

Altova FlowForce Server 2021 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.

公開日: 2015-2021

(C) 2015-2021 Altova GmbH

目次

1 はじめに 14 1.1 新規機能 15 1.2 基本概念 19 1.3 セキュリティ概念 21 1.4 動作の説明 22 1.5 Web 管理インターフェイス 24 1.6 FlowForce Server へのログオン 26

2 インストール

2.1	システム	.必要条件	28
2.2	プロセッ	サーコアとライセンス	29
2.3	重要なノ	パス	30
2.4	Linux		31
	2.4.1	FlowForce Server のインストール	31
	2.4.2	FlowForce Server 2021 への移行	32
	2.4.3	現在インストールされている製品の確認	33
	2.4.4	FlowForce Server のアンインストール	33
	2.4.5	インストール ディレクトリ	33
2.5	macOS.		34
	2.5.1	FlowForce Server のインストール	34
	2.5.2	FlowForce Server 2021 への移行	35
	2.5.3	FlowForce Server のアンインストール	36
	2.5.4	インストールディレクトリ	36
2.6	Windows		37
	2.6.1	FlowForce Server のインストール	37
	2.6.2	Windows サービスの構成	41
	2.6.3	ネットワーク構成 (Windows のみ)	42
	2.6.4	データを手動で移行する	45
	2.6.5	インストールフォルダー	46

100

3	サーバ	「一の構成	49
3.1	セットアン	ップ	50
3.2	ネットワ・	ーク設定の定義	54
	3.2.1	ファイル参照の構成	57
3.3	SSL 暗枭	骨化のセットアップ	63
	3.3.1	証明書機関により SSL 証明書を署名する方法	63
	3.3.2	自己署名済みの SSL 証明書の作成方法	65
	3.3.3	秘密キーの必要条件	75
	3.3.4	FlowForce Web サーバーのために SSL を有効化する方法	
	3.3.5	FlowForce サーバーのために SSL を有効化する方法	78
	3.3.6	FlowForce Web Serverと FlowForce サーバー間に SSL を有効化する方法	
3.4	デフォル	トのタイムゾーンの設定	81
3.5	電子メー	ルのパラメーターの設定	82
3.6	ディレク	~リサービスの設定	83
3.7	ログ設定	<u> </u>	86
3.8	サービス	の開始と停止 (Linux)	91
3.9	サービス	の開始と停止 (macOS)	92
3.10	サービス	の開始と停止 (Windows)	93
3.11	FlowFor	ce Server アプリケーションデータ	94
3.12	FlowFor	ce Server のローカライズ	
3.13	バックア	ップと復元	97

4 配布済みの実行

4.1	配布済みの実行に関連する用語	101
4.2	「マスター」モードでの FlowForce Server の作動	102
4.3	「ワーカー」モードに FlowForce Server を変更する	103
4.4	Windows 上でのクラスターインストール	106
4.5	配布された実行のセットアップ	107
4.6	「ワーカー」モードを停止する方法	111

2.7

5	ユ-+	ゲーアクセスの管理	112
5.1	<u>ユ</u> ーザ-	ーとロール	113
	5.1.1	ユーザーの作成	113
	5.1.2	ロールの作成	114
	5.1.3	ドメインユーザーとロールをインポート	114
	5.1.4	デフォルトのユーザーとロール	115
	5.1.5	ユーザーとロールの名前の変更	116
	5.1.6	ユーザーにロールを割り当てる	117
	5.1.7	ロールを他のロールに割り当てる	117
	5.1.8	ルートパスワードの再設定	118
5.2	特権		119
	5.2.1	特権の作動方法	119
	5.2.2	特権レポートの閲覧	122
5.3	パーミッ	ションとコンテナー	125
	5.3.1	パーミッションの動作方法	125
	5.3.2	コンテナーを理解する	128
	5.3.3	コンテナーの作成、名前変更、および移動	130
	5.3.4	コンテナーパーミッションの閲覧	133
	5.3.5	コンテナーパーミッションの設定	135
	5.3.6	/public コンテナーへのアクセスの制限	136
5.4	パスワー	ードポリシー	139
	5.4.1	パスワードポリシーの動作方法	139
	5.4.2	パスワードポリシーの作成と割り当て	139

6 ジョブの構成

6.1	6.1 資格情報		
	6.1.1	資格情報の定義	144
	6.1.2	OAuth 2.0 資格情報	145
	6.1.3	ジョブから資格情報を参照する方法	146
6.2	ジョブの	作成	148
6.3	ジョブの	複製	149
6.4	入力パー	ラメーターの管理	150

ステップの管理152
順番にステップを処理する154
条件付でステップを処理156
ステップのエラーの処理
6.8.1 エラー時に再試行する160
6.8.2 実行する際の注意点162
延期されたステップ
ジョブの戻り値の型を宣言する170
データ型をステップで処理する171
Windows ネットワークパスの参照174
トリガーの管理
6.13.1 タイマートリガー
6.13.2 ファイルシステムトリガー177
6.13.3 HTTP トリガー179
ジョブを Web サービスとして公開する181
ジョブの結果をキャッシュする185
キュー設定の定義
Windows ネットワークパスの参照191
FTP サポート
ジョブのインポートとエクスポート193
6.19.1 構成データのエクスポート198
6.19.2 機密な情報を含む、または含まないオプション
6.19.3 構成データのインポート
6.19.4 不足する依存関係の処理方法

ジョブの実行のモニタリング 7.1 分散済みのジョブの実行のモニタリング......

/.1	万敗済みのショノの美门のモーダリング	210
7.2	ジョブのステータス	211
7.3	ジョブの停止	214
7.4	メインログの確認	215
7.5	インスタンスログの確認	217
7.6	統計詳細ページ	219

8	FlowForce 式	223
8.1	式の計算	224
8.2	式言語のルール	225
8.3	文字列フィールド内に式を埋め込む	228
8.4	式関数の呼び出し	229
8.5	FlowForce データ型	230
8.6	演算子	232

9 Altova 製品との統合 233 9.1 サーバー実行のためにファイルを準備する 234 9.2 ElowEarce Server にマッピングをデプロイオる 239

9.2	FlowFor	ce Server にマッピングをデプロイする	. 239
9.3	マッピン	グと変換をジョブとして実行する方法	. 243
	9.3.1	マッピング関数内の資格情報	. 246
	9.3.2	例: OAuth 2.0 認証	. 248
	9.3.3	動的な認証	. 266
	9.3.4	リソース	. 267
9.4	マッピン	グ/変換結果へのアクセス	. 274
9.5	RaptorX	ML Server との統合	. 276
9.6	ツールフ	ファイル	. 279
	9.6.1	環境変数の設定	. 280

10 AS2 との統合

10.1	概要: AS2 データの送信	283
10.2	概要: AS2 データの受信	285
10.3	MapForce と MapForce Server を使用した AS2 との統合	287
10.4	AS2 概念	292
10.5	AS2 証明書の構成	293
10.6	AS2 パートナーの構成	297
10.7	AS2 メッセージの送信	304
10.8	AS2 メッセージの受信	308
10.9	例: フル AS2 メッセージの交換 (シンプル)	313
10.10	例: フル AS2 メッセージ交換 (高度)	323

11 コマンド ライン インターフェイス

334

11.1	help	338
11.2	assignlicense	339
11.3	compactdb	340
11.4	createdb	341
11.5	debug	342
11.6	exportresourcestrings	343
11.7	foreground	344
11.8	initdb	345
11.9	install	346
11.10	licenseserver	347
11.11	migratedb	348
11.12	repair	349
11.13	resetpassword	350
11.14	setdeflang (sdl)	351
11.15	start	352
11.16	uninstall	353
11.17	upgradedb	354
11.18	verifylicense	355

12 ビルトイン関数

12.1	/system		357
	12.1.1	abort	357
	12.1.2	compute	358
	12.1.3	compute-string	360
12.2	/system	ı/as2	363
	12.2.1	send	363
12.3	./system	ı/filesystem	365
	12.3.1	сору	365
	12.3.2	delete	366
	12.3.3	mkdir	366
	12.3.4	move	367

	12.3.5	rmdir	368
12.4	./system	ı/ftp	370
	12.4.1	delete	371
	12.4.2	delete-wildcard	373
	12.4.3	mkdir	375
	12.4.4	move	378
	12.4.5	list	380
	12.4.6	retrieve	383
	12.4.7	retrieve-wildcard	386
	12.4.8	rmdir	388
	12.4.9	store	391
	12.4.10	store-wildcard	395
12.5	./system	n/sftp	398
	12.5.1	connect	399
	12.5.2	delete	400
	12.5.3	delete-wildcard	400
	12.5.4	list-directories	401
	12.5.5	list-files	402
	12.5.6	mkdir	402
	12.5.7	move	403
	12.5.8	retrieve	404
	12.5.9	retrieve-wildcard	404
	12.5.10	rmdir	405
	12.5.11	rmdir-wildcard	406
	12.5.12	store	407
	12.5.13	store-wildcard	407
12.6	./system	n/mail	409
	12.6.1	send	409
	12.6.2	send-mime	410
12.7	./system	n/maintenance	415
	12.7.1	archive-log	415
	12.7.2	cleanup-files	415
	12.7.3	truncate-log	416
12.8	./system	n/shell	417
	12.8.1	commandline	417

13	式関数	牧	419
13.1	ステッフ	パ結果関数	
	13.1.1	error-message	
	13.1.2	exitcode	
	13.1.3	failed-step	
	13.1.4	results	
	13.1.5	retry-count	
	13.1.6	stdout	
	13.1.7	stderr	
13.2	ストリー	-ム関数	
	13.2.1	as-file	
	13.2.2	content	
	13.2.3	empty-stream	
	13.2.4	stream-from-string	
	13.2.5	stream-open	
13.3	ファイル	シシテム関数	430
	13.3.1	list-files	
	13.3.2	list-directories	430
	13.3.3	read-lines	
13.4	ファイル	ノパス関数	
	13.4.1	extension	
	13.4.2	filename	
	13.4.3	filename-with-extension	
	13.4.4	join-paths	
	13.4.5	parent-directory	
13.5	リスト関	数	
	13.5.1	char	437
	13.5.2	code	438
	13.5.3	from-to	
	13.5.4	join	439
	13.5.5	length	
	13.5.6	list	441
	13.5.7	nth	

	13.5.8	slice	442
13.6	文字列	對数	444
	13.6.1	concat	444
	13.6.2	contains	444
	13.6.3	ends-with	445
	13.6.4	find-all	446
	13.6.5	number	446
	13.6.6	split	447
	13.6.7	starts-with	447
	13.6.8	string	448
	13.6.9	string-join	448
	13.6.10	string-length	449
	13.6.11	substring	449
	13.6.12	trim	450
	13.6.13	trim-start	451
	13.6.14	trim-end	451
13.7	ブール値	直関数	452
	13.7.1	all	452
	13.7.2	any	452
	13.7.3	false	453
	13.7.4	if	453
	13.7.5	not	454
	13.7.6	true	454
13.8	ランタイ	ム情報関数	455
	13.8.1	instance-id	455
	13.8.2	log	455
	13.8.3	slot-number	457
13.9	AS2 式	男数	459
	13.9.1	as2-disposition	459
	13.9.2	as2-http-status	459
	13.9.3	as2-mdn-serialize	460
	13.9.4	as2-message-id	460
	13.9.5	as2-partner-local-name	460
	13.9.6	as2-partner-remote-name	461
	13.9.7	as2-success	464

	13.9.8	as2-signed	464
13.10	MIME 式	関数	465
	13.10.1	add-mime-header	465
	13.10.2	add-mime-headers	465
	13.10.3	current-message-id	466
	13.10.4	get-mime-content-disposition-param	467
	13.10.5	get-mime-content-id	467
	13.10.6	get-mime-content-type-param	468
	13.10.7	get-mime-header	469
	13.10.8	get-mime-headers	469
	13.10.9	get-stream-filename	470
	13.10.10	is-file	470
	13.10.11	is-mime-content-type	470
	13.10.12	mime-content-encode	471
	13.10.13	mime-flatten	472
	13.10.14	mime-multipart	472
	13.10.15	mime-multipart-related	473
	13.10.16	mime-multipart-from-list	475
	13.10.17	mime-parse	475
	13.10.18	mime-split-multipart	475
	13.10.19	new-message-id	476
	13.10.20	reset-mime-headers	476
	13.10.21	set-mime-content-disposition	477
	13.10.22	set-mime-content-id	477
	13.10.23	set-mime-header	478
	13.10.24	set-mime-headers	479

14	ジョブの構成の例	480
14.1	˝Hello, World!˝ ジョブの作成	481
14.2	パスが存在するか確認する	484
14.3	ファイルをコピーする	488
14.4	MapForce マッピングからジョブを作成する	493
14.5	ジョブを他のジョブのステップとして使用する	500
14.6	ディレクトリ ポーリング ジョブを作成する	503

14.7	エラー処理をジョブに追加する	509
14.8	Web サービスとしてジョブを公開する	515
14.9	JSON を FlowForce Web サービスヘポストする方法	520
14.10	ジョブの結果をキャッシュする	527
14.11	StyleVision 変換からジョブを作成する	531
14.12	RaptorXML を使用してドキュメントを検証する	539
14.13	エラーログのある XML を検証する	541
14.14	RaptorXML を使用して XSLT を実行する	545
14.15	複数の PDF を複数の XML から生成する	550

15 用語

562

15.1	C	563
15.2	.E	564
15.3	F	565
15.4	I	566
15.5	.J	567
15.6	M	568
15.7	.P	569
15.8	Q	570
15.9	R	571
15.10	S	572
15.11	Т	573
15.12	U	574
15.13	W	575

インデックス

1 はじめに

FlowForce Server はWindows、Linux、およびmacOS サーバー上、および、Web インターフェイスを介したフークステーションでタ スクを自動化するために使用されるクロスプラナトフォームノストウェアノリューションです。

FlowForce Server は他のAltova サード製品 (MapForce Server、StyleVision Server、および、Raptor XML Server の 両方の種類)と統合することができ、Webサービスとして作動するジョブを含む再帰的ホオンデマドのジョブとして機能を拡張することがで きます。例えば、MapForce Server とStyleVision Server との統合のために、MapForceマナング、おまよ StyleVision 変換 を再帰的な FlowForce ジョブとして作動することができます。RaptorXML Server との統合でも同様に、XML おままJSON ファイルを Webサービスとして公開されているオンデマドのジョブとしてアイルを検証することができます。

FlowForce Sever を使用すると、電子メールの送信、ローカルシステムまけはネットワーク上でのファイルの管理、ファイル転送プロトコル (FTP)を使用したファイルの管理、シェルスクレプトの実行など他の共通サーバー、マスを作成し自動化することができます。FlowForce Server のAdvanced Edition はAS2 メッセーンを受け入れ、送信することができ、クラスターとして作動する複数のサーバー上でのジョ ブの配布された実行へのサポートを追加します。



最終更新日: 2021 年 02 月 25 日

1.1 新規機能

FlowForce Server 2021 R2

- <u>/systm/ftp</u> ライブラカらの既存のFlowForce ビルトイン関数はFTPS (SSL を介したFTP)を介したサーバーへの接続のオプシュンをサポートします。
- FlowForce Server Advanced Edition は安全な FTP (SFTP とて既知の、おはSSH を介した FTP)をサポートします。FTP サーバーにSFTP を介して接続し、オペレーションを実行できるように新規の関数を/system/sftp コンテナー内で使用できるようになりました。
- SFTP、SSH キー、のための新規の資格情報の型が使用できるようておりまた。
- ワイルドカードを使用してノモートのFTP サードー上でファイルをアップロード、抽出、削除することを可能にする新規のFTP 関数を使用できるようこなりました。具体的にはFTP サードをFTP おけはFTPS を介して接続する場合、関数 delete-wildcard、retrieve-wildcard、とstore-wildcard を/system/ftp ライブラルら使用することができます。
 SFTP を介して接続する場合 /system/sftp ライブラルら同じ名前の関数を使用することができます。
- ジョブの実行と他のジョブは関連した情報の結果のサマレーを表示するために<u>統計とチャート</u>をWeb管理インターフェイスで使用 できるようになりました。
- <u>ファイルシステムトリガー</u>を作成する際、最小のポーレンプ間隔を1秒に設定できるようになりました(以前は、最小の間隔は30 秒でした)。
- <u>/system/mail/send</u> 関数は[メッセージボディ] フィールドのための改善された複数行のエディターを提供します。
- 統計詳細ページ: 色スキームトラベル内の変更
- ファイル ス関数: join-paths は引数とて提供された マを一つに結合する新規機能です。
- /system/sftp にフイルドカードに一致するデルクトリをSFTP サーバーから削除するrmdir-wildcard 新規の関数が搭載 されました。

FlowForce Server 2021

- 更に素早くレコードをロードするために<u>ログビュー</u>ページは最適化され、新規のナビゲートとフィルターオプション、およびログの現在の状態を永久リンクとて保存する機能が搭載されました。
- 一度にこつのログされナジョブをビューするオメの専用の新規のログインスタンスページを使用することができます。後に他の受取人に送信するオメリエのページからログされた情報を.zipアーカイブミジップすることができます。以前にエクスポートされナジョブインスタンスをログインスタンスページコロードし、後/ニデバッグするオメリエニーするこなどができます。
- ログのナダの特定のログの詳細が保管ませるエキップされるかを決定できる新規のログ設定を使用できるようしておりました。ジョブの結果をベースはしてログ詳細のレベリを構成することもできます。ログのトレース情報を保管することが奨励されますが、実行の成功の場合、最も基本的な項目のみを保管することが奨励されます。
- 特定のログ設定をアプリケーションレベルはすではなく、特定のFlowForce Server ジョブレベルで構成できるようなりまた。 オブシェクトレベルでのログルールを参照してくたさい。

FlowForce Server 2020 R2

- エラーが発生した場合、1つおけは複数のステップの実行を再試行することが可能です。エラー時に再試行するを参照してくたさい。
- ジョブはステップを延期して実行することができます。これはジョブがWebサービスの呼び出しを介して呼び出される場合に適してします。
 <u>延期されたステップ</u>を参照してくたさい。
- 新規の任意のホスト名フィールドがセナア・プページで使用できるよう」なりました。<u>やホワーク設定の定義</u>を参照してください、SSL構成が更に柔軟になり、Webサービスを直接ショブ構成ページからテストランできるようになりました。
- 新規のファイルませまディンケリが指定されたディンケリに追加される際ファイルシステムトリガーを実行するように構成できます。
 このトリガーはポールされたディンケリが後に変更された場合トリガーを実行しない既存の変更された日付」とは異なります。
- AS2 パトナー構成ページはAS2 メッセージ正規の書式に再フォーマナの能にする相互運用性の設定新規のオプタンを 提供します。

- 型 OAuth 2.0 の資格情報を定義する際 POST リケエストボデイニ認証の詳細を構成することができます。これは POST リケ エスト ヘッダー内に認証の詳細を提供するサポート済みの標準メンド以外の任意のメンドです。 OAuth 2.0 証明書を参照し てくたさい。
- セトアップページか簡素化されたかを評価するプロシージャです。

FlowForce Server 2020

- Webサービスを呼び出すFlowForce Server ジョブはOAuth 2.0 プロトコルを使用してサービスプロ・イダーと共に認証することができます。FlowForce内の「資格情報」エトリはOAuth 2.0 フィールドもサポートするように拡張されました。
 <u>OAuth</u>
 <u>2.0 資格情報</u>も参照してくたさい。
- MapForce とFlowForce Server 内の両方で資格情報を定義することができ、デザイン時にマッピングは埋め込む、まけよ FlowForce Server 内の実行ステップに、ラメーターとして提供することができます。マッピング関数内の資格情報も参照してく たさい。
- <u>資格情報オブンエクトを定義中</u>使用の特定のドメインを制限することができます。「使用方法」は以下の「つまけ」複数であることができます: ジョブ実行、FTP、HTTP。
- (「グロー、シリソース」とて既知の MapForce 内で定義されているポータブルファイル、フォルダー、データベースレファレンスを FlowForce Server にデプロイレ、マピング関数により使用することができます。必要な場合、FlowForce 内で直接マピン グ関数により使用されるリソース(ファイル、フォルダー、データベースレファレンス)を変更することができます。これはこの関数を使用 する FlowForce ジョブのすべてご影響を与えます。FlowForce Server 内で直接リソースを作成ませば編集することができます。 ます。リソースを参照してくたさい。
- ジョブ構成データを他のFlowForce Server インスタンスませま.zip アーカイブニエクスポートする場合、機密なデータを任意 で選択することもできます。構成データのインポートとエクスポートを参照してくたさい。

FlowForce Server 2019 R3

- FlowForce Server を使用して作成されたWeb サービスは HTTP POST リクエストのボディをパラメーターとして受け入れることができます。Web サービス、ウメーターを参照してください。例えば JSON をFlowForce Web サービス、ポストする方法を参照してください。
- ログ式関数と新規のログ設定により、FlowForce Serverのログ機能が拡張されまた。ログ設定の変更を参照してくたさい。

FlowForce Server 2019

- FlowForce Server をW indows Active Directory おけではよくLDAP (ライトウェイト ディレクトリアクセスプロトコリを サポートする Directory Service プロバイダー 注統合することができます。ディレクトリサービスの設定を変更するを参照してく たさい。
- FlowForce Server Advanced Edition はショブ配布済みの実行の恩恵を受けることができます。複数のFlowForce Server インスタンスをクラスターとしてセナアップすることができ、複数のマシン上でショブ処理ワークロードに再配布することができます。 配布済みの実行と、ランスのロードを参照してください。
- 新規のFlowForce Server 内蔵関数はFlowForce Server に比送信される電子メールシャセージ本文をカスタマイズすることができる<u>send-mime</u>で使用することができます(例えば、StyleVision Server 変換を実行してHTML 本文、おこは、添付を準備する場合など。新規関数によりSMTP ヘッダーをFlowForceで使用することのできる<u>MIME 式関数</u>を使用してカスタマイズすることができます。

