返します。
- **add-minutes-to-time**(xs:time("14:10:00"), -5) 14:05:00 を返します。

add-seconds-to-time [altova:]

add-seconds-to-time(Time *as xs:time*, **Minutes** *as xs:integer*) を**xs:time** とする **XP3.1** **XQ3.1**

時間までの期間を秒数で表示します。第 2 の引数は、第 1 の引数として与えられた xs:time に追加される秒数です。結果は xs:time 型です。第 2 のコンポーネントは 0 から59.999 の範囲であることができます。

サンプル

- **add-seconds-to-time**(xs:time("14:00:00"), 20) は14:00:20 を返します。
- **add-seconds-to-time**(xs:time("14:00:00"), 20.895) は14:00:20.895 を返します。

[**Top**]

現在の日付 /時刻を生成する関数からタイムゾーンを削除する **XP3.1** **XQ3.1**

これらの関数は、現在の**xs:dateTime**、**xs:date**、または**xs:time** 値からそれぞれタイムゾーンを削除します。xs:dateTime とxs:dateTimeStamp の差異は、後者のタイムゾーンが必要な場合です。(前者の場合は任意です。) xs:dateTimeStamp 値のフォーマットはCCYY-MM-DDThh:mm:ss.sss±hh:mm またはCCYY-MM-DDThh:mm:ss.sssZ です。、日付と時刻がxs:dateTimeStamp としてシステムクロックから読み込まれる場合、current-dateTime-no-TZ() 関数がタイムゾーンを削除するために使用されます。

current-dateTime-no-TZ [altova:]

current-dateTime-no-TZ() を**xs:dateTime** とする **XP3.1** **XQ3.1**

この関数は引数を必要としません。current-dateTime() (システムクロックによる現在の時刻) のタイムゾーンの部分を削除し、xs:dateTime の値を返します。

サンプル

現在のdateTime が**2014-01-15T14:00:00+01:00** の場合:

- **current-dateTime-no-TZ**() は2014-01-15T14:00:00 を返します。

current-date-no-TZ [altova:]

current-date-no-TZ() を**xs:date** とする **XP3.1** **XQ3.1**

この関数は引数を必要としません。current-date() (システムクロックによる現在の時刻) のタイムゾーンの部分を削除し、xs:date の値を返します。

サンプル

現在のdate が**2014-01-15+01:00** の場合:

- **current-date-no-TZ**() は2014-01-15 を返します。

current-time-no-TZ [altova:]

current-time-no-TZ() を**xs:time** とする XP3.1 XQ3.1
この関数は引数を必要としません。current-time() (システムクロックによる現在の時刻) のタイムゾーンの部分を削除し、xs:time の値を返します。

サンプル

現在のtime が**14:00:00+01:00** の場合:

- **current-time-no-TZ**() は14:00:00 を返します。

[Top]

---

xs:dateTime または xs:date として曜日を返す XP3.1 XQ3.1
これらの関数は、xs:dateTime またはxs:date から曜日を(整数として)を返します。曜日は(米国式フォーマットを使用して) 1 から7 と番号づけられています。 日曜=1 と番号付けられます。欧州のフォーマットでは月曜 (=1) として番号付けられます。日曜=1 である米国フォーマットは整数 0 がフォーマットを表示するために使用できる箇所で設定することができます。

weekday-from-dateTime [altova:]

weekday-from-dateTime(DateTime *as xs:dateTime***)** を**xs:integer** とする XP3.1 XQ3.1

時刻付きの日付を単一の引数として、この日付の曜日を正数として返します。曜日は 日曜=1 から開始して番号を付けます。(月曜=1 とする) ヨーロッパのフォーマットが必要な場合、この関数の他の署名を使用します (*下の次の署名を参照*)。

サンプル

- **weekday-from-dateTime**(xs:dateTime("2014-02-03T09:00:00")) は月曜を表示する 2 を返します。

weekday-from-dateTime(DateTime *as xs:dateTime*, **Format** *as xs:integer***)** as **xs:integer** とする XP3.1 XQ3.1
時間月の日付を最初の引数として、この日付の曜日を正数として返します。曜日は月曜=1 から開始して番号を付けます。第 2 の引数(整数)が0 の場合、日曜=1 から開始して、曜日は1から7 と番号を付けます。第 2 の引数が0 以外の整数の場合、月曜=1です。第 2 の引数がない場合、関数はこの関数の他の署名を持つと読み込まれます (*前の次の署名を参照*)。

サンプル

- **weekday-from-dateTime**(xs:dateTime("2014-02-03T09:00:00"), 1) は月曜を表示する1 を返します。
- **weekday-from-dateTime**(xs:dateTime("2014-02-03T09:00:00"), 4) は月曜を表示する1 を返します。
- **weekday-from-dateTime**(xs:dateTime("2014-02-03T09:00:00"), 0) は月曜を表示する2 を返します。

weekday-from-date [altova:]

weekday-from-date(Date *as xs:date***)** を **xs:integer** とする XP3.1 XQ3.1
日付を単一の引数として、この日付の曜日を正数として返します。曜日は 日曜=1 から開始して番号を付けます。(月曜=1 とする) ヨーロッパのフォーマットが必要な場合、この関数の他の署名を使用します (*下の次の署名を参照*)。

サンプル

- `weekday-from-date(xs:date("2014-02-03+01:00"))` は月曜を表示する `2` を返します。

`weekday-from-date(Date as xs:date, Format as xs:integer)` を `xs:integer` とする XP3.1 XQ3.1

日付を最初の引数とし、この日付の曜日を正数として返します。曜日は月曜=`1` から開始して番号を付けます。第 2 (フォーマット) 引数が `0`,の場合、日曜=`1` から開始し、曜日は `1` から7で番号付けられます。第 2 引数が整数で `0` 以外の場合、月曜=1です。第2 の引数がない場合、関数はこの関数の他の署名を持つと読み込まれます (*前の署名を参照*)。

サンプル

- `weekday-from-date(xs:date("2014-02-03"), 1)` は月曜を示す `1` を返します。
- `weekday-from-date(xs:date("2014-02-03"), 4)` は月曜を示す `1` を返します。
- `weekday-from-date(xs:date("2014-02-03"), 0)` は月曜を示す `2` を返します。

[Top]

`xs:dateTime` または `xs:date` から週数を返す XP2 XQ1 XP3 XQ3

これらの関数は週数(整数として)を`xs:dateTime` から`xs:date`.から返します。週の番号付けは、米国、欧州、イスラムのカレンダーフォーマットで使用することができます。週の始まりが異なるため、週数の番号付けは、カレンダーのフォーマットにより異なります。(米国フォーマットでは、日曜、欧州フォーマットでは月曜、イスラムフォーマットでは土曜が週の開始日です)。

weeknumber-from-date [altova:]

`weeknumber-from-date(Date as xs:date, Calendar as xs:integer)` を`xs:integer` とする XP2 XQ1 XP3 XQ3

正数として提出された`Date` 引数の週数を返します。第 2 の引数(**カレンダー**)は続くカレンダーシステムを指定します。
サポートされる**カレンダー** の値は次のとおりです:

- **`0` = US 米国のカレンダー** (週の始まりは日曜日)
- **`1` = ISO 標準、 欧州のカレンダー** (週の始まりは月曜日)
- **`2` = イスラムのカレンダー** (週の始まりは土曜日)

デフォルトは`0` です。

サンプル

- `weeknumber-from-date(xs:date("2014-03-23"), 0)` は`13` を返します。
- `weeknumber-from-date(xs:date("2014-03-23"), 1)` は`13` を返します。
- `weeknumber-from-date(xs:date("2014-03-23"), 2)` は`13` を返します。
- `weeknumber-from-date(xs:date("2014-03-23")  )` は`13` を返します。

上のサンプル(`2014-03-23`) の`date` の曜日は日曜日です。米国およびイスラムのカレンダーは欧州カレンダーのこの日付よりも一週間先です。

weeknumber-from-dateTime [altova:]

weeknumber-from-dateTime(DateTime *as xs:dateTime*, **Calendar** *as xs:integer*) を **xs:integer** とする **XP2 XQ1 XP3 XQ3**
正数として提出された**DateTime** 引数の週数を返します。第 2 の引数(**カレンダー**)は続くカレンダーシステムを指定します。
サポートされる**カレンダー** の値は次のとおりです:

- **0 = US 米国のカレンダー** (週の始まりは日曜日)
- **1 = ISO 標準、 欧州のカレンダー** (週の始まりは月曜日)
- **2 = イスラムのカレンダー** (週の始まりは土曜日)

Default is **0**.

サンプル

- **weeknumber-from-dateTime**(xs:dateTime("2014-03-23T00:00:00"), 0) は13 を返します。
- **weeknumber-from-dateTime**(xs:dateTime("2014-03-23T00:00:00"), 1) は13 を返します。
- **weeknumber-from-dateTime**(xs:dateTime("2014-03-23T00:00:00"), 2) は13 を返します。
- **weeknumber-from-dateTime**(xs:dateTime("2014-03-23T00:00:00")) は13 を返します。

上のサンプル(2014-03-23T00:00:00) のdateTime の曜日は日曜日です。米国およびイスラムのカレンダーは欧州カレンダーのこの日付よりも一週間先です。

[トップ]

各型の構文コンポーネントから日付、時刻、期間の型を構築する XP3.1 XQ3.1

関数は、xs:date, xs:time または xs:duration の構文コンポーネントを入力引数とし、引数を結合させて対応するデータ型を構築します。

build-date [altova:]

build-date(Year *as xs:integer*, **Month** *as xs:integer*, **Date** *as xs:integer*) を **xs:date** とする **XP3.1 XQ3.1**
第 1 、第 2 、第3 の引数は、それぞれ、年、月、日を表します。xs:date 型の値を構築するため結合されます。整数の値は、特定の日付の一部の適正な範囲内である必要があります。例えば第 2 の引数 (月の部分) は12 以上であってはなりません。

サンプル

- **build-date**(2014, 2, 03) は2014-02-03 を返します。

build-time [altova:]

build-time(Hours *as xs:integer*, **Minutes** *as xs:integer*, **Seconds** *as xs:integer*) を**xs:time** とする **XP3.1 XQ3.1**
第 1 、第 2 、第3 引数はそれぞれ時間数 (0 から23)、分数 (0 から59)、および秒数 (0 から59) の値です。これらの値は、xs:time 型の値を作成するために結合されます。整数の値は、それぞれの部分の正しい範囲内である必要があります。例えば、秒 (分) 引数は、59 以上であってはなりません。値にタイムゾーンを追加するには、この関数の他の署名を使用してください(*次の署名を参照*)。

サンプル

- **build-time**(23, 4, 57) は23:04:57 を返します。

build-time(Hours *as xs:integer*, **Minutes** *as xs:integer*, **Seconds** *as xs:integer*, **TimeZone** *as xs:string*) を**xs:time** とする **XP3.1** **XQ3.1**
第 1 、第 2 、第 3 引数はそれぞれ時間数 (0 から23)、分数 (0 から59)、および秒数 (0 から59) の値です。第四の引数は、値の一部としてタイムゾーンを与えます。4 つの引数が結合され、xs:time 型の値を構築します。例えば、第 2 の引数 (分数) は59 以上であってはなりません。

サンプル

- **build-time**(23, 4, 57, '+1') は23:04:57+01:00 を返します。

build-duration [altova:]

build-duration(Years *as xs:integer*, **Months** *as xs:integer*) を **xs:yearMonthDuration** とする **XP3.1** **XQ3.1**
xs:yearMonthDuration 型の値を構築するためには2 つの引数が必要です。最初の引数は期間の年数 値の部分を与え、第 2 の引数は、月数 値の部分を与えます。第 2 の引数 (月数) が同じまたはより大きくなると、整数は12 により分割されます。 商は、年数 の部分を表す 最初の引数に加算され、(除算の) 残りは の部分は月数 の部分を表すために使用されます。期間のxs:dayTimeDuration 型を作成するには、次の署名を参照してください。

サンプル

- **build-duration**(2, 10) はP2Y10M を返します。
- **build-duration**(14, 27) はP16Y3M を返します。
- **build-duration**(2, 24) はP4Y を返します。

build-duration(Days *as xs:integer*, **Hours** *as xs:integer*, **Minutes** *as xs:integer*, **Seconds** *as xs:integer*) を**xs:dayTimeDuration** とする **XP3.1** **XQ3.1**
xs:dayTimeDuration 型の値を構築するためには4 つの引数が必要です。最初の引数は期間の日数 値の部分で、第 2 、第 3 、第 4 引数は、それぞれ期間の時間数、分数、秒数 値のです。これらの3 つの時間 引数は、次の高い単位の値に加算され、結果は期間の全体の値を計算するために使用されます。例えば、72 秒は、1M+12S (1 分 と12 秒) に変換され、この値が期間全体の値を計算するために使用されます。
xs:yearMonthDuration 型の期間を構築するためには、次の署名を参照してください。

サンプル

- **build-duration**(2, 10, 3, 56) はP2DT10H3M56S を返します。
- **build-duration**(1, 0, 100, 0) はP1DT1H40M を返します。
- **build-duration**(1, 0, 0, 3600) はP1DT1H を返します。

[トップ]

文字列入力から日付、日付時刻 または 時刻 を構築する **XP2** **XQ1** **XP3** **XQ3**

関数は、文字列を引数として、xs:date、xs:dateTime またはxs:time データ型を構築します。文字列は、データ型のコンポーネントを提出されたパターン引数をベースにして分析されます。

parse-date [altova:]

parse-date(Date *as xs:string*, **DatePattern** *as xs:string*) を**xs:date** とする **XP2** **XQ1** **XP3** **XQ3**
Date 入力文字列を**xs:date** の値として返します。第二の引数である**DatePattern** は、入力文字列のパターン (コンポーネントのシーケンス) を指定します。**DatePattern** は、下にリストされるコンポーネント指定子および任

意の文字であるコンポーネントセパレータと共に説明されています。下のサンプルを参照してください。

D 日付
M 月
Y 年

DatePattern のパターンは、**Date** のパターンに一致する必要があります。出力は**xs:date** 型であるため、出力は常に **YYYY-MM-DD** 構文フォーマットになります。

サンプル

- **parse-date**(xs:string("06-03-2014"), "[D]-[M]-[Y]") は2014-03-06 を返します。
- **parse-date**(xs:string("06-03-2014"), "[M]-[D]-[Y]") は2014-03-06 を返します。
- **parse-date**("06/03/2014", "[M]/[D]/[Y]") は2014-03-06 を返します。
- **parse-date**("06 03 2014", "[M] [D] [Y]") は2014-03-06 を返します。
- **parse-date**("6 3 2014", "[M] [D] [Y]") は2014-03-06 を返します。

parse-dateTime [altova:]

parse-dateTime(DateTime *as xs:string*, **DateTimePattern** *as xs:string*) を **xs:dateTime** とする **XP2 XQ1 XP3 XQ3**

DateTime 入力文字列を**xs:dateTime** の値として返します。第二の引数である**DateTimePattern** は、入力文字列のパターン(コンポーネントのシーケンス) を指定します。**DateTimePattern** は、下にリストされるコンポーネント指定子および任意の文字であるコンポーネントセパレータと共に説明されています。下のサンプルを参照してください。

D 日
M 月
Y 年
H 時間
m 分
s 秒

DateTimePattern のパターンは、**DateTime** のパターンに一致する必要があります。出力は**xs:dateTime** 型であるため、出力は常に**YYYY-MM-DDTHH:mm:ss** 構文フォーマットになります。

サンプル

- **parse-dateTime**(xs:string("06-03-2014 13:56:24"), "[D]-[M]-[Y] [H]:[m]:[s]") は2014-03-06T13:56:24 を返します。
- **parse-dateTime**("time=13:56:24; date=06-03-2014", "time=[H]:[m]:[s]; date=[D]-[M]-[Y]") は2014-03-06T13:56:24 を返します。

parse-time [altova:]

parse-time(Time *as xs:string*, **TimePattern** *as xs:string*) を**xs:time** とする **XP2 XQ1 XP3 XQ3**

Time 入力文字列を**xs:time** の値として返します。第二の引数である**TimePattern** は、入力文字列のパターン(コンポーネントのシーケンス) を指定します。**TimePattern** は、下にリストされるコンポーネント指定子および任意の文字であるコンポーネントセパレータと共に説明されています。下のサンプルを参照してください。

H 時間
m 分

s 秒

TimePattern のパターンは、**Time** のパターンに一致する必要があります。出力は**xs:time** 型であるため、出力は常に**YYYY-HH:mm:ss** 構文フォーマットになります。

サンプル

- **parse-time**(xs:string("13:56:24"), "[H]:[m]:[s]") は13:56:24 を返します。
- **parse-time**("13-56-24", "[H]-[m]") は13:56:00 を返します。
- **parse-time**("time=13h56m24s", "time=[H]h[m]m[s]s") は13:56:24 を返します。
- **parse-time**("time=24s56m13h", "time=[s]s[m]m[H]h") は13:56:24 を返します。

[トップ]

年齢に関連した関数 XP3.1 XQ3.1

関数は計算された年齢(i) 入力引数の日付と現在の日付の期間 (ii) 2 つの入力引数の日付の期間 を返します。**altova:age** 関数は、年齢を年数で返します、**altova:age-details** は年齢を年数、月数、日数からなる3つの整数のシーケンスで返します。

---

age [altova:]

age(StartDate *as xs:date***)** を**xs:integer** とする XP3.1 XQ3.1

引数として提出された開始日から(システムクロックから取得された)現在の日付 までの日数を数えて、あるオブジェクトの年齢の年数である正数を返します。入力引数が1 年よりも大きい場合、または一年の場合、返される値は負の数です。

サンプル

If the current date is **2014-01-15**:

- **age**(xs:date("2013-01-15")) は1 を返します。
- **age**(xs:date("2013-01-16")) は0 を返します。
- **age**(xs:date("2015-01-15")) は -1 を返します。
- **age**(xs:date("2015-01-14")) は0 を返します。

age(StartDate *as xs:date*, **EndDate** *as xs:date***)** を**xs:integer** とする XP3.1 XQ3.1

最初の引数として提出された開始日から第 2 の引数の終了日 までの日数を数えて、あるオブジェクトの年齢の年数である正数を返します。最初の引数が1 年または第 2 の引数よりも後の日付の場合、返される値は負の数です。

サンプル

If the current date is **2014-01-15**:

- **age**(xs:date("2000-01-15"), xs:date("2010-01-15")) は10 を返します。
- **age**(xs:date("2000-01-15"), current-date()) は、現在の日付が2014-01-15 の場合 14 を返します。
- **age**(xs:date("2014-01-15"), xs:date("2010-01-15")) は -4 を返します。

age-details [altova:]

age-details(InputDate *as xs:date*) を(**xs:integer)*** とする **XP3.1** **XQ3.1**
引数と(システムクロックから取得された)現在の日付として提出された日付の間の年数、月数、日数の 3 つの整数を返します。返されたyears+months+days の合計は、2 つの日付 (入力の日付と現在の日付) の時間差です。入力の日付は現在の日付よりも早いまたは遅い値をもつことができますが、入力の日付が早いか遅いかは返される値で示されていません。戻される値は常に正の数です。

サンプル

現在の日付が**2014-01-15** の場合:

- **age-details**(xs:date("2014-01-16")) は(0 0 1)を返します。
- **age-details**(xs:date("2014-01-14")) は(0 0 1) を返します。
- **age-details**(xs:date("2013-01-16")) は(1 0 1) を返します。
- **age-details**(current-date()) は(0 0 0)を返します。

age-details(Date-1 *as xs:date*, **Date-2** *as xs:date*) を(**xs:integer)*** とする
XP3.1 **XQ3.1**
二つの引数間の年数、月数、日数の 3 つの整数を返します。 返されたyears+months+days の合計は、2 つの入力の日付 の時間差です。最初の引数として提出される二つの日付はどちらが速くても、また遅くてもかまいません。返される値は入力の日付が現在の日付より早いかまたは遅いかを示しません。戻される値は常に正の数です。

サンプル

- **age-details**(xs:date("2014-01-16"), xs:date("2014-01-15")) は(0 0 1) を返します。
- **age-details**(xs:date("2014-01-15"), xs:date("2014-01-16")) は(0 0 1)を返します。

[Top]

XPath/ XQuery 関数: 位置情報

以下の位置情報 XPath/XQuery 拡張関数は、MobileTogether Designer の現在のバージョンによりサポートされています。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)**。

XPath 関数 (XSLT 内のXPath 式で使用):	**XP1** **XP2** **XP3.1**
XSLT 関数 (XSLT 内のXPath 式で使用):	**XSLT1** **XSLT2** **XSLT3**
XQuery 関数 (XQuery 内のXQuery 式で使用):	**XQ1** **XQ3.1**

parse-geolocation [altova:]

parse-geolocation(GeolocationInputString *as xs:string***)** を**xs:decimal+** とする
XP3.1 **XQ3.1**

`GeolocationInputString` 引数を解析して、位置情報の緯度と経度 (この通りの順番) を2 つの `xs:decimal` アイテムのシーケンスとして返します。位置情報入力文字列が提供されることのできるフォーマットは以下のリストの通りです。

メモ `image-exif-data` 関数とExif メタデータの`@Geolocation` 属性を位置情報入力文字列を提供する際に使用することができます。

サンプル

- **parse-geolocation**("33.33 -22.22") は2 つの`xs:decimals (33.33, 22.22)` のシーケンスを返します。
- **parse-geolocation**("48°51'29.6""N 24°17'40.2""") は2 つの `xs:decimals (48.8582222222222, 24.2945)` のシーケンスを返します。
- **parse-geolocation**('48°51''29.6"N 24°17''40.2"') は2 つの `xs:decimals (48.8582222222222, 24.2945)` のシーケンスを返します。
- **parse-geolocation**(**image-exif-data**(//MyImages/Image20141130.01)/ **@Geolocation**) は2 つの `xs:decimals` のシーケンスを返します。

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は `+90` から `-90` (北 から南) です。経度の値の範囲は `+180` から `180` (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(**"**) でハイライトされており、エスケープした単位インジケーターは青い(**""**) でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(`N/S`, `W/E`)

D°M'S.SS"N/S D°M'S.SS"W/E
サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:
33.33N -22°44'66.66"
33.33 22°44'66.66"W
33.33 22.c

Altova Exif 属性: 位置情報

Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つのExif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S 151° 13'11.73"E

geolocation-distance-km [altova:]

geolocation-distance-km(GeolocationInputString-1 *as xs:string*, **GeolocationInputString-2** *as xs:string*) を**xs:decimal** とする **XP3.1** **XQ3.1**
2 つの位置情報の間の距離をキロメートルで計算します。位置情報入力文字列を提供することのできるフォーマットは下にリストされています。緯度の値の範囲は+90 から-90 (北 から南) です。経度の値の範囲は+180 から -180 (東 から西)です。

メモ image-exif-data 関数とExif メタデータの@Geolocation 属性を位置情報入力文字列を提供する際に使用することができます。

サンプル

- **geolocation-distance-km**("33.33 -22.22", "48°51'29.6""N 24° 17'40.2""") は xs:decimal 4183.08132372392 を返します。

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ: 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(") でハイライトされており、エスケープした単位インジケーターは青い("") でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:
33.33N -22°44'66.66"
33.33 22°44'66.66"W
33.33 22.c

Altova Exif 属性: 位置情報

Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つの Exif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S

				151° 13'11.73"E

geolocation-distance-mi [altova:]

geolocation-distance-mi(GeolocationInputString-1 *as xs:string*, **GeolocationInputString-2** *as xs:string*) を**xs:decimal** とする XP3.1 XQ3.1

2 つの位置情報の間の距離をマイルで計算します。位置情報入力文字列を提供することのできるフォーマットは下にリストされています。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ image-exif-data 関数とExif メタデータの @Geolocation 属性を位置情報入力文字列を提供する際に使用することができます。.

サンプル

- **geolocation-distance-mi**("33.33 -22.22", "48°51'29.6""N 24° 17'40.2""") は xs:decimal 2599.40652340653 を返します。

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(") でハイライトされており、エスケープした単位インジケーターは青い("") でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:
33.33N -22°44'66.66"
33.33 22°44'66.66"W
33.33 22.c

Altova Exif 属性: 位置情報

Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つのExif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S 151° 13'11.73"E

geolocation-within-polygon [altova:]

geolocation-within-polygon(Geolocation *as xs:string*, **((PolygonPoint** *as xs:string*)**+))** を**xs:boolean** とする **XP3.1** **XQ3.1**

PolygonPoint 引数により説明されている**Geolocation** (最初の引数) が多角形のエリア内に存在するかを決定します。もし、PolygonPoint 引数が(最初と最後のポイントが同じ場合に作成される)閉じられたフィギュアを作成しない場合、フィギュアを閉じるために、最初のポイントが明示的に最後のポイントとして追加されます。全ての引数(Geolocation および PolygonPoint+) は、(*下にリストされるフォーマットの*) 位置情報入力文字列により提供されます。Geolocation 引数が多角形エリア内にある場合、関数は true(); を返します。その他の場合は false() を返します。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から -180 (東 から西) です。

メモ image-exif-data 関数とExif メタデータの @Geolocation 属性を位置情報入力文字列を提供する際に使用することができます。

サンプル

- **geolocation-within-polygon("33 -22", ("58 -32", "-78 -55", "48 24", "58 -32"))** は true() を返します。
- **geolocation-within-polygon("33 -22", ("58 -32", "-78 -55", "48 24"))** は true() を返します。
- **geolocation-within-polygon("33 -22", ("58 -32", "-78 -55", "48° 51'29.6""N 24°17'40.2""))** は true() を返します。

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(**"**) でハイライトされており、エスケープした単位インジケーターは青い(**""**) でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:
33.33N -22°44'66.66"
33.33 22°44'66.66"W
33.33 22.c

Altova Exif 属性: 位置情報

Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つの Exif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S 151° 13'11.73"E

geolocation-within-rectangle [altova:]

geolocation-within-rectangle(Geolocation *as xs:string*, **RectCorner-1** *as xs:string*, **RectCorner-2** *as xs:string*) を**xs:boolean** とする **XP3.1 XQ3.1**
長方形の対格の角を指定する第 2 及び第 3 引数(**RectCorner-1**、**RectCorner-2**)により説明されてい

る**Geolocation** (最初の引数) が長方形のエリア内に存在するかを決定します。全ての引数(Geolocation、**RectCorner-1** および**RectCorner-2**) は、(*下にリストされるフォーマットの*) 位置情報入力文字列により提供されます。Geolocation 引数が長方形エリア内にある場合、関数は true(); を返します。その他の場合は false() を返します。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ image-exif-data 関数とExif メタデータの @Geolocation 属性を位置情報入力文字列を提供する際に使用することができます。

サンプル

- **geolocation-within-rectangle(**"33 -22", "58 -32", "-48 24") は true() を返します。
- **geolocation-within-rectangle(**"33 -22", "58 -32", "48 24") は false() を返します。
- **geolocation-within-rectangle(**"33 -22", "58 -32", "48°51'29.6""S 24° 17'40.2"""") は true() を返します。

位置情報入力文字列フォーマット:

位置情報入力文字列は空白で区別された緯度と経度 (この通りの順番) を含む必要があります。緯度と経度は以下のフォーマットをとることができます。組み合わせることも可能です。緯度が1つのフォーマットで、経度が他のフォーマットをとることができます。緯度の値の範囲は +90 から -90 (北 から南) です。経度の値の範囲は +180 から 180 (東 から西) です。

メモ 単一およびダブル引用符が入力文字列引数を区切るために使用されていると、使用されている単一およびダブル引用が、それぞれ、分の値と秒の値、不一致をもたらします。この様な場合、分の値と秒の値を表すための使用されている引用符は、ダブルにしてエスケープされる必要があります。このセクションのサンプルでは、入力文字列を区別するために使用されている引用符は黄色い(") でハイライトされており、エスケープした単位インジケーターは青い("") でハイライトされています。

- 度、分、10 進の秒、方角のサフィックス付き(N/S, W/E)
  D°M'S.SS"N/S D°M'S.SS"W/E
  サンプル: **33°55'11.11"N 22°44'66.66"W**

- 度、分、10 進の秒、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M'S.SS" +/-D°M'S.SS"
  サンプル: **33°55'11.11" -22°44'66.66"**

- 度、分、10 進の分、方角のサフィックス付き(N/S, W/E)
  D°M.MM'N/S D°M.MM'W/E
  サンプル: **33°55.55'N 22°44.44'W**

- 度、分、10 進の分、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D°M.MM' +/-D°M.MM'
  サンプル: **+33°55.55' -22°44.44'**

- 10 進の度、方角のサフィックス付き(N/S, W/E)
  D.DDN/S D.DDW/E
  サンプル: **33.33N 22.22W**

- 10 進の度、プレフィックスサイン付き(+/-); (**N/W**) のためのプラスサインは任意です。
  +/-D.DD +/-D.DD
  サンプル: **33.33 -22.22**

フォーマットの組み合わせのサンプル:

```
33.33N  -22°44'66.66"
33.33  22°44'66.66"W
33.33  22.c
```

Altova Exif 属性: 位置情報

Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つの Exif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S 151° 13'11.73"E

[Top]

XPath/ XQuery 関数: イメージに関連

以下のイメージに関連したXPath/XQuery 拡張関数は、MobileTogether Designer の現在のバージョンによりサポートされています。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)**。

XPath 関数 (XSLT 内のXPath 式で使用):	**XP1** **XP2** **XP3.1**
XSLT 関数 (XSLT 内のXPath 式で使用):	**XSLT1** **XSLT2** **XSLT3**
XQuery 関数 (XQuery 内のXQuery 式で使用):	**XQ1** **XQ3.1**

suggested-image-file-extension [altova:]

suggested-image-file-extension(Base64String *as string***)** を**string?** とする
XP3.1 **XQ3.1**
イメージファイルのBase64 エンコードを引数として、イメージのファイル拡張子を、イメージのBase64エンコード内の記録として返します。返された値は、エンコード内で使用することのできるイメージ型情報を基にしたヒントです。この情報が使用できない場合、空の文字列が返されます。この関数は、Base64 イメージをファイルとして保存し、適切なファイル拡張子を動的取得する際に役に立ちます。

サンプル

- **suggested-image-file-extension**(/MyImages/MobilePhone/Image20141130.01) は 'jpg' を返します。
- **suggested-image-file-extension**($XML1/Staff/Person/@photo) は '' を返します。

上のサンプルでは、関数の引数として与えられたノードはBase64 エンコードイメージを含むと仮定します。最初のサンプルは jpg をファイルの型および拡張子として取得します。二番目のサンプルでは、与えられたBase64 エンコードは使用できる拡張子の情報を提供しません。

mt-transform-image [altova:]

mt-transform-image(Base64Image *as Base64BinaryString*, **Size** *as item()+*, **Rotation** *as xs:integer*, **Quality** *as xs:integer***)** を**Base64BinaryString** とする
XP3.1 **XQ3.1**
Base64-エンコードイメージを最初の引数として、変換されたBase64-エンコードイメージを返します。第 2、第 3、第 4 引数は変換された以下のイメージパラメーターです: サイズ、回転、およびクオリティ。

- サイズ引数には3 つのサイズ変更のオプションがあります。

(X,Y)	絶対ピクセルの値。アスペクト率は保持されません。高さ幅は自動的にイメージの長いおよび短い辺に逢わされるため、高さ幅の指示は関係ありません。2 つの整数アイテムのシーケンスとして値が入力されます。かっこが必要です。
X	X をピクセルの新しい長い辺として、イメージの縦横比のサイズ変更が行われます。アスペクト率は保持されます。値は整数で引用符なしで入力されます。

'X%'	元のディメンションの与えられたパーセンテージにイメージのサイズを変更します。値は文字列として引用符を付けて入力される必要があります。

- 回転は以下の値を持つことができます: 90、180、270、-90、-180、-270。これらの値は回転の度数です。正の値はイメージを時計回りに回転させます。負の値はイメージを反時計回りに回転させます。Altova Exif 属性 **OrientationDegree** を使用して、イメージの現在の回転の度数(0, 90, 180, 270) をイメージの Exif **Orientation** タグから取得することができます。ですが、**OrientationDegree** 属性はデータの **Orientation** タグから取得されるため、Exif データ内に **Orientation** タグが存在する場合のみ使用することができます。(*下の OrientationDegree 説明を参照してください*)。
- クオリティは、0 から100 の値で、JPEG 圧縮の IJG クオリティスケールの値を参照していますが、クオリティのパーセンテージのインジケーターではありません。サイズとクオリティのどちらかを優先すると、もう一方の優先度が下がります。フルカラーソースのイメージでは、75 が通常最高値と見なされています。もし、75 が満足のいく結果をもたらさない場合は、値を上げてください。

メモ Exif データが元のイメージに存在する場合、変換時に削除され、変換されたイメージには Exif データは存在しません。

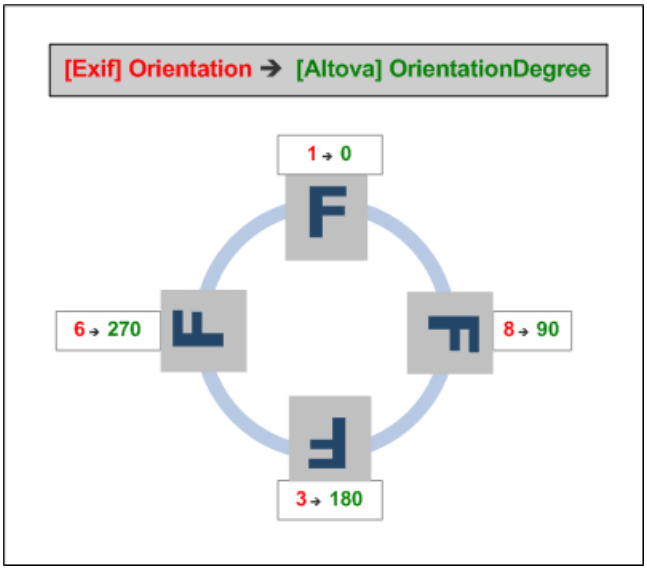
サンプル

- **mt-transform-image**(Images/Image[@id='43'], '50%', 90, 75)
  関数は、43 の@id 値を持つイメージ子孫ノード内で Base64-エンコード文字列として保管されているイメージを入力とします。関数は変換されたイメージを返します。変換されたイメージは50% まで、サイズ変更され、時計回りに90 度回転され、75 のクオリティレベルを与えられます。
- **mt-transform-image**(Images/Image[@id='43'], 400, 90, 75)
  関数は前のサンプルと同じ結果を出しますが、長い辺は400 ピクセルの特定の値に設定されています。元のイメージのアスペクト率は保持されます。
- **mt-transform-image**(Images/Image[@id='43'], (400, 280), image-exif-data($XML1/$XML1/Images/ReferenceImage)/@OrientationDegree, 75)
  このサンプルは、前のサンプルと同じイメージを選択し、同じクオリティ値 (75) を設定します。イメージのサイズは400x280 ピクセルに設定されています。回転値は、ノード内の Base64-エンコード イメージの **@OrientationDegree** 属性から得られます。

Altova Exif 属性: OrientationDegree

Altova XPath/XQuery エンジンはカスタム属性 **OrientationDegree** を Exif メタデータタグ **Orientation** から生成します。

OrientationDegree は標準 Exif タグ Orientation を正数の値 (1、8、3 または6) からそれぞれ対応する度数の値 (0, 90, 180, 270) へ下の図に示されているように変換します。2、4、5、7 の Orientation 値の変換はないことに注意してください。(これらの向きはイメージ 1 を垂直方向中央軸で反転して、2 の値を持つイメージを取得します。そして、このイメージを90-度ごとに時計回りにジャンプさせそれぞれ7、4、および5 の値を取得します。)。