FlowForce Server 2018 R2

- (Windows 固有) Windows Active Directory 統合の拡張。Windows マンがFlowForce Server かドメインの一部として作動する場合、「設定」ページ内で選択するためにこれらのドメインか自動的に使用することができます(次を参照:ディレントリサービス設定の変更)。
- AS2 データを送信するジョブコルえ、AS2 サービスを実行するジョブを作成し、パトトナーからAS2 データを受信することができます。HTTP(S) Web サービスの受け取りご類似し、FlowForce Server は、構成されたアドレスとポートでパトナーからAS2 リンエストをリッスします。これは、効果的にFlowForce Server をAS2 データを処理し保管するAS2 サーバーはます。次を参照してくたさいAS2 との統合。

FlowForce Server 2018

- FlowForce Server Advanced Edition か既存のFlowForce Server 「コルえご利用 サゴナます。FlowForce Server Advanced Edition のインストール ジケージ は個別 こうイセンス供与されて る 次の追加製品 と ジドルされて ます: MapForce Server Advanced Edition、RaptorXML Server、および Style Vision Server。FlowForce Server のインストール ジケージ は個別 こうイセンス供与されて る 次の追加製品 と ジドルされて ます: MapForce Server、RaptorXML Server、Style Vision Server。
- FlowForce Server Advanced Edition はAS2 (Applicability Statement) メッセージをAS2 取引相手を送信する ためにサポートされています。次を参照してくたさい<u>AS2 との統合</u>。
- 内部の更新と最適化。

FlowForce Server 2017r3

- FlowForce Server とFlowForce Web Sever は共に暗号化されてないプレーンな HTTP および(SSL-暗号化された) HTTPS 接続を受け入れることができます。HTTPS 接続設定は、セオアップページ(次を参照: <u>ネオワークの設定の定義</u>)、おけよ構成ファイルを使用して定義することができます(次を参照: <u>ファイル参照の構成</u>)。ウイドウでは、接続設定はインストール中に定義することができます。次を参照してくたさい:<u>ネオワークの構成(Windows のみ)</u>。
- Windows 上でのFlowForce Server インストール中、FlowForce Server とFlowForce Web Server に対応する Windows サービスを特定のオペレーティングシステムユーザーとして実行するように構成することができます。サービスの開始(自動、手動)もセメアップすることができます。次を参照してくたさい、Windows サービス構成。
- 新規のシステム管理コマイドがコマイドラインで使用できます: <u>compactdb</u>。

FlowForce Server 2017

- システムシャットダウンでの改善された振る舞い
- 内部の更新と最適化。

FlowForce Server 2016 R3

- ジョブのステップ、トリガー、および、ウメーターを新規「複製」(¹) ポタンを使用することこと、複製することができます。また、
 ステップ、トリガー、まけよ、ウメーターを削除した後に「元に戻す」(¹) ポタンを使用することができます、以下を参照、入力パ
 - スパッシストリンテ、おうよくシーテーを削除しるようプローチャリー() パッシアを使用することができまり、以下を参照、スリン <u>ラメーターの管理</u>、ステップの管理、およびトリガーの管理)。 他のショブコマジリンヤセンショブナ ちたする場合、ディーターの説明レント ビ てまテレキオ・ディーターの説加しまする話しい
- 他のジョブコ呼び出されたジョブは存在する場合、ウメーターの説明とこととして表示します。、ウメーターの追加に関する詳しい 情報は以下を参照してくたさい、入力、ウメーターを追加する。

FlowForce Server 2016

• 既存のジョブか新規の名前で保存されることができ、新規のジョブのテンプレートとしての役割を果たします(以下を参照: ジョブの 複製)。

- ジョブの管理の強化:コンテナー内の特定のジョブのみが引けてはなく、サブコンテナーを含むコンテナー全体を削除ませるエクスポート することが可能はないました(以下を参照:ジョブのインポートとエクスポート)。
- コマボラインインターフェイスからルートパワードのセルを行うことができます(以下を参照:ルトパマワードの再設定)。
- パフォーマンスの向上:集中したショブの実行によるWeb インターフェイスのタイムアナトは大幅に削減されました。
- ビルイン
 <u>/system/mail/send</u> 関数は SMTP サーバーへのSSL/TLS 接続をサポートします(以下を参照:<u>電子メ</u> <u>ール・ラメーターの設定</u>)。

FlowForce Server 2015 R4

- Mac のためのFlowForce Server インストールのディスクイメージ(.dmg) ファイルをご利用したおけます(以下を参照上で 実行されている<u>FlowForce Server のインストール</u>)。
- ・ 失敗したジョブの出力を処理する柔軟性。ジョブの実行が失敗すると、新規「エラー時には中断」、ウメーターは、ジョブが中断されるか、関数がfalseを返すかを決めます。この、ウメーターは/system/filesystem, /system/ftp, /system/mail, and /system/shell エレテナーからの関数に適用されます。
- パフォーマノスの向上:ジョブ構成ページ内で大きなジョブが速くロードすることができます。

FlowForce Server 2015 R3

• Web 管理インターフェイスから、(ジョブまけよ関数などの) オブジェクトが作成された、おけよ 変更された日時、およい変更を 行オニューザー名を確認することができます(以下を参照:<u>コンテナーを理解する</u>)。

FlowForce Server 2015

- <u>Windows Active Directory</u> 統合
- ユーザー ペワードの複雑性の構成
- システム特権のしポート
- ジョブ<u>インポートとエクスポート</u>
- 実行中のジョブの停止

FlowForce Server 2014 R2

- FlowForce <u>キャッシュ拡張</u>
- に名前を付ける能力 <u>ユーザーとロール</u>
- <u>コンテナー/オブジェクト</u>の名前変更機能
- 新規のステップ 式 <u>関数</u>
- RaptorXML(キー/値/デメーターペア)のための、デメーターのノスト

FlowForce Server 2014

- 新規のサーメー<u>管理</u>関数
- ジョブの結果のキャッシュ機能と、結果を他のジョブで使用する機能
- 拡張された<u>RaptorXML</u>へのサポート

FlowForce Server 2013 R2

- RaptorXML Server との統合
- 条件をベースコンナジョブステップショブの実行を許可するフローの管理実行ステップの繰り返しを無制限に行う機能
- 実行を繰り返す ステップを無制限に使用する機能
- 1つのステップを次のジョブステップで使用することを許可する<a>

 ステップ変数の定義
- 電子メールの通知、FTP サーバー インタラグション、式の計算を許可するビルトインステップの拡張されたセメ

1.2 基本概念

ジョブ

ジョブとはFlowForce Server 内のコア概念です。この概念サーメートで実行されるはタスクませ」はタスクのシーケンスを表します。ジョブは、 電子メールの送信などしつのステップから構成されるジョブであることができます。しかしながら、複数のアケンタンを実行し結果(例えば、ファイ ルを他のジョブの、ラメーターとして、マするジョブを作成することもできます。ジョブは<u>入力、ラメーター、ステップ、トリガー</u>および他の設定から構成されます。



FlowForce ジョブの構成

入力パラメーター

FlowForce Server ジョブのエテキストで、入力、ラメーターは、どのジョブが特定の方法で実行されるかを表したジョブに与えられた任意の精報を意味します。ジョブの入力、ラメーターは、関数引数にプログラム言語の観点から類似してします。入力、ラメーターは多種の型(例えば、ファイルまだはディレクトリルファレンス、テキスト、番号、ブール値、その他)であることができます。一部のジョブの型に対しては、FlowForce Server は、入力、デメーターを自動的に作成します。

トリガー

ジョブを作成する際、トリガーとしてジョブをトリガーする条件(おけは基準)を指定する必要があります。 FlowForce Server は、定義されたリガーを監視し、トリガーの条件が満たされるとショブを実行します。

ステップ

ステップよ FlowForce Server ジョブが実際に行う内容を定義します、例えば、ファイルの判除、MapForce マピングの実行、お子ま 電子メールの送信など)。最も簡単な形式としては、ステップは失敗お子は成功した成果を持つオペレーションで、実行されるために<u>関数</u>を 必要とします(以下を参照:<u>実行ステップ</u>)。しかしながら、ループとして他のステップを条件付きで実行する方法を与えるステップも存在しま す(以下を参照:<u>選択ステップ</u>、エラー/成功の処理ステップ および For Each ステップ)。同じジョブ内で必要な数だけステップを作成 し、実行される順序を設定することができます。

関数

ジョブのエレテキストでは、関数は、FlowForce Server によい理解される、ターゲットファイルシステム上のオペレーションを実行するさせの 命令です。関数は以下であることができます:

- ビルインFlowForce 関数(以下を参照 ビルイン関数)
- StyleVision 変換
- MapForce マピング
- ジョブの実行ステップ

関数の多くは入力、デメーターを有します。必須インプト、デメーターは、ステップを成功裏に実行するため呼び出し側により与えられる必要 かあります。

実行の結果

FlowForce Server 内では、ステップの実行はステップの実行後に返される内容を定義します(例えば、ファイル、およも、テキスト)。ジョ ブと作業する際、ステップの実行の結果を特定のデータ型(文字列ませよブール値)になるように、おせな破棄されるように、明示的に宣言する ことができます。通常、結果を他のジョブで使用する場合、おせよ、結果をキャッシュする場合、実行の結果を宣言する必要かあります。

資格情報

資格情報オブジェクトは、認証に関する情報を保管します。これはFlowForce Server ジョブが作動する箇所でのオペレーティングシステム上のユーザーアカウトに関連するユーザー名と マワードの組み合わせですがHTTP おけはFTP 資格情報 おけはOAuth セキュリティ の詳細のセナ であることもできます。

設定

ジョブの作成時、以下の任意の設定を構成することができます

- Web サービスとしてショブを使用できるようえます(以下も参照してくたさい)ジョブをWeb サービスとして表示する)。
- 同じジョブのために平行して実行されているインスタンスの数を制限します(以下も参照してくたさい)<u>キュー設定の定義)。</u>
- ジョブには返された結果をキャッシュします(以下も参照してくたさい、ジョブの結果をキャッシュする)。

1.3 セキュリティ概念

FlowForce Server は、企業まどはビジネスモデルのエーズと構造に従って構成されるロールをベースはたユーザーのアクセスを管理するメ カニズムを使用しています。例えば、ジョブと資格情報を特別なデータエンテナーに整理し、閲覧まどは変更するはよアクセス権利を必要と するパッケージ化することができます。対応するアクセス権利を有するユーザーのみがゴンテナー内部のデータにアクセスすることができます。

コンテナー

名前か意味するように、コンテナーオン・ゲージされたデータです。FlowForce Server では、コンテナーは、オペレーシングシステム上のフォ ルダーと大まかに比較することができます。コンテナーオ以下を含むことができます:ジョブ、資格情報、関数と他のコンテナー。コンテナーで・パ <u>ーミッション</u>を設定することに以、コンテナー内のデータを閲覧およびデータニアクセスする権利を管理することができます。デーダをコンテナー別 に整理し、対応するパーミッションを各コンテナーのためにセットアップすることは、は、セキュリティ対策*と*なるでしょう。

ユーザー

ユーザーとはFlowForce Server イエグオーレジョブを構成、おけはMapForce おけはStyleVision 変換をデプロイレ、FlowForce Server を管理する個人を指します。ユーザーが使用することのできるアグションは以下には影響されません

- a) 割り当てられたパーミッション おけは特権
- b) ユーザーがシンバーであるロールに割り当てられているパーミッションと特権

ロール

ロールはビジネスニースをベースしたセキュリティの強化を手助けする特権のセルです。標準的なロールをベースしたセキュリティイスは 最低2つのロールが含まれます:管理者と一般ユーザー。各ロールは、ロールに与えられた特権により定義されています。例えば、管理者は、 自身のおよび他のユーザーのマワードを変更することができますが、一般ユーザーは自身のマワードのみしか変更することができません。ユー ザーコロールを割り当て、必要に口応じて、ユーザーのロールを取り消すことができます。

特権

特権は、FlowForce Server 内でユーザーが行うことのできる内容を定義します、例えば、自身の ダワード設定、ユーザーとロールの読み 取り、ジョブの停止、など)。特権は、パーミッションはユンテナーへのユーザーアクセスを管理し、特権はグロー・ ジルこ FlowForce Server 全体に対して効果がある点で<u>パーミッション</u>と異なります。次の簡単なルールは、特権とパーミッションの違いを理解する助けてよります。特権 はグロー・ ジルであり、パーミッションはローカルです。

パーミッションに類似して、特権は個々のユーザーとユールに割り当てることができます。ですから、ユーザーがFlowForce Server にエグオン すると、有効な特権はおよ下により決定されます。

a)直接割り当てられた特権

b)ユーザがシンバーであるロールに割り当てられた特権

パーミッション

パーミッションは、ユーザーのエレテナーへのアクセスを管理します。特権と同様、ノ<u>パーミッション</u>は、ユーザーとユールに与えられることができます。ですから、ユーザーがロールの一部である場合、ロールに与えられたパーミッションは、ユーザーにも自動的に適用されます。

デフォルトで、コンテナーに設定されたパトミッションは、親コンテナーから継承されます。例えば、コンテナーA には、子コンテナーB かあると します。コンテナーA にアクセスするパーミッションを持つユーザーは、コンテナーB にもアクセスすることのできるデフォルトのパーミッションを持つ ことったります。しかしなから、管理者は、コンテナーの階層の全てのレベルで、全てのユーザーおよびロールのパーミッションを再定義することが できます。

パマードポルシー

FlowForce Servers は、管理者が複雑なユーザー ペワードを管理する ペワードポリシーを使用します。 パワードポリシーとは、有効になるためにユーザー ペワードが必要とする最低条件です (例えば、最低 N 文字のなど)。

1.4 動作の説明

Altova Web サイト: 🥜 <u>ワークフロー自動化ツール</u>

標準的な FlowForce Server インストールは、下の図に表されています。任意でライセンスが必要なコンポーネトは点線で囲まれています。



標準的な FlowForce Server インストール

図に示されているように、標準的な FlowForce Server インストールは、FlowForce Server とて同じサードーマンとにクストールされている複数のサードを製品(おけま 技術的な用語では、サービス)にお構成されて、ます。FlowForce Server ソリューションは、以下の2つのサービスから構成されています。FlowForce Web Server と FlowForce Server。これらの2つのサービスは、異なるサービスとして作動し、個別に構成、開始、停止することができます。これら2つのサービスの管理は、オペレーティングシステム(Linux、Windows、macOS)におしてなり、また、このドキュメントで更に詳しく説明されています。

FlowForce Web Server は、HTTP クライアトからのソフエストを受け入れ検証し、FlowForce Server に やします。および FlowForce Server は、FlowForce Server ソリューションのコアであり、視覚的なユーザーインターフェイスを使用することなく背景のサー ビスとして作動します。FlowForce Server は、継続的はトリガーの条件をチェックし、ジョブ実行を開始および監視し、詳細なログを作成 します。更に、FlowForce Server は、Web サービスとして公開されてしるジョブのナダのソフエストをリッスします(ローカルマシンとノモ ートのクライアントからのHTTP リクエストを受け入れるように構成することができます、次を参照してくたさい、<u>ネットワーク設定の定義</u>)。

FlowForce W eb Server、はその一方、ジョブを監視し、多種のFlowForce 設定を管理するW eb 管理インターフェイスからのリクエ ストを処理します。FlowForce W eb Server は、HTTP(おけはHTTPS)接続を次のプライアントの型から受け入れます。

Web ブラウザー	Web ブラウザーを使用して、FlowForce Server ジョブと他の設定を構成することがで
	きます(概要に関しては、以下を参照してくたさい、Web管理インターフェイス)。

MapForce Enterprise おけよ Professional Edition	MapForce は、Altova の視覚的なデータマメングのオメのデストップアプリケーションです。MapForce で作成されたマメングをFlowForce Server にデプロイレ、柔軟に構成することのできるショオ 変換することができます。例えば、マメングショズを毎日指定された時間、ませまファイルの監視されるディレクトリニ追加されると実行するようご構成することができます。デプロイされた MapForce マメングからジョブを実行するはまま、FlowForce Server は、役割がマメングを実行し、結果出カファイルを作成するMapForce Server を呼び出します。詳細に関しては、MapForce とMapForce Server 製品ページを参照してくたさい。
StyleVision Enterprise おは Professional Edition	Style Vision は、XML、SQL データベース および XBRL 入力をベース してしポートをデザインするデストップアプリケーションです。スタイルシート かテストされ、デバックされる と、FlowForce Server にデプロイすることができます。デプロイされ ファイルは、サーバ ー上で変換ショブのナムリゴ使用することができます。デプロイされた Style Vision 変換から 作成され ジョブを実行する コよ、FlowForce Server は変換と結果出力 ファイルを 作成する Style Vision Server を呼び出します。詳細に関しては、 <u>Style Vision</u> と <u>Style Vision Server</u> 製品ページを参照してくたさい。

各製品の詳細に関しては Altova ドキュメトのページを参照してくたさい (https://www.altova.com/ja/documentation.html)。

RaptorXML Server

(略して Raptor XML と呼ばれる) Altova Raptor XML Server は パラレルコンピューティング環境と最新の標準のよめに最適化された Altova の第3世代の、高速 XML プロセッサーです。高水準なクロスプラナトフォームの機能性のためにデザインされたエンシンは、XML の 高速な処理を行うために、現在のマルチコアコンピュータの利点を活用しています。Raptor XML には2つのエディションが存在します:(i) Raptor XML Server と(ii) Raptor XML+XBRL Server。Raptor XML+XBRL Server エディションは、XML に加え、 XBRL (eXtensible Business Reporting Language) ドキュメトの検証と処理へのサポートを含んでいます。

RaptorXML Server がFlowForce Server と同じサーバーにインストールされている場合、関数は、ビルトインFlowForce Server 関数として使用することができます。これは、XMLドキュメントの整形式をチェックする、おけは検証する、おけよ、XSLTとXQueryドキュ メートを変換するジョブを作成することができることを意味します。詳細に関しては、<u>RaptorXML Server との統合</u>を参照してくたさい。

1.5 Web 管理インターフェイス

FlowForce Server W eb 管理インターフェイスは、サーバーを管理し、ジョブを構成するFlowForce Server のフロントエンドです。 W eb 管理インターフェイスに<u>アドレスとポートの構成</u>のW eb のブラウザーからアクセスすることができます。

以下のセクタンではWeb 管理インターフェイス内で使用することのできるメニューアイテムこして説明しています。

- <u>*-</u>4
- <u>構成</u>
- <u>ログ</u>
- <u>管理</u>
- <u>ヘル</u>
- メモ Web 管理インターフェイスから使用することのできるハノースとアクションへのアクセスは、ユーザーアクセス管理メカニズムによど管理されています。これは、割り当てられたパトミッションの許可する限り、構成データにアクセスし変更することができます。同じように、対応する特権が与えられている場合、アグションを実行し(対応するメニューアイテムを閲覧することができます。

ホーム

最新の統計とチャート、最近および実行中のショブのリスト、アクティブよトリガーとサービスのリストを表示します。詳細に関してはショブの実 行のモニタレノブを参照してくたさい。

構成

現在定義されている Flow Force ユンテナー、ジョブ、 資格情報、および 関数 を表示します。オブジェクト のエレテンソと詳細を確認するこ は、対応する記録をクリックしてください。

デフォルトで以下のコンテナーを使用することができます:

- /public
- /system
- */RaptorXML (ライセンスが与えられた RaptorXML Server を使用できる場合)*

コンテナー・「関する詳細は、以下を参照してくたさい、コンテナーを理解する。

関連するアクセスの権利があると前提して、構成ページから、コンテナー、ジョブ、資格情報および関数、を管理し、コンテナーインペーミッションを設定することができます。

ログ

サーバーログを表示します。表示されたログエトリはサーバー、関連した、また、ジョブ、関連したメッセージを含みます。内容はアクセス権利にお見ないます。詳細に関しては、ジョブのログの確認を参照してくたさい。

管理

管理エリアから、サーイ・構成とユーザーの管理に関連したアウションを実行することができます。管理エリアは以下のメニューアイテムから構成 されます:

ユーザー

ユーザーの作成、削除、および、特権、ロールとマワードポルシーの変更を含む、管理を行います。詳細に関しては、以下を参照してくたさい、ユーザーアクセスの管理。

レポート

アクセスコトロールのロールを作成、削除および保持することができます。詳細に関しては、以下を参照してください、ユーザーとユール。

パマワードポトシー

パマワードの難易度のポパシーを作成、削除、および保持することを許可します。詳細に関しては、以下を参照してくたさい、パマワードポパシー。

レポート

現在割り当てられているユーザーと中に関するレポートの閲覧を許可します。詳細に関しては、以下を参照してください、特権レポートの閲覧。

設定

FlowForce Server とActive Directory おはLDAP 準拠したサイトを統合し、デフォルトのタイムゾーン、電子メールサイト設定を定義することを許可します。詳細に関しては、以下を参照してくたさい、サイトの構成。

ケラスター

FlowForce Server の複数のインスタンス全体のジョブの実行を配布することを許可します。配布済みの実行を参照してくたさい。

へしプ

FlowForce Server ドキュメイを別のブラウザータブまけまナイドウで開きます。

1.6 FlowForce Server へのログオン

(ジョブを作成し、ユーザーを追加するなどの) FlowForce Server を管理するコよ (例えば、<u>http://localhost:8082</u>のよう な)構成されたHTTP(S) アドレスとポートでW eb 管理インターフェイスコングオンする必要があります。このURL の構成に関しては、次 を参照してください、<u>ネットワープ設定の定義</u>(「FlowForce W eb Server」の下でグループ化された設定を検索してくたさい)。

デンォル・では、FlowForce Serverを新規にインストールすることには、ユーザー名 root と マワード root を使用してログオンすることができます。

ログイン	/
接続中: 127.0.0.	1:4646 。 オンライン
ログイン名:	root
パスワード:	••••
	ログイン

セキュリティ上の理由のため、FlowForce Server に最初にエグイン後、デフォリトのrootノマワードを変更してくたさい。

(Active Directory などの) ディレクトリサービスプロレイダー か構成されている場合、ドメインユーザーは、FlowForce にログオンすることができます。この場合、ログインページコよ、Windows ドメインを選択することのできるドロップダウノリストか含まれています。ディレクトリサ ービス認証の代わりにHTTP 認証を使用する場合、ログインドロップダウノリストから、ディレクトリを選択してください。