- *標準 Exif メタタグ*
  - ImageWidth
  - ImageLength
  - BitsPerSample
  - Compression
  - PhotometricInterpretation
  - Orientation
  - SamplesPerPixel
  - PlanarConfiguration
  - YCbCrSubSampling
  - YCbCrPositioning
  - XResolution
  - YResolution
  - ResolutionUnit
  - StripOffsets
  - RowsPerStrip
  - StripByteCounts
  - JPEGInterchangeFormat
  - JPEGInterchangeFormatLength
  - TransferFunction
  - WhitePoint
  - PrimaryChromaticities
  - YCbCrCoefficients
  - ReferenceBlackWhite
  - DateTime
  - ImageDescription
  - Make
  - Model
  - Software
  - Artist
  - Copyright

- ExifVersion
- FlashpixVersion
- ColorSpace
- ComponentsConfiguration
- CompressedBitsPerPixel
- PixelXDimension
- PixelYDimension
- MakerNote
- UserComment
- RelatedSoundFile
- DateTimeOriginal
- DateTimeDigitized
- SubSecTime
- SubSecTimeOriginal
- SubSecTimeDigitized
- ExposureTime
- FNumber
- ExposureProgram
- SpectralSensitivity
- ISOSpeedRatings
- OECF
- ShutterSpeedValue
- ApertureValue
- BrightnessValue
- ExposureBiasValue
- MaxApertureValue
- SubjectDistance
- MeteringMode
- LightSource
- Flash
- FocalLength
- SubjectArea
- FlashEnergy
- SpatialFrequencyResponse
- FocalPlaneXResolution
- FocalPlaneYResolution
- FocalPlaneResolutionUnit
- SubjectLocation
- ExposureIndex
- SensingMethod
- FileSource
- SceneType
- CFAPattern
- CustomRendered
- ExposureMode
- WhiteBalance
- DigitalZoomRatio
- FocalLengthIn35mmFilm
- SceneCaptureType
- GainControl
- Contrast
- Saturation
- Sharpness
- DeviceSettingDescription
- SubjectDistanceRange

(C) 2018 Altova GmbH

- ImageUniqueID

- GPSVersionID
- GPSLatitudeRef
- GPSLatitude
- GPSLongitudeRef
- GPSLongitude
- GPSAltitudeRef
- GPSAltitude
- GPSTimeStamp
- GPSSatellites
- GPSStatus
- GPSMeasureMode
- GPSDOP
- GPSSpeedRef
- GPSSpeed
- GPSTrackRef
- GPSTrack
- GPSImgDirectionRef
- GPSImgDirection
- GPSMapDatum
- GPSDestLatitudeRef
- GPSDestLatitude
- GPSDestLongitudeRef
- GPSDestLongitude
- GPSDestBearingRef
- GPSDestBearing
- GPSDestDistanceRef
- GPSDestDistance
- GPSProcessingMethod
- GPSAreaInformation
- GPSDateStamp
- GPSDifferential

image-exif-data [altova:]

image-exif-data(Base64BinaryString *as string*) を**element?** とする **XP3.1** **XQ3.1**

Base64 エンコード イメージを引数として、イメージの Exif メタデータを含む **Exif** という名の要素を返します。The Exif メタデータは Exif 要素の属性の値ペアとして作成されます。属性名は、Base64 エンコード内で検出された Exif データタグです。Exif 仕様タグのリストは以下の通りです。ベンダー特有のタグが Exif データ内に存在する場合、タグとその値も属性値のペアとして返されます。標準 Exif メタデータタグに追加して (*下のリスト参照*) Altova 特有の属性値のペアもまた生成されます。これらの Altova Exif 属性は以下のとおりです。

サンプル

- 属性にアクセスするには、以下の関数を使用します:
  image-exif-data(//MyImages/Image20141130.01)/**@GPSLatitude**
  image-exif-data(//MyImages/Image20141130.01)/**@Geolocation**
- 全ての属性にアクセスするには、以下の関数を使用します:
  image-exif-data(//MyImages/Image20141130.01)/**@***
- 全ての属性の名前にアクセスするには、以下の式を使用します:
  for $i in image-exif-data(//MyImages/Image20141130.01)/**@* return**

name($i)
関数により返される属性の名前を検出するために役に立ちます。

Altova Exif 属性: 位置情報

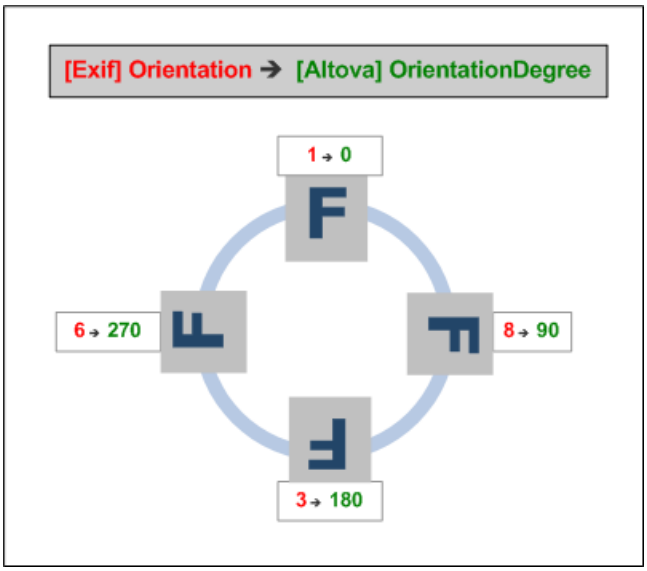
Altova XPath/XQuery エンジンはカスタム属性 **Geolocation**を標準 Exif メタデータタグから生成します。**Geolocation** は、4 つの Exif タグの連結です: 単位の追加された(*下のテーブル参照*) GPSLatitude、GPSLatitudeRef、GPSLongitude、GPSLongitudeRef。

GPSLatitude	GPSLatitudeRef	GPSLongitude	GPSLongitudeRef	Geolocation
33 51 21.91	S	151 13 11.73	E	33° 51'21.91"S 151° 13'11.73"E

Altova Exif 属性: OrientationDegree

Altova XPath/XQuery エンジンはカスタム属性 **OrientationDegree** を Exif メタデータタグ **Orientation** から生成します。

OrientationDegree は標準 Exif タグ Orientation を正数の値 (1、8、3 または 6) からそれぞれ対応する度数の値 (0, 90, 180, 270) へ下の図に示されているように変換します。2、4、5、7 の Orientation 値の変換はないことに注意してください。(これらの向きはイメージ 1 を垂直方向中央軸で反転して、2 の値を持つイメージを取得します。そして、このイメージを 90-度ごとに時計回りにジャンプさせそれぞれ 7、4、および 5 の値を取得します。)。



標準 Exif メタデータタグのリスト

- ImageWidth
- ImageLength
- BitsPerSample
- Compression

- PhotometricInterpretation
- Orientation
- SamplesPerPixel
- PlanarConfiguration
- YCbCrSubSampling
- YCbCrPositioning
- XResolution
- YResolution
- ResolutionUnit
- StripOffsets
- RowsPerStrip
- StripByteCounts
- JPEGInterchangeFormat
- JPEGInterchangeFormatLength
- TransferFunction
- WhitePoint
- PrimaryChromaticities
- YCbCrCoefficients
- ReferenceBlackWhite
- DateTime
- ImageDescription
- Make
- Model
- Software
- Artist
- Copyright

- ExifVersion
- FlashpixVersion
- ColorSpace
- ComponentsConfiguration
- CompressedBitsPerPixel
- PixelXDimension
- PixelYDimension
- MakerNote
- UserComment
- RelatedSoundFile
- DateTimeOriginal
- DateTimeDigitized
- SubSecTime
- SubSecTimeOriginal
- SubSecTimeDigitized
- ExposureTime
- FNumber
- ExposureProgram
- SpectralSensitivity
- ISOSpeedRatings
- OECF
- ShutterSpeedValue
- ApertureValue
- BrightnessValue
- ExposureBiasValue
- MaxApertureValue
- SubjectDistance

- MeteringMode
- LightSource
- Flash
- FocalLength
- SubjectArea
- FlashEnergy
- SpatialFrequencyResponse
- FocalPlaneXResolution
- FocalPlaneYResolution
- FocalPlaneResolutionUnit
- SubjectLocation
- ExposureIndex
- SensingMethod
- FileSource
- SceneType
- CFAPattern
- CustomRendered
- ExposureMode
- WhiteBalance
- DigitalZoomRatio
- FocalLengthIn35mmFilm
- SceneCaptureType
- GainControl
- Contrast
- Saturation
- Sharpness
- DeviceSettingDescription
- SubjectDistanceRange
- ImageUniqueID

- GPSVersionID
- GPSLatitudeRef
- GPSLatitude
- GPSLongitudeRef
- GPSLongitude
- GPSAltitudeRef
- GPSAltitude
- GPSTimeStamp
- GPSSatellites
- GPSStatus
- GPSMeasureMode
- GPSDOP
- GPSSpeedRef
- GPSSpeed
- GPSTrackRef
- GPSTrack
- GPSImgDirectionRef
- GPSImgDirection
- GPSMapDatum
- GPSDestLatitudeRef
- GPSDestLatitude
- GPSDestLongitudeRef
- GPSDestLongitude
- GPSDestBearingRef

- GPSDestBearing
- GPSDestDistanceRef
- GPSDestDistance
- GPSProcessingMethod
- GPSAreaInformation
- GPSDateStamp
- GPSDifferential

[トップ]

XPath/ XQuery 関数: 数値

Altova の数値拡張関数は XPath とXQuery 式で使用することができ、XML スキーマの異なる日付および時刻データ型で保存されているデータを処理するための追加機能を提供します。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、および XSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)**。

XPath 関数 (XSLT 内のXPath 式で使用):	XP1 XP2 XP3.1
XSLT 関数 (XSLT 内のXPath 式で使用):	XSLT1 XSLT2 XSLT3
XQuery 関数 (XQuery 内のXQuery 式で使用):	XQ1 XQ3.1

自動付番関数

generate-auto-number [altova:]

generate-auto-number(ID *as xs:string*, **StartsWith** *as xs:double*, **Increment** *as xs:double*, **ResetOnChange** *as xs:string*) を xs:integer とする XP1 XP2 XQ1 XP3.1 XQ3.1

関数が呼び出される度に番号を生成します。関数が初めて呼び出される際に生成される最初の番号は StartsWith 引数により指定されます。その後の関数の呼び出しは新しい番号を生成します。この番号は前に生成された番号を、Increment 引数で指定された値ごとにインクリメントします。実質的には、generate-auto-number 関数は、ID 引数により指定されたカウンターを作成し、このカウンターが関数が呼び出されるごとにインクリメントされます。ResetOnChange 引数の値が、前の関数呼び出しから変更されると、生成される番号の値が StartsWith 値にリセットされます。自動付番は reset-auto-number 関数を使用してリセットすることができます。

サンプル

- **generate-auto-number**("ChapterNumber", 1, 1, "SomeString") は、関数が呼び出される度に番号を1 つ返します。starting with 1, から始まり、関数が呼び出される度に1 ずつインクリメントします。その後の呼び出しの4 番目の"SomeString"引数がである限り、インクリメントは継続されます。4 番目の引数の値が変更されると、(ChapterNumber と呼ばれる) カウンターは1 にリセットされます。ChapterNumber の値もreset-auto-number 関数の呼び出しによりの reset-auto-number("ChapterNumber") ようにリセットされます。

reset-auto-number [altova:]

reset-auto-number(ID *as xs:string*) XP1 XP2 XQ1 XP3.1 XQ3.1

この関数は、ID 引数内で名づけられた自動付番カウンターの番号をリセットします。引数内のカウンター名を作成した関数の引数により指定された数値にリセットされます。generate-auto-number

サンプル

- **reset-auto-number**("ChapterNumber") は、generate-auto-number 関数により作成されたChapterNumber という名の自動付番カウンターをリセットします。
- generate-auto-number 関数のStartsWith 引数の値に数値をリセットします。

[Top]

数値関数

hex-string-to-integer [altova:]

hex-string-to-integer(HexString *as xs:string***)** を**xs:integer** とする **XP3.1** **XQ3.1**

10 進のシステム(10進)内の正数の16 進の同値の文字列引数を必要とし、10 進の整数を返します。

サンプル

- **hex-string-to-integer**('1') は1 を返します。
- **hex-string-to-integer**('9') は9 を返します。
- **hex-string-to-integer**('A') は10 を返します。
- **hex-string-to-integer**('B') は11 を返します。
- **hex-string-to-integer**('F') は15 を返します。
- **hex-string-to-integer**('G') はエラーを返します。
- **hex-string-to-integer**('10') は16 を返します。
- **hex-string-to-integer**('01') は1 を返します。
- **hex-string-to-integer**('20') は32 を返します。
- **hex-string-to-integer**('21') は33 を返します。
- **hex-string-to-integer**('5A') は90 を返します。
- **hex-string-to-integer**('USA') はエラーを返します。

integer-to-hex-string [altova:]

integer-to-hex-string(Integer *as xs:integer***)** を**xs:string** とする **XP3.1** **XQ3.1**

正数を引数として必要とし、文字列として自身のベース16 の同値を返します。

サンプル

- **integer-to-hex-string**(1) は'1' を返します。
- **integer-to-hex-string**(9) は'9' を返します。
- **integer-to-hex-string**(10) は'A' を返します。
- **integer-to-hex-string**(11) は'B' を返します。
- **integer-to-hex-string**(15) は'F' を返します。
- **integer-to-hex-string**(16) は'10' を返します。
- **integer-to-hex-string**(32) は'20' を返します。
- **integer-to-hex-string**(33) は'21' を返します。
- **integer-to-hex-string**(90) は'5A' を返します。

[トップ]

数値をフォーマットする関数

mt-format-number [altova:]

mt-format-number(Number *as xs:numeric*, **PictureString** *as xs:string***)** を **xs:string** とする **XP3.1** **XQ3.1**

第 2 (PictureString) 引数に従い、数値を最初の引数、フォーマットとし、フォーマットされ数値を文字列として返します。これは、読みにくい番号を読み取りやすくするためにフォーマットする際にとても役に立ちます。ピクチャ文字列は通貨シンボルなどの文字を含むことができ、フォーマットされた出力でこれらの文字を使用することができます。その位置の入力番号に小数がない場合は小数の場所にゼロを挿入することができ、ピクチャ文字列の小数の位置にゼロを使用することができます(*下のサンプル参照*)。ゼロ(または他の文字) の使用を強制しない場合は、番号記号 (#) を使用します。

小数区切りの前の小数は決して短縮することができません。(小数区切りの右側の) 数値の小数の部分、および、(小数区切りの左側の) 単位桁は、ピクチャ文字列の小数部分が入力数値の小数の数より短い場合、四捨五入されます。

メモ モバイルデバイス上でのフォーマットされた出力でのグループ区切りと小数区切りは、モバイルデバイスで使用される言語の区切りです。

サンプル

- **mt-format-number**(12.3, '$#0.00') は $12.30 を返します。
- **mt-format-number**(12.3, '$00.00') は $12.30 を返します。
- **mt-format-number**(12.3, '$0,000.00') は $0,012.30 を返します。
- **mt-format-number**(12.3, '$#,000.00') は $012.30 を返します。
- **mt-format-number**(1234.5, '$#,##0.00') は $1,234.50 を返します。
- **mt-format-number**(1234.5, '$#0.00') は $1234.50 を返します。
- **mt-format-number**(123.4, '$0') は $123 を返します。
- **mt-format-number**(1234.5, '$0') は $1235 を返します。
- **mt-format-number**(1234.54, '$0.0') は $1234.5 を返します。
- **mt-format-number**(1234.55, '$0.0') は $1234.6 を返します。

generate-auto-number [altova:]

generate-auto-number(ID *as xs:string*, **StartsWith** *as xs:double*, **Increment** *as xs:double*, **ResetOnChange** *as xs:string*) を xs:integer とする **XP1 XP2 XQ1 XP3.1 XQ3.1**

関数が呼び出される度に番号を生成します。関数が初めて呼び出される際に生成される最初の番号は StartsWith 引数により指定されます。その後の関数の呼び出しは新しい番号を生成します。この番号は前に生成された番号を、Increment 引数で指定された値ごとにインクリメントします。実質的には、generate-auto-number 関数は、ID 引数により指定されたカウンターを作成し、このカウンターが関数が呼び出されるごとにインクリメントされます。ResetOnChange 引数の値が、前の関数呼び出しから変更されると、生成される番号の値が StartsWith 値にリセットされます。自動付番は reset-auto-number 関数を使用してリセットすることができます。

サンプル

- **generate-auto-number**("ChapterNumber", 1, 1, "SomeString") は、関数が呼び出される度に番号を 1 つ返します。starting with 1, から始まり、関数が呼び出される度に 1 ずつインクリメントします。その後の呼び出しの 4 番目の "SomeString" 引数がである限り、インクリメントは継続されます。4 番目の引数の値が変更されると、(ChapterNumber と呼ばれる) カウンターは 1 にリセットされます。ChapterNumber の値も reset-auto-number 関数の呼び出しによりの reset-auto-number("ChapterNumber") ようにリセットされます。

[トップ]

XPath/ XQuery 関数: シーケンス

Altova のシーケンス拡張関数は XPath とXQuery 式で使用することができ、XML スキーマの異なる日付および時刻データ型で保存されているデータを処理するための追加機能を提供します。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)**。

XPath 関数 (XSLT 内のXPath 式で使用):	XP1 XP2 XP3.1
XSLT 関数 (XSLT 内のXPath 式で使用):	XSLT1 XSLT2 XSLT3
XQuery 関数 (XQuery 内のXQuery 式で使用):	XQ1 XQ3.1

attributes [altova:]

attributes(AttributeName *as xs:string*) を**attribute()*** とする XP3.1 XQ3.1

入力引数 AttributeName 内で与えられた名前と同じローカル名を持つすべての属性を返します。検索は大文字と小文字を区別し、attribute:: 軸に対して行われます。これは、コンテキストノードが親要素ノードである必要があることを意味します。

サンプル

- **attributes**("MyAttribute") はMyAttribute()* を返します。

attributes(AttributeName *as xs:string*, **SearchOptions** *as xs:string*) as **attribute()*** XP3.1 XQ3.1

入力引数 AttributeName 内で与えられた名前と同じローカル名を持つすべての属性を返します。検索は大文字と小文字を区別し、attribute:: 軸に対して行われます。コンテキストノードが親要素ノードである必要があります。第 2 引数はオプションのフラグを含みます。使用することのできるフラグは以下の通りです:

r = 正規表現検索を切り替えます; AttributeName は正規表現検索文字列である必要があります;

f = このオプションが指定されている場合、AttributeName は完全一致を提供します。それ以外の場合は、AttributeName は属性名に部分的に一致するとその属性を返します。例えば **f** が指定されていない場合、MyAtt はMyAttribute を返します。

i = 大文字と小文字を一致させる検索に切り替えます。

p = 検索に名前空間プレフィックスを含みます。AttributeName は、名前空間プレフィックスを含みます。例えば : altova:MyAttribute。

フラグは順序に関わりなく書き込むことができます。無効なフラグはエラーを生成します。1つまたは複数のフラグを省略することができます。空の文字列は許可されていますが、引数を1つしか持たない関数と同じ効果を持ちます(*前の署名*)、ですが、空のシーケンスは第 2 の引数としては許可されていません。

サンプル

- **attributes**("MyAttribute", "rfip") はMyAttribute()* を返します。
- **attributes**("MyAttribute", "pri") はMyAttribute()* を返します。
- **attributes**("MyAtt", "rip") はMyAttribute()* を返します。
- **attributes**("MyAttributes", "rfip") は不一致を返します。
- **attributes**("MyAttribute", "") はMyAttribute()* を返します。
- **attributes**("MyAttribute", "Rip") 認識されないフラグエラーを返します。
- **attributes**("MyAttribute",) 見つからない第 2 引数を返します。

elements [altova:]

elements(ElementName *as xs:string*) を**element()*** とする **XP3.1** **XQ3.1**

入力引数 ElementName で与えられた名前と同じローカル名を持つすべての要素を返します。検索は大文字と小文字を区別して child:: 軸に対して実行されます。コンテキストノードは、検索される要素の親ノードである必要があります。

サンプル

- **elements**("MyElement") は MyElement()* を返します。

elements(ElementName *as xs:string*, **SearchOptions** *as xs:string*) as **element()*** **XP3.1** **XQ3.1**

入力引数 ElementName 内で与えられた名前と同じローカル名を持つすべての属性を返します。検索は大文字と小文字を区別し、child:: 軸に対して行われます。コンテキストノードが親要素ノードである必要があります。第 2 引数はオプションのフラッグを含みます。使用することのできるフラッグは以下の通りです:

r = 正規表現検索を切り替えます; ElementName は正規表現検索文字列である必要があります;

f = このオプションが指定されている場合、ElementName は完全一致を提供します。それ以外の場合は、ElementName は属性名に部分的に一致するとその属性を返します。例えば **f** が指定されていない場合、MyElem は MyElement を返します。

i = 大文字と小文字を一致させる検索に切り替えます。

p = 検索に名前空間プレフィックスを含みます。ElementName は、名前空間プレフィックスを含みます。例えば altova:MyElement。

フラッグは順序に関わりなく書き込むことができます。無効なフラッグはエラーを生成します。1 つまたは複数のフラッグを省略することができます。空の文字列は許可されていますが、引数を 1 つしか持たない関数と同じ効果を持ちます(*前の署名*)、ですが、空のシーケンスは第 2 の引数としては許可されていません。

サンプル

- **elements**("MyElement", "rip") は MyElement()* を返します。
- **elements**("MyElement", "pri") は MyElement()* を返します。
- **elements**("MyElement", "") は MyElement()* を返します。
- **attributes**("MyElem", "rip") は MyElement()* を返します。
- **attributes**("MyElements", "rfip") 不一致を課wします。
- **elements**("MyElement", "Rip") 認識されないフラッグエラーを返します。
- **elements**("MyElement",) 見つからない第 2 引数を返します。

find-first [altova:]

find-first((Sequence *as item()**), **(Condition(** **Sequence-Item** *as xs:boolean*))) を**item()?** とする **XP3.1** **XQ3.1**

この関数は 2 つの引数を必要とします。最初の引数は 1 つ、または 1 つ以上のデータ型のアイテムのシーケンスです。第 2 の引数Condition は、(**1** のアリティを持つ) 1 つの引数 を必要とし、boolean を返す XPath 関数に対する参照です。Condition で参照された関数の代わりに、**Sequence** の各アイテムが提出されます。(*注意*: この関数は 1 つの引数のみを必要とします。) **Condition** 内の関数に **true()** と評価させる最初の **Sequence** アイテムは、**find-first**、反復の終了の結果として返されます。

サンプル

- **find-first**(5 to 10, function($a) {$a mod 2 = 0}) は xs:integer 6 を返します。

  Condition 引数は、**$a** という名のインライン関数を宣言し、定義します。XPath 3.0 インライン関数 **function()** を参照します。**Sequence** 引数内の各アイテムでは、**find-first** がパスされ、代わりに、**$a** を入力値とします。入力値は関数定義($a mod 2 = 0)内の条件に対してテストされます。この条件を満たす最初の入力値が **find-first** (この場合は **6**) の結果として返されます。

- **find-first**((1 to 10), (function($a) {$a+3=7})) はxs:integer 4 を返します。

更なるサンプル
ファイル**C:\Temp\Customers.xml** が存在する場合:

- **find-first**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) はxs:string C:\Temp\Customers.xml を返します。

ファイル**C:\Temp\Customers.xml** が存在せず、**http://www.altova.com/index.html** が存在する場合:

- **find-first**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) はxs:string http://www.altova.com/index.html を返します。

ファイル**C:\Temp\Customers.xml** が存在せず、**http://www.altova.com/index.html** も存在しない場合:

- **find-first**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) 結果無しを返します。

上のサンプルについての注意点

- XPath 3.0 関数 doc-available はURI として使用され、ドキュメントノードが提出されたURI で検出される場合 true を返す単一の引数を必要とします。(ですから、提出されたURI でのドキュメントはXML ドキュメントである必要があります。)
- doc-available 関数は、find-first の第 2 引数である **Condition** で使用することができます。これは、1 つの引数 (アリティ=1) のみを必要とするからであり、item()を入力(URI として使用される文字列) として、boolean の値を返すからです。
- doc-available 関数は、参照されているだけで、呼び出されていない点に注意してください。アタッチされている#1 サフィックスは関数が 1 つのアリティであることを表示するためです。doc-available#1 の意味は以下のとおりです: アリティ=1 を持つ doc-availabe() 関数を使用し、最初のシーケンスの各アイテムの代わりに単一引数としてパスします。 この結果、 2 つの文字列の各自つは、文字列をURI として使用し、URI にドキュメントノードが存在するかテストする**doc-available()**にパスされます。1 つが各当する場合、**doc-available()** 関数はtrue() を評価し、シーケンス内のその文字列のインデックス ポジションは、 **find-first** 関数の結果として返されます。doc-available() 関数に関しての注意点: 相対パスは、デフォルトで関数がロードされるXML ドキュメントの現在のベースURI に対して相対的に解決されます。

find-first-combination [altova:]

find-first-combination((Seq-01 *as item()**), **(Seq-02** *as item()**), **(Condition(Seq-01-Item, Seq-02-Item** *as xs:boolean*)) を**item()*** とする **XP3.1 XQ3.1**

この関数は3 つの引数を必要とします:

- 最初の2 つの引数 **Seq-01** と**Seq-02**, は、1 つまたは1 つ以上のデータ型のアイテムです。
- 第 3 の引数 **Condition** は、(**2** のアリティを持つ) 2 つの引数を必要とし、boolean を返すXPath 関数に対しての参照です。

Seq-01 と**Seq-02** のアイテムは指定された組み合わせ(各シーケンスからの1つずつにアイテムで構成されるペア)で、**Condition** 内の関数の引数としてパスされました。組み合わせは以下のように指定されています。

```
If   Seq-01 = X1, X2, X3 ... Xn
And  Seq-02 = Y1, Y2, Y3 ... Yn
Then (X1 Y1), (X1 Y2), (X1 Y3) ... (X1 Yn), (X2 Y1), (X2 Y2) ... (Xn Yn)
```

Condition 関数に**true()** と評価するように指示する最初のペアは**find-first-combination** の結果として返されます。以下の点に注意してください: (i) **Condition** 関数が提出された引数ペア内で繰り返され、**true()** を評価しない場合、**find-first-combination** は、結果を返しません; (ii) **find-first-combination** の結果が常に(データ型の)アイテムのペアである場合またはアイテムでない場合。

サンプル

- **find-first-combination**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 32}) はxs:integers (11, 21) のシーケンスを返します。
- **find-first-combination**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 33}) はxs:integers (11, 22) のシーケンを返します。
- **find-first-combination**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 34}) は xs:integers (11, 23) のシーケンを返します。

find-first-pair [altova:]

find-first-pair((Seq-01 *as item()**), **(Seq-02** *as item()**), **(Condition(Seq-01-Item, Seq-02-Item** *as xs:boolean*)) を**item()*** とする **XP3.1** **XQ3.1**

この関数は3 つの引数を必要とします:

- 最初の2 つの引数 **Seq-01** と**Seq-02**, は、1 つまたは1 つ以上のデータ型のアイテムです。
- 第 3 の引数 **Condition** は、(**2** のアリティを持つ) 2 つの引数を必要とし、boolean を返す XPath 関数に対しての参照です。

Seq-01 と**Seq-02** のアイテムは指定された組み合わせで、**Condition** 内の関数の引数としてパスされました。組み合わせは以下のように指定されています。

```
If   Seq-01 = X1, X2, X3 ... Xn
And  Seq-02 = Y1, Y2, Y3 ... Yn
Then (X1 Y1), (X2 Y2), (X3 Y3) ... (Xn Yn)
```

Condition 関数に**true()** と評価するように指示する最初のペアは**find-first-pair** の結果として返されます。以下の点に注意してください: (i) **Condition** 関数が提出された引数ペア内で繰り返され、**true()** を評価しない場合、**find-first-pair** は、結果を返しません; (ii) **find-first-combination** の結果が常に(データ型の) アイテムのペアである場合またはアイテムでない場合。

サンプル

- **find-first-pair**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 32}) は xs:integers (11, 21) のシーケンスを返します。
- **find-first-pair**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 33}) は*結果無し*を返します。

上の2 つのサンプルに表示される通り、ペアの順序は以下の通りです: (11, 21) (12, 22) (13, 23)...(20, 30) 。(33 を返す指示されたペアがないため) この理由で第 2 のサンプルは*結果無し*を返します。

find-first-pair-pos [altova:]

find-first-pair-pos((Seq-01 *as item()**)**, (Seq-02** *as item()**)**, (Condition(Seq-01-Item, Seq-02-Item** *as xs:boolean*)**)** を**xs:integer** とする
XP3.1 **XQ3.1**
この関数は3 つの引数を必要とします:

- 最初の2 つの引数 **Seq-01** と**Seq-02**, は、1 つまたは1 つ以上のデータ型のアイテムです。
- 第 3 の引数 **Condition** は、(**2** のアリティを持つ) 2 つの引数を必要とし、boolean を返す XPath 関数に対しての参照です。

Seq-01 と**Seq-02** のアイテムは指定された組み合わせで、**Condition** 内の関数の引数としてパスされました。組み合わせは以下のように指定されています。

```
If   Seq-01 = X1, X2, X3 ... Xn
And  Seq-02 = Y1, Y2, Y3 ... Yn
Then (X1 Y1), (X2 Y2), (X3 Y3) ... (Xn Yn)
```

Condition 関数に**true()**を評価させる、最初に指示されたペアのインデックスポジションは、は**find-first-pair-pos** の結果として返されます。関数が提出された引数ペア内で繰り返され、**true()**を一度も評価しない場合、**find-first-pair-pos** *結果無し*が返します。

サンプル

- **find-first-pair-pos**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 32}) は1 を返します
- **find-first-pair-pos**(11 to 20, 21 to 30, function($a, $b) {$a+$b = 33}) は*結果無し*を返します。

上の2 つのサンプルに表示される通り、ペアの順序は以下の通りです: (11, 21) (12, 22) (13, 23)...(20, 30)。最初のサンプルでは、最初のペアは**Condition** 関数に**true()**を評価させ、シーケンス内のインデックスポジションには**1** が返されます。第 2 のサンプルでは、33 を返すペアがないため、*結果無し*が返します。

find-first-pos [altova:]

find-first-pos((Sequence *as item()**)**, (Condition(Sequence-Item** *as xs:boolean*)**)** を**xs:integer** とする **XP3.1** **XQ3.1**
この関数は2 つの引数を必要とします。最初の引数は1 つ、または1 つ以上のデータ型のアイテムのシーケンスです。第 2 の引数 Condition は、(**1** のアリティを持つ) 1 つの引数 を必要とし、boolean を返す XPath 関数に対する参照です。Condition で参照された関数の代わりに、**Sequence** の各アイテムが提出されます。(*注意: この関数は1 つの引数のみを必要とします。*) **Condition** 内の関数に**true()**と評価させる最初の**Sequence** アイテムは、**find-first-pos** の結果として返された**Sequence** 内インデックスポジションを持ちます。

サンプル

- **find-first-pos**(5 to 10, function($a) {$a mod 2 = 0}) はxs:integer 2 を返します。
  Condition 引数は、**$a** という名のインライン関数を宣言し、定義します。XPath 3.0 インライン関数 **function()**を参照します。 **Sequence** 引数内の各アイテムでは、**find-first-pos** がパスされ、代わりに、**$a** を入力値とします。入力値は関数定義($a mod 2 = 0)内の条件に対してテストされます。この条件を満たす最初の入力値が**find-first-pos** ((シーケンス内で)条件を満たす最初の値である6 がシーケンスのインデックス位置 2 にあるためこの場合は2、) の結果として返されます。

- **find-first-pos**((2 to 10), (function($a) {$a+3=7})) はxs:integer 3 を返します。

更なるサンプル
ファイル**C:\Temp\Customers.xml** が存在する場合:

- **find-first-pos**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) returns 1

ファイル**C:\Temp\Customers.xml** が存在せず、**http://www.altova.com/index.html** が存在する場合:

- **find-first-pos**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) returns 2

ファイル**C:\Temp\Customers.xml** が存在せず、**http://www.altova.com/index.html** も存在しない場合:

- **find-first-pos**(("C:\Temp\Customers.xml", "http://www.altova.com/index.html"), (doc-available#1)) 結果無しを返します。

上のサンプルについての注意点

- XPath 3.0 関数 doc-available はURI として使用され、ドキュメントノードが提出されたURI で検出される場合 true を返する単一の引数を必要とします。(ですから、提出されたURI でのドキュメントはXML ドキュメントである必要があります。)
- doc-available 関数は、find-first-pos の第 2 引数である **Condition** で使用することができます。これは、1 つの引数 (アリティ=1) のみを必要とするからであり、item()を入力(URI として使用される文字列) として、boolean の値を返すからです。
- doc-available 関数は、参照されているだけで、呼び出されていない点に注意してください。アタッチされている#1 サフィックスは関数が 1 つのアリティであることを表示するためです。doc-available#1 の意味は以下のとおりです: アリティ=1 を持つ doc-availabe() 関数を使用し、最初のシーケンスの各アイテムの代わりに単一引数としてパスします。 この結果、 2 つの文字列の各自つは、文字列をURI として使用し、URI にドキュメントノードが存在するかテストする**doc-available()**にパスされます。1 つが各当する場合、**doc-available()** 関数はtrue() を評価し、シーケンス内のその文字列のインデックス ポジションは、 **find-first-pos** 関数の結果として返されます。doc-available() 関数に関しての注意点: 相対パスは、デフォルトで関数がコードされるXML ドキュメントの現在のベースURI に対して相対的に解決されます。

substitute-empty [altova:]

substitute-empty(FirstSequence *as item()**, **SecondSequence** *as item()***)** を **item()*** とする **XP3.1** **XQ3.1**

FirstSequence が空の場合、SecondSequence を返します。FirstSequence が空でない場合、FirstSequence を返します。

サンプル

- **substitute-empty**((1,2,3), (4,5,6)) は(1,2,3) を返します。
- **substitute-empty**((), (4,5,6)) は(4,5,6) を返します。

XPath/ XQuery 関数: 文字列

Altova の文字列拡張関数は XPath とXQuery 式で使用することができ、XML スキーマの異なる日付および時刻データ型で保存されているデータを処理するための追加機能を提供します。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)。**

XPath 関数 (XSLT 内のXPath 式で使用):	XP1 XP2 XP3.1
XSLT 関数 (XSLT 内のXPath 式で使用):	XSLT1 XSLT2 XSLT3
XQuery 関数 (XQuery 内のXQuery 式で使用):	XQ1 XQ3.1

camel-case [altova:]

camel-case(InputString *as xs:string***)** を**xs:string** とする XP3.1 XQ3.1

入力文字列を**InputString** をキャメルケースで返します。文字列は(空白スペースのショートカットである) 正規表現 **'\s'** を使用して分析されます。空白文字または連続する空白文字のシーケンスの後の最初の非空白スペース文字は大文字です。出力文字列の最初の文字は大文字です。

サンプル

- **camel-case**("max") Max を返します。
- **camel-case**("max max") Max Max を返します。
- **camel-case**("file01.xml") File01.xml を返します。
- **camel-case**("file01.xml file02.xml") File01.xml File02.xml を返します。
- **camel-case**("file01.xml file02.xml") File01.xml File02.xml を返します。
- **camel-case**("file01.xml -file02.xml") File01.xml -file02.xml を返します。

camel-case(InputString *as xs:string*, **SplitChars** *as xs:string*, **IsRegex** *as xs:boolean***)** を**xs:string** とする XP3.1 XQ3.1

SplitChars を使用して、次の大文字をトリガーする文字を決定し、入力文字列を**InputString** キャメルケースに変換します。 **SplitChars** は、**IsRegex = true()**の場合、または**IsRegex = false()**の場合、プレーン文字は正規表現として使用されます。出力文字列の最初の文字は大文字です。