ログイン		
接続中: 127.0.0	.1:4646 ₀ オンライン	
ログイン名: パスワード:	domain-user01	
ログイン	Domain: kg.bole.com (kg)	
	ログイン	

FlowForce Server によい公開されるWeb サービスにアクセスするクライアントはHTTP 認証への代替とて Active Directory 認証 を使用することができまず(通常 http://local host: 4646/service/SomeService などのURL)。Active Directory 認 証を可能にするコま、ユーザー名はNT/と共にプレフィックスされている必要があり、NT/john.doe@my.domain.com などのドメイン 名を含む必要があります。次を参照してくたさい、Webサービスとしてジョブを公開する。

Authentication Required X		
?	http://127.0.0.1:4646 is requesting your username and password. The site says: "Altova FlowForce"	
User Name:	NT/john.doe@my.domain.com	
Password:	•••••	
	OK Cancel	

Windowsドメイン認証の構成方法の詳細に関しては、次を参照してくたさい、ディンケリサービス設定の変更。

2 インストール

2.1 システム必要条件

Windows	プラオフォーム更新済みのWindows 7 SP1、Windows 8、 Windows 10
Windows Server	プラオフォーム更新済みのWindows Server 2008 R2 SP1 ま だは以降
Linux	 CentOS 7 おけお以降 RedHat 7 おけお以降 Debian 8 おけお以降 Ubuntu 16.04 LTS おけお以降
macOS	macOS 10.13 およお以降

Web 管理インターフェイスはGoogle Chrome、Mozilla Firefox、および Microsoft Edge の最新バージョンよアクセス可能です。

2.2 プロセッサーコアとライセンス

Altova サード、製品へのライセス供与は(ロジカルはコアの数量ではなく製品マシントで使用することのできる物理的なプロセッサーの コアの数量をベースしています。例えば、デュアルコアプロセッサーイコは、ケアドコアプロセッサー4コア、ヘキサコアるコアなどの2つのコアか存在 します。製品にライセンス供与されたコア数は、物理ませま板想マシントに関わらず、サードーマシントで使用することのできるコアの数量と同 数、ませま、それ以上である必要があります。例えば、サードーが8コア持つ場合、8コアライセンを購入する必要があります。コアの数量を 満たすためにライセンスを組み合わせることもできます。ですから、2つの4コアライセンスを使用して、8コアライセンスの代わり13コアを達成する こともできます。

処理する容量は少ないが、CPUコアの大きなコンピューターサーバーを使用している場合、少ない数量のコアに割り当てられている仮想マ シンを作成し、その数量のライセンスを購入することもできます。このようなデプロイの場合、もちろん、コンピューターのすべてのコアを使用する場 合に北ベニのようなデプロイの処理スピードは遅いです。

メモ 各 Altova サーバー製品ライセンスは、使用されていないライセンス供与能力が存在する場合でも、Altova サーバー製品がインストールされているマシンで、一度にコンのグライアントマシンのためにのみ使用することができます。例えば、10 コアライセンスが6 CPU コアを持つマシンのために使用されている場合、残りのライセンス許与能力の4 コアは、他のグライアントマシンのために同時 に使用することはできません。

単ースレイ実行

Altova サーン・製品が単一スレンドの実行を許可すると*単一スレンドの実行を*使用できるようズンはます。この場合、Altova サーン・製品ライセンスようイセンスプール内で「コアのみ使用することができます。複数のコアを持つマンノはこの「コアライセンス」「割り当てることができます。このような場合、その製品を作動するマンノは単一コアで作動します。(マルチコアでは使用可能な)複数スレンドを使用できな、サメシ処理 「よこのような」場合、表の製品にそのマンン上で単一スレンドモードで実行されます。

マルチコアマシノニ単一コラテイセンスを割り当てるゴよその製品のためにLicenseServer内で「単一スレンド実行に制限する」チェックボックスを選択してくたさい。

必要なコア数の見積もり

サーバーが扱うことのできるデータの量と処理回数を影響する多くの外部要素が存在します。(例えば、ハードウエア、CPU 上の現在のコード、サーバー上で作動する他のアプリケーションのメモルの割り当てなどか挙(形れます)。パフォーマンスを正確に測定するために、データの量、 条件、およびビジネスの環境にらい状態でアプリケーションをテストしてくたさい。

2.3 重要なパス

FlowForce Server のインストール後以下のファイルに重要な情報が保管されます:

- 个水一/デル/ INSTALLDIR)
- **アプリケーションデータディレクトリ**(DATADIR)

FlowForce Server インストールディレクトリ(INSTALLDIR)

Linux	/opt/Altova/FlowForceServer2021/
macOS	/usr/local/Altova/FlowForceServer2021/
Windows	C:\Program Files\Altova\FlowForceServer2021\ C:\Program Files (x86)\Altova\FlowForceServer2021\

Linux	/var/opt/Altova/FlowForceServer2021/data
macOS	/var/Altova/FlowForceServer2021/data
Windows	C:\ProgramData\Altova\FlowForceServer2021\data

2.4 Linux

2.4.1 FlowForce Server のインストール

システムの必要条件

- - CentOS 7 おけ以降
 - RedHat 7 おけお以降
 - Debian 8 お北以降
 - Ubuntu 16.04 LTS おけお以降

次のライブラルはアプリケーションをインストールし実行するけっと要とされるライブラリです。下のパッケージが使用中 Linux のマシンで使用できない場合、yum(まけよ、適用できる場合、apt-get を)コマンドを実行してインストールしてくけさい。

サーバー	CentOS, RedHat	Debian	Ubuntu
LicenseServer	krb5-libs	libgssapi-krb5-2	libgssapi-krb5-2

必要条件

- root ユーザーおけは sudo 特権を持つユーザーとてインストールを行ってくたさい。
- FlowForce Server の前の、ージョンか新しい、ージョンがインストールされる前にアンインストールされる必要があります。
- FlowForce Server を他のAltova サーン製品とクストールする場合は、FlowForce Server を最初にクストールすることが奨励されます。

Flow Force Server のインストール

- 1. https://www.altova.com/download#server からローカルディレクトリにインストールッケージンダウレロードします。
- 2. インストールッケージをダウレロードするデルクトリニ変更します。例

cd /home/User/Downloads

3. FlowForce Server パッケージをインストールます。

[Debian, Ubuntu] sudo dpkg --install flowforceserveradv-2021-debian.deb [CentOS, RedHat] sudo rpm -ivh flowforceserveradv-2021-1.x86_64.rpm

メモ 現在のリレース おけよ サービス シクバージョンニー 致するかがに 上記の シケージの名前を調整する必要がある場合があます。

Flow Force Server のライセンス供与

- 1. Altova LicenseServer の最新バージョンがインストールされており、ローカルズシン、おけよ ネトワーク上で実行されていること を確認してくたさい(<u>https://www.altova.com/ja/licenseserver</u>)。
- 2. FlowForce Server をAltova LicenseServer に登録します。このためには、が作動中のエレビューターの木木名、おけよアドレスを知る必要があります。例えば、LicenseServer が127.0.0.1 上で作動する場合、登録におい下のように行うことがで

きます:

sudo ./flowforceserver licenseserver 127.0.0.1

メモ 上記のコマイは、プログラムインストーリディレクトリから実行可能ファイルを呼び出していることが前提です。

3. LicenseServer 管理者インターフェイスコングオン、FlowForce Server か実行されるマシノこライセンを割り当てます。

ライセスの詳細に関しては、Altova LicenseServerドキュメトを参照してなさい (https://www.altova.com/ja/documentation)。

インストール後のタスク

- 1. セオアップモードを入力してセオアップページを参照してくたさい。
- 2. Web 管理インターフェイスのホットワークアドレスとポートを構成します FlowForce Server を Altova License Server に登録します。 違します。 ネットワーク設定の定義を参照してくたさい。
- 3. Web 管理インターフェイスコンゲオンし、(前のステップ内でイドレスとポートを変更しな)限り、デフォルトでは http://localhost:8082、です)デフォルトのFlowForce Server root / マワードを変更します。デフォルトのログイン名とパ スワードはそれぞれ、root とroot です。
- 4. FlowForce Server と共に他のAltova 製品がインストールされている場合、(例えば、MapForce Server、StyleVision Server、RaptorXML Server)環境変数を設定する場合、環境編集の設定を参照してくたさい。

2.4.2 FlowForce Server 2021 への移行

必要条件:

- FlowForce Server の前の、ージョンをアンヘノストールします。アンインストールはアプリケーションデータディレオリを削除することを意味しません。重要ないなも参照してくたさい。アプリケーションデータディレオ・リへの、みはFlowForce Server のメジャー、バージョンイニと異なります(例、/var/opt/FlowForceServer2017)。
- FlowForce Server 2021 を <u>インストール</u>します。これはデフォルトの構成データを持つ新規のアプリケーションデータディレクトリ を作成します(例、/var/opt/FlowForceServer2021).

FlowForce Server 2021 へのデータの移動の方法:

1. 実行中の場合 FlowForce Web Server サービスを停止します。

sudo systemctl stop flowforcewebserver

- 2. 実行中の場合 FlowForce Server サービスを停止します。上記と同じコマイを使用します。 flowforcewebserver を flowforceserver と置き換えます。
- 3. インストール中に作成された新規のデータディレクトリを、削除ませる、別の名前を与えます。

sudo rm -rf /var/opt/Altova/FlowForceServer2021/data

4. FlowForce Server のコマドラインインターフェイス内で使用可能な <u>migratedb</u> コマドを実行して EXISTING デーダを移行します。例:

sudo /opt/Altova/FlowForceServer2021/bin/flowforceserver migratedb
--olddatadir=/var/opt/Altova/FlowForceServer2017/data

--datadir=/var/opt/Altova/FlowForceServer2021/data

5. FlowForce Web Server サービスを開始します。

sudo systemctl start flowforcewebserver

6. FlowForce Server サービスを開始します。上記と同じコマイドを使用します。flowforcewebserver を flowforceserver と置き換えます。

2.4.3 現在インストールされている製品の確認

Altova サードー製品がインストールされているか確認するコカ以下のコマンドを実行します:

[Debian, Ubuntu] dpkg --list | grep Altova [CentOS, RedHat] rpm -qa | grep server

2.4.4 FlowForce Server のアンインストール

FlowForce Server をアンインストールするコよ 以下のコマンドを実行します:

[Debian, Ubuntu] sudo dpkg --remove flowforceserveradv [CentOS, RedHat] sudo rpm -e flowforceserveradv

2.4.5 インストール ディレクトリ

Linux 上での FlowForce Server インペールディンケリの場所は 以下の通りです:

/opt/Altova/FlowForceServer2021/bin

2.5 macOS

2.5.1 FlowForce Server のインストール

システム必要条件

▼ <u>macOS</u>

macOS 10.13 封出以降

必要条件

- Altova LicenseServer がインストールされており、ローカルズシン、おけよ ネトワーク上で実行されていることを確認してくたさい。
- 管理(ルート)特権を持つユーザーとしてインストールします。
- FlowForce Server を他のAltova サーン製品とインストールする場合は、FlowForce Server を最初にインストールすることが奨励されます。
- macOS マンは、IP アドレスマオルて名前か解決されるように構成される必要かあります、すなわち、コマイ ping <hostname> を使用してターミナルからオスト名をping することに成功する必要かあります)。

Flow Force Server のインストール

- 1. Altova Web ページカビ <u>http://www.altova.com/ja/download.html</u>) から、ディスクイメージファイル(.dmg) をダウロードし、開くためにクリックします。 これによりインストーラーがコンピューター上で新規ドライブとして表示されます。
- 2. パッケージ(.pkg) ファイルをダブルクトックして、インストールンナザードに従いインストールします。これらのステップは、説明を必要と せず、続行する前に、ライセンス使用許諾書に同意するステップ1つが含まれています。ウィザードを完了すると、FlowForce Server アイエッカアプリケーション内に表示されます。パッケージは以下のディレオトリ に/usr/local/Altova/FlowForceServer2021/。
- Altova License Server がインストールされていない場合、おけよ最新バージョンにアップグレードする場合、対応するパッケージ(.pkg)ファイルをダブルクトック、画面の説明に従ってくたさい。Altova LicenseServer は、FlowForce Server を含む Altova サーバーアプリケーションを実行するために必須です。。
- 4. アプリケーション内でFlowForce Server アイコンをダブルクリックします。これによりFlowForce Server 構成ページか開かれます。

Mac マンンの名前がIP アドレスを解決できない場合、ブラウザーは次のメッセージを表示するページを開きます:

「http://<hostname>:<port> でFlowForceW eb を使用できません。再起動の後、このページを再ロードしてくたさい。」このメッセージが表示されると以下を行ってくたさい、次を参照してくたさい、必要条件):

- a. このメッセージ内のレクをクリックする。
- b. ブラウザーのアドレスバー内の<hostname>を使用中のMac のIP アドレスと置き換えてくたさい。
- 5. 構成ページから、Altova LicenseServer を使用して、FlowForce Server を登録します。
- 6. Web 管理インターフェイスのホットワーケアドレスとポートを構成します FlowForce Server を Altova License Server に登録します。<u>ネットワーク設定の定義</u>を参照してくたさい。
- 7. Web 管理インターフェイスコングオンし、(前のステップ内でアドレスとポートを変更しない限り、デフォルトでは、 http://localhost:8082、です) デフォルトのFlowForce Server root パマワードを変更します。デフォルトのログイン名とパ スワードはそれそれ、root とroot です。
- メモ FlowForce Server をインネールする前に、Altova サーバー製品 がインネールされている場合、FlowForce Server /tools ディレネリへ 対応する製品の/etc ディレオリから.tool ファイルをエピーしてください。.tool ファイルよ 各

Altova サーン・製品の実行可能ファイルへの ひを含みます。ツールファイルをユピーするコよ ターミナル内で cp コマンドを実行 してくたさい。例えば、MapForce Server インストール ひから、ファイルをユピーする場合、以下を実行します:

cp /usr/local/Altova/MapForceServer2021/etc/*.tool /usr/local/Altova/FlowForceSe rver2021/tools

Flow Force Server のライセンス供与

- 1. Altova LicenseServer の最新バージョンがインストールされており、ローカルズシン、おけよ ネ・トワーク上で実行されていること を確認してくたさい (<u>https://www.altova.com/ja/licenseserver</u>)。
- 2. FlowForce Server をAltova LicenseServer に登録します。このためには、が作動中のエレビューターの木木名、おけよ アドレスを知る必要があります。例えば、LicenseServer が127.0.0.1 上で作動する場合、登録におい下のように行うことができます。

sudo flowforceserver licenseserver 127.0.0.1

3. LicenseServer 管理者インターフェイスコングオン、FlowForce Server か実行されるマンノこライセンを割り当てます。

詳細に関しては、Altova LicenseServerドキュメトを参照してたさい(<u>https://www.altova.com/ja/documentation</u>)。

2.5.2 FlowForce Server 2021 への移行

必要条件

- FlowForce Server 2021 がインストールされている必要があります(以下を参照: FlowForce Server のインストール)。
- データの移行を管理者(ルート)の特権を持つユーザーとして行します。

Flow Force Server 2021 にデータを移行する

1. FlowForce Server サービスを停止します。

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceServer.plist

2. FlowForce W eb Server サービスを停止します。

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceWebServer.plist

3. インストール中に作成されたデータディレクトリの名前を変更するか削除します。これにより、手動のアップグレードを行う前のインスト ールの後に作成された新しし v シージョン内のオブジェクト か削除されます:

sudo rm -rf /var/Altova/FlowForceServer2021/data

4. migratedb コマドを実行します:

```
sudo /usr/local/Altova/FlowForceServer2021/bin/FlowForceServer migratedb
--olddatadir=/var/Altova/FlowForceServer2017/data --
datadir=/var/Altova/FlowForceServer2021/data
```

5. FlowForce Server サービスを開始します。

sudo launchctl load /Library/LaunchDaemons/com.altova.FlowForceServer.plist

6. FlowForce Web Server サービスを開始します。

sudo launchctl load /Library/LaunchDaemons/com.altova.FlowForceWebServer.plist

2.5.3 FlowForce Server のアンインストール

FlowForce Server のアンインストール

1. FlowForce Server サービスを停止します。

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceServer.plist

2. FlowForce Web Server サービスを停止します。

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceWebServer.plist

サービスが停止下かを確認するために、アクティビティモニターターミナルを開き、サービスがリストに存在したいことを確認します。アク ティビティモニターをダブルクトックすると、「マイプロセス」以外の全てのプロセスが表示されます(「表示」メニューの「全てのプロセ ス」をクトックします)。

- 3. 「アプリケーション」内のFlowForce Server アイコンを右クトックし、「ゴミ箱に移動する」を選択します。FlowForce Server 構成データは別途に保管されており、この段階では削除されないてとご注意してくたさい(次のステップを参照してくたさい)。
- 4. 任意で、FlowForce Server 構成データ北削除する場合、以下のコマイを実行します:

sudo rm -rf /usr/local/Altova/FlowForceServer2021/

メモ 上記のステップは/var/Altova/FlowForceServer2021からFlowForce Server 構成データを削除しません。 FlowForce Server 構成データが削除されると、後で復元できないてという言してくたさい。

2.5.4 インストールディレクトリ

macOS 上でFlowForce Server インストールディレクトリの場所は、以下の通りです。

/usr/local/Altova/FlowForceServer2021
2.6 Windows

2.6.1 FlowForce Server のインストール

システム必要条件

- プラオーム更新済みのW indows 7 SP1、W indows 8、W indows 10
- プラオフォーム更新済みのW indows Server 2008 R2 SP1 おけお以降

必要条件

- 管理特権を持つユーザーとてんみールを行います。
- バージョン 20217.1 以降から Flow Force Server の32 ビナバージョンは64 ビナバージョンにクレストールすることはできません。(i) 新しい ビジンを 32 ビナバージョンにクレストールすることはできません。(i) 新しい ビジンを インストールする 前に古い ビジンを削除する おとは(ii) 古い インストールと同じビナバージョンを新規のビージョンに更新する必要があります。

Flow Force Server のインストール

- 1. Altova Web サイト(<u>https://www.altova.com/download#server</u>)からインストールファイルをダウンロードして実行します。
- 2. 任意で、ウィザードの左下のボックスからインストール言語を選択し、「次へ」をクリックします。ここで選択される言語は、Web ブラ ウザー内のFlowForce Server ユーザーインターフェイスの言語も決定します。必要であれば、後に、<u>コマイドライン</u>から言語を変 更することができます。
- 3. 以下の内のつを行います:
 - a. Altova LicenseServer かまだインストールされていない場合、デフォルの設定を保持してくたさい。ウィザードか最新バージョンのAltova LicenseServer をウィザードを作動しているコンピューターにインストールにます。
 - b. Altova LicenseServer かおデインストールされてなく、他のエレビューターにAltova LicenseServer をインストールする場合、「使用中のコンピューターにAltova LicenseServer をインストールする」チェックボックスをクレアして、「後で登録」を選択します。この場合、Altova LicenseServer のインストールと FlowForce Server の登録を個別に行う必要があります。

 - d. Altova LicenseServer が既に使用中のエンピューター、おけよ ネットワークにインストールされており、ウィザードに示される デジョンと同じの場合は 以下を行ってください
 - i. 「使用中のコンピューターにAltova LicenseServer をインストールする」チェックボックスをクリックしてクリアします。
 - ii. 「この製品を登録する」からFlowForce Serverを登録するAltova LicenseServer インスタンスを選択、おこは「後で登録」を選択してくたさい。LicenseServer への関連を無視し、FlowForce Server のインストールを継続する場合、「後で登録」を常に選択することができます。

Altova FlowForce Server - Installer Wizard		
ALTOVA		
This product requires registration with Altova LicenseServer This installer contains Altova LicenseServer		
Register this product with:		
Register later		
< <u>B</u> ack <u>N</u> ext > Cancel		

- 4. 「次へ」をクリックします。
- 5. 任意で、インストールする追加 Altova サーバー製品を選択してくたさい。
- 6. FlowForce Server とFlowForce Web Server に適用することのできるやホワークの設定とWindows サービスの構成 を任意で定義することができます。詳細に関しては、<u>やホワーク構成</u>とWindows サービスの構成を参照してくたさい。 FlowForce のインストール時に同じダイアログボックスから、それぞれのダブをクリックすることにより、他のAltova 製品のためにオプ ションでやホワークとサービス設定を定義することもできます。これらの設定をインストール後に行う場合、「次へ」をクリックします。

Altova FlowForce Server 2020 Advanced Edition - インストーラー ウィザード			
ALTOV	A		
FlowForce Serve	FlowForceWeb Server	RaptorXML Server	r
ウェブサーバーへの ポート番号: 4646	暗号化されていない接続		変更
ウェブサーバーへの ポート番号: 4647	暗号化されている SSL		変更
サービスの構成 開始の型:自動,『	コグオンアカウント: ローカル	システム	変更
	< 戻る(B)	次へ(N) >	キャンセル

- 7. 任意で、FlowForce ケラスター構成オプションをセナアップします。このインストールステップは、FlowForce Server Advanced Edition のためこのが適用することができます。<u>Windows 上でクラスターをインストールする</u>を参照してくたさい。 FlowForce Server をクラスターで作動する必要がない場合、「スタンドアロン」を選択します。
- 8. ウィザードに従いインストールを完了してくたさい。インストールが完了すると、システム通知エリアに、Altova ServiceController アイコン(🌆) が追加されます。

Windows サーバーコア上でのインストール

W indows Server Core は多数のGUI 機能を使用しない最低限のW indows インストールです。W indows Server Core マシン 上にFlowForce Server を以下のようにインストールすることができます:

- 1. Altova ダウルードセター(<u>https://www.altova.com/ja/download</u>) からFlowForce Server 実行可能ファイルをダウン ロードして実行します。使用中のサーバープラナトフォーム(32-ビナトおけよ64-ビナト)にマッチする実行可能ファイルが選択されて いることを確認して下さい。
- 2. グラスカルなユーザーインターフェイスが搭載されているWindows エピューターでコマドプロンプトウイドウを管理者とて開き インストール実行可能ファイル上の以下のコマドを実行します:

FlowForceServerAdv<version>.exe /u

- 3. 次のステップのために必要な.msi ファイルが解凍されます。Windows Server コアで作動するコンピューターに.msi ファイルを コピーします。
- 4. FlowForce Server の前のレージョンを更新する場合 FlowForce Server を次のステップを実行する前にシャナメダウムます。
- 5. .msi ファイルを以下のコマイを使用して FlowForce Server をインストールます:

msiexec /i FlowForceServerAdvanced.msi

インストールの戻り値をテストするには、以下に類似するスクリプトを実行する必要がある場合があります:

```
start /wait msiexec /i FlowForceServerAdvanced.msi /q
echo %errorlevel%
```

インストールオペレーションのリターシュードはSerrorlevels 環境変数内で使用することができます。リターシュード 0 は成功を示しています。

リターシュードとインストーリレプロセスのログを持つサイレントインストールのために以下を実行します。

start /wait msiexec /i FlowForceServerAdvanced.msi /q /L*v! <pathToInstallLogFile>

メモ msi ファイルを安全な場所に保管してくたさい。後にアンインストール、修復、おけな変更する場合これらが必要になります。

インストールを変更するコお以下を実行します:

msiexec /m FlowForceServerAdvanced.msi

インストールを修復するコお以下を実行します:

msiexec /r FlowForceServerAdvanced.msi

FlowForce Server をアンインストールするココお以下を実行します:

msiexec /x FlowForceServerAdvanced.msi

おけよ以下のコマドはFlowForce Server を確認せずアンインストールしてログファイルに結果の詳細をレポートします:

start /wait msiexec /x FlowForceServerAdvanced.msi /q /L*v! <pathToUninstallLogFile>

Flow Force Server のライセンス供与

- 1. Altova LicenseServer の最新バージョンがインストールされており、ローカルズシン、おけよ ネトワーク上で実行されていることを確認してくがさい(<u>https://www.altova.com/ja/licenseserver</u>)。
- 2. インストール中に行っている、場合、FlowForce Server をAltova LicenseServer に登録します。このためゴよが作動中 のエピューターのオスト名、おゴよ、アドレスを知る必要があります。例えば、LicenseServer が127.0.0.1 上で作動する場合、登録しまり下のようご行うことができます。

FlowForceServer.exe licenseserver 127.0.0.1

3. LicenseServer 管理者インターフェイスコングオン、FlowForce Server か実行されるマンノこライセンを割り当てます。

詳細に関しては、Altova LicenseServerドキュメトを参照してたさい(<u>https://www.altova.com/ja/documentation</u>)。

インストール後のタスク

1. インストール中に行っていない場合、ベトワーク設定を構成します

2. Web 管理インターフェイスコングオン、デフォルのFlowForce Server root パマワードを変更します(アドレスとポートを変更していない場合、デフォルトではhttp://localhost:8082です)。デフォルトのログイン名とマワードはそれぞれ root とrootです。

2.6.2 Windows サービスの構成

Windows オペノーティングシステム上にFlowForce Server をインストール中、以下の設定を構成することができます:

- FlowForce Server をWindows サービスとして開始する方法 自動、おさよ オンデマイ、遅延された自動、おさよ 無効 化。
- FlowForce Server によりオペレーティングシステムユーザーアカウストをWindows サービスとして使用する方法 ローカルシステム、ローカルサービス、ネットワークサービス、おけは他のユーザー。
- FlowForce Server か他のユーザーとて実行されるように構成されていると、Windows Services 管理コンノールと同様 にこのユーザーのユーザー名と ペワードを設定することができます。

If you select 他のユーザーを選択するとユーザーアカウト(おけおシンドーであるグループ)はC:\ProgramData\Altovaへの読 み取り書き込みアクセスを有している必要がありますそれ以外の場合、インストールまけは起動の失敗を引き起こす可能性があります。

上にリストされる設定は、FlowForce Web Server のためご構成することもできます。

設定には、Windows Services 管理コンノールからサービスプロレテクを変更することと同じ意味があります。

インストール中に上記の設定を定義するコよインストールダイアログボックスの関連するタブをクリックします(FlowForce Server お」は FlowForce Web Server)。そして、「サービスの構成」の横の「変更」をクリックします。

Altova FlowForce Server	- インストーラー ウィザード
ALTOV	0
FlowForce Server	FlowForceWeb Server RaptorXML Server
ウェブサーバーへの暗 ポート番号: 4646	号化されていない接続変更
ウェブサーバーへの暗 無効化済み	号化されている SSL 変更
サービスの構成 開始の型:自動,ログ	変更 [*] オンアカウント: ローカルシステム
	< 戻る(B) 次へ(N) > キャンセル

インストール後に設定を変更する場合、Windows Services 管理コンノールを開き、必要とするサービスのプロ 守を変更します。 Windows Services 管理コンノールを開くには、コマイラインケイドでを開き、Services.msc を入力します。

2.6.3 ネットワーク構成 (Windows のみ)

Windows オペレーティングシステム上にFlowForce Server をインストール中、以下の設定を構成することができます

• FlowForce Server への暗号化されてなは接続か許可されている場合のポート番号。



• FlowForce Server への安全な(SSL-暗号化済み) 接続が許可されているポート。デフォルトで安全な接続は無効化されま す。SSL-暗号化済みの接続を有効化するコよ、適用可能な場合、証明書ファイル、秘密キー、中間証明書が必要です。 FlowForce Serve のためこSSL を有効化する
て説明されているとおり、他の必要条件が適用されます。

ウェブサーバーへの暗号化	されている SSL	8	
FlowForce Server			
クライアントがこのウェブサーバーと連絡を取るために使用するポート番号を入力し てください。 安全な <mark>(SSL)</mark> コミュニケーションのために必要な証明書と秘密キーを 選択してください。			
	■ 有効化済み		
术-卜番号:	4647		
証明書:	C:¥path_to¥certificate	参照	
秘密キー:	C:¥path_to¥PrivateKey	参照	
	ОК	キャンセル	

メモ 上にリストされる設定は FlowForce Web Server のために構成することできます。

インストール中に上記の設定を定義するコよ、インストールダイアログボックスの関連するタブ(FlowForce Server お」はFlowForce Web Server)をクリックし、「変更」をクリックします。

Altova FlowForce Server 2017 rel. 3 - インストーラー ウィザード
FlowForce Server FlowForceWeb Server LicenseServer RaptorXML Server
ウェブサーバーへの暗号化されていない接続 変更 ポート番号: 4646
ウェブサーバーへの暗号化されている SSL 変更 変更
サービスの構成 開始の型:自動,ログオンアカウント:ローカルシステム
< 戻る(B) 次へ(N) キャンセル

上記の設定を後で行う場合は、「次へ」をクリックします。この場合、FlowForce Server は、ダイアログボックスの対応するタブの下に表示 されるデフォルトのオプションを使用してインストールされます。インストール後、<u>ネ・トワーク設定の定義</u>で説明されているとおと設定を変更する ことができます。

2.6.4 データを手動で移行する

W indows では、構成デーを手動で移行する必要は通常ありません。FlowForce Server の新しい メジャー・ デュンをインストールする際に、前の、デジョンカ既にインストールされている場合、インストールウィザードは、構成データの移行に関してプロンプトします。

構成データを手動で移行する場合、以下の手順を踏んでくたさい。

構成データを手動で移行する方法:

- 1. システム通知エリアで、Altova ServiceController Altova ServiceController 開始してくたさい。それ以外の場合、Altova ServiceController 開始してくたさい。
- 2. FlowForce Server サビスとFlowForce Web Server サビスを停止します。
- 3. 2021 インストールウィザードによりインストールされた Flow Force Server データフォルダーを削除してくたさい。Windows バージョンによりデータフォルダーへの いは異なります (Flow Force Server の構成データの保管方法を参照してくたさい)。
- 4. コマドプロンプ・でmigratedb コマドを使用して Flow Force 実行可能ファイルを実行します。

"C:\Program Files(x86)\Altova\FlowForceServer2021\bin\FlowForceServer.exe" migratedb

--datadir=C:\ProgramData\Altova\FlowForceServer2021\data --olddatadir=C:\ProgramData\Altova\FlowForceServer2017\data

5. FlowForce Server サービスとFlowForce Server サービスを開始します。

2.6.5 インストールフォルダー

Windows 上でFlowForce Server のデフォルトのインホールディレケーリの いは 以下の通りです:

Windows 7、8、10	C:\Program Files\Altova\
64 ビナ OS 上の32 ビナバージョン	C:\Program Files (x86)\Altova\

2.7 Altova Server の移行

このトピックでは、1つのサーバーマンカビる種のサポートされるプラナフォームを含む)他のマンへのAltova サーバーの移行について説明されています。これらの手順は、「古いサーバー」おゴゴ古いマシン」とう用語をデータを移行するたのマンクを説明するナックを説明するナックで使用し、「新しいサーバー」おゴゴ新し、マシン」と言う用語をデータを移行するたのマンクを説明するナックコ使用します。

Altova LicenseServer

Altova LicenseServerを1つのサーバーマンクから、他のダウクタイムの少ないものこ移行するコよ以下のステップを実行してくたさい

- 1. Altova LicenseServer を新しいマシントインストールします。
- 2. 新しいマンのAltova LicenseServer を停止し、licenseserver.db データベースファイルを古いマンカら新しいマンイコ ビーします。後に、何らかの理由で最初の状態に戻す必要かあるかもしれないので、licenseserver.db を新しいサーバーで バッグアップしてください。licenseserver.db ファイルは、LicenseServer アプリケーションディレオトリ内にあり、パンオオペレ ーティングシステムこと異ないます:

Windows C:\ProgramData\Altova\LicenseSe:	
Linux	/var/opt/Altova/LicenseServer
macOS	/var/Altova/LicenseServer

- 3. 古マンのAltova LicenseServer を停止し、新しマンでAltova LicenseServer サビスを起動してたさい。 LicenseServer の2つのインスタンスが同時に作動していることを確認してくたさい。
- 4. すべてのAltova サーバー製品を新しいAltova LicenseServer に登録します。移行するライセンスの数が少な 場合は、手動で行うことができます。移行するライセンスの数が多し場合、Altova LicenseServer 管理下にあるそれぞれのサーバー製品のために "register" コマイドを呼び出すコマイドラインスクリプトを作成し登録を自動化することができます

Altova FlowForce Server

バッケアップとカバーを参照してくたさい。

Altova MobileTogether Server

Mobile Together Server ドキュメント内で説明されているとおり、バックアップと復元の機能を使用してください。

Altova MapForce Server

MapForce Server、では新しマンンへの移行は古いマンンから新し、マンンへのライセンスの再割り当てから構成されています。手順は以下の通りです。

- 1. MapForce Server を新しマンノニインストールます。FlowForce Server インストールの一部として既にインストール済みの場合、このステップを無視してくたさい。
- 2. 新しマシンで、MapForce Server をAltova LicenseServer に登録します。
- 3. 古、マシンでは、ケライアトかサーバーを使用していないことを確認してくたさい。(例えば、マピングが実行されない場合)。
- 4. Altova LicenseServer 管理ページを開き、古いMapForce Server マシンからライセンスを非アクティブ化し、新し、マシン に再割り当てします。

Altova RaptorXML(XBRL) Server

RaptorXML(XBRL) Server では新しマンへの移行は古マンから新しマンへのライセンスの再割り当てから構成されています。手順は以下の通りです:

- 1. RaptorXML(XBRL) Server を新しいマシンにインストールします。FlowForce Server インストールの一部とて既にインストール済みの場合、このステップを無視してくたさい。
- 2. 新しいマシンで、RaptorXML(XBRL) Server をAltova LicenseServer に登録します。
- 3. 古いマシンでは クライアト かサーバーを使用していない ことを確認してくたさい。(例えば 変換 か実行されない場合)。
- 4. Altova LicenseServer 管理ページを開き、古いRaptorXML(XBRL) Server マンカシライセンを非アクティブ化し、 新しいマンノニ再割り当てします。

Altova StyleVision Server

StyleVision Server、では新しいマンンへの移行は古いマンンから新しいマンンへのライセンスの再割り当てから構成されています。手順は以下の通りです:

- 1. Style Vision Server を新し、マシンにインストールます。FlowForce Server インストールの一部として既にインストール済みの場合、このステップを無視してくたさい。
- 2. 新しいマシンで、StyleVision ServerをAltova LicenseServer に登録します。
- 3. 古、マシンでは クライアト かサーバーを使用していない こを確認してくたさい。(例えば 変換 か実行されない場合)。
- 4. Altova LicenseServer 管理ページを開き、古いStyleVision Server マンカシライセンを非アクティブ化し、新しいマンン に再割り当てします。

3 サーバーの構成

このチャプターはFlowForce Server 構成の命令について説明されています。これはまインストール後にすくに行う構成、およびサービスの開始と停止、データのバックアップなど多種のサーバーメンテナンス、ませまレーチンタスクなどが含まれています。

FlowForce Server とその設定を以下のアプローチを使用して管理することができます。

セットアップページから	新規にFlowForce Server をインストール後、最初に定義されるのはFlowForce Web Server とFlowForce Server をリンスンする通常ホスト名(おけはIP アドレス)とポーです。これらに関する詳細については、以下を参照くたさい。 ・ セルアンプ ・ セルアンプ ・ ないワーク設定の定義 ・ SSL暗号化のセルアップ
W eb 管理インターフェイスから	上記のやトワーク設定が設定されると、以下の順序で構成することができます:
コマンドラインインターフェイスから	コマバラインインターフェイスを参照してくたさい。
構成ファイルの編集をして	<u>FlowForce Server アプリケーションデータを参照してくたさい。</u>

3.1 セットアップ

FlowForce Server とFlowForce W eb Server がリンスナるインターフェイスとポートを含む、FlowForce Server セトア・プページは、W eb 管理インターフェイスのデフォルトのホトワークバレスとポートとLicenseServer のホトワークバレスを指定することができます。フラウザー内でセトア・プページを開く(おけまURLを取得する)最も簡単な方法は、以下で説明される通り管理特権を使用してセットア・プモードを実行することです。更に高度な構成に関しては以下の「高度のオプション」を参照してくたさい。

セルアップページから構成することの可能なすべての設定への参照はホルワーク設定の定義を参照してください。

Linux

Linux でセオアップモードを開く方法

1. ターミナルを開き、アプリケーションのデータディレクトリニ変更します。

cd /var/opt/Altova/FlowForceServer2021

- 2. 以下の内のつを行います:
 - a. Linux をグランカルなインターフェイスで作動する場合、FlowForce Web server 実行可能ファイルを setup コマナド で実行します:

sudo /opt/Altova/FlowForceServer2021/bin/flowforcewebserver setup

b. Linux をグラフィカルなユーザーインターフェイスで作動する場合、<u>--listen</u> オプションを追加しつつ上記と同じ setup コ マイドを作動します。後者はセナアップページを使用できるインターフェイスとポートを指定します。これが行われると、他のマシ ン上のブラウザーからセナアップページは接続することができます。

上記のステップを実行すると、ブラウザーのアドレスバーにコピーして貼り付けることのできる2つの代替 URL がターミナルで開かれます。最初の URL が作動しない場合、2番目のものを使用してください。

macOS

macOS でセオアップモードを開く方法:

• アプリケーション内で、FlowForce Server 2021 Advanced Edition アイコンをダブルクトックします。

緊急の場合セナアップモードを以下のように入力することができます。

1. ルート特権を使用してターミナルを開き、アプリケーションのデータディレクトリニ変更します:

cd /var/Altova/FlowForceServer2021

2. setup コマイを使用して Flow Force Web サーバー実行可能ファイルを実行します

/usr/local/Altova/FlowForceServer2021/bin/flowforcewebserver setup

上記のステップを実行すると、ブラウザーのアドレスバーにコピーして貼り付けることのできる2つの代替 URL がターミナルで開かれます。最初の URL が作動しない場合、2番目のものを使用してくたさい。

Windows

Windows でセオアップモードを開く方法:

• スタートメニューカら Altova FlowForce Server 2021 > Flow Force Server セットアップページを選択します。

緊急の場合セルアップモードを以下のように入力することができます

1. 管理者としてコマドプロンプトウィドウを開き、アプリケーションのデータディレクトリニ変更します:

cd C:\ProgramData\Altova\FlowForceServer2021

2. setup コマイを使用して Flow Force Web サーバー実行可能ファイルを実行します:

"C:\Program Files\Altova\FlowForceServer2021\bin\FlowForceWebServer.exe" setup

セトアップページ

上記のステップを実行すると新規のブラウザーウィンドウ内でセットアップページが開かれます(おけよ URL がターミナルで表示され、これをコ ピーしてブラウザーウィンドウニ張り付けることができます)。

LicenseServer			
VIEPDEV02.vie.altova.com	▼ ₽ ₽		
LicenseServer により登録			
FlowForce Web Serve 暗号化されていない接続	er		
有効化済み:	<u>r</u>		
バインドアドレス:	● ローカルのみ (127.0.0.1) ・	◎ その他:	ポート: 8082
SSL 暗号化された接続			
有効化済み:			
バインドアドレス:	● ローカルのみ (127.0.0.1) ・	◎ その他:	ポート: 8089
証明書ファイル:			
秘密キーファイル:			

Windows では詳細コマイプロンプトウィンドウが開かれます。例:



このウイドウはセオアップ中開かれており、セオアップページで設定を適用し Flow Force Server サービスを再起動するボターかう リックされると通常自動的に閉じられます。

設定を適用し Flow Force Server サービスを再起動する ポタンをセットアップページからクリックすると、構成は.ini ファイルニ保存されます。 ます。具体的にはセットアップページを管理者特権で上記の通り開くと、以下のファイルは更新されます。

- DATADIR\flowforcewebserver.ini
- DATADIR\flowforceserver.ini

DATADIR が以下のディレクトリを参照する箇所:

- /var/opt/Altova/FlowForceServer2021/data (Linux)
- /var/Altova/FlowForceServer2021/data (macOS)
- C:\ProgramData\Altova\FlowForceServer2021\data (Windows)

柔軟性が必要な場合、高度なオプションを使用して setup コマイを以下のように実行することができます。

高度なセトアップオプション

setup コマイは下にリストされる高度なオプションの一部をサポートします。

datadir	このオプシンを使用して.ini構成ファイルが書き込まれる個所のデルノケリーのの やを提供できます(通常は上記の DATADIR です)。ターゲメディレクトリが高度な「書き込み」特権を必要とする場合セムアップモードをそのデルノケリ に書き込みが可能な特権 (root おけま administrator) アカウトを使用して実行する必要かあります。 このオプションが設定されていない場合.ini ファイルは現在のディレクトリの.\data サブディレクトリに保存されます。この ためセムアップコマンドを(上記の通り)アプリケーションディレクトリから実行、おけまdatadir オプションを提供して 実行します。		
listen	デフォルトではセナアップが実行される都度、セナアップページのURLはフレーのランダムはポートに再生成されます。 例: http://localhost:50492/setup。		
	このオプションにより(localhost おけは127.0.0.1以外)をリッスンする代替の インターフェイス/ポート 組み合わせを 設定することができます。他のマシン上のブラウザーからセットアップページにアクセスする場合これは役に立ちます。		
	例えばコマイ:		
	flowforcewebserver setuplisten=0.0.0.0:10008		

	全てのインターフェイス上でポート 10008 上でリッスンのセナアップが作成されます。以下の点に注意してください
	 実際のデータディレクトリと外部やトワークインターフェイスへのゾインディングと共にセットアプの実行を特権を 使用するようご設定することが奨励されます。これを行う場合次のオプタン(key) は役に立ちます。代替 としてdatadir オプションを使用して一時的なデータディレクトリを提供します。これにより公開済みに公開 ページから実際のFlowForce Server 構成を更新することを回避できます。 ポートは既に使用されているため、ノーマルな(セットアップではよし、)FlowForce Web Server おさま FlowForce Server インスタンスと同じポートを使用しないでくたさい。
	バインドアドレス(インターフェイス)がローカルではない場合、指定されたポートを介してアクセスが可能になるように オペレーティングシステムのファイアーウォールを構成する必要がある場合があります。
key	このオプションにはレセトア・プページのナッカニアクセスキーを設定することができます。この場合、URL内に正確なアクセ スキーが提供されている場合のみ・セットア・プページを保存することができます。キーはURL内に含まれる必要のある任 意の文字列であることができます。例えば以下のようなLinuxコマンドを実行すると
	flowforcewebserver setuplisten=wild.berries.com:8015 key=all_cats_love_fish datadir=/var/opt/Altova/FlowForceServer2021/data
	接続するURL は以下の通りよります。
	http://wild.berries.com:8015/setup?key=all_cats_love_fish
	目的のためにキーフレーズが安全であることを確認してくたさい。
	HTTPS パラメーターを構成するなっとに自身を使用しているなった、セオアップページはHTTPS を使用しないてとご注意 してくたさい。セオアップページは継続的に使用することができず、セオアップ中だけに開かれていることご注意してくたさい。

3.2 ネットワーク設定の定義

FlowForce Server とFlowForce W eb Server がリンスナタイオンとポート、およびネットワークス関連した設定をセットアップページから指定することができます。セットアップページをオペレーティングシステムゴ従い異なる方法で開くことができます。または、設定の大半は、構成ファイルを用いて定義することができます。構成ファイルの参照を参照してくたさい、セットアップページ内で定義されている設定は FlowForce Server の新規のマイナー・デジョンをインストール場合保管されます。メジャー・デジョンのマイナー・デジョンをインストールすると、インストール中、前のメジャー・デジョンカらデーダを移行する場合のみ設定は保管されます。