サンプル

- **camel-case**("setname getname", "set|get", true()) setName getName を返します。
- **camel-case**("altova\documents\testcases", "\", false()) Altova \Documents\Testcases を返します。

char [altova:]

char(Position *as xs:integer***)** を**xs:string** とする XP3.1 XQ3.1

xs:string.Position に対してのコンテキストアイテムの値を変換することにより得られた文字列内の Position 引数により指定されたポジションにある文字を含む文字列を返します。引数により提出されたインデックスに文字が存在しない場合、結果文字列は空です。

サンプル

コンテキスト アイテムが **1234ABCD** の場合:

- **char**(2) は2 を返します。
- **char**(5) はA を返します。
- **char**(9) は空の文字列を返します。
- **char**(-2) は空の文字列を返します。

char(InputString *as xs:string*, **Position** *as xs:integer*) を**xs:string** とする **XP3.1** **XQ3.1**

引数として提出された文字列内のPosition 引数により指定されたポジションでの文字を含む文字列を返します。Position 引数により提出されたインデックスに文字が存在しない場合、結果文字列は空です。

サンプル

- **char**("2014-01-15", 5) は- を返します。
- **char**("USA", 1) はU を返します。
- **char**("USA", 10) は空の文字列を返します。
- **char**("USA", -2) は空の文字列を返します。

first-chars [altova:]

first-chars(X-Number *as xs:integer*) を**xs:string** とする **XP3.1** **XQ3.1**

xs:string に対するコンテキストアイテムの値を変換することにより得られた最初のX-Number 文字を含む文字列を返します。

サンプル

コンテキストアイテムが **1234ABCD** の場合:

- **first-chars**(2) は12 を返します。
- **first-chars**(5) は1234A を返します。
- **first-chars**(9) は1234ABCD を返します。

first-chars(InputString *as xs:string*, **X-Number** *as xs:integer*) を**xs:string** とする **XP3.1** **XQ3.1**

InputString 引数として提出された文字列の最初の文字を含む文字列を返します。

サンプル

- **first-chars**("2014-01-15", 5) は2014- を返します。
- **first-chars**("USA", 1) はU を返します。

last-chars [altova:]

last-chars(X-Number *as xs:integer*) を**xs:string** とする **XP3.1** **XQ3.1**

xs:string に対してのコンテキストアイテムの値の変換より取得された文字列の最後のX-Number 文字を含んでいる文字列を返します。

サンプル

コンテキストアイテムが **1234ABCD** の場合:

- **last-chars**(2) はCD を返します。
- **last-chars**(5) は4ABCD を返します。
- **last-chars**(9) は1234ABCD を返します。

last-chars(InputString *as xs:string*, **X-Number** *as xs:integer*) as **xs:string** とする XP3.1 XQ3.1
引数として提出された文字列の最後の X-Number 文字を含んでいる文字列を返します。

サンプル

- **last-chars**("2014-01-15", 5) は 01-15 を返します。
- **last-chars**("USA", 10) は USA を返します。

pad-string-left [altova:]

pad-string-left(StringToPad *as xs:string*, **StringLength** *as xs:integer*, **PadCharacter** *as xs:string*) を **xs:string** とする XP3.1 XQ3.1
PadCharacter 引数は1文字です。文字列の左側にパッドされ、この数が、StringLength 引数の整数の値と等しくなるように StringToPad の文字の数を増やします。StringLength 引数は任意の整数の値 (正数または負数) を持つことができますが、パディングは、StringLength の値が StringToPad 内の文字数よりも多い場合の場合のみ発生します。もし、StringToPad が StringLength の値よりも多くの文字数も持つ場合、StringToPad は変更されません。

サンプル

- **pad-string-left**('AP', 1, 'Z') は 'AP' を返します。
- **pad-string-left**('AP', 2, 'Z') は 'AP' を返します。
- **pad-string-left**('AP', 3, 'Z') は 'ZAP' を返します。
- **pad-string-left**('AP', 4, 'Z') は 'ZZAP' を返します。
- **pad-string-left**('AP', -3, 'Z') は 'AP' を返します。
- **pad-string-left**('AP', 3, 'YZ') は [パッド文字が長すぎます] エラーを返します。

pad-string-right [altova:]

pad-string-right(StringToPad *as xs:string*, **StringLength** *as xs:integer*, **PadCharacter** *as xs:string*) を **xs:string** とする XP3.1 XQ3.1
PadCharacter 引数は1文字です。文字列の右側にパッドされ、この数が、StringLength 引数の整数の値と等しくなるように StringToPad の文字の数を増やします。StringLength 引数は任意の整数の値 (正数または負数) を持つことができますが、パディングは、StringLength の値が StringToPad 内の文字数よりも多い場合の場合のみ発生します。もし、StringToPad が StringLength の値よりも多くの文字数も持つ場合、StringToPad は変更されません。

サンプル

- **pad-string-right**('AP', 1, 'Z') を 'AP' を返します。
- **pad-string-right**('AP', 2, 'Z') を 'AP' を返します。
- **pad-string-right**('AP', 3, 'Z') を 'APZ' を返します。
- **pad-string-right**('AP', 4, 'Z') を 'APZZ' を返します。
- **pad-string-right**('AP', -3, 'Z') を 'AP' を返します。
- **pad-string-right**('AP', 3, 'YZ') は [パッド文字が長すぎます] エラーを返します。

repeat-string [altova:]

repeat-string(InputString *as xs:string*, **Repeats** *as xs:integer*) を **xs:string** とする XP2 XQ1 XP3 XQ3
最初の InputString 引数により構成される文字列、Repeats 回繰り返してを生成します。

サンプル

- **repeat-string**("Altova #", 3) は "Altova #Altova #Altova #" を返します。

substring-after-last [altova:]

substring-after-last(MainString *as xs:string*, **CheckString** *as xs:string*) を **xs:string** とする **XP3.1 XQ3.1**

CheckString が MainString 内で検出された場合、MainString 内の CheckString が発生した後のサブ文字列が返されます。MainString 内で CheckString が検出されない場合、空の文字列が返されます。CheckString が空の文字列の場合、MainString 全体が返されます。一度以上発生する場合、CheckString の最後の発生の後のサブ文字列が返されます。

サンプル

- **substring-after-last**('ABCDEFGH', 'B') は 'CDEFGH' を返します。
- **substring-after-last**('ABCDEFGH', 'BC') は 'DEFGH' を返します。
- **substring-after-last**('ABCDEFGH', 'BD') は '' を返します。
- **substring-after-last**('ABCDEFGH', 'Z') は '' を返します。
- **substring-after-last**('ABCDEFGH', '') は 'ABCDEFGH' を返します。
- **substring-after-last**('ABCD-ABCD', 'B') は 'CD' を返します。
- **substring-after-last**('ABCD-ABCD-ABCD', 'BCD') は '' を返します。

substring-before-last [altova:]

substring-before-last(MainString *as xs:string*, **CheckString** *as xs:string*) を **xs:string** とする **XP3.1 XQ3.1**

CheckString が MainString 内で検出された場合、MainString 内の CheckString が発生する前のサブ文字列が返されます。MainString 内で CheckString が一度以上発生する場合、CheckString の最後の発生の前のサブ文字列が返されます。

サンプル

- **substring-before-last**('ABCDEFGH', 'B') は 'A' を返します。
- **substring-before-last**('ABCDEFGH', 'BC') は 'A' を返します。
- **substring-before-last**('ABCDEFGH', 'BD') は '' を返します。
- **substring-before-last**('ABCDEFGH', 'Z') は '' を返します。
- **substring-before-last**('ABCDEFGH', '') は '' を返します。
- **substring-before-last**('ABCD-ABCD', 'B') は 'ABCD-A' を返します。
- **substring-before-last**('ABCD-ABCD-ABCD', 'ABCD') は 'ABCD-ABCD-' を返します。

substring-pos [altova:]

substring-pos(StringToCheck *as xs:string*, **StringToFind** *as xs:string*) を **xs:integer** とする **XP3.1 XQ3.1**

StringToCheck 内での StringToFind の最初の発生の文字位置を整数として返します。StringToCheck の最初の文字は、位置 1 にあります。StringToFind が StringToCheck 内で発生しない場合、整数 0 が返されます。第 2 またはその後の StringToCheck, 発生を確認するには、この関数の次の署名を確認してください。

サンプル

- **substring-pos**('Altova', 'to') 3 を返します。
- **substring-pos**('Altova', 'tov') 3 を返します。
- **substring-pos**('Altova', 'tv') returns 0 を返します。
- **substring-pos**('AltovaAltova', 'to') 3 を返します。

substring-pos(StringToCheck *as xs:string*, **StringToFind** *as xs:string*, **Integer** *as xs:integer*) を**xs:integer** とする **XP3.1** **XQ3.1**

StringToCheck 内でのStringToFind の最初の発生の文字位置を返します。Integer 引数により与えられた、文字位置からStringToFind の検索が開始されます。この位置の前の文字サブ文字列は検索されません。しかし、返された整数は、全体文字列 StringToCheck の検索された文字列の位置です。この署名は、StringToCheck 内で複数回発生する、第 2 または後の発生を検索する際に役に立ちます。StringToFind がStringToCheck 内で発生しない場合、整数 0 が返されます。

サンプル

- **substring-pos**('Altova', 'to', 1) 3 を返します。
- **substring-pos**('Altova', 'to', 3) 3 を返します。
- **substring-pos**('Altova', 'to', 4) 0 を返します。
- **substring-pos**('Altova-Altova', 'to', 0) 3 を返します。
- **substring-pos**('Altova-Altova', 'to', 4) 10 を返します。

trim-string [altova:]

trim-string(InputString *as xs:string*) を**xs:string** とする **XP3.1** **XQ3.1**

この関数はxs:string 引数を必要とし、先頭または後続の空白を削除し、トリミングされたxs:string を返します。

サンプル

- **trim-string**(" Hello World ")) は"Hello World" を返します。
- **trim-string**("Hello World ")) は"Hello World" を返します。
- **trim-string**(" Hello World")) は"Hello World" を返します。
- **trim-string**("Hello World")) は"Hello World" を返します。
- **trim-string**("Hello World"))は "Hello World" を返します。

trim-string-left [altova:]

trim-string-left(InputString *as xs:string*) を**xs:string** とする **XP3.1** **XQ3.1**

この関数はxs:string 引数を必要とし、先頭または後続の空白を削除し、トリミングされたxs:string を返します。

サンプル

- **trim-string-left**(" Hello World ")) は "Hello World " を返します。
- **trim-string-left**("Hello World ")) は"Hello World " を返します。
- **trim-string-left**(" Hello World")) は"Hello World" を返します。
- **trim-string-left**("Hello World")) は"Hello World" を返します。
- **trim-string-left**("Hello World")) は"Hello World" を返します。

trim-string-right [altova:]

trim-string-right(InputString *as xs:string*) を**xs:string** とする **XP3.1** **XQ3.1**

この関数はxs:string 引数を必要とし、先頭または後続の空白を削除し、トリミングされたxs:string を返します。

サンプル

- **trim-string-right**(" Hello World ")) は" Hello World" を返します。
- **trim-string-right**("Hello World ")) は"Hello World" を返します。
- **trim-string-right**(" Hello World")) は" Hello World" を返します。
- **trim-string-right**("Hello World")) は"Hello World" を返します。

- `trim-string-right("Hello World")` は`"Hello World"` を返します。

XPath/ XQuery 関数: その他

以下の一般使用のためのXPath/XQuery 拡張関数は、MobileTogether Designer の現在のバージョンによりサポートされています。

関数の名前指定と言語の適用性

Altova 拡張関数はXPath/XQuery 式で使用することができ、XPath、XQuery、およびXSLT 関数の標準ライブラリで使用可能な機能に更なる機能性を与えます。

- Altova 拡張関数を標準関数ライブラリと区別するために、このドキュメントでは、Altova 拡張関数にサフィックス **[altova:]**を付けて名前指定されています。例: **add-years-to-date [altova:]**。
- XPath/XQuery 式でAltova 拡張機能を使用する場合、プレフィックスまたはサフィックス**なし**で、関数を使用する同様、Altova 拡張関数を使用することができます。Altova 拡張子を次のように使用してください: **add-years-to-date(xs:date("2014-01-15"), 10)**。

XPath 関数 (XSLT 内のXPath 式で使用):	**XP1** **XP2** **XP3.1**
XSLT 関数 (XSLT 内のXPath 式で使用):	**XSLT1** **XSLT2** **XSLT3**
XQuery 関数 (XQuery 内のXQuery 式で使用):	**XQ1** **XQ3.1**

URI 関数

get-temp-folder [altova:]

get-temp-folder() を`xs:string` とする **XP2** **XQ1** **XP3** **XQ3**

この関数は引数を必要としません。この関数は現在のユーザーの一時的なフォルダーへのパスを返します。

サンプル

- **get-temp-folder**() は、マシーン上で`xs:string` として`C:\Users\<UserName>\AppData\Local\Temp\` と類似したパスを返します。

[Top]

22.2 ライセンス情報

ライセンス情報

- ソフトウェアの配布 に関する情報
- ソフトウェアのアクティベーションとライセンスの計測 に関する情報
- このソフトウェアに関連する知的所有権 に関する情報
- このソフトウェア製品の使用に関するエンドユーザー使用許諾契約書

本製品を使用する前に、上記の情報をよくお読みください。ソフトウェアのインストール時に上記のすべての条件に同意したとみなされ、お客様は上記の条件に拘束されることを同意したとみなされます。

電子的なソフトウェアの配布

この製品は電子的なソフトウェアの配布により利用することが可能で、この配布方法により、以下のユニークなメリットがあります：

- 購入を決定する前に、無料でソフトウェアを試用することができます。
- ソフトウェアの購入を決定した際には、Altova Web サイトにて注文を行います。すぐにライセンス登録された製品の使用を開始することができます。
- オンラインにて注文を行うと、常に最新のソフトウェアをご利用いただけます。
- 製品パッケージには包括的なヘルプシステムが画面上に表示されます。最新バージョンのユーザーマニュアルは www.altova.com 上にあり、(i) HTML フォーマットによる閲覧、ならびに (ii) PDF フォーマットのダウンロードと印刷に対応しております。

組織内でソフトウェアの評価を行う

評価版のソフトウェアを組織内のネットワークにて配布したい場合、またはインターネットに接続されていないコンピューターにてソフトウェアを使用する場合、どのような状態でも改変されていないことを条件に、セットアッププログラムだけの配布を行うことが可能です。ソフトウェアインストーラーへアクセスした人は、例外なく30日間の評価ライセンスキーコードをリクエストして、試用期間が経過した後は、製品を使い続けるためにライセンスの購入を行う必要があります。

詳細については、このセクション最後にあるAltova ソフトウェアライセンス使用許諾契約書 を参照ください。

ソフトウェアのアクティベーションとライセンスの計測

Altova のソフトウェアアクティベーションの一部として、ソフトウェアにより内部ネットワークまたはインターネットへの接続を行い、インストール時、登録時、Altova により使用されるライセンスサーバーの更新やライセンスの正当性を検証することで、ソフトウェアの不正な使用を防ぎ、顧客サービスを向上するため、ライセンスに関する情報を送信することがあります。アクティベーションにより、オペレーティングシステムや IP アドレス、日付 時刻、ソフトウェアのバージョン、コンピュータ名などのライセンスに関する情報が、お使いのコンピューターとAltova ライセンスサーバー間にでやり取りされます。

お使いのAltova 製品にはライセンス計測モジュールが内蔵されており、エンドユーザー使用許諾契約書の意図しない違反を防ぎます。お使いの製品はシングルユーザーまたはマルチユーザーとしてインストールされており、ライセンス計測モジュールにより、ライセンスされている数を超えたユーザーが同時に製品を使用することが無いことが保証されます。

このライセンス計測技術により、ローカルエリア接続 (LAN) において、別々のコンピューター間で動作しているアプリケーションインスタンス間の通信が行われます。

シングルライセンス

ライセンス計測プロセスの一部としてアプリケーションが起動すると、ソフトウェアにより短いデータグラムがブロードキャストにより送信され、同一のネットワークセグメントにある他のコンピューターにてプログラムが動作していないかのチェックが行われます。応答が無い場合は、アプリケーションの他インスタンスから送信される信号に応えるため、ポートが開かれます。

マルチライセンス

同一のLAN 内にて2つ以上のアプリケーションインスタンスが使用された場合、スタートアップ時に、これらインスタンス間において通信が行われます。これらのインスタンス間にてキーコードのやりとりが行われ、購入された数のライセンスを超えてインスタンスが起動しないように保証することができます。このようなライセンス計測システムはUNIX やデータベース開発ツールにて広く使用されているもので、Altova ユーザーはリーズナブルな価格にて同時使用マルチユーザーライセンスを購入することができます。

弊社はアプリケーションのデザインも行っており、少数の小さなネットワークパケットを送信することで、ネットワークに対する負荷を最小限に抑えております。Altova により使用される2799番 TCP/IP ポートはIANA により公式登録されており（詳細は（ IANA Web サイト(http://www.iana.org/) を参照ください）、弊社のライセンス計測モジュールも既にテストされたものです。

ファイヤーウォールを使用している場合、2799番ポートにて Altova 製品が動作しているコンピューター同士が通信しているのに気づかれるかも知れません。その他の手段によりライセンス使用許諾書の内容が守られることを保証できる限り、組織間の異なるグループにおいてこのようなトラフィックをブロックすることは勿論可能です。

お使いのコンピューターがオンラインの場合、ライセンス計測技術とは関係の無い数々の便利な機能を利用することができるようになります。

知的所有権

Altova ソフトウェアならびにAltova により認められたコピーは、Altova ならびにその供給者が知的所有権を有しています。ソフトウェアの構造や構成、コードはAltova ならびにその供給者にとって価値のある企業秘密かつ機密情報となります。このソフトウェアは米国の著作権法により保護されており、国際条項ならびにソフトウェアが使用されている国の法律が適用されます。Altova はこのソフトウェアに関する特許、企業秘密、商標に関する所有権ならびにこれらに限定されないその他の知的財産権を有しており、Altova の所有権には、このソフトウェアならびにそれに付随する印刷物に抱合された画像、写真、アニメーション、映像、音声、音楽、テキスト、そして「アプレット」が含まれます。著作権侵害に関する届け出は、Altova Web Site に記されているAltova 著作権エージェントへお送りください。

Altova ソフトウェアには国際条項により保護されているサードパーティソフトウェアも含まれており、http://www.altova.com/legal_3rdparty.html にて詳細が記されている著作権法なども含め、しかしこれに限定されないかたちで知的所有権により保護されます。

その他全ての名前または商標は、対応する所有者が有するものです。

Altova MobileTogether Designer エンドユーザー使用許諾契約書

THIS IS A LEGAL DOCUMENT - - RETAIN FOR YOUR RECORDS

ALTOVA® MOBILETOGETHER DESIGNER END USER LICENSE AGREEMENT

Licensor:
Altova GmbH
Rudolfsplatz 13a/9
A-1010 Wien
Austria

Important - Read Carefully. Notice to User:

This End User License Agreement ("Agreement") is a legal document between you and Altova GmbH ("Altova"). It is important that you read this document before using the Altova-provided software ("Software") and any accompanying documentation, including, without limitation printed materials, 'online' files, or electronic documentation ("Documentation"). By clicking the "I accept" and "Next" buttons below, or by installing, or otherwise using the Software, you agree to be bound by the terms of this Agreement as well as the Altova Privacy Policy ("Privacy Policy") including, without limitation, the warranty disclaimers, limitation of liability, data use and termination provisions below, whether or not you decide to purchase the Software. You agree that this agreement is enforceable like any written agreement negotiated and signed by you. If you do not agree, you are not licensed to use the Software, and you must destroy any downloaded copies of the Software in your possession or control. You may print a copy of this Agreement as part of the installation process at the time of acceptance. Alternatively, a copy of this Agreement may be found at http://www.altova.com/legal.html and a copy of the Privacy Policy may be found at http://www.altova.com/privacy.

1. SOFTWARE LICENSE

(a) License Grant.

(i) Upon your acceptance of this Agreement Altova grants you a non-exclusive, non-transferable (except as provided below), limited license, without the right to grant sublicenses, to install and use a copy of the Software on one compatible personal computer or workstation.

(ii) You may not use the Software to develop and distribute other software programs that directly compete with any Altova software or service without prior written permission. Altova reserves all other rights in and to the Software.

(b) AppStore Apps.

(i) The AppStore App feature in the Software enables users to generate source code. Your license to install and use a copy of the Software as provided herein permits you to generate source code based on (i) Altova Library modules that are included in the Software (such generated code hereinafter referred to as the "Restricted Source Code") and (ii) schemas or mappings that you create or provide (such code as may be generated from your schema or mapping source materials hereinafter referred to as the "Unrestricted Source Code"). In addition to the rights granted herein, Altova grants you a non-exclusive, non-transferable, limited license to compile the complete generated code (comprised of the combination of the Restricted Source Code and the Unrestricted Source Code) into executable object code form, and to use, copy, distribute or license that executable. You may not distribute or redistribute, sublicense, sell, or transfer the Restricted Source Code to a third-party in the un-compiled form. Notwithstanding anything to the contrary herein, you may not distribute, incorporate or combine with other software, or otherwise use the Altova Library modules or Restricted Source Code, or any Altova intellectual property embodied in or associated with the Altova Library modules or Restricted Source Code, in any manner that would subject the Restricted Source Code to the terms of a copyleft, free software or open source license that would require the Restricted Source Code or Altova Library modules source code to be disclosed in source code form. Notwithstanding anything to the contrary herein, you may not use the Software to develop and distribute other

software programs that directly compete with any Altova software or service without prior written permission. Altova reserves all other rights in and to the Software.

(ii) In the event Restricted Source Code is incorporated into executable object code form, you will include the following statement in (1) introductory splash screens, or if none, within one or more screens readily accessible by the end-user, and (2) in the electronic and/or hard copy documentation: "Portions of this program were developed using Altova® MobileTogether and includes libraries owned by Altova GmbH, Copyright © 2007-2016 Altova GmbH (www.altova.com)."

(iii) You agree not to use the Software to create an AppStore App for any unlawful or illegal activity, nor to develop any AppStore App that would commit or facilitate the commission of a crime, or other tortious, unlawful, or illegal act. You agree that, to the best of your knowledge and belief, your AppStore App will not violate, misappropriate, or infringe any Altova or third party copyrights, trademarks, rights of privacy and publicity, trade secrets, patents, or other proprietary or legal rights (e.g. musical composition or performance rights, video rights, photography or image rights, logo rights, third party data rights, etc. for content and materials that may be included in your AppStore App). Further, you agree not to use the Software to create any AppStore App or other software program that would disable, hack or otherwise interfere with any security, digital signing, digital rights management, content protection, verification or authentication mechanisms implemented in or by the Software or by other third party software, services or technology, or enable others to do so, unless otherwise permitted by the Altova in writing. AppStore Apps must not contain any malware, malicious or harmful code, program, or other internal component (e.g. computer viruses, trojan horses, "backdoors") and may not use any Software in a way that could damage, destroy, or adversely affect Altova software or services, or any other software, firmware, hardware, data, systems, services, or networks.

(iv) You agree that you are solely responsible for any data, content, or resources that you create, transmit or display through the AppStore App, and for the consequences of your actions.

(c) Backup and Archival Copies. You may make one (1) backup and one (1) archival copy of the Software, provided your backup and archival copies are not installed or used on any computer and further provided that all such copies shall bear the original and unmodified copyright, patent and other intellectual property markings that appear on or in the Software.

(d) Title. Title to the Software is not transferred to you. Ownership of all copies of the Software and of copies made by you is vested in Altova, subject to the rights of use granted to you in this Agreement. As between you and Altova, documents, files, stylesheets, generated program code (including the Unrestricted Source Code), schemas that are authored or created by you via your utilization of the Software, and AppStore Apps, in accordance with its Documentation and the terms of this Agreement, are your property. "AppStore Apps" means the app authored or created by you via your utilization of the "create AppStore App" option in Altova MobileTogether Designer and connected to the Altova MobileTogether Server.

(e) Reverse Engineering. Except and to the limited extent as may be otherwise specifically provided by applicable law in the European Union, you may not reverse engineer, decompile, disassemble or otherwise attempt to discover the source code, underlying ideas, underlying user interface techniques or algorithms of the Software by any means whatsoever, directly or indirectly, or disclose any of the foregoing, except to the extent you may be expressly permitted to decompile under applicable law in the European Union, if it is essential to do so in order to achieve operability of the Software with another software program, and you have first requested Altova to provide the information necessary to achieve such operability and Altova has not made such information available. Altova has the right to impose reasonable conditions and to request a reasonable fee before providing such information. Any information supplied by Altova or obtained by you, as permitted hereunder, may only be used by you for the purpose described herein and may not be disclosed to any third party or used to create any software which is substantially similar to the expression of the Software. Requests for information from users in the European Union with respect to the above should be directed to the Altova Customer Support Department.

(f) Other Restrictions. You may not loan, rent, lease, sublicense, distribute or otherwise

transfer all or any portion of the Software to third parties except as otherwise expressly provided. You may not copy the Software except as expressly set forth herein, and any copies that you are permitted to make pursuant to this Agreement must contain the same copyright, patent and other intellectual property markings that appear on or in the Software. You may not modify, adapt or translate the Software. You may not, directly or indirectly, encumber or suffer to exist any lien or security interest on the Software; knowingly take any action that would cause the Software to be placed in the public domain; or use the Software in any computer environment not specified in this Agreement. You may not permit any use of or access to the Software by any third party in connection with a commercial service offering, such as for a cloud-based or web-based SaaS offering.

You will comply with applicable law and Altova's instructions regarding the use of the Software. You agree to notify your employees and agents who may have access to the Software of the restrictions contained in this Agreement and to ensure their compliance with these restrictions.

(g) NO GUARANTEE. THE SOFTWARE IS NEITHER GUARANTEED NOR WARRANTED TO BE ERROR-FREE NOR SHALL ANY LIABILITY BE ASSUMED BY ALTOVA IN THIS RESPECT. NOTWITHSTANDING ANY SUPPORT FOR ANY TECHNICAL STANDARD, THE SOFTWARE IS NOT INTENDED FOR USE IN OR IN CONNECTION WITH, WITHOUT LIMITATION, THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION, COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL EQUIPMENT, MEDICAL DEVICES OR LIFE SUPPORT SYSTEMS, MEDICAL OR HEALTH CARE APPLICATIONS, OR OTHER APPLICATIONS WHERE THE FAILURE OF THE SOFTWARE OR ERRORS IN DATA PROCESSING COULD LEAD TO DEATH, PERSONAL INJURY OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE. YOU AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR THE ACCURACY AND ADEQUACY OF THE SOFTWARE AND ANY DATA GENERATED OR PROCESSED BY THE SOFTWARE FOR YOUR INTENDED USE AND YOU WILL DEFEND, INDEMNIFY AND HOLD ALTOVA, ITS OFFICERS AND EMPLOYEES HARMLESS FROM ANY THIRD PARTY CLAIMS, DEMANDS, OR SUITS THAT ARE BASED UPON THE ACCURACY AND ADEQUACY OF THE SOFTWARE IN YOUR USE OR ANY DATA GENERATED BY THE SOFTWARE IN YOUR USE.

2. INTELLECTUAL PROPERTY RIGHTS

You acknowledge that the Software and any copies that you are authorized by Altova to make are the intellectual property of and are owned by Altova and its suppliers. The structure, organization and code of the Software are the valuable trade secrets and confidential information of Altova and its suppliers. The Software is protected by copyright, including without limitation by United States Copyright Law, international treaty provisions and applicable laws in the country in which it is being used. You acknowledge that Altova retains the ownership of all patents, copyrights, trade secrets, trademarks and other intellectual property rights pertaining to the Software, and that Altova's ownership rights extend to any images, photographs, animations, videos, audio, music, text and "applets" incorporated into the Software and all accompanying printed materials. You will take no actions which adversely affect Altova's intellectual property rights in the Software. Trademarks shall be used in accordance with accepted trademark practice, including identification of trademark owners' names. Trademarks may only be used to identify printed output produced by the Software, and such use of any trademark does not give you any right of ownership in that trademark. Altova®, XMLSpy®, Authentic®, StyleVision®, MapForce®, UModel®, DatabaseSpy®, DiffDog®, SchemaAgent®, SemanticWorks®, MissionKit®, Markup Your Mind®, Nanonull™, MobileTogether®, MobileTogether Server®, MobileTogether Designer®, MobileTogether Mobile App®, RaptorXML®, RaptorXML Server®, RaptorXML +XBRL Server®, Powered By RaptorXML®, FlowForce Server®, StyleVision Server®, and MapForce Server® are trademarks of Altova GmbH. (pending or registered in numerous countries).Unicode and the Unicode Logo are trademarks of Unicode, Inc. Windows, Windows XP, Windows Vista, Windows 7, and Windows 8 are trademarks of Microsoft. W3C, CSS, DOM, MathML, RDF, XHTML, XML and XSL are trademarks (registered in numerous countries) of the World Wide Web Consortium (W3C); marks of the W3C are registered and held by its host institutions, MIT, INRIA and Keio. Except as expressly stated above, this Agreement does not grant you any intellectual property rights in the Software. Notifications of claimed copyright infringement should be sent to Altova's copyright agent as further provided on the Altova Web Site.

3. LIMITED WARRANTY AND LIMITATION OF LIABILITY

(a) Limited Warranty and Customer Remedies. YOU ACKNOWLEDGE THAT THE SOFTWARE IS PROVIDED TO YOU **"AS-IS" WITH NO WARRANTIES FOR USE OR PERFORMANCE,** AND ALTOVA DISCLAIMS ANY WARRANTY OR LIABILITY OBLIGATIONS TO YOU OF ANY KIND, WHETHER EXPRESS OR IMPLIED. WHERE LEGALLY LIABILITY CANNOT BE EXCLUDED FOR THE SOFTWARE, BUT IT MAY BE LIMITED, ALTOVA'S LIABILITY AND THAT OF ITS SUPPLIERS SHALL BE LIMITED TO THE SUM OF FIFTY DOLLARS (USD $50) IN TOTAL.

(b) No Other Warranties and Disclaimer. THE FOREGOING LIMITED WARRANTY AND REMEDIES STATE THE SOLE AND EXCLUSIVE REMEDIES FOR ALTOVA OR ITS SUPPLIER'S BREACH OF WARRANTY. ALTOVA AND ITS SUPPLIERS DO NOT AND CANNOT WARRANT THE PERFORMANCE OR RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE. EXCEPT FOR THE FOREGOING LIMITED WARRANTY, AND FOR ANY WARRANTY, CONDITION, REPRESENTATION OR TERM TO THE EXTENT WHICH THE SAME CANNOT OR MAY NOT BE EXCLUDED OR LIMITED BY LAW APPLICABLE TO YOU IN YOUR JURISDICTION, ALTOVA AND ITS SUPPLIERS MAKE NO WARRANTIES, CONDITIONS, REPRESENTATIONS OR TERMS, EXPRESS OR IMPLIED, WHETHER BY STATUTE, COMMON LAW, CUSTOM, USAGE OR OTHERWISE AS TO ANY OTHER MATTERS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, ALTOVA AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, INFORMATIONAL CONTENT OR ACCURACY, QUIET ENJOYMENT, TITLE AND NON-INFRINGEMENT, WITH REGARD TO THE SOFTWARE, AND THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS, WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.

(c) Limitation of Liability. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW EVEN IF A REMEDY FAILS ITS ESSENTIAL PURPOSE, IN NO EVENT SHALL ALTOVA OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE OR THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES, EVEN IF ALTOVA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Because some states and jurisdictions do not allow the exclusion or limitation of liability, the above limitation may not apply to you. In such states and jurisdictions, Altova's liability shall be limited to the greatest extent permitted by law and the limitations or exclusions of warranties and liability contained herein do not prejudice applicable statutory consumer rights of person acquiring goods otherwise than in the course of business. The disclaimer and limited liability above are fundamental to this Agreement between Altova and you.

(d) Waiver and Indemnity. BY USING THE SOFTWARE, YOU AGREE, TO THE EXTENT PERMITTED BY LAW, TO SAVE AND PROTECT, HOLD HARMLESS, INDEMNIFY AND DEFEND ALTOVA, ITS DIRECTORS, OFFICERS, EMPLOYEES, AFFILIATES, AGENTS, CONTRACTORS, AND LICENSORS AGAINST ANY AND ALL LIABILITY, CAUSES OF ACTION, CLAIMS, LOSS DAMAGE OR COST AND EXPENSE ARISING FROM, ALLEGEDLY ARISING FROM, OR RESULTING DIRECTLY OR INDIRECTLY FROM ANY ACTS OF THE LICENSEE OR ANY OF ITS OFFICERS, EMPLOYEES, INDEPENDENT CONTRACTORS OR AGENTS DONE IN THE PERFORMANCE, OPERATION, OR USE OF THE SOFTWARE, OR ANY ACT DONE UNDER PRETENDED AUTHORITY OF THIS LICENSE. THIS AGREEMENT TO INDEMNIFY AND HOLD ALTOVA HARMLESS SHALL INCLUDE ANY COSTS INCURRED BY ALTOVA IN DEFENDING ANY ACTION INVOLVING AN ACT BY THE LICENSEE OR ANY OF ITS OFFICERS, EMPLOYEES, INDEPENDENT CONTRACTORS OR AGENTS, AND SHALL INCLUDE ANY ATTORNEY'S FEES INCURRED BY ALTOVA.

4. SUPPORT AND MAINTENANCE

Altova offers "Support & Maintenance Package(s)" ("SMP") for the Software only if you have obtained a valid license for MobileTogether Server, which may be obtained from Altova at

www.altova.com. The Support Period shall coincide with the MobileTogether Server Software license term. The terms of SMP are set forth in the Altova MobileTogether Server Software License Agreement located at http://www.altova.com/license_agreements.html.

5. SOFTWARE ACTIVATION AND UPDATES

(a) License Metering. The Software includes a built-in license metering module that is designed to assist you with monitoring license compliance in small local networks. The metering module attempts to communicate with other machines on your local area network. You permit Altova to use your internal network for license monitoring for this purpose. This license metering module may be used to assist with your license compliance but should not be the sole method. Should your firewall settings block said communications, you must deploy an accurate means of monitoring usage by the end user and preventing users from using the Software more than the Permitted Number.

(b) License Compliance Monitoring. You are required to utilize a process or tool to ensure that the Permitted Number is not exceeded. Without prejudice or waiver of any potential violations of the Agreement, Altova may provide you with additional compliance tools should you be unable to accurately account for license usage within your organization. If provided with such a tool by Altova, you (a) are required to use it in order to comply with the terms of this Agreement and (b) permit Altova to use your internal network for license monitoring and metering and to generate compliance reports that are communicated to Altova from time to time.

(c) Software Activation. **The Software may use your internal network and Internet connection for the purpose of transmitting license-related data at the time of installation, registration, use, or update to an Altova-operated license server and validating the authenticity of the license-related data in order to protect Altova against unlicensed or illegal use of the Software and to improve customer service. Activation is based on the exchange of license related data between your computer and the Altova license server. You agree that Altova may use these measures and you agree to follow any applicable requirements. You further agree that use of license key codes that are not or were not generated by Altova and lawfully obtained from Altova, or an authorized reseller as part of an effort to activate or use the Software violates Altova's intellectual property rights as well as the terms of this Agreement. You agree that efforts to circumvent or disable Altova's copyright protection mechanisms or License Server violate Altova's intellectual property rights as well as the terms of this Agreement. Altova expressly reserves the rights to seek all available legal and equitable remedies to prevent such actions and to recover lost profits, damages and costs.**

(d) LiveUpdate. Altova provides a new LiveUpdate notification service to you, which is free of charge. Altova may use your internal network and Internet connection for the purpose of transmitting license-related data to an Altova-operated LiveUpdate server to validate your license at appropriate intervals and determine if there is any update available for you.