構成可能な設定は下にリストされる通りです。

LicenseServer

FlowForce Server は LicenseServer に登録されている必要が対応す(次を参照するAltova LicenseServer)。 インストール 中に LicenseServer ホストが指定されていない場合、 Altova LicenseServer が作動するマシンのエコニアドレス おけよ ホスト名を 入力します。(LicenseServer がローカルでインストールされている場合)これはローカルマシン、おけよ ネットワークアドレスであることができ ます。

LicenseServer	
localhost	
LicenseServer により登録	

FlowForce Web Server

設定のこのグループはFlowForce Server W eb 管理インターフェイスへのブラウザーからのHTTP(S) リクエストを処理するナメのサービ スである FlowForce Web Server サービスに適用可能です。(「FlowForce W eb Server」と「FlowForce Server」の違いを理 解するコよ 次を参照してください<u>作動の仕組み</u>)。

FlowForce Web Ser 暗号化されていない接続	ver 売
有効化済み: バインドアドレス: ホスト名:	 ✓ ● 全てのインターフェイス (0.0.0.0) ∨ ○ その他: ポート: 8082
SSL 暗号化された接続	
有効化済み:	
バインドアドレス:	 ● 全てのインターフェイス (0.0.0.0) ∨ ○ その他: ポート: 8089
ホスト名:	example.name.com
証明書ファイル:	C:\secure\flowforceweb.crt
秘密キーファイル	C:\secure\flowforceweb.key
証明書チェーンファイル:	C:\secure\intermediate.pem

暗号化されていない接続 / 有効化済み	FlowForce Web Server へのプレーンな HTTP(暗号化されていない)接続を有効化するオンタンニ のチェックボックスを選択します。 インストーリゅ中、 おさは構成ファイルから変更されていない 場合、 デフォル トでは(暗号化されていない) プレーンな HTTP からの接続か有効化されています。 構成ファイルの参照。を参照してくたさい。
	*FlowForce Server インストール中のネットワークシステムの変更はWindows のみで許可されています。
ハイントアドレス	Windows 上では、FlowForceWebサード管理者インターフェイスは、デフォルトでポト 8082 上のすべてのネトワークインターフェイスで使用することができます。Linux とMac OS ではポト番 号は、インストール中にランダムに選択されます。「ローカルのみ」ませは「全てのインターフェイス」以外に カスタムのアドレスを指定するはよ「他の」テキストボックスに入力します。
	バインドアドレス(インターフェイス)がローカルではおよい場合、指定されたポートを介してアクセスが可能してなるようにオペレーティングシステムのファイアーウォールを構成する必要がある場合かあります。
*+	FlowForce W eb Server がリッスンするTCP ポートを指定します。ポートは使用済みであってはな りません。
<i>木</i> スト <i>名</i>	ホスト名 フィールドは空では、場合、バインドのナカリニ使用される固定ホスト名を設定します。ネトワーク上の他のマシンカ接続するナカリニ使用する FlowForce Web Server を作動するマシンの名前を 設定します。
	通常、FlowForce は使用する適切なオストを自動的に検知します。このフィールドを明示的に設定した場合、自動検知が優先されます。所属機関内のホルワーク構成により somehost おさま somehost.example.org などの値を使用する必要がある場合があります。
	バインドに関連したオスト名はSSL のために使、およびWindows 上のAltova Service

Controller によ使用されて、ます(<u>FlowForce Web Server のためにSSLを有効化</u> 照してくたさい)。SSL か有効化されている場合、ホスト名は証明書の共通の名前 プロ・デ する必要かあります。	
	バインドアドレスがローカルではない場合(すなオカ・バインドアドレス フィールドが[Local (127.0.0.1)] 以外に設定されている場合)ホスト名の設定は意味があります。
SSL 暗号化された接続	FlowForce Web サーバーのためにSSLを有効化する方法も参照してくたとい。

FlowForce Server

FlowForce ジョブからHTTP(S) クライアントに作成されたサービスを公開するサービスである FlowForce Server ナ	ナビス:設定のこ
のグループを適用することが可能です。	

FlowForce Server 暗号化されていない接続			
	有効化済み: バインドアドレス: ホスト名:	 ○ □ーカルのみ (127.0.0.1) ○ その他: ポート: 4646 	
SS	SSL 暗号化された接続		
	有効化済み:		
	バインドアドレス:	● □ーカルのみ (127.0.0.1) ∨ ○その他: ポート: 4647	
	ホスト名:	example.name.com	
	証明書ファイル:	C:\secure\flowforce.crt	
	秘密キーファイル	C:\secure\flowforce.key	
	証明書チェーンファイル:	C:\secure\intermediate.pem	

暗号化されて ない 接続 / 有効化済み	FlowForce Server へのプレーンな HTTP (暗号化されていない) 接続インストールド中、ませば構成 ファイルから変更されていない場合、デフォルトでは、(暗号化されていない) プレーンな HTTP からの接続が有効化されています。 構成ファイルの参照。を参照してくたさい。 * FlowForce Server インストール中のネットワークシステムの変更は Windows のみで許可さ れています。
バイントアドレス	FlowForce Server のデフォルトの設定は、ポト 4646 上の同じマンン(127.0.0.1) からのリクエ ストを暗号化されていたは 接続を使用して受け入れます。 リモートのマンクからプレーシ HTTP を使用し て、Web サービスとしてショブを開始する場合は、 バインド アドレス コンボドックスから「全てのインター フェイス(0.0.0.0)」を選択します。
	バインドアドレス(インターフェイス)がローカルではない場合、指定されたポートを介してアクセスか可能になるようにオペレーティングシステムのファイアーウォールを構成する必要かある場合かあります。

<i>ポ</i> +	FlowForce Server がルスンするTCP ポートを指定します。ポートは使用済みであってしばいません。
ホスト名	フィールド ホスト名 はFlowForce Server がWeb サービスとして公開されているジョブにアクセスするフライアントからの接続をリッスンするインターフェイスコップナイされるホスト名を指定します。
	バインドアドレスが[ローカル(127.0.0.1)]に設定されている場合ホスト名の設定は意味があります。 所属機関内のホトワーク構成により somehost おは somehost.example.org などの値を 使用する必要がある場合があります。
	バイナに関連したホスト名はSSLのために使用されています(<u>FlowForce Server のためにSSL</u> <u>を有効化する</u> を参照してくたさい)。SSL が有効化されている場合、ホスト名は証明書の共通の名 前プロンティニー致する必要があります。
	ホスト名は上のでも使用されています。ホスト名を設定しないと、FlowForce はWindows 上の Altova Service Controller により使用されている最初の適切なホスト名を自動的に検知します。
	hostname か構成されている場合 FlowForce Web インターフェイスはホームページのアクティブよト リガーとサービス セクション内のレノを含むWeb サービスとして公開されているショブをナビゲートするた
	めニクトック可能なレンを表示することができます。Web サービスの呼び出し ▶ ポタンが新規のブ ラナザーウィンドウ内でWeb サービスの呼び出しを可能にするジョブ構成ページのサービスセクション内で 使用できるよう」こなります。詳細に関しては <u>Web サービスの公開</u> を参照してくたさい。
SSL 暗号化された接続	FlowForce Server のためこSSLを有効化する方法も参照してくたさい。

マスターインスタンス暗号化済み接続

FlowForce Server がFlowForce Server を作動している複数のマシンのケラスター内のマスターインスタンスの場合、下の設定は構成 されている必要があります。配布済みの実行と、マンスのロードを参照してくたさい。これらの設定は、flowforce.ini構成ファイル内で使用 することができます。構成ファイルの参照を参照してくたさい。

マスターインスタン	ス暗号済み接続	
有効化済み:		
バインドアドレス:	● 全てのインターフェイス(0.0.0.) > ○ その他:	ポート: 4645

3.2.1 ファイル参照の構成

セルアップページませま.ini 構成ファイルからFlowForce Server とFlowForce W eb Server のネルワーク設定を以下で説明されるように構成することができます。

上記のディレケリ内では 2つの.ini ファイルが存在します。1つは Flow Force Server (flow force.ini) のかかにもう一つは Flow Force Web Server (flow force web.ini) のかかです: ini 構成ファイルは次の 次で見つ おことができます:

Linux	/var/opt/Altova/FlowForceServer2021/data/flowforce.ini /var/opt/Altova/FlowForceServer2021/data/flowforceweb.ini
macOS	/var/Altova/FlowForceServer2021/data/flowforce.ini

	/var/Altova/FlowForceServer2021/data/flowforceweb.ini	
Windows	C:\ProgramData\Altova\FlowForceServer2021\data\flowforce.ini C:\ProgramData\Altova\FlowForceServer2021\data\flowforceweb.ini	

上記のディレケリ内では、2つのサンプル.iniファイルが存在し、テンプレートとして使用することができます:

- flowforce.ini.template
- flowforceweb.ini.template

.ini ファイルの編集後、対応するサービス(FlowForce Server おけまFlowForce Web Server)を再起動してくたさい。詳細 は関しては、次を参照してくたさい

- <u>サービスの開始と停止 (Linux)</u>
- サービスの開始と停止 (macOS)
- サービスの開始と停止(Windows)

サンプルのflowforce.ini ファイルは以下のよう」てなります:

```
[Listen]
active=1
host=127.0.0.1
port=4646
hostname=
[ListenSSL]
active=1
SSL=1
host=0.0.0.0
port=4647
hostname=
[SSL]
certificate=/path/to/certificate.crt
private_key=/path/to/private_key.key
certificate_chain=/path/to/certificate_chain
[Master]
host=0.0.0.0
port=4645
active=1
```

サンプルのflowforceweb.ini ファイルは以下のよう」ないます:

```
[Listen]
active=1
host=0.0.0.0
port=8082
hostname=example.domain.org
[ListenSSL]
```

```
active=1
SSL=1
```

host=0.0.0.0 port=8083
hostname=example.domain.org
<pre>[SSL] certificate=path/to/certificate.crt private_key=path/to/private_key.key certificate_chain=/path/to/certificate_chain</pre>
[FlowForce]

host=127.0.0.1
port=4646
hostname=

.ini ファイルは、下に説明されるようにセクションで整理されています。2つのファイルの違いは下で説明されています。

[Listen]

[Listen] セクションは、HTTP 接続設定を定義します。複数の[Listen] セクションを定義することも可能です。各[Listen] セクションを定義することも可能です。

active	(任意)[Listen] セクションを有効化、おけは無効化します。有効な値	
	0 無効化済み	
	1 有効化済み	
	例えば active=1 は HTTP 接続が有効化済みであることを意味します。	
ssl	(任意)この[Listen] セクションのためのSSL サポートを有効化します。有効な値	
	0 無効化済み	
	1 有効化済み	
	SSL サポートを有効化するコよ ssl=1 を設定し、[SSL] セクションを以下に示されるようご作成します。	
host	FlowForce (Web) Server のホトワーク・イイドアドレスを指定します例えば 127.0.0.1。 これはIPv4 おけはIPv6 アドレスであることができます。0.0.0.0 をすべてのインターフェイスをリッス ンするけっかに使用します。ローカルアクセスのけっかには、127.0.0.1 を使用します。	
port	FlowForce (Web) Server がリッスンするポートを指定します。このポートが既に使用されていな してとを確認してください。	
	バインドブドレス(インターフェイス)がローカルではよい場合、指定されたポートを介してアクセス が可能になるようにオペレーティングシステムのファイアーウォールを構成する必要がある場合があ ります。	
hostname	ホスト名 フィールドは空では、場合、バインドのナダロ・使用される固定ホスト名を設定します。ネオワーク上の他のマンカ接続するナメロロ使用する FlowForce Web Server を作動するマンクの名前を設定します。	

	通常、FlowForce は使用する適切なホストを自動的に検知します。このスイールドを明示的に設定した場合、自動検知が優先されます。所属機関内のネトワーク構成により somehost おけま somehost.example.org などの値を使用する必要がある場合があります。 バインドに関連したホスト名はSSL のために使、およびW indows 上のAltova Service Controller により使用されています(FlowForce Web Server のためにSSL を有効化するを参照してくたさい)。SSL が有効化されている場合、ホスト名に証明書の共通の名前 プロ・ディ に一致する必要があります。
<pre>max_request_body_size</pre>	このオプションによりFlowForce Server おけまFlowForce Web Server, へのHTTP リケエ ストの最大サイズを、小いて指定できるようプロジます。例: max_request_body_size=50000000
	デフォルトでは、明示的なにたかは100 MB (100,000,000 バイト)くらいです。このオプタンを以下のシチュエーションで明示的に設定する必要がある場合があります。 1. FlowForce Web サービスをショブとして公開するようコ呼び出し、HTTP リクエストボ ディがデフォルトのにかんと見たも、場合。 2. MapForce からFlowForce Server にマッピングをデプロイすると、入力ファイルがデ フォットのにかんと見たも、場合。
	ケース1の場合、オプションはflowforce.iniファイル内でのみ設定される必要があます。ケース 2の場合、オプションはflowforce.iniとflowforceweb.iniファイル内の両方で設定される必要があます。

[SSL]

このセクションはSSL/HTTPS 接続設定を定義します。

certificate	PEM 書式の証明書ファイルへの絶対ルマを指定します。		
private_key	秘密キーファイルへの絶対パンを指定します。		
certificate_chain	証明書チェーンファイルへの任意のパジを指定します。		

[FlowForce]

このセクションはFlowForce Web Server (flowforceweb.ini ファイル)のみに適用することができます。FlowForce Web サービスとFlowForce Server サービス間の接続の詳細を定義します。

ssl	(オプノョン)FlowForce Web Server とFlowForce Server の間の接続のナメのSSL サポートを有効化します。有効な値		
	0 無効化済み 1 有効化済み		
host	FlowForce Server のIP アドレスを指定します。		

	FlowForce Server がすべてのインターフェイスコン・インドされていない場合、この値は flowforce.iniの"[Listen]"セクションの一つと同じである必要があります。それ以外の場合は 127.0.0.1 です。
	SSL がそのポートで有効化されていると、この値はflowforce.iniの"[SSL]"セクション内で構成済 みの証明書の「共通名]プロ、ティニー致する必要があります。
port	FlowForce W eb Server がFlowForce Server に接続するTCP ポーを指定します。
	この値は同じポート番号を持つflowforce.iniの対応する"[Listen]" おけは"[ListenSSL]" セク ションの一つと同じである必要があります。
	SSL がのポート上で有効化されているとhost とhostname (封けは hostname が存在しな い場合 host) は flowforce.ini の "[SSL]" セクション内で構成されている証明書の[共通名]に 一致する必要があります。
hostname	空ではよい場合、Webサービスとして公開されているFlowForceジョブは接続するけっとのフィー ルドはノットワーク上の他のマシンにより使用されている固定ホスト名を設定します(<u>Webサービスと</u> て公開するを参照してくたさい)。
	所属機関内のホナワーク構成により somehost ませま somehost.example.org などの植を 使用する必要かある場合かあります。
	SSL がそのポートで有効化されていると、この値はflowforce.iniの"[SSL]"セクション内で構成 済みの証明書の【共通名】プロ・ティニー致する必要があります。
	Altova Service Controller も木木名を使用してます。ホスト名を設定しないと、FlowForce はAltova Service Controller によ使用されている最初の適切なホスト名を自動的に検知します。
	hostname か構成されている場合 FlowForce Web イノターフェイスはホームページのアクティブな トリガーとサービス セクション内のレノを含むWeb サービスとして公開されているジョブをナビゲートする
	ためコクトック可能なレンを表示することができます。Webサービスの呼び出し おなっか新規のブラウザーウィドウ内でWebサービスの呼び出しを可能にするジョブ構成ページのサービスセクション内で使用できるようコンはます。
certificate	(オプション)FlowForce Server により受け入れられるサーバー証明書を定義します。証明書が与 えられていない場合、システムルート CA 証明書が使用され、サーバー証明書が検証されます。存在 する場合、この値は(flowforce.ini ファイル内の)FlowForce Server を使用する証明書にこ の値が一致する必要があります。

[FlowForceWeb]

このセクタンはFlowForce Web Server (flowforceweb.ini ファイル)のみに適用することができます。

timezone	FlowForce W eb Server のデフォルトのタイムゾーンを指定します。例えば	
	timezone=Europe/Berlin	

[Master]

このセクションはflowforce.ini ファイルのみに適用することができます。複数のFlowForce Server インスタンスがプラスター内で実行されている、現在のインスタンスがマスター インスタンスである場合のみ関連性があります。配布済みの実行と、ランスのロードを参照してくたさい。

active	このマスターインスタンスへの暗号化済みの接続を有効化します。有効な値	
	0 無効化済み 1 有効化済み	
binding address	マスターFlowForce Server インスタンスのバイボブドレスを指定します。0.0.0.0 をすべてのイン ターフェイスをリッスンするすっかに使用します。	
port	<u>ワーカー</u> インスタンスからのリクエスト のためにマスターインスタンスがリッスンするポート。	

3.3 SSL 暗号化のセットアップ

FlowForce を構成し、次のHTTP 接続をSSL 証明書を使用して暗号化することができます。

- 1. ブラウザーとFlowForce Web Server 間の接続。
- 2. (例えば ケライア・トアプリケーションなどの) Web サービスコンシューマーと Flow Force Server サービス間の接続。
- 3. FlowForce W eb Server と FlowForce Server 間の内部接続 (FlowForce Server が FlowForce W eb Server とのように異なるが 理する詳細は 次を参照してくたい、 しくみ)。

AS2 データの交換のためこFlowForce を使用する場合、SSL 証明書を任意で使用して、AS2 サービスの一部としてデータを暗号化 することができます。次を参照してくたさい、AS2 統合。

上記の1 と2 接続のために、その証明書に対応するSSL 証明書と秘密キーが必要にないます。セキュレティ上の理由から、各接続のため に個別のSSL 証明書と秘密キーが必要にないます。両方の接続のために同じ 証明書と秘密キーを使用する場合、FlowForce Server とFlowForce Web Server の両方は同じ 完全修飾ドメイン名 (FQDN) を必要とします。例えば、FlowForce Web Server がhttps://somehost:8083 をリンスンする場合、FlowForce Server はhttps://somehost:4647 をリンスンする 必要があります。ポートは後で変更することができるために、この場合ホスト名のみか重要であることに意してくたさい。

上記の接続3に関しては3番目の証明書と秘密キーのペアは必要無く、同じSSL証明書をFlowForce Server のために使用することができ、この場合、FlowForce Web Server はFlowForce Server のHTTP クライアントとしての役割を果たします。

FlowForce Server 内のSSL 接続を暗号化するためこ必要な証明書を取得するコよ 以下のオプノョンかあります:

- CSR (証明書署名要求)を生成し、DigiCert、Comodo などの公開されている証明書機関(CA)に比署名してくたい、 ブラウザー(まけま、オペレーティングシステム)はすで「証明書機関を信頼するため、CAブラウザーの大多数はこのような証明書 機関により署名されているサーバー証明書を信頼します。証明書機関により署名済みの証明書を取得する方法に関しては、次 を参照してくたさい、証明書機関によりSSL 証明書を署名する方法。
- 2. おけよ FlowForce Server がプイベートのやトワーク上で作動する場合、そして、この期間内でこれを行う権利を有してい る場合、自身のSSL ルート証明書機関を構成することが可能です。ブラウザーおけはオペレーティングシステムはこのようは機関を デフォルトでは信頼せず、自己署名 ルート証明書を信頼するために FlowForce Server に接続する各マンン(おけよ 場合 によってはブラウザー)を構成する必要あみります。それ以外の場合、ブラウザーばこのサイトは安全ではありません」おけば Web サービスの呼び出しに失敗しました」などの警告を表示します。詳細に関しては、次を参照してくたさい、自己署名済みのSSL 証明書の作成方法。

3.3.1 証明書機関により SSL 証明書を署名する方法

信頼されている証明書機関(CA)からSSL 証明書を購入する場合、秘密キーとCSR (証明書署名要求) が必要です。秘密キーは 安全に保管され、公開されないようしてください。処理中に証明書機関によりCSR が必要とされます。

秘密キーとCSR をオペレーティングシステム上に既存のソール、おけよ、第三者パーティソールを使用して作成することができます(Mac 上のKeychain Access、Linux 上のopenssl)。このサンプルはOpenSSL ツールキナト(<u>https://www.openssl.org/</u>)を使用しています。OpenSSL は オープンノースライブラルがリーズで使用する前にエンパイルされる必要があります。OpenSSL のすめのエンパイルとインストール命令は、各オペレーションシステムで異なり、このドキュメントのスコープではありません。Linux とMac マシン上で、OpenSSL はすでは使用できる可能性があります。それ以外の場合、コマイドラインからインストール、更新することができます。 OpenSSL が存在するか、下記のコマイを入力して確認することができます(このコマイドにより現在のOpenSSL バージョンカ表示されます):

openssl version

W indows では 公式の OpenSSL ソースコードから イナ をコン イル おけよ OpenSSL を含む イナ 配布をダウンロード すること もできます。 https://www.openssl.org/community/binaries.html を参照してくたさい。

署名済みのSSL 証明書を取得する方法:

1. 秘密キーを作成します。次のOpenSSL コマンドは、2048 ビットのサイズである flowforce.key 呼ばれるキーを作成します (証明書機関には通常受け入れられる最低限の暗号化です):

openssl genrsa -out flowforce.key 2048

灹

- 秘密キーはPEM (プライバシー強化メール) フォーマナ である必要があります。PEM ファイルのファイル拡張子 は通常.pem ですが.key.cert.cer、おさは.crt であることもできます。
- 秘密キーは安全に保存されている必要があります。
- 2. 既に生成されている秘密キーのナカに証明書署名要求(CSR)を生成します。SSL 証明書を購入する場合、CSR が必要になる場合があります。次を参照してくたさい、次のOpenSSL コマンドは、CSR とう名前のmyserver.csr をキー flowforce.key のナカン作成します:

openssl req -new -nodes -key flowforce.key -out myserver.csr

プロンプトされると、機関に関する情報を入力します。例

```
Country Name (2 letter code) [AU]: AT

State or Province Name (full name) [Some-State]: .

Locality Name (eg, city) []: Vienna

Organization Name (eg, company) [Internet Widgits Pty Ltd]: MyCompany Ltd

Organizational Unit Name (eg, section) []: IT

Common Name (eg, YOUR name) []: server.my.domain.com

Email Address []: test@example.org
```

メモ

- フィールド 共有される名前 のために Flow Force Server が作動するホストマンの FQDN (完全修飾ドメイン名) が 入力されていることを確認してください。
- プロンプトされると、チャレンジ・ペワードフィールドをそのまましておいてください。
- 3. 証明書機関から証明書を注文します。注文の過程で、CSRを提供する必要があります。これを行うには、myserver.csrを Notepad などのテキストエディターで開き、クトップボードに内容を張り付け、オンラインの注文フォームに張り付けます。
- 4. 証明書機関か所属企業を検証すると、「中間」証明書と呼ばれる購入済みの証明書が提供されます。<u>中間証明書を準備</u> <u>する方法</u>で示されるようコ中間の証明書の内容を一つのファイルにコピーして貼い付けます。

概要

上記のステップに従うと、次の証明書とキーを所有している必要があります。

• flowforce.key - この秘密キーはFlowForce には使用される証明書に当います。

- certificate.crt (ファイル拡張子は異なります) これはブラウザーとFlowForce Web Server 間の接続、おさよ FlowForce Server により公開されているサービスは接続するクライア・トアプリケーションを暗号化する購入済みの証明書です。
- intermediate.pem このファイルコよ 証明書機関から受信されたすべての中間の証明書か含まれます。

FlowForce Server、FlowForce Web Server、およびHTTP のために以下のようにSSL を有効化することができます:

- FlowForce Web サーバーのためにSSL を有効化する方法
- <u>FlowForce</u> サーバーのためこSSL を有効化する方法
- <u>FlowForce Web Server とFlowForce サード間にSSL を有効化する方法</u>

3.3.1.1 中間証明書を準備する方法

証明書機関と証明書を署名する場合、サーバーと証明書機関の間の信頼のチェーンを作成する証明書を受信します。FlowForce Server 内て証明書をすくに使用するコよ、すべての証明書を一つのファイルしサンゆる証明書チェーンファイルルに以下に示されるように結合 する必要があります:

- 1. Notepad などのテキストエディターを使用して、新規のテキストファイルを作成します(intermediary.pem と呼ぶことしますが、他のファイル名と拡張子を選択することもできます)。
- 2. 各中間の証明書をテキストエディターで開き、intermediary.pem ファイル内にコンテンンをコピーして貼り付けます。重要な点 は、証明書テキストが逆の順番でコピーされる必要がある点です(すなオち、2番目の中間証明書が最初にコピーされ、プライマリ の証明書は2番目にコピーされます)。例

```
--BEGIN CERTIFICATE--
... (secondary intermediary certificate) ...
--END CERTIFICATE--
--BEGIN CERTIFICATE--
... (primary intermediary certificate) ...
--END CERTIFICATE
```

3. intermediary.pem ファイルを保存します。FlowForce セオアップページからこのファイルを参照する必要があります。

3.3.2 自己署名済みの SSL 証明書の作成方法

プライベートのホトワーク上で作動するのFlowForce Server ために、自己署名済みSSL 証明書を作成する方法をこのデモは説明しています。このデモは内部で簡素化されており、生産の過程で使用されるこは最適でない場合があります。所属機関には、特定のセキュリティポシーに関連するSSL 証明書が存在し、下記以外のSSL ツールを使用する場合があります。信頼されている証明書機関により署名されるSSL 証明書を取得する情報に関しては、次を参照してください、証明書機関によりSSL 証明書を署名する方法。

前提条件

このサンプルはOpenSSLツールキナ(https://www.openssl.org/)を使用して自己署名済み証明書を生成します。OpenSSL は、オープンノースライブラバあり、コマ・ドラインで使用する前にコンパイルされる必要があります。OpenSSLのオメのコンパイルとインストー ル命令は、各オペレーションシステムで異なり、このドキュメントのスコープではありません。Linux とMac マシン上で、OpenSSL はすでに 使用できる可能性があります。それ以外の場合、コマ・ドラインからインストールし、更新することができます。OpenSSL が存在するか、下 記のコマンドを入力して確認することができます(このコマンドにとい現在のOpenSSL バージョンが表示されます):

openssl version

Windows では公式のOpenSSL ソースコードからイナルをコンパル おけよ OpenSSL を含む イナ 配布をダウンロードすること

もできます。https://www.openssl.org/community/binaries.html を参照してくたさい。

ルート証明書を作成する方法

1. このデモはいで使用されるすべての証明書を補完するディレクトリを作成します(例、「C: &ecure」)。これは、全てのOpenSSL コマナド(関連するファイル)な)のすめの作業ディレクトリです。このすめ、コマ・ドラインからのディレクトリを変更します:

cd C:\secure

- 2. このデモのために、OpenSSL 拡張子を使用して証明書を作成します。これを可能にするこは、OpenSSL ディストリビューションのopenssl.cnf ファイルを検索し、前のステップで作成された作業ディレクトリニピーします。
- 3. ルート秘密キーを作成します。安全な環境内で生成され保管するために、ルート秘密キーか公開キー構造の最も機密な部分であることに注意してくたさい(このデモでは、「C:%ecure」内に保管されます)。

openssl genrsa -aes256 -out root.key 2048

プロンプトされると、ルートキーを保護するオンガン・スワードを入力します。証明書 リクエストを署名するオンガニの、スワードが後に 必要にないます。

4. ルート証明書を作成する方法。下のコマイは、上記で作成された秘密キーのために3650日間有効な自己署名済みの証明書を生成します。-config パラメーターは、同じデルノオリ内のopenssl.cnf ファイルを指していることを確認してくたさい。 -extensions パラメーターは openssl.cnf 内で定義されている「v3_ca」 拡張子(セクション)を参照します。

openssl req -config openssl.cnf -extensions v3_ca -x509 -new -nodes -key root.key -sha256 -days 3650 -out root.pem

プロンプトされると、所属機関の情報を入力します。例:

```
Country Name (2 letter code) [AU]: AT

State or Province Name (full name) [Some-State]: .

Locality Name (eg, city) []: Vienna

Organization Name (eg, company) [Internet Widgits Pty Ltd]: MyCompany Ltd

Organizational Unit Name (eg, section) []: IT

Common Name (eg, YOUR name) []: Demo CA

Email Address []: test@example.org
```

所属機関に適用することのできる必要とされるフィールドに記入します。フィールド共通名に自己署名済みの証明書機関(この デモの場合、「Demo CA」)を入力します。

FlowForce 証明書を作成する方法

(FlowForce Server、おうまFlowForce Web Server、おうま、両方) SSL 暗号化のために使用される実際の証明書を作成する ことができます。次のOpenSSL コマンドは秘密キーを作成します。

openssl genrsa -out flowforce.key 2048

ᆇ

- 秘密キーはPEM (プライバシー強化メール) フォーマナ である必要があります。PEM ファイルのファイル拡張子 は通常.pem ですが.key.cert.cer、おはt.crt であることもできます。
- FlowForce 内で秘密キーを使用可能にするコよ、パワードで保護されている必要があります。秘密キーの必

要条件を参照してくたさい。 • 秘密キーは安全に保存されている必要が決ます。

作業中のopenssl.cnf ファイルを開き、次のセクションを追加します:

```
[ server_cert ]
# Extensions for server certificates (`man x509v3_config`).
basicConstraints = CA:FALSE
nsCertType = server
nsComment = "OpenSSL Generated Server Certificate"
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid,issuer:always
keyUsage = critical, digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth
subjectAltName=DNS:server.my.domain.com
```

FlowForce Server が作動するマシンのFQDN (完全修飾ドメイン名)に対応するように subject AltName (「Subject Alternative Name」) か変更されていることを確認してくたさい。このサンプルで、「server.my.domain.com」に設定されています。サブジェクトの代替名は Google Chrome 58 おけよ以降を必要とします。それ以外の場合、自己署名済みの証明書は NET::ERR_CERT_COMMON_NAME_INVALID エラーを生成します (次を参照してくたさい) https://support.google.com/chrome/a/answer/7391219?hl=en)。

証明書署名要求 (CSR) を下記のとお作成します。-config パラメーターは以前に編集されている openssi.cnf ファイルを指しています。-extension パラメーターは openssi.cnf 内で定義されている「server_cert」 拡張子を参照しています。

```
openssl req -config openssl.cnf -extensions server_cert -new -nodes -key flowforce.key
-out flowforce.csr
```

プロンプトされると、機関に関する情報を入力します。例

Country Name (2 letter code) [AU]: AT State or Province Name (full name) [Some-State]: . Locality Name (eg, city) []: Vienna Organization Name (eg, company) [Internet Widgits Pty Ltd]: MyCompany Ltd Organizational Unit Name (eg, section) []: IT Common Name (eg, YOUR name) []: server.my.domain.com Email Address []: test@example.org

と

- フィールド 共有される名前 のために Flow Force Server が作動するオストマンクの FQDN (完全修飾ドメイン名) が 入力されていることを確認してくたさい。
- · プロンプトされると、チャレンジ・マワードフィールドをそのまましておいてください。

このデモのために、ルート証明書を使用して FlowForce 証明書を直接署名します。生産環境では、ルート証明書はサーバー証明書を直接署名しません。中間の証明書が代わりに使用されます。下のコマンドは前に作成されているルート証明書に対して flowforce.csr 証明書リクエストを署名し、flowforce.crt ファイル(FlowForce Server で必要とされるサーバー証明書)を作成します。

openssl x509 -extfile openssl.cnf -extensions server_cert -req -in flowforce.csr -CA root.pem -CAkey root.key -CAcreateserial -out flowforce.crt -days 365 -sha256

概要

上記のステップに従うと、次の証明書とキーを所有している必要がみます

- root.key 証明書機関(CA)の秘密キーです。このファイルを安全な場所に保管します。このキーが危害を受けると、プラウザ ーイことに「頼される証明書が使用者として生成できるよう」こなります。
- root.pem これは証明書機関の公開証明書です。FlowForce に安全にアクセスする各マシン(おけま ブラウザー)の言頼されている証明書にこの証明書をインストール(インポート)する必要があります。次を参照してください、ルート証明書のインポート.
- flowforce.key この秘密キーはFlowForce には使用される自己署名済みの証明書に伴います、次のアイテムを参照してくたさい。
- flowforce.crt これはFlowForce Server、FlowForce Web Server おは、両方には使用される自己署名済みの 証明書です。

FlowForce Server、FlowForce Web Server およびHTTP 接続間のためにSSLを以下のように有効化することができます:

- <u>FlowForce Web サードのためにSSL を有効化する方法</u>
- <u>FlowForce サーバーのためにSSL を有効化する方法</u>
- <u>FlowForce W eb Server と FlowForce サード間にSSL を有効化する方法</u>

3.3.2.1 ルート証明書のインポート

自身の証明書機関(CA)を作成する場合、ルート証明書は自己署名です。ですから、デフォルトではブラウザーは信頼されません。すなわち、FlowForce Server は接続するブラウザーは、「このサイト」は信頼されていません」などの警告を表示します。(ブラウザーなどの) HTTP クライア・トが自己署名済みの証明書を信頼するコよ、証明書を以下のようにインポートしてくたさい

- ブラウザーが後者を使用する場合、オペレーティングシステムの信頼されている証明書ストア。Windows 上では、例えば Google Chrome とMicrosoft Edge はオペレーティングシステムの証明書ストアを使用しますが、Mozilla Firefox は自身 のストアを使用します。Linux 上では、Google Chrome とMozilla Firefox は自身の証明書ストアを使用します(次のア イテムを参照してくたさい)。Mac 上では、オペレーティングシステムの証明書ストア(Keychain Access)を使用します。
- ブラウザー自身の信頼されている証明書ストア。

メモ

- 各クライア・トマンン、おけ、ブラウザー)のためにFlowForce Server にアクセスするこのステップを実行する必要かあます。
- FlowForce W eb Server と FlowForce Server の間で、SSL 暗号化を有効化すると、ブラウザーに証明書をインポートするだけでは十分ではかません。自身で署名済みのルート CA 証明書はオペレーティングシステムによれ信頼されている必要があります。

Linux

Linux 上では信頼される証明書をシステムの証明書ストアニ以下のようこへんポートすることができます。

サーバーの証明書の信頼性が確実の場合、次のステップをおこなってくたさい。

Debian とUbuntu では、以下のステップに従います:

1. Web サーバーの証明書ファイルを次のディノケリにコピーします。

sudo cp /home/downloads/server_cert.crt /usr/local/share/ca-certificates/

2. 証明書を以下のように更新します

sudo update-ca-certificates

CentOS では、以下のステップに従います:

1. ca-certificates パッケージをヘノストールします:

yum install ca-certificates

2. 動的な証明機関構成機能を有効化します:

update-ca-trust enable

3. サーバーの証明書を次のデルノトリニュピーします:

cp server_cert.crt /etc/pki/ca-trust/source/anchors/

4. コマドを使用します:

update-ca-trust extract

ブラウザーからのみサーバーにアクセスする必要かある場合、各ブラウザーストアに証明書をインポートするのみで十分です。各ブラウザーにより 正確な手続きは異なります。例えば、Firefox 59.0.2 では、以下を行うことができま

- 1. 「オプション | プライバシーとセキュリティ」から、「証明書を表示」をクリックします。
- 2. 「認証局証明書」ダブで、「インポート」をクリックし、以前に作成されているルート証明書ファイルを参照します。
- 3. プロンプトされるとこの証明書をウェブサイトの識別に使用する」を選択します。

Downloading Certificate			
You have been asked to trust a new Certificate Authority (CA).			
Do you want to trust "ODIN CA" for the following purposes?			
Trust this CA to identify websites.			
Trust this CA to identify email users.			
□ Trust this CA to identify software developers.			
Before trusting this CA for any purpose, you should examine its certificate and its policy and procedures (if available).			
View Examine CA certificate			
Cancel OK			

Mac

macOS では、信頼される証明書をKeychain Access に以下のようにんポートすることができます。

- 1. Keychain Access を実行します。
- 2. 「システム」をクリックして、「証明書」をクリックします。
- 3. 「ファイル」メニューからアイテムのインポート」をクトックします。
- 4. 信用される証明書を参照し、「開く」をクルクします。
- 5. Keychain Access パワードをプロンプトされると入力し、「キーチェーンの変更」をクリックします。
- 6. 証明書をダブルクトックし、「信頼」セクションを拡張し、「常に信頼する」を選択します。

APOLLO CA			
Certificate Root certificate authority Expires: Thursday, 15. June 2028 at 09:53:18 Central European Summer Time This certificate has custom trust settings			
▼ Trust			
When using this certificate: Use Custom Settings ᅌ ?			
Secu	re Sockets Layer (SSL)	Always Trust	
Secure Mail (S/MIME)		Never Irust	
Extensible Authentication (EAP)		Never Trust	
	IP Security (IPsec)	Never Trust	
	Code Signing	Never Trust	
	Time Stamping	Never Trust	
	X.509 Basic Policy	Always Trust	
▶ Details			

Windows

Windows では、信頼された証明書をシステム証明書ストアコ以下のようこインポートすることができます:

1. Windows 証明書ストアをコンピューターアカウナのために開きます、Windows 証明書ストアへのアクセスを参照してくたさい。

Web サーバー証明書の認証を確認するおけの場合、次のステップを実行します。

2. 「信頼されナーレト証明書機関」で「証明書」を右クルクレ、「すべてのタスク | インポート」を選択し、証明書のインポート ウィザードに従ってくたさい。

🚟 ファイル(F) 操	作(A) 表示(V) お気に入り(O)	ウィンドウ(W) ヘルプ(H)		
🗢 🄿 🔁 🗖	🗊 🏾 📾 🖬			
וע-ארב 🚞		発行先 ^	発行者	有効期限
→ 🚽 証明書 (ロ-	カル コンビューター)	AddTrust External CA Root	AddTrust External CA Root	2020/05/30
> 🧾 個人		🔄 Altova GmbH Enterprise CA	Altova GmbH Enterprise CA	2011/01/19
✓ ■ 信頼され	たルート証明機関	Altova GmbH Enterprise CA	Altova GmbH Enterprise CA	2013/01/10
		· · · · · · · · · · · · · · · · · · ·	Altova GmbH Enterprise CA	2014/12/19
> <u>1</u> 29	/う 9へ(のタスク(K)	> 1 2/1/= P(1)	Altova GmbH Enterprise CA	2021/02/02
> 🔜 中間部	⁹ 表示(V)	> ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2009/01/24
> 🔜 1日根さり	パーラー ここから新しいウインドウ(W)	ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2003/04/05
> 🔜 1言根され	, (223, 54, 100, 7171, 7(0)	ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2021/09/24
> ソードハ	新しいタスクパッド表示(T)	ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2021/09/30
> 15/尺で1		ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2021/09/24
> 📫 ブルドュー		ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2011/01/19
> 10 ほかの人	- 一覧のエクスホート(L)	ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2013/01/10
s 📫 eSIM G	rt ヘルプ(H)	ova GmbH Enterprise CA	Altova GmbH Enterprise CA	2014/12/19
Homed	roup Machine Certificates		Altova GmbH Enterprise CA	2017/12/15
> 📔 Local N	onRemovable Certificates	Altova GmbH Enterprise CA	Altova GmbH Enterprise CA	2007/00/13
MSIEHi	story Journal	Altere Crebil Enterprise CA	Alteria Cashili Enterprise CA	2005/04/15
> □ INSICHISTORYSOUTHAI ■ II∓−ト デスクトップ		Altere Cashi Esternice CA	Altova Grobili Enterprise CA	2007/01/24
> 🎬 証明書(0登録要求		Altova GmbH Enterprise CA	2007/08/15
> <u></u> ZZ-h	hードの信頼されたルート	Altova GmbH Enterprise CA	Altova GmbH Enterprise CA	2021/02/02
		🟹 Altova GmbH Enterprise CA	Altova GmbH Enterprise CA	2021/09/23

詳細に関しては<u>https://technet.microsoft.com/en-us/library/cc754489(v=ws.11).aspx</u>を参照してたきい。

3.3.2.2 Windows 証明書ストアにアクセスする方法

W indows 上では Microsoft 管理コンノール(MMC) スナップイン、おけは Internet Explorer から証明書を管理することができます。

(現在 Windows を使用するユーザーのための)証明書 スナップインを開く方法:

• certmgr.msc をコマイラインで実行します。

(コンピューターのアカウントのため)ご証明書スナップインを開く方法:

- 1. mmc をコマイラインで実行します。
- 2. 「ファイル」メニューから、「スナップインの追加/削除」をクトックします。
- 3. 「証明書」をクリックして、「追加」をクリックします。


4. 「コンピュータのアカウナト」をクリックして、「次へ」をクリックします。

証明書スナップイン		×
このスナップインで管理する証明書:		
〇 ユーザー アカウント(M)		
○ サービス アカウント(S)		
● コンピューター アカウント(C)		
	. = 3 (0)	
	< 戻 つ(B) 次へ(N)	> キャンセル

5. 「ローカルコンピューター」をクリックして、「完了」をクリックします。

コンピューターの選択	×
このスナップインで管理するコンビューターを選択してください。	
このスナップインで管理するコンピューター:	
● ローカル コンピューター(L): (このコンソールを実行しているコンピューター)	
○別のコンピューター(A):	参照(R)
□ コマンドラインから起動したときは選択されたコンピューターを変更できるようにする(W) これは、コンソールを保存した場合にのみ適用されます。	
< 戻る(B) 完了	キャンセル

3.3.3 秘密キーの必要条件

FlowForce Server は無人で作動するため、enabling SSL の有効化は証明書の秘密キーが暗号化なしてあることを必要とします。 すなわち、パマワードで保護されていてはなりません。それ以外の場合、FlowForce Server によし使用されることはできません。この理由 のために、秘密キーを保管するファイルは制限されたアクセスを持ち、期間内で権限を持つ者のみかアクセス可能である必要があります。

秘密キーが、アワードにて保護されている、ませ」お暗号化なしてあるかなとを確認するこは、テキストエディターませ」はコマンドランで秘密キー ファイルを開いて確認します。暗号化された秘密キーイは以下のラインで開始します。