(e) Use of Data. The terms and conditions of the Privacy Policy are set out in full at http://www.altova.com/privacy and are incorporated by reference into this Agreement. By your acceptance of the terms of this Agreement and/or use of the Software, you authorize the collection, use and disclosure of information collected by Altova for the purposes provided for in this Agreement and/or the Privacy Policy. Altova has the right in its sole discretion to amend this provision of the Agreement and/or Privacy Policy at any time. You are encouraged to review the terms of the Privacy Policy as posted on the Altova Web site from time to time.

(f) Audit Rights. You agree that Altova may audit your use of the Software for compliance with the terms of this Agreement at any time, upon reasonable notice. In the event that such audit reveals any use of the Software by you other than in full compliance with the terms of this Agreement, you shall reimburse Altova for all reasonable expenses related to such audit in addition to any other liabilities you may incur as a result of such non-compliance.

(g) Notice to European Users. Please note that the information as described in paragraph 5(d) above may be transferred outside of the European Economic Area, for purposes of processing, analysis, and review, by Altova, Inc., a company located in Beverly, Massachusetts,

U.S.A., or its subsidiaries or Altova's subsidiaries or divisions, or authorized partners, located worldwide. You are advised that the United States uses a sectoral model of privacy protection that relies on a mix of legislation, governmental regulation, and self-regulation. You are further advised that the Council of the European Union has found that this model does not provide "adequate" privacy protections as contemplated by Article 25 of the European Union's Data Directive. (Directive 95/46/EC, 1995 O.J. (L 281) 31). Article 26 of the European Union's Data Directive allows for transfer of personal data from the European Union to a third country if the individual has unambiguously given his consent to the transfer of personal information, regardless of the third country's level of protection. By agreeing to this Agreement, you consent to the transfer of all such information to the United States and the processing of that information as described in this Agreement and the Privacy Policy.

6. TERM AND TERMINATION

This Agreement may be terminated (a) by your giving Altova written notice of termination; or (b) by Altova, at any time without prior notice. Upon any termination of the Agreement, you must cease all use of the Software that this Agreement governs, destroy all copies then in your possession or control and take such other actions as Altova may reasonably request to ensure that no copies of the Software remain in your possession or control. The terms and conditions set forth in Sections 1(c), 1(d), 1(e), 1(l), 2, 3, 5, 7, 8, and 9 survive termination as applicable.

7. RESTRICTED RIGHTS NOTICE AND EXPORT RESTRICTIONS

The Software was developed entirely at private expense and is commercial computer software provided with **RESTRICTED RIGHTS**. Use, duplication or disclosure by the U.S. Government or a U.S. Government contractor or subcontractor is subject to the restrictions set forth in this Agreement and as provided in FAR 12.211 and 12.212 (48 C.F.R. § 12.211 and 12.212) or DFARS 227. 7202 (48 C.F.R. § 227-7202) as applicable. Consistent with the above as applicable, Commercial Computer Software and Commercial Computer Documentation licensed to U.S. government end users only as commercial items and only with those rights as are granted to all other end users under the terms and conditions set forth in this Agreement. Manufacturer is Altova GmbH, Rudolfsplatz, 13a/9, A-1010 Vienna, Austria/EU. You may not use or otherwise export or re-export the Software or Documentation except as authorized by United States law and the laws of the jurisdiction in which the Software was obtained. In particular, but without limitation, the Software or Documentation may not be exported or re-exported (i) into (or to a national or resident of) any U.S. embargoed country or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Department of Commerce's Table of Denial Orders. By using the Software, you represent and warrant that you are not located in, under control of, or a national or resident of any such country or on any such list.

8. U.S. GOVERNMENT ENTITIES

Notwithstanding the foregoing, if you are an agency, instrumentality or department of the federal government of the United States, then this Agreement shall be governed in accordance with the laws of the United States of America, and in the absence of applicable federal law, the laws of the Commonwealth of Massachusetts will apply. Further, and notwithstanding anything to the contrary in this Agreement (including but not limited to Section 5 (Indemnification)), all claims, demands, complaints and disputes will be subject to the Contract Disputes Act (41 U.S.C. § § 7101 *et seq.*), the Tucker Act (28 U.S.C. § 1346(a) and § 1491), or the Federal Tort Claims Act (28 U.S.C. § § 1346(b), 2401-2402, 2671-2672, 2674-2680), FAR 1.601(a) and 43.102 (Contract Modifications); FAR 12.302(b), as applicable, or other applicable governing authority. For the avoidance of doubt, if you are an agency, instrumentality, or department of the federal, state or local government of the U.S. or a U.S. public and accredited educational institution, then your indemnification obligations are only applicable to the extent they would not cause you to violate any applicable law (e.g., the Anti-Deficiency Act), and you have any legally required authorization or authorizing statute.

9. THIRD PARTY SOFTWARE

The Software may contain third party software which requires notices and/or additional terms and conditions. Such required third party software notices and/or additional terms and conditions are located at our Website at http://www.altova.com/legal_3rdparty.html and are made

a part of and incorporated by reference into this Agreement. By accepting this Agreement, you are also accepting the additional terms and conditions, if any, set forth therein.

10. GENERAL PROVISIONS

If you are located in the European Union and are using the Software in the European Union and not in the United States, then this Agreement will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht, Wien (Commercial Court, Vienna) in connection with any such dispute or claim.

If you are located in the United States or are using the Software in the United States then this Agreement will be governed by and construed in accordance with the laws of the Commonwealth of Massachusetts, USA (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the federal or state courts of the Commonwealth of Massachusetts and you further agree and expressly consent to the exercise of personal jurisdiction in the federal or state courts of the Commonwealth of Massachusetts in connection with any such dispute or claim.

If you are located outside of the European Union or the United States and are not using the Software in the United States, then this Agreement will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht Wien (Commercial Court, Vienna) in connection with any such dispute or claim. This Agreement will not be governed by the conflict of law rules of any jurisdiction or the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded.

This Agreement contains the entire agreement and understanding of the parties with respect to the subject matter hereof, and supersedes all prior written and oral understandings of the parties with respect to the subject matter hereof. Any notice or other communication given under this Agreement shall be in writing and shall have been properly given by either of us to the other if sent by certified or registered mail, return receipt requested, or by overnight courier to the address shown on Altova 's Web site for Altova and the address shown in Altova 's records for you, or such other address as the parties may designate by notice given in the manner set forth above. This Agreement will bind and inure to the benefit of the parties and our respective heirs, personal and legal representatives, affiliates, successors and permitted assigns. The failure of either of us at any time to require performance of any provision hereof shall in no manner affect such party 's right at a later time to enforce the same or any other term of this Agreement. This Agreement may be amended only by a document in writing signed by both of us. In the event of a breach or threatened breach of this Agreement by either party, the other shall have all applicable equitable as well as legal remedies. Each party is duly authorized and empowered to enter into and perform this Agreement. If, for any reason, any provision of this Agreement is held invalid or otherwise unenforceable, such invalidity or unenforceability shall not affect the remainder of this Agreement, and this Agreement shall continue in full force and effect to the fullest extent allowed by law. The parties knowingly and expressly consent to the foregoing terms and conditions.

Last updated: 2015-11-01

(C) 2018 Altova GmbH

Index

$

$MT_SERVICE ページソース ,1097
シミュレーション内 ,996

A

addEventListener(), 1008
ADO,
データ接続インターフェイスとして ,834
接続のセットアップ ,840
ADO.NET,
接続のセットアップ ,847
Altova グローバルリソース ,
グローバルリソース ,954
グローバルリソース を参照する ,954
Altova 拡張関数 ,
チャート関数 (チャート関数を参照), 1235
Android ビーム ,532

B

Base64 エンコード イメージ ,
イメージ ,677

D

DatabaseSpy, 752
3D チャート ,761
チャートX- 軸 ,758
チャートY- 軸 ,760
チャートZ- 軸 ,761
チャートグリッド ,758, 760, 761
チャートサイズ ,762
チャートのタイトル ,750
チャートの色 ,756
チャートの背景 ,750
チャートの凡例 ,750
チャートフォント ,763
チャートフォントオプション ,763
DB,
データベースを参照 ,828
DB クエリ ビュー ,181
DB コミット トランザクションアクション ,596
DB データソース ,97
DB トランザクション ,588, 596, 597
DB トランザクションの開始アクション ,588
DB ロールバック トランザクションアクション ,597
DB 実行アクション ,590

E

Exif イメージ ,
イメージ ,679

F

Firebird,
ODBC を使用しての接続 ,868
FTP,
ページソースとして追加 ,233, 234
リクエストのための設定 ,234

G

GUI,
ユーザー インターフェイスを参照 ,176

H

HTTP,
ページソースとして追加 ,233, 234
リクエストのための設定 ,234

I

IBM DB2,
ODBC を使用しての接続 ,873

IBM DB2 for i,
ODBC を使用しての接続 ,878
IBM Informix,
JDBC を使用しての接続 ,881
If- Then アクション ,623
If- Then- Else アクション ,624

J

JDBC,
OS X Yosemite 上での Oracle 接続のセットアップ ,919
Teradata への接続 ,914
データ接続インターフェイスとして ,834
接続のセットアップ (Linux), 918
接続のセットアップ (macOS), 918
接続の設定 (Windows), 856
JWT,
作成 ,1015
JWT 認証 ,1013
MobileTogether Server 設定 ,1015, 1017
共有された秘密 ,1015
対称キー ,1015
秘密 ?公開キー ,1017
非対称暗号化 ,1017
JWT 認証のサンプル ,1039

L

Let アクション ,627
Linux,
サーバー実行ファイルをデプロイ ,916
サポートされるデータベース ,916
データベース接続のセットアップ ,916
Loop アクション ,625

M

Mac OS,
サーバー実行ファイルをデプロイ ,916
サポートされるデータベース ,916
データベース接続のセットアップ ,916
MapForce 転送アクション ,490
MariaDB,
ODBC を使用した接続 ,883
MFX ファイル ,490
Microsoft Access,
ADO を介しての接続 ,840
ADO を使用しての接続 ,885
Microsoft SQL Server,
ADO を使用しての接続 ,887
ODBC を使用しての接続 ,890
MobileTogether,
システム必要条件 ,21
用語 ,23
MobileTogether Client, 21
MobileTogether Server, 21
MobileTogether Server Advenced Edition, 1096
MobileTogether 概要 ,21
MobileTogetherのための XPath 拡張関数 ,798
MobileTogetherのための拡張関数 ,798
MT 記録ファイル ,1050
MT_SERVICE ページソース ,
$MT_SERVICE ページソース参照 ,996
MTPNSIM ファイル ,719, 992
MySQL,
ODBC を使用しての接続 ,892

N

NDEF 技術 ,703
NFC,
MobileTogether サポート ,703
NFC タグからデータを読み取る ,705
タグ ,703, 705
データの送信 ,532, 706
デザインコンポーネントの概要 ,709
デバイス ,703
デバイス上での使用 ,703
と Android ビーム ,532, 703
プッシュアクション ,532, 706
開始 /停止 アクション ,530
関連したイベント ,707
技術 ,703
NFC アクション ,529
NFC サンプルファイル ,990
NFC ツリー構造 ,530, 705

O

ODBC,
MariaDB への接続 ,883
Teradata への接続 ,910
データ接続インターフェイスとして ,834
接続のセットアップ ,854
ODBC ドライバー ,
使用可能か確認する ,854
OLE DB,
データ接続インターフェイスとして ,834
OnServiceRunning, 1097
Oracle データベース ,
JDBC を使用しての接続 ,900
ODBC を使用しての接続 ,895
OriginalRowSet, 82
Outlook の連絡先 ,994, 995

P

PDF,
デザインを使用して生成する ,485
PostgreSQL,
ODBC を使用しての接続 ,902
直接 (ネイティブ)接続 ,858
直接接続する(ネイティブ), 861
postMessage(), 1007
Progress OpenEdge データベース ,
JDBC を使用しての接続 ,907
ODBC を使用しての接続 ,904
PXF ファイル ,
PDF、Word、RTF 出力の生成のため ,485

R

REST,
ページソースとして追加 ,233, 236
リクエストの設定 ,236
REST リクエスト内の OAuth, 236
RTF,
デザインを使用して生成する ,485

S

SMS, 480
SOAP,
ページソースとして追加 ,233
リクエストのための設定 ,243
SOAP チュートリアル ,148
SQL Server,
ADO を介しての接続 ,840
ADO.NET を使用して接続する ,847
SQL ステートメント,590
SQLite,
外部キーの無効化 ,865
接続のセットアップ(Linux), 917
接続のセットアップ(macOS), 917
接続の設定 (Windows), 865
StyleVision Server,
PDF、Word、RTF 出力の生成のため ,485
Sybase,
JDBC を使用しての接続 ,909

T

Teradata,
JDBC を使用した接続 ,910, 914

U

URL を開くアクション ,482

V

Visibility,
of controls, 86
of table columns, 86

W

WADL,

WADL,
REST リクエスト内 ,236
Web ページへのソリューションの埋め込み ,1019
Windows,
サーバー実行ファイルをデプロイ ,916
Word,
デザインを使用して生成する ,485
WSDL,
SOAP リクエスト内 ,243

X

XML ツリー ,248
XPath コンテキスト ノード ,37
XPath 式 ,
MobileTogether デザイン内での使用 ,27
XPath/ XQuery 関数 ,
MobileTogetherのための拡張関数 ,798
ユーザー定義 ,814
XPath/ XQuery 式エバリュエータ ,796
XPath/ XQuery 式ダイアログ ,796
XPath/ XQuery 式ビルダー ,796
XPath/ XQuery 条件式の編集ダイアログ ,796

Z

アクション ,466
DB トランザクションの開始 ,588
DB コミット トランザクション ,596
DB バルク挿入 ,594
DB ロールバック トランザクション ,597
DB 実行 ,590
HTTP/ FTP をロード/保存 ,569
If- Then, 623
If- Then- Else, 624
Let, 627
Loop, 625
MapForce 転送 ,490
REST リクエストの実行 ,581
SOAP からロード/保存 ,573
SOAP リクエストの実行 ,579
URL を開く ,482
アクションの実行のキャンセル ,601
イベントの設定 ,47
イメージの表示 ,508
オーディオ (再生), 511
オーディオの録音 ,515
カレンダーへのアクセス ,495
キーボードを隠す ,551
コメント ,599
サーバー接続エラーの トライ /キャッチ ,632
サブページへ移動 ,545
サブページを閉じる ,549
スロー ,629
ソリューションの実行 ,604
に SMS を送信する ,480
に通話を発信する ,481
ノードの置換 ,620
ビデオ(再生), 519
ファイルからロード ,557
ファイルにイメージを保存 ,503
ファイルに保存 ,557
ファイルをロード/保存 ,557
ファイルを開く ,482
フォルダーの読み取り ,584
ページソースの保存 /復元 ,586
ページのロード時 ,56
ページへ移動 ,544
マップで位置情報を表示 ,526
メッセージ ボックス ,470
ユーザーキャンセルの振る舞い ,602
ユーザーによるバーコードのスキャン ,509
ユーザーによる時刻の選択 ,500
ユーザーによる日付の選択 ,499
ユーザーのイメージ選択 ,502
リセット ,578
リターン ,633
位置情報データの読み込み ,522
位置情報トラッキングの開始 ,521
位置情報トラッキングの停止 ,521
一番下へスクロール ,550
印刷 ,485
共有 ,477
更新ノード ,608
再ロード ,556
削除ノード ,619
実行 ,600, 690
挿入ノード ,612
待機カーソルの表示 /非表示 ,493
電子メールの送信 ,472
読み上げ ,518
表示の更新 ,552
付加ノード ,615
文字列からロード ,571

アクション ,466
文字列に保存 ,571
保存 ,69, 575
例外のトライ /キャッチ ,630
連絡先の読み取り ,494
アクショングループ ,634
呼び出し、Let アクションの変数の変数値の設定 ,644
作成と編集 ,637
使用 ,639
アクションの実行のアクションのキャンセル ,601
アクションの保存 ,69
アサーション ,306
アサーション メッセージ コントロール ,306
アドレスブックのサンプルファイル ,994
アドレス帳からの連絡先 ,
データソースとして並べ替え ,494
読み取り ,494
イベント,
アクション設定 ,47
コントロールイベント ,462
ページ イベント ,286
イメージ ,45, 674
Base64, 679, 685
Base64 エンコード ,677
Base64 または URL, 675
Exif, 677, 679
URL の変更 ,59
サーバーで変換 ,600
サイズ変更 ,690
ソース ,675
データベース内 ,691
ファイルに保存 ,503
ファイル拡張子 ,503
ユーザーにより選択された ,502, 685
回転 ,690
変換 ,690
イメージ コントロール ,378, 675
イメージアクション ,501
イメージの表示アクション ,508
イメージの変換 ,690
エイリアス ,
グローバルリソース参照 ,954
エスケープショートカット,
ページ上のコントロールの表示 ,1175
エリアチャートの機能 ,752
ゲージチャートの機能 ,752
ローソク足チャートの機能 ,752
円グラフの機能 ,752
折れ線グラフの機能 ,752
棒グラフの機能 ,752
エンドユーザー使用許諾契約書 ,1282
オーディオ ,
ファイルフォーマット、エンコード ,701
再生アクション ,511
録音 アクション ,515
オーディオ 再生 ,
の概要 ,693
オーディオの再生 ,
開始、一時停止 、再開、停止、へスキップ ,511
オーディオの録音 ,
の概要 ,696
開始、停止 ,515
オプション ,
3D チャート ,761
エリアチャートの機能 ,752
ゲージチャートの機能 ,752
チャートX- 軸 ,758
チャートY- 軸 ,760
チャートZ- 軸 ,761
チャートグリッド ,758, 760, 761
チャートサイズ ,762
チャートのタイトル ,750
チャートの色 ,756
チャートの背景 ,750
チャートの凡例 ,750
チャートフォント ,763
ローソク足チャートの機能 ,752
円グラフの機能 ,752
折れ線グラフの機能 ,752
棒グラフの機能 ,752
カレンダー ,
イベント情報を書き込む ,495
イベント情報を読み取る ,495
ページソースからデータを読み取る ,495
カレンダーのサンプルファイル ,995
カレンダー内のイベント,
読み取りと書き込み ,495
キーボードを隠すアクション ,551
キャッシュ ,217, 259
クエリ ,936
クライアントでの試用 ,984
クライアント側の永続データ ストレージ ,219
グローバル リソース ,213
定義 ,955
グローバル リソース XML ファイル ,955
グローバルリソース ,
データベースの型の定義 ,963
ファイルの型の定義 ,957

グローバルリソース ,
フォルダーの型の定義 ,961
構成の変更 ,969
使用 ,966, 968, 969
グローバルリソースの ,
構成 ,955
グローバルリソースの構成 ,969
グローバル変数 ,817
MT_Android, 819
MT_AudioChannel, 822
MT_Browser, 819
MT_ButtonBackgroundColor, 819
MT_ButtonTextColor, 819
MT_CameraAvailable, 819
MT_CanvasX, 822
MT_CanvasY, 822
MT_ClientLanguage, 819
MT_ControlKind, 822
MT_ControlName, 822
MT_ControlNode, 822
MT_ControlValue, 822
MT_ControlValueBeforeChange, 822
MT_DBExecute_Result, 822
MT_DeviceHeight, 819
MT_DeviceWidth, 819
MT_DPIX, 819
MT_DPIY, 819
MT_EditFieldBackgroundColor, 819
MT_EditFieldTextColor, 819
MT_FirstPageLoad, 822
MT_GeolocationAvailable, 819
MT_HTTPExecute_Result, 822
MT_InputParameters, 819
MT_iOS, 819
MT_iPad, 819
MT_IsAppStoreApp, 819
MT_IsEmbedded, 819
MT_LabelBackgroundColor, 819
MT_LabelTextColor, 819
MT_Landscape, 822
MT_NFCAvailable, 819
MT_PageBackgroundColor, 819
MT_PageName, 822
MT_Portrait, 822
MT_ServerConnectionErrorLocation, 822
MT_SimulationMode, 819
MT_SMSAvailable, 819
MT_TableColumnContext, 822
MT_TargetNode, 822
MT_TelephonyAvailable, 819
MT_UserName, 819
MT_UserRoles, 822
MT_WindowHeight, 822
MT_Windows, 819
MT_WindowsPhone, 819
MT_WindowWidth, 822
ユーザー定義 ,825
グローバル変数 (静的), 819
グローバル変数 (動的), 822
ゲージチャートの機能 ,752
コメントアクション ,599
コントロール ,300
アサーション メッセージ ,306
イメージ ,378
コンボ ボックス ,337
スペース ,421
チェック ボックス ,328
チャート ,322
テーブル ,431
ビデオ ,454
ボタン ,312
ラジオボタン ,398
ラベル ,389
リッチテキスト ,408
共通のコンテキストメニューコマンド ,301
時刻 ,443
垂直線 ,451
水平線 ,372
水平線スライダー ,375
切り替え ,423
日時(iOS), 356
日付 ,347
編集フィールド ,363
コントロール アクション ,47
コントロール イベント ,47
コントロール ペイン ,186
コントロールイベント ,462
とアクション ,462
コンボ ボックス ,42
イベントとアクション ,47
コンボ ボックス コントロール ,337
コンボボックス ,
ソースノード ,59
ドロップダウンリストの定義 ,101
ドロップダウンリストの編集 ,63
サーバー ,
インタラクションのレベル ,197
サーバーアクション ,600

サーバーサービス ,
サービスのデプロイ ,1099
サービスの実行 ,1100
トリガーデータを持つシミュレーション ,996
概要 ,1096
新規サービスの作成 ,1097
サーバーシミュレーションのためのファイルの場所 ,978
サーバーにデプロイされた ,
デプロイするファイル ,184
サーバーへの接続エラー時 ,293
サーバー上のデータ ストレージ ,217
サーバー上の作業用ディレクトリの設定 ,
サーバーシミュレーション ,978
サーバー接続エラーの トライ /キャッチアクション ,632
サービス ,
サーバーサービスを参照 ,1096
サブページ ,182
サブページ チュートリアル ,118
サブページへ移動アクション ,545
サブページを閉じる アクション ,549
システム DSN,
セットアップ ,854
システム必要条件 ,21
シミュレーション ,972
MobileTogether Designer 内 ,973
Outlook カレンダー ,995
クライアントで ,984
サーバー上 ,978
サービストリガー上 ,996
デバイスのアドレス帳から連絡先を読み取る ,994
デバイスのカレンダーからカレンダーを読み取る ,995
ファイルのロケーション(サーバー), 978
ファイルの場所 (デザイナー), 973
メッセージ ペイン ,998
位置情報コンポーネント付きデザイン ,986
結果と理由 ,61
実行 ,50, 61
出たの読み取りのための NFC タグ ,990
ショートカットの挿入 ,
ページ上のコントロールのための表示 ,1175
シリアル化 ,571
スクロールすることのできるテーブル ,168
スタイル & プロパティ ペイン ,190
スペース コントロール ,421
スローアクション ,629
ソースデータ ,
データソース参照 ,222
ソリューション ,
ソリューションの終了後の他のソリューションの開始 ,604
最小化して他のアクションを開始する ,604
ソリューションの実行アクション ,604
ソリューションへのハイパーリンク ,789
ソリューションへのリンク ,789
タブされたページ ,182
タブの順序 ,
シーケンス内でのコントロールの定義 ,1175
タブ分割 ,182
チェック ボックス コントロール ,328
チャート ,721
3D 設定 ,761
X- 軸 ,758
Y- 軸 ,760
Z- 軸 ,761
エリアチャートの機能 ,752
グリッドライン ,758, 760, 761
ゲージチャートの機能 ,752
サイズ ,762
タイトル ,750
データの選択 ,724
データの選択: シンプル ,727
データの選択: フレキシブル ,731
フォント ,763
ローソク足チャートの機能 ,752
円グラフの機能 ,752
外観 ,741
系列色 ,756
作成と構成 ,722
色の定義 ,756
色の範囲 ,756
折れ線グラフの機能 ,752
塗りつぶし ,756
背景色 ,750
棒グラフの機能 ,752
凡例の削除 ,750
凡例の追加 ,750
目盛り間隔 ,762
余白 ,762
チャート コントロール ,322
チャート チュートリアル ,105
チュートリアル ,32
SOAP, 148
クイックスタート(パート1), 34
サブページ ,118
ファイルの場所 ,34
可視性 ,118
ツリーデータ ,248
ツリーデーター ,
のためのデフォルトファイル ,253

ツリーデーター ,
ファイルからデータをインポート ,253
手動で入力 ,253
ツリー構造 ,248
について ,251
手動で作成 ,251
データ ,
クライアント上で永続的 ,219
データ ソース ,
種類 ,225
追加 ,224
データソース ,97, 222
HTTP/ FTP、REST および SOAP, 233
ページ データ ソースを参照 ,37
データソースツリー ,
変更 ,56
データソースツリーのデフォルトのファイル ,253
データツリー ,248
データの更新アクション ,607
データベース ,594, 828
DB データの編集 ,922
DB 実行アクション ,930
MobileTogether 内でのデータの表示 ,934
NULL 値 ,575
OriginalRowSet, 830
SELECT ステートメント付きのデータソースを選択する , 920
SQL ステートメントの実行 ,590
アクション ,587
コミット トランザクション ,596
データ ソースとして ,830
データソースとして DB オブジェクトを選択する ,920
データソースとしてテーブルを選択する ,920
データを保存 ,926
とグローバル リソース ,968
とテーブル ,104, 109, 113
トランザクションの開始 ,588
ロールバック トランザクション ,597
自動インクリメント フィールド ,830
主キーの値 ,830
内のイメージ ,691
編集の有効化 ,109, 113
データベースドライバー ,
概要 ,837
データベース接続 ,
ウィザードの開始 ,836
グローバル リソースからの再使用 ,866
セットアップサンプル ,867
設定 ,834
データをデフォルトファイルに保存する ,69
テーブル ,649
コンテキストメニュー ,671
スクロールすることのできる ,168
とデータベース ,104, 109, 113
プロパティ ,665
繰り返し ,653
静的 ,651
追加 ,41
動的な行 ,657
動的な行と列 ,661
動的な列 ,661
テーブル コントロール ,431
テキストサイズ ,
自動調整グループのリスト ,1177
デザイナーシミュレーターのためのファイルの場所 ,973
デザイン に XML を埋め込む ,215
デザイン ファイル ,
新規作成 ,35
デザインのステップ ,25
デザインのテスト ,
シミュレーションを参照する ,984
デザインのフォーマット ,41
テストケース ,
サーバーにデプロイ ,1054
記録 ,1046
再生 ,1047
テストケースとテストラン ,
管理 ,1050
テストケースをサーバーにデプロイ ,1054
テストラン ,
比較 ,1058
デバイスのプレビュー ,
選択 ,36
デフォルトの XPath コンテキスト ノード ,37
デフォルトファイル ,65
トップページ ,182
に SMS を送信するアクション ,480
に通話を発信するアクション ,481
ノード ,
新規作成 ,612, 615
ノードの置換アクション ,620
バーコード ,509
パフォーマンス ,214
ビデオ ,
ファイル フォーマット、エンコード ,462, 701
再生 アクション ,519
ビデオ 再生 ,
の概要 ,699

ビデオコントロール ,454
ビデオの再生 ,
開始、一時停止、停止 ,519
ファイル DSN,
セットアップ ,854
ファイルにイメージを保存アクション ,503
ファイルペイン ,184
ファイルを開くアクション ,482
プッシュ通知 ,
$MT_PUSHNOTIFICATION, 714
OnPushNotificationReceived, 714
シミュレーション ,719, 992
と $MT_PUSHNOTIFICATION ツリー ,537
プッシュ通知 ,714
プッシュ通知 トピック ,542
ペイロード ,537
ボタン ,537, 714
外部プッシュ通知キー ,541, 714
概要 ,711
受信のメカニズム ,714
送信アクション ,537
送信のメカニズム ,712
送信方法 ,712, 714
プッシュ通知アクションの送信 ,537
プッシュ通知 トピック ,542
プッシュ通知 トピックの登録 ,542
プッシュ通知のシミュレーション ,719, 992
プレビューデバイス ,
選択 ,179
プロジェクト ,196
デプロイ ,200
のローカライズ ,211
プロジェクトファイルの場所 ,198
プロパティ ,204
内の名前空間 ,212
プロジェクトのシミュレーション ,
シミュレーション参照 ,50
プロジェクトのデプロイ ,200
プロジェクトの検証 ,49
プロジェクト構造 ,94
プロジェクト内の名前空間 ,258
ページ イベント ,286
とアクション ,286
ページ コントロール ,186
ページ シーケンス ,94
ページ ソース ,
ルートノード ,245
ページ デザイン ビュー ,179
ページ ビュー設定 ,179
ページ ペイン ,182
ページコントロール ,301
ページソース ,
オプション ,231
ツリーの構造とデータ ,248
内部メモリに状態を一時的に保存 ,586
内部保存のの保存 /復元 ,586
ページソース ペイン ,
ツリー構造の作成 ,187
機能 ,187
ページソースアクション ,554
ページソースツリー ,248
XML 構造のインポート ,251
のコンテキストメニュー ,262
ファイルからデータをインポート ,253
構造 ,251
読み取り専用および編集可能 ,253
ページソースツリー内のツリーノード ,262
のコンテキストメニュー ,262
ページソースリンク ,59
ページデータソース ,
追加 ,37
ページデザイン ,278
ページに関連するアクション ,543
ページのセットアップ ,36
ページプロパティ ,283
ページへ移動 アクション ,544
ボタンコントロール ,
ボタンのクリック時 ,312
マップで位置情報を表示アクション ,526
メイン ウィンドウ ,
説明と機能 ,178
メッセージ ボックス アクション ,470
メッセージペイン ,192
ユーザー DSN,
セットアップ ,854
ユーザー インターフェイス ,
しくみの説明 ,176
概要 ,176
ユーザーキャンセルの振る舞いアクション ,602
ユーザーによるバーコードのスキャンアクション ,509
ユーザーによる時刻の選択アクション ,500
ユーザーによる日付の選択アクション ,499
ユーザーのイメージ選択 ,502
ユーザーログイン ,1012
ユーザー定義 XPath/ XQuery 関数 ,814
ユーザー変数 ,825
ライセンス ,
情報 ,1282

ライセンス計測 ,
Altova 製品にて ,1284
ラジオボタン コントロール ,398
ラベル コントロール ,389
リセット アクション ,578
リターン アクション ,633
リッチテキスト,
ソリューション内での編集 ,786
リッチテキストコントロール ,408
セットアップ ,779
ソリューション内での編集 ,786
リッチテキストスタイルシートの割り当て ,780
リッチテキストスタイルシート,
リッチテキストコントロールの割り当て ,780
作成 ,780
定義 ,782
リッチテキスト機能 ,778
ローカライズ ,211, 1158
シミュレーション ,1164
ソリューションの言語の切り替え ,605
ローカライズのためのシミュレーション言語 ,1164
ローカル動的編集 ,
グローバル変数 (動的), 822
ローソク足チャートの機能 ,752
ワークフロー ,94
トップページのシーケンスで定義された ,182
ワークフロー シミュレーション ,
シミュレーション参照 ,50
位置情報アクション ,520
位置情報データ,
$MT_GEOLOCATION ツリーに入力 ,522
位置情報データの読み込みアクション ,522
位置情報トラッキング ,
開始と停止 ,521
位置情報トラッキングの開始アクション ,521
位置情報トラッキングの停止アクション ,521
一番下へスクロールアクション ,550
印刷アクション ,485
永続データ,219
円グラフの機能 ,752
可視性プロパティチュートリアル ,118
外部キー ,
SQLite 内で無効化 ,865
外部プッシュ通知キー ,541
外部プッシュ通知キーの登録 ,541
概要ペイン ,189
共有アクション ,477
言語 ,
ランタイムでの切り替え ,605
更新ノード アクション ,608
混在したコンテンツ要素 ,
シリアル化 ,816
混在したコンテンツ要素のテキストコンテンツ ,
FAQ, 816
再ロード アクション ,556
削除ノード アクション ,619
試用期間 ,
Altova ソフトウェア製品の試用 ,1283, 1285
時刻 ,
ユーザーによる選択 ,500
時刻 コントロール ,443
自動化されたテスト,1044
テストケースとテストランの管理 ,1050
テストケースの記録 ,1046
テストケースの再生 ,1047
テストケースをサーバーにデプロイ ,1054
テストランの比較 ,1058
自動調整グループのリスト,1177
実行 ,690
実行アクション ,600
新機能 ,6
垂直線コントロール ,451
推奨されているイメージファイル拡張子 ,503
水平線コントロール ,372
水平線スライダーコントロール ,375
切り替え コントロール ,423
折れ線グラフの機能 ,752
設定 ,
3D 設定 ,761
エリアチャートの機能 ,752
ゲージチャートの機能 ,752
チャートX- 軸 ,758
チャートY- 軸 ,760
チャートZ- 軸 ,761
チャートグリッド ,758, 760, 761
チャートサイズ ,762
チャートのタイトル ,750
チャートの色 ,756
チャートの背景 ,750
チャートの凡例 ,750
チャートフォント ,763
ローソク足チャートの機能 ,752
円グラフの機能 ,752
折れ線グラフの機能 ,752
棒グラフの機能 ,752
挿入ノード アクション ,612
待機カーソル ,493
待機カーソルの表示 /非表示 ,493

著作権情報 ,1282
定義 ,
3D 設定 ,761
X- 軸設定 ,758
Y- 軸設定 ,760
Z- 軸設定 ,761
エリアチャートの機能 ,752
グリッドライン ,758, 760, 761
ゲージチャートの機能 ,752
チャートサイズ ,762
チャートのタイトル ,750
チャートの色 ,756
チャートフォント ,763
チャート内のフォント ,763
ローソク足チャートの機能 ,752
円グラフの機能 ,752
折れ線グラフの機能 ,752
棒グラフの機能 ,752
電子メール ,472
電子メールの送信アクション ,472
電話 ,481
匿名ログイン ,1011
日時コントロール (iOS), 356
日付 ,
ユーザーによる選択 ,499
日付コントロール ,347
配布 ,
Altova ソフトウェア製品の配布 ,1283, 1285
Altova のソフトウェア製品 ,1282
表示の更新アクション ,552
評価期間 ,
Altova のソフトウェア製品 ,1282
付加ノード アクション ,615
文字列アクションからロード ,296
文字列からロード ,571
文字列に保存 ,571
編集フィールド コントロール ,363
保存アクション ,575
法的な情報 ,1282
棒グラフの機能 ,752
埋め込まれた Web ページソリューション ,
埋め込まれたソリューション ,1000
埋め込まれた Web ページソリューションの認証 ,1010
JWT 認証 ,1013
ユーザーログイン ,1012
匿名ログイン ,1011
埋め込まれたソリューション ,1000
addEventListener(), 1008
IFrame への Web ページデータの設定 ,1034
IFrames, 1003
JSON サンプル ,1020
JSON ページソースの自動的なロード ,1034
JWT 認証のサンプル ,1039
postMessage(), 1007
Web ページからサーバーにデータを送信する ,1020, 1027
Web ページからデータを送信 /受信する(JSON サンプル), 1020
Web ページからデータを送信 /受信する(XML サンプル), 1027
Web ページからポスト ,1007
Web ページとの通信 ,1006
Web ページにソリューションを埋め込む ,1003
XML サンプル ,1027
XML データへのアクセスと送信 ,571
サーバーから Web ページにデータを送信する ,1020, 1027
サーバーから Web ページへのメッセージ ,1008
サンプル ,1018
データの通信 ,1006
ページソース ,296, 571
メッセージの送信 ,296
埋め込みの例 ,1019
埋め込まれたメッセージイベント ,296
埋め込まれたメッセージページソース ,296
埋め込まれたメッセージをバックするアクション ,606
名前空間 ,212
用語 ,23
例外のトライ /キャッチアクション ,630
連絡先のサンプルファイル ,994
連絡先の読み取りアクション ,494