-----BEGIN RSA PRIVATE KEY-----Proc-Type: 4,ENCRYPTED DEK-Info: AES-256-CBC,DFC3FAD546517ED6336CFF72AA23F6C7

秘密キーの暗号化を解除する方法の一つはOpenSSLツールキナの使用です(<u>https://www.openssl.org/</u>)。具体的には、以下のコマイを実行して秘密キーの暗号化を解除することができます。

openssl rsa -in enc.key -out dec.key

```
enc.key が暗号化されているキーで dec.key がコマンドにお出力される暗号化されていないキーである個所。例えばffenc.key は暗
号化されているキーで、秘密キーが保管されているディレクトリ内の以下のコマンドを実行して暗号化されていないキーを取得することができます。
```

openssl rsa -in ffenc.key -out ffdec.key

コマドの実行後 ffdec.key ファイルはENCRYPTED とは表記しなないます。

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAzCCedru/oKzaSiwh6avtf9eMPix99RKpd07fWtwstkuglAdi
--
--
------END RSA PRIVATE KEY-----
```

3.3.4 FlowForce Web サーバーのために SSL を有効化する方法

下記に、FlowForceのWeb管理インターフェイスを作動する「FlowForceWebServer」サービスのためにSSLを有効化する方法について説明されています。

必要条件

- (DigiCert、Comodo など) ブラウザーには信頼されている証明書機関には、屠名された秘密キーと対応する証明書が必要です。証明書機関には、提供されるすべての中間証明書が必要になります。取得方法に関しては、証明書機関によりSSL 証明書が必要になります。取得方法に関しては、証明書機関によりSSL 証明書の作成方法を参照してくたさい。
- 証明書ファイルと秘密キーはPEM (プライバシー強化メール)フォーマトである必要があります。PEM ファイルのファイル拡張子 は通常.pem ですが.key.cert.cer、おは.crtであることもできます。
- 証明書はFlowForce Server が作動するドメイン名のために発行されます。
- 証明書の秘密キーは、アワードで暗号化されてはなりません。秘密キーの必要条件を参照してくたさい。
- 自己署名済み証明書を作成した場合、各クライア・トブラウザーは自己署名済みの証明書の機関を信頼するように構成されて いる必要あります。

前提条件が満たされると、クライア・トマシンとFlowForce Web Serve 間の接続を次のように確保することができます:

- 1. FlowForce Server セオアップページを開きます。
- 2. 「FlowForce Web Server」の下でグループ化されている設定を検索し、次を行います:
 - a. Enabled有効化済みチェックボックスを「SSL暗号化された接続」の下で選択します。
 - b. 「バインドアドレス」の横で、すべてのインターフェイス(0.0.0.0)を選択します。(現在のマシンからローカルにアクセス可能 な打きておよくFlowForce Web Server が外部からアクセス可能なことを想定としています)。
 - c. ホスト名とポートフィールド内でFlowForce Web Server がSSL 暗号化済み接続のためコルスンする箇所でホスト(ドメイン)名とポートを入力します。

ホスト名 フィールド内に入力されているドメイン名はSSL 証明書の共通名に対応している必要が決ます。ポートは使用されていてはなりません。

場合によっては、「その他」を選択し、FlowForce Server がSSL 暗号化された接続をリッスンするアドレスを入力します。ホスト名を入力せずゴ 他の」フィーリド内にIP アドレスを入力すると、このIP アドレスはSSL 証明書の共通名に対応する必要があります。

d. 証明書と秘密キーファイルを対応するテキストボックスに入力します。

ょ

- 証明書はPEM (プライバシー強化メール)フォーマトである必要がかます。PEM ファイルのファイル拡張子は 通常.pem ですが.key、.cert、.cer、おは.crt であることもできます。
- 秘密キーはPEM (プライバシー強化メール)フォーマナである必要があります。PEM ファイルのファイル拡張子は 通常.pem ですが.key.cert.cer、おけは.crt であることもできます。
- FlowForce内で秘密キーを使用可能にするコよノアワードで保護されている必要があります。

 <u>秘密キーの必</u>
 <u>要条件</u>を参照してくたさい。
- 秘密キーは安全に保存されている必要があります。
- e. 証明書チェーンファイルフィールド内の証明書チェーンファイルへの やを入力します。これは中間証明書ではよく、このフィー ルドを空白のままこすることができます。しくつかの中間ファイルを使用できる場合、これられたから全ての中間ファイルを含む [チェーンファイル]に結合する必要があります。中間ファイルの準備を参照してくたさい。
- f. 任意で、「暗号化されていない接続」の下の効化済みチェックボックスをクリアします。この操作によりFlowForce Server がプレーンな HTTP によし使用できなくなるため、SSL 暗号化されている接続が作動することを確認した後にのみこのステッ プを実行してくたさい、HTTP 接続を完全に無効化する代わりに、下のイメージで表示されているように、ローカルの接続を制限してくたさい。

F	lowForce Web nencrypted Conn	ection		
	Enabled: Bind address: Host name:	 ✓ ✓ Local only (127.0.0.1) ▼ ○ other: 	Port:	8082
S	SL Encrypted Con	nection		
	Enabled:			
	Bind address:	● All interfaces (0.0.0.0) ▼ ○ other:	Port:	8083
	Host name:	server.example.org		
	Certificate file:	C:\secure\flowforceweb.crt		
	Private Key file:	C:\secure\flowforceweb.key		
	Certificate Chain file:	C:\secure\intermediate.pem		

3. 設定を適用し Flow Force サービスを再起動するをクルクします。

SSL 有効化 チェックボックスを選択し 設定を適用し Flow Force サービスを再起動するを選択すると、ブラウザーは ("http "ではない" https" URL に再ダイレクトされます。

以下の点に注意してくたさい

- ブラウザー(まけよ 接続するケライアント) はSSL 証明書の共通名 (CN) がFlowForce Server か実行されるドメイン名 とIP アドレスに一致しない場合、警告を表示します。
- 自己署名済み証明書を使用する際、オペレーティングシステムの証明書ストア、おゴよ、ブラウザーの証明書ストアにCAルト証明書を追加していない場合、ブラウザー(おゴよ、接続するクライアント)は警告を表示します(次を参照:<u>ルト証明書のへポ</u> <u>ト</u>)。

3.3.5 FlowForce サーバーのために SSL を有効化する方法

このトピックでは、FlowForce ジョブからHTTP(S) クライアントに作成されたサービスを公開するサービスである「FlowForce Server」 サービスのためこSSLを有効化することここして説明されています。ブラウザーとFlowForce Web 管理インターフェイス間のSSLを有 効化する場合、FlowForce Web Server のためこSSLを有効化するを参照してくたさい。

必要条件

- (DigiCert、Comodo など) ブラウザーには信頼されている証明書機関には、署名された秘密キーと対応する証明書が必要です。証明書機関には、提供されるすべての中間証明書が必要になります。取得方法に関しては、証明書機関によりSSL 証明 書を署名する方法 おけよ自己署名済みのSSL 証明書の作成方法を参照してくたさい。
- 証明書ファイルと秘密キーはPEM (プライバシー強化メール)フォーマトである必要かあります。PEM ファイルのファイル拡張子 は通常.pem ですが.key.cert.cer、おけは.crtであることもできます。
- 証明書はFlowForce Server が作動するドメイン名のために発行されます。
- 証明書の秘密キーは マワードで暗号化されてはなりません。<u>秘密キーの必要条件</u>を参照してくたさい。
- 自己署名済み証明書を作成した場合、各ケライアントプラウザーは自己署名済みの証明書の機関を信頼するように構成されて いる必要あります。

前提条件が満たされると、クライア・トマシンとFlowForce Server 間の接続を次のように確保することができます。

- 1. FlowForce Server セトアップペーンを開きます。
- 2. 「FlowForce Server」の下でグループ化されている設定を検索し、次を行います:
 - a. Enabled 有効化済み チェックボックスを「SSL 暗号化された接続」の下で選択します。
 - b. 「バインドブドレス」の横で、すべてのインターフェイス(0.0.0.0) を選択します。(現在のマシンからローカルにアクセス可能 なだうけではなく FlowForce Server か外部からアクセス可能なことを想定としています)。
 - c. ホスト名とポートフィールド内でFlowForce Web Server がSSL 暗号化済み接続のためコルスンする箇所でホスト(ドメイン)名とポートを入力します。

ホスト名フィールド内に入力されているドメイン名はSSL証明書の共通名に対応している必要が設ます。ポートは使用されていてはないません。

場合によっては、「その他」を選択し、FlowForce Server がSSL 暗号化された接続をリッスンするアドレスを入力します。ホスト名を入力せずゴ 他の」フィールド内に IP アドレスを入力すると、この IP アドレスは SSL 証明書の共通名に対応する必要があります。

d. 証明書と秘密キーファイルを対応するテキストボックスに入力します。

メモ

- 証明書はPEM (プライバシー強化メール)フォーマトである必要があります。PEM ファイルのファイル拡張子は 通常.pem ですが.key.cert.cer、おさは.crtであることもできます。
 - 秘密キーはPEM (プライバシー強化メール)フォーマナである必要があります。PEM ファイルのファイル拡張子は 通常.pem ですが.key.cert.cer、お」は.crtであることもできます。
 - FlowForce内で秘密キーを使用可能にするコよ、パワードで保護されている必要があります。

 <u>秘密キーの必</u>
 <u>要条件</u>を参照してくたさい。
 - 秘密キーは安全に保存されている必要があります。
- e. 証明書チェーンファイルフィールド内の証明書チェーンファイルへの 父を入力します。これは中間証明書ではなく、このフィー ルドを空白のままこすることができます。 しくつかの中間ファイルを使用できる場合、これらをしかゆる全ての中間ファイルを含む [チェーンファイル]に結合する必要があります。 <u>中間ファイルの準備</u>を参照してください。

f. 任意で、「暗号化されていない接続」の下の効化済み チェックボックスをクリアします。 プレーン HTTP を介して FlowForce Server を使用できなくなります。

FlowForce Server 暗号化されていない接続	壳
有効化済み: バインドアドレス: ホスト名:	 □ーカルのみ (127.0.0.1) ○ その他: ポート: 4646
SSL 暗号化された接続	
 有効化済み: パインドアドレス: ホスト名: 証明書ファイル: 秘密キーファイル: 証明書チェーンファイル: 	 ○ □ーカルのみ (127.0.0.1) ○ その他: ポート: 4647 example.name.com C:\secure\flowforce.crt C:\secure\flowforce.key C:\secure\intermediate.pem

3. 設定を適用し Flow Force サービスを再起動するをクルクします。

以下の点に注意してくたさい

- ブラウザー(おけよ 接続するクライアント)はSSL 証明書の共通名 (CN) がFlowForce Server か実行されるドメイン名 とIP アドレスに一致しない場合、警告を表示します。
- 自己署名済み証明書を使用する際、オペレーティングシステムの証明書ストア、おさよ、ブラウザーの証明書ストアにCAルート証明書を追加していない場合、ブラウザー(おさよ、接続するクライアント)は警告を表示します(次を参照 <u>ルート証明書のインポ</u> <u>ー</u>)。

3.3.6 FlowForce Web Server と FlowForce サーバー間に SSL を有効化する方法

FlowForce W eb Server とFlowForce Server の.ini ファイルをそれそれ編集し FlowForce W eb Server とFlowForce Server 間のSSL を有効化することができます。FlowForce Server アプリケーションデータ ディレオ・リのデータ サブディレオ・リ内で両方のファイルを見つけることができます。

flowforce.ini ファイル内で次を行います:

1. [ListenSSL] セクション内で、次の デメーターのための値を入力します: ssl、active、host、port、例:

```
[ListenSSL]
ssl=1
active=1
host=server.my.domain.com
port=4647
```

• ssl とactive / デメーターは1(有効化済み)に設定されている必要があります

- host はFlowForce Server は使用されるSSL 証明書の共通名に一致するサーチー名(おけよ 適用できる場合は IP アドレス)に対応する必要があります。
- port は暗号化されていない 接続のために使用されるデフォルトの4646 ポート以外のポートである必要があります。 例えば、使用されていない 場合、4647 に設定することができます。
- 2. [SSL] セグションでは、FlowForce Server のために使用することのできる証明書と秘密キーへの やを入力します。 <u>FlowForce Server のよめにSSL を有効化する方法</u>内で記載されている証明書と秘密キーのペアと同じです。例:

```
[SSL]
certificate_chain=
certificate=C:\secure\flowforce.crt
private key=C:\secure\flowforce.key
```

flowforceweb.ini ファイル内で次を行います:

1. 既存ではは、場合、[FlowForce] と呼ばれるセクションを追加し、次の3つの、ラメーターのために値を入力します: ssl、host、 およびport,例:

```
[FlowForce]
ssl=1
host=server.my.domain.com
port=4647
```

- SSL 接続を有効化するコよ ssl パラメーターを有効化(1 に設定)する必要がおよす。
- host はflowforce.ini ファイルの [ListenSSL] セクション内の一つと同じである必要があります。
- host は FlowForce Server のために使用することのできる SSL 証明書の共通名に一致するサーバー名 (おけよ 適用 てきる場合は IP アドレス) に対応する必要がおます。
- port は暗号化されていない、接続のために使用されているデンオルトの4646 ポートとは異なる必要があります。通常この値は、4647 です。

メモ .ini ファイルの編集が完了すると、FlowForce Server とFlowForce W eb Server サービスの両方を再起動します。詳細 に関しては、次を参照してくたさい

- サービスの開始と停止(Linux)
- ・ サービスの開始と停止 (macOS)
- <u>サービスの開始と停止 (Windows)</u>

3.4 デフォルトのタイムゾーンの設定

時間をベースリユナム・リガーを使用するショブを作成すると、適用されるタイムゾーンを指定する必要があります。ショブの構成ページのデフォルトの設定が選択されていると、タイムゾーンをグロー・シルニ構成することができます。

デフォルト のタイムゾーンの設定:

- 1. FlowForce Web 管理インターフェイスからログオンします。
- 2. 「管理」をクリックします。
- 3. 「設定」をクリックします。
- 4. 「入力フォーマット」から、デフォルトのタイムゾーンを選択してくたさい。
- 5. 「保存」をクルクします。

3.5 電子メールのパラメーターの設定

電子メールを送信するジョブを作成する場合、メールサー、一のSMTP プドレスとポート、およびSMTP 資格情報を構成する必要かあります。

FlowForce は、TLS おけよSSL を介して、暗号化された接続を構築しようと試みます。暗号化された接続に失敗すると、FlowForce は、暗号化と無して接続を開始します。そして、SMTP サーバーが明示的に要求する場合、接続を暗号化する可能性があります。それ以外の場合、SMTP 接続は、テキスト書式のまま保持されます。

電子メール設定の変更:

- 1. FlowForce Web 管理インターフェイスからログオノします。
- 2. 「管理」をクルクします。
- 3. 「設定」をクルクします。
- 4. 「SMTP サー・・・」と「SMTP ポート」に、電子メールサー・・の名前とポートをそれそれ入力します。標準のSMTP サー・・ ーはポート 25 で接続を受け入れます。 SSL/TLS プロトコールで暗号化される接続を必要とする SMTP サー・・・ーは、通常 465 まけは 587 である、他のポートからの接続を受け入れます。
- 5. SMTP サーバーが認証を必要とする場合、「ユーザー認証」の横のプラスのアイコンをクトックし、ユーザー名と マワードを入力します。
- 6. 任意で、RFC2822-compliant メールボックスアドレスの植を「デフォルトの送信元」フィールドに入力します。ここで入力され るファイルは ①/system/mail/send と①/system/mail/send-mime 関数の ラメーターのデフォルト「差出 人」として使用されます。

3.6 ディレクトリサービスの設定

所属機関がMicrosoftActive Directory おけよApache Active Directory、OpenLDAP Server、Oracle Unified Directory などのLDAP-準拠デルノオ・リサービスプロ・イダーを使用する場合、FlowForce Server ど統合することができます。 FlowForce Server パースペクティブからのDirectory Service プロ・イダーとの統合は以下を意味します:

- ユーザーは、ドメインユーザー名となワードを使用して、FlowForce Server にエクオンすることができます。
- 管理者は、既存のドメインユーザーボンイン資格情報(つま)、明示的なユーザーインポート)を使用して、FlowForce Server へのログオンを許可、おけは、ドメインユーザーとグループをFlowForce Server に明示的にインポートすること許可 することができます(以下を参照: ドメインユーザーとクループを FlowForce Server のユーザーアカウト 管理ページで表示されています。これにより、管理者は、ビルトイン FlowForce Server アカウトと同様に、グループ おけはユーザーアカウト に特権と パージッションを追加、おけは、削除することができます (以下を参照: 特権の作動方法、とパージッションのしくみ)。管理者は、およ、FlowForce Server ロールを グループ おけは ユーザーアカウト割り当てることができます(以下を参照: ユーザーロールを割り当てる)。
- 管理者は、FlowForce Server にんポートされナジメインユーザーの名前を変更、おけお タワードを変更することはできません。
- 管理者は、FlowForce Server にんポートされオジメイングループの名前を変更、オイコメンバーシップを変更することはできません。
- 管理者 FlowForce Server からんポートされナジメインアカウトを削除することができます。これは、¥ドメインからアカウトを削除することは無く、関連した特権を変更することはかりません。
- インポートされナボメインアカナナトが、割り当てられたFlowForce Server 特権というシシシンを保有する場合、これらは、特権 レポートに表示されます(以下を参照: 特権レポートの閲覧)。

ディレクトリサービスの設定を変更する方法

- 1. FlowForce Web 管理インターフェイスからログオノします。
- 2. 「管理」をクトックし、「設定」をクトックします。

5	- ィレクト	リサービス			
	☑ 有効化				
	次を使用して 接続:	○ Active Directory			
		◉ ライトウェイト ディレクトリ アクセス プロ	ホスト:	10.100. 389	
		トコル (LDAP)		ホスト名、ドメイン、または、 /P アドレス。 カスタムポートを以下で指定	することができます \:389'
			ユーザー:	cn=name	
				'name@domain.com' または 'cn=name,dc=domain,dc=com' の書式のユーザ-	-名
			パスワー ド:	ここに入力してパスワードをセッ	
			SSL:		
				SSL が作動するには、 LDAP サーバーの CA 証明書が FlowForce サーバー が 必要があります	存在するマシン上にインストールされている
		ーザーにログインを許可			

使用することのできる設定は、以下のとおりです

有効化

FlowForce Server はメインユーザー名と、なワードを使用してユーザーがログオンできるようにこのチェックボックスを選択します。このチェックボックスをチェックすると、以下に説明されているとおり、 Active Directory おはライトウェイト ディレクトリアクセス プロトコル (LDAP) オプションを選択する必要があります。 ライトウェイト ディレクトリアクセスプロトコル(LDAP)オブションを選択した場合、(ユーザー名、ハウワードなどの) 接続の詳細が正確 であることを確認してくたさい。「保存」 をクルクすると、FlowForce は指定された LDAP サーバーと通信を試み、接続の詳細が有効で ない場合、エラーカ表示されます。 FlowForce Server のLDAP サーバーへの接続が成功しない限り、 LDAP 設定を保存できない事 に注意してくたさい。

Active Directory オプションを選択した場合、FlowForce Server が作動するマンノは Active Directory には管理されているド メインの一部である必要があります。

ディレクトリリービスの認証を有効化後、FlowForce Server ログインページで「ログイン」と呼ばれる追加ドロップダウノリストか表示される ようになります。「ログイン」ドロップダウノリストによりユーザーが、次のアイテムを含む認証オプションを選択することができます。

- 直接 これはデフォルトのFlowForce Server 認証オプノョンです。
- [特定のWindows ドメイン]. FlowForce Server が作動するマンメンマれて適用できます。

次も参照してくたさい FlowForce Server へのログオン。

接続

Active Directory を選択して、Active Directory との直接の統合を有効化します。これはFlowForce Server がWindows 上で作動中にのみ適用することができます。

ライトウェイト ディレクトリアクセスプロトコル(LDAP)を選択して、LDAP に準拠するディレクトリサービスとの統合を有効化します。 詳細を以下の通り入力します:

- ホスト LDAP サーバーのホスト名、ドメイン名、おけはIP アドレスを入力します。ポート番号を追加するけは、ポート番号の 後にコロン文字を追加します。例えば、somehost:10389
- ユーザー デル/トリサービスをクロける管理者の権利を持つユーザー名を入力します。ユーザー名は「Distinguished-Name」(例えば、cn=name, dc=domain, dc=com)の書式、おけよActive Directoryのみに適用される「ユーザープリンシットル名」(例えば、user@some.domain.com)書式を使用することができます。メモ 他のLDAP サービスに関しては「識別名」書式を使用します。
- パマワード ユーザーのマワード。メモノマワードの入力を複数回間違えると、LDAP サーバーはアカウトをロックする場合 からります。この場合、継続する前にアカウトがロックされていないことを確認してくたさい。
- SSLの使用LDAPサイトかライアトからSSL暗号化済みの接続を受け入れるようご構成されている場合のみこのチェックボックズを選択してくたさい。このオプランを選択する場合、安全な接続のためにLDAPサイドーこと使用されているポート番号(通常ポート 636) に変更してくたさい。所属機関が両方のマシン上で信頼されるルレート証明書を既に使用する場合、追加の構成手順は存在しません。それ以外の場合、LDAPサイドのルート(CA)証明書は、FlowForce Server が作動するマンノニインストールされている必要があります。
 - a. LDAP サーバーが存在するマシン上で、信頼されてる証明書ストアからルート証明書をエクスポートします。使用中のオペレ ーティングシステムニ固有のソールを使用します(例えば、Windows上では<u>証明書スナップイン</u>を使用します).
 - b. FlowForce Server か存在するマシン上では、<u>ルト証明書のインポト</u>で説明されている通り、信頼されている証明書ストアから証明書をインポートします。

ー部の場合、LDAPサイドーは特定の標準に一致しない任意のスキーマが存在します。FlowForce Server が使用中のLDAPプロ バイダーのスキーマを検知できない場合、「ディレオトリサービスは無効な LDAP スキーマを検知しました。」に類似したエラーが表示されま す。この場合、directoryservice.cfg ファイルを FlowForce Server 実行可能ファイルと同じディレオトリニニピーします。このファイル が存在すると、FlowForce Server はプロレバダーを自動的に検知しようとしません。

全てのドメインユーザーコングインを許可

ドメイン資格情報を使用してユーザーがFlowForceに最初にエグオンするとユーザーのドメインアカウトがFlowForceユーザーデータベ ースにインポートされる場合、このチェックボックスを選択します。

このオプションが無効化されていると、管理者によりドメインアカウストが既にFlowForce Server にインポートされている場合、ドメインユーザーはFlowForce Server にエグオンすることができます。次も参照してください、ドメインとユーザートロールのインポート。

デフォルトのログインドメイン

有効化 チェックボックスが選択され設定が保存された後にこのオプションは表示されます。

ドロップダウノストは、マンカジンバーのすべてのドメインを表示しています。ディレオトリサービス認証が有効化されていると(上記の最初のオ プロンを確認してください) FlowForce ログインページ内のユーザーがドメインのリストを確認することができます。

ドメインがFlowForce Server 認証ページ内のログインドロップダウノルト内のデフォルトの選択として選択される場合、「ドメインのログインをデフォルトとする」チェックボックスを選択してくたさい。

「ドメインのログインをデフォルトとする」チェックボックスをクルアすると、ビルトイン Flow Force Server 認証(「Directly」) かデフォルトの 選択です。

3.7 ログ設定

FlowForce Server は(ジョブ出カイベト、変更イベトの構成、エラーなどの多種のイベトと発生の時刻を登録するログインメカニズ ムを提供します。専用のページから ログイベトを確認することができます。ジョブログの確認 を参照してくたさい。ログイベトは FlowForce Server 内部データベースのサイズを大幅に増量することができることにきることできるしてくたさい。このためにarchive-log または truncate-log /system/maintenance 関数を使用して、ログは定期的に達成され、クレーンアップされる必要かあります。ログを妥当 な制限内のサイズコ収める手助けをする他の設定を使用することができ、このページの下の部分で説明されています。

FlowForce Server 内でのログロはつの種類が存在します

- 1. 手動の介入を全く必要としないデフォルトのシステムログ。この種類のログはすべてのイベトを登録しますが、更に良い状態のシステムの安定度と、フォーマンスのために各ログレコードを一定の尽小まで保持します。システムログが十分な詳細のレベルを提供しない場合、おさは、(ステップ内の、ラメーター値などの)ログエントリが切り捨てられている場合、明示的なログを下記の通り選択することができます。
- 2. ジョブ構成ページから強制することのできるオプションの(明示的な)ログ。ログ構成ページは、ログ内でトラックする各パテメータの横

にあり、オプションで有効化することのできるログ 2 ボタンを提供します。これを行うことで、ジョブが実行されると、その、ラメーターのフル値がログされます。更に、その関数を明示的にコングするオーカに、log 式関数内で Flow Force 関数を埋め込むことができます。 おっ、これは式を切り捨てることなくフルコングします。Flow Force Server は明示的なログの結果としてログされるエトリのサイズを制限しません。

ログの制限

デフォルトのログサイズを高い値に変更すると、システムの安定度と、プォーマンスに影響を及ぼす可能性があります。

デフォルトのログサイズを高い値に変更すると、システムの安定度とり、フォーマンスに影響を及ぼす可能性があります。ですから、このオプションの使用にはお主意してくたさい。明示的なログを使用するオーダの奨励されるアプローチは上記の通りです。

ログエトリのデフォルトのサイズをビューおけま変更する方法

- 1. FlowForce Web 管理イクターフェイスからログオします。
- 2. 「管理者 | 設定」に移動して、「ログの、ミナ」の下でグループ分けされている デメーターを監視します。

Lo	ogging limi	ts		
	String type limits:			
		Default string value logging limit:	50	characters
	List type limits:			
		Default list value logging limit:	10	items
		Recurse into sublists:	2	levels

ログ済みのエントリニコ 2つの種類が存在することご注意してくたさい文字列の型とリストの型。結果、各型のサイズを管理するために2つのパラメーターが存在します。

デフォルトの文字列の値ログミット	型「文字列」であるログエトリのデフォルトの長さを指定します。ログエトリかこの値を超すと、 ファイルレ ななどの長い任意の値が切り捨てられます。

デフォルトのルトの値ログミット	上記同様、型「リスト」であるログエトリに適用されます。
サブリストに再帰する	この設定は地のノストを子として含むノストを操作するジョブニ影響を与えます。この値を設定して、FlowForce にNレベリの深さをログインのナムンニ命令します。

インスタンスのログ

インスタンスログのセグションは特に<u>インスタンスログ</u>ページ内でしポートされた情報のレベルよ影響を与えます。

ログされナメッセージコお以下の順序の需要度が存在します(低度から高度へ):情報、警告、エラー「インスタンスのログ」の、ラメーターは重要度に従い特定のメッセージのログをスキップすることを不可能してます。FlowForce Server により保管されるトレース情報の量を構成、ませっちログの保管を無効化することができます。下のイメージはデフォルトの設定を表示しています。

Instance logging		
Default:		
Retain log:		
	Messages severity:	information 🗸
	Execution trace:	streams 🗸
	I	

ログ保持 チェックボックスをクリアすると ヘノスタレスログ ページで情がしポートされません。

メッセージの重要度 オプノョンは保持されるメッセージを指定します:

無し	メッセージは保持されません
<i>I</i> 7 -	エラーと重要なメッセージを保持
警告	エラー、重要なメッセージ、および警告を保持
情報	エラー、重要なメッセージ、警告、情報メッセージを保持
すべて	これは最も詳細なオプションです。重要度に関わらず、全てのメッセージが保管されます。

トレースの実行、ウメーターは保管されるトレーシングの詳細の量を指定します

無し	トレーシングの情報は保管されません
ズリーム	ストリームを保持し、トレースを除外します
FL	トレーズを保持し、ストリームを除外します
	トレース情報の可能なすべてのレベルを保持します

ルール

上記で説明されているインスタンスログ」の設定はエルールを構成します。 🎦 ボタンをクトックしてデフォハト につれえカスタムルールを作成することができます。 以下のようなジョブの結果をベース してこれによリルールを条件づけて適用することが可能にないます:

実行の成功	ジョブの結果は成功としてみなされます。
実行の失敗	泊辺実行に失敗しまた。
	ユーザーアグライにより泊ブは停止されまた。 ジョブの停止を参照してくたさい。
中断済み	ジョブの完了前にサービスが中断されまた。おけまFlowForce Server クラッシュしまた。, おけまフーカーインスタンスへの接続が、クラスター済み、セナアップ内で失われまた。

このページで定義済みのルールは上から下へ評価されます。ジョブの結果が上記の結果のどれかに一致する場合、ルールは一致します。最初に一致するルールは優先されます。

例えば、上記の構成はジョブの実行に成功しない場合フレダッセージログを保持します。すなオオ、結果が「実行の失敗」、「ユーザーによ将 止」、および「中断済み」の場合最初のルールカトリガーされます。実行に成功した場合、デフォルトのルールが代わりにトリガーされ、ログメッセ ージは保持されますが、トレーシング情報を使用することしてきません。

Instance logging	
Enable	1 4 🕤 🖬
When: Then:	
Successful execution	
Failed execution Messages severity: all)
Stopped by user Execution trace: full	
☐ Interrupted	
Default:	
Retain log:	
Messages severity: information 🗸	
Execution trace: none 🗸	

カスタムリレールをデフォリトのルールの後ではよく、前のみに追加することができます。ルールの順序を変更するには上

と下

・
ボタ

ンを使用してくたさい。これらのボタンはつまたはそれ以上のルールが存在する場合のみ有効化されます。

以前のルールが一致しなく、カスタムルールを定義する場合、全てをキャッチするフィルターとしてデフォルトのルールを使用することが奨励されます。

プロジェクトレベルでのログルール

アプリケーションノベルでグロー・シレビナではなく特定のFlowForce Server ジョブのナムリエレグルールを作成することができます。サブショブを 持つショブのナムリコレールを作成すると、ルールはすべてのサブショブにも適用されます。 ジョブ構成ページを開く方法:

- 1. ジョブ構成ページが開かれます。
- 2. ログ設定 ボタンをジョブ構成ページ内でクトックします。
- 3. ログルールの新規インスタンスをクトックします。

複数のジョブのためにログルールを設定する方法:

- 1. 構成をクルクレコンテナーを開きます。
- 2. 1つまけは複数のショブ(まけはエンテナー全体)を選択し選択されたショブのためのログ設定をクリックします。必要であれば選択 を詳細設定することのできるダイアログボックスが表示されます。

Change options for selected objects	×
Name	
CompletePO.job	
CopyFiles	
CopyOutput	
CheckPath	
	_
Change Logging Settings Cancel	

- 3. ログ設定の変更をクリックします。
- 4. ログルールの新規インスタンスをクトックします。

Instance Logging Settings for Objects				
Name 🗢	Туре			
D /public/Examples/CheckPath	job			
Dublic/Examples/CompletePO.job	јођ			
Dublic/Examples/CopyFiles	job			
Dublic/Examples/CopyOutput	job			
Logging Rules:				
☐ Enable				
When: Then:				
Successful execution Retain log:				
✓ Failed execution Messages	severity: all			
Stopped by user Execution	trace: full			
☑ Interrupted				
new Instance Logging rule				

ログの構成設定はルールのセクション内で説明されている設定と同様に作動します。

オブシェクトレベルとアプリケーションレベルの両方でログルールを定義した場合、優先順序は北下のように確立されます

- オブジェクトレベルで定義されているログルールが最初にチェックされます。
- このレベルで見つかった一致が存在する場合、ルールは適用されアプリケーションレベルでのルールはチェックされません。
- このレベルで一致が見つからな、場合、アプリケーションレベルでの一致からエックされます。

3.8 サービスの開始と停止 (Linux)

FlowForce Server ソリューションはつのサービスこと構成されてます:

- 1. flowforcewebserver
- 2. flowforceserver

flowforcewebserver サービスを停止するために下のコマイを実行します。 flowforceserver サービスを開始、おけよ 停止 するけよ 下のコマイト内で flowforcewebserver を flowforceserver と置き換えます。

Flow Force Web Server サービスの開始方法:

sudo systemctl start flowforcewebserver

Flow Force Web サービスの停止方法:

sudo systemctl stop flowforcewebserver

サービスが作動中か確認するコよ次のコマイを実行してくたさい(flowforcewebserver おけまflowforceserver と servicename を置き換えます)。

sudo service servicename status

3.9 サービスの開始と停止 (macOS)

Flow Force Server サービスの開始:

sudo launchctl load /Library/LaunchDaemons/com.altova.FlowForceServer.plist

FlowForce Web Server サービスの開始:

sudo launchctl load /Library/LaunchDaemons/com.altova.FlowForceWebServer.plist

FlowForce Server サービスの停止:

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceServer.plist

FlowForce Web Server サービスの停止:

sudo launchctl unload /Library/LaunchDaemons/com.altova.FlowForceWebServer.plist

3.10 サービスの開始と停止 (Windows)

デフォルトでFlowForce Server サービスは自動的にWindows か開始する際に開始されます。サービスを手動で管理する必要がある 場合、以下の命令に従います。

Flow Force Server サービスの開始:

システム通知エリアの、ServiceController アイエン() をクルクします。そして、「Altova FlowForce Web > Setup」を選択します。

FlowForce Web Server サービスの開始:

• システム通知エリアの、ServiceController アイエン(🚠) をクルクします。そして、「Altova FlowForce Web > Start service」を選択します。

FlowForce Server サービスの停止:

• システム通知エアの Service Controller アイエン(🌆) をクリックします。そして、「Altova Flow Force Server > Stop service」を選択します。

FlowForce Web Server サービスの停止:

• システム通知エルアの、ServiceController アイエン(🌆) をクルックします。そして、「Altova FlowForce Web > Stop service」を選択します。

Microsoft 管理コンノールを使用して Flow Force Server サービスを開始することもできます (コトロール ネル> 管理ソール> サービス。

3.11 FlowForce Server アプリケーションデータ

このセクションでは、FlowForce Server アプリケーションデータディレクトリのコンテンソニン、て説明されています。この清報は、手動でメ ジャーな FlowForce Server バージョンにデーダを移行する場合、おさよ、ini ファイルを編集することにより FlowForce Server 構成 設定を変更する場合、役しささます。

アプリケーションデータディレクトリは、FlowForce Serverとジョブ、トリガー、システム関数、サーバーログ、他のファイルなど、そのユーザーにお生成されたデータを保管します。

アプリケーションデータディレクトリは、複数の.ini スタイル構成ファイルも含んでいます。管理者は、<u>Web管理インターフェイス、セナアップ</u> ページまけは、コマンド ライン・インターフェイスから設定を変更する代替として、.ini 構成ファイルをテキストエディターで編集することができます。 す。

アプリケーションデータディレクトリへのやは、オペレーションシステムとプラナフォームにより以下のように異なります。

Linux	/var/opt/Altova/FlowForceServer2021	
macOS	/var/Altova/FlowForceServer2021	
Windows	C:\ProgramData\Altova\FlowForceServer2021	

下のテーブルはアプリケーションデータディレケリ内のメインのファイルとフォルダーのノストを表しています。

flowforceserver.ini	FlowForce Server のプローノ N構成設定を保管します。(現在サーノ デログとエ ラーメッセージニ使用されている言語です)。
flowforceweb.ini	FlowForce W eb Server のブロー・ い構成設定を保管します。(W eb 管理インターフェイスの言語)。
	メモ(下て説明されているとおり) data ディレクトリニ保存されている flowforceweb.ini ファイルと区別してくたさい
data/cache.db	このデータベースファイルは、FlowForceのキャッシュ機能に関連したデータを保管 します、以下を参照・ジョブの結果をキャッシュする)。
data/state.db	このデータベースファイルは、FlowForceの可変の状態のまり、構成されていない 状態を保管します。
data/ffweb.log data/flowforce.log	これらのファイルは、FlowForce Web Server とFlowForce Server からの ログメッセージをそれそれ保管します。これはWindows とmacOS システムに対 して適用されます。
	Debian、Ubuntu おびCentOS おけお以降に対しては、ログはシステムログ に書き込まれます(/var/log/syslog)。
data/flowforce.db	このデータベースファイルは、FlowForce Server オブジェクトシステム、アクティブな データ、アクティブなジョブ、ロール、およびその他を保管します。
data/flowforcelog.db	このデータベースファイルは、FlowForce Server ログを保管します。
data/flowforce.ini	この構成ファイルは、FlowForce Server のポートとのリッスンインターフェイスを定義します。ファイルの参照の構成も参照してくたさい。

data/flowforceweb.ini	この構成ファイルは、FlowForce Web Server のポートとのトルスンインターフェイ スを定義します。ファイルの参照の構成も参照してくたさい。
data/files	デプロイされた機能に関連したファイルを保管します。
data/logs	ジョブ実行ステップからのキャプチャされた出力を含みます。
data/tmp	一時ファイルを保管します。
data/tools	(MapForce Server おさまStyleVision Server などの他のAltova サーバ 一製品がFlowForce Server と共にインストールされる場合、このディレクトリは FlowForce Server がこれらの製品作業を可能にする.tool ファイルを保管しま す。
	このデルクトリが空の場合、各製品のインストーリディレクトリから各製品のソール ファイルを手動でコピーすることができます。
	MapForce マピング、おけよ StyleVision 変換の実行に必要な.tool ファイ ルを環境変数を設定するために編集する必要かある可能性かあります(以下を参 照:環境変数の設定)。

3.12 FlowForce Server のローカライズ

FlowForce Server は以下の言語へのサポーを搭載しています:英語、フランス語、ドイン語、スペイン語、および日本語。これらの言語のローカライズされた、・・ジョンを作成する必要はありません。これらみつの言語をデフォルトの言語として設定する場合は、FlowForce Server のsetdeflang コマンドを使用してくたさい。

Flow Force Server のローカライズされたバージョンを作成する

- 1. exportresourcestrings コマイドを使用してリノース文字列を含むXML ファイルを生成します。
- 2. リノース文字列をターゲオ言語に翻訳します。リノース文字列は XML ファイル内の<string> 要素のエンテンソです。 {option} おは{product} などの、中かに内の変数は翻訳しないでくたとい。
- 3. 翻訳された XML ファイルからローカライズされた DLL ファイルを生成する JL Altova サポートに連絡を取ってくたさい。
- 4. Altova サポト(<u>https://www.altova.com/support</u>) かローカライズされたDLL ファイルを受信すると、 <INSTALLATION FOLDER>\FlowForceServer2021\bin フォルダーにDLL を保存してください。DLL ファイルは、 以下のフォームを使用します: FlowForceServer2021_1c.dll。名前の_1c 部分は 言語コードを含んています。例え ば、FlowForceServer2021 de.dll では、de パト がイン語(Deutsch)のオンの言語コードです。
- 5. 使用する FlowForce Server アプルして、ローカライズされた DLL ファイルを設定する コス、 <u>setdeflang</u> コマイドを使用 します。setdeflang コマイ・の引数して DLL 名の一部である言語コードを使用します。

3.13 バックアップと復元

FlowForce Server 内のデータをバックアップするコセンの方法があります:

- Web 管理インターフェイス。この種類の、シグアップはは構成データのみか含まれます:ショブ、資格情報、デプロイ済みの MapForce マンピングおけま Style Vision 変換、リノース AS2 証明書、AS2 パトナー。アプリケーション設定、おけまユ ーザーは含まれません。相対するオブジェクト上で、トミッションの許可する場合、FlowForce Server ユーザーは構成データの インポートおけまコクスポートを実行することができます。
- アプリケーションデータディレクトリの管理バックアップ。このアプローチはFlowForce Server がインストールされてしるマシン上のFlowForce Server アプリケーションデータディレクトリ へのアクセスが必要となります。アプリケーションデータディレクトリ前のドットからのすべてのデータが含まれており、ユーザーと(Active Directory などのDirectory Service からインポートされたユーザーとロールを含む)ロールおよび電子メール、おけは、LDAP サーバー設定、パスワードポパンー、クラスター設定などのアプリケーションノベルの設定が含まれます。
- メモ このトピックでは(ジョブ、封まはFlowForce リソース、封まれーカルファイルベースのデータベースに求して入力/出力であるファイ ルませまデルクトリなど)FlowForce Server の外部のデータの、シクアップと復元に関しては説明されていません。このデータを 個別に、シクアップする必要があります。ベストプラクティスとして、お簡単なバックアップと管理のために、可能であれば、このような外 部データを全て同じディノクトリに保管してくたさい。

ビナ

デーダを新規のマンメニ移行、ませま、バックアップから復元する場合、以下のヒートを考慮してデータのポータブル性を保証してくたさい

- LicenseServer が(しかゆる「フェールオー・デー」サー・ビージーがしかの2番目のサー・を構成することが奨励されます。詳細に関しては、LicenseServerドキュメト(<u>https://www.altova.com/documentation</u>)を参照してがさい。
- 全てのジョブは(インラインではな」、ノスタンドアロン資格情報を使用することが奨励されます。そのオペーティングシステムコンクされているユーザー資格情報をマッチするためコンゴブ内のローカル(インライン)資格情報を使用する場合、新規のサーバーマシン上で変数する必要があります。反対に、スタンドアロン資格情報を使用する場合、新しいサーバーマンン上でスタンドアロン資格情報を振集するおされです。
- MapForce からデプロイされたマンピング関数を実行する場合、絶対的なレファレンスの代わりに

 ダー・スとデータベースを参照することを考慮してくたさい。
- FlowForce Server 内で直接ユーザーとロールを作成し管理する代替として、W indows Active Directory おけま Directory Services のためのサポート付きの他のLDAP Server を使用してくたさい。詳細に関しては、ディレクトリサービス 設定の変更を参照してくたさい。

Web 管理インターフェイスからの部分的ない シクアップ

選択されたオブンエクトをバックアップする場合、FlowForce Web 管理インターフェイスコログインして、「エクスポート」機能を使用します。エーダを復元するばあい、インポート機能を使用してくたさい。詳細に関しては、構成データのインポートとエクスポートを参照してください。

メモ エクスポートされるデータと同じませ」は以降の、デジョンである FlowForce Server インスタンスは構成デーダをインポートすることができます。構成デーダを FlowForce Server の前の、デジョンにインポートすることは成功するかもしれませんが回避されるべきです。

全てのFlowForce アプリケーションデータのバックアップ

全てのアプリケーションデータのバックアップは基本的にFlowForce Server データベース(DATADIR)のコピーを後に必要に応じて安全な場所に作成することには構成されます。

下の命令内では、DATADIR は以下のディレクトリを参照します。

- Linux: /var/opt/Altova/FlowForceServer2021/data
- macOS: /var/Altova/FlowForceServer2021/data
- Windows: C:\ProgramData\Altova\FlowForceServer2021\data

時間とディスクのスペースを節約するために、DATADIR ディレクトリを可能な限リロンパクトにまとめます。以下の追加ステップを実際の、シクアップの前に行い節約を行います:

- 1. ビル・インのarchive-log 関数を実行するショブを作成し、古いログレコードをアーカイブ化します。
- 2. ビルトインのtruncate-log 関数を実行するションを作成し、古いログレコードを削除します。
- 3. ビルトクのcleanup-files 関数を実行するジョブを作成し、使用されていないファイルを削除します。
- 4. compactdb を使用して Flow Force Server 実行可能ファイルを実行します。

実際のバックアップステップを続行することができます

- 1. FlowForce Server と FlowForce Web Server サービスを停止します。 Linux、 macOS、 と Windows のための命令を 参照してくたさい。
- 2. DATADIR のエピーを作成し、安全なディレクトリニエピーします(異なるマンンませるオンスか良いでしょう)。慣例では、このエピーを以降の手順で DATADIR_BACKUP と呼びます。

DATADIR 内のprivate.db ファイルは、パワード秘密キーなどの機密な情報を含んでいます。バックアップが安全な場所に保管されていることを保証してくたさい。

データの復元

DATADIR_BACKUP か同じ、デションで現在作動している FlowForce Server と同じマンン上の場合、以下を簡単に復元すること かできます:

- 1. FlowForce Server サービスが作動中の場合、停止してくたさい。Linux、macOS、とWindows のための命令を参照してくたい。
- 2. DATADIR を例えば temp_data に名前を変更してくたさい。
- 3. DATADIR_BACKUPをDATADIR にピーしてけさい。
- 4. FlowForce Web Server とFlowForce Server サービスを開始します。

古いデータベースバージョンを有する他のマシンからの、シクアップも復元することができます。FlowForceデータを新規のサーバーに移行する場合、おけよハードウェアの障害が発生した場合などに下の手順は役に立つ可能性があります。

灹

- 同じおけば異なるオペレーティングシステムを作動するマシンで復元を実行することができます。後者の場合、ジョブ内で使用されてしるすべての、マは新規のオペレーティングシステムで有効です。場合かります。この場合、更新を手動で行う必要がおります。
- オペレーティングシステムユーザーアカウトの操作に関連する資格情報「ジョブ実行のための使用方法を許可する」オプション か有効化されている個所での資格情報」は新規のマシンで有効ではおよ、場合かあり、この場合手動で更新される必要かあります。

新規のFlowForce Server インストールませま デジョンに復元する方法

1. FlowForce Server をインホールに、適応できる場合、以下もインホールにます: MapForce Server, Style Vision Server および RaptorXML Server。LicenseServer もインホールする場合、FlowForce Server インホールの一部と して行います(Windows のみ実行可能です)。他のプラホフォームでは、LicenseServerを個別にインホールする必要が あります。

- 2. LicenseServer Web 管理インターフェイスコングオンし、古いマシンから登録されている製品すべてを登録解除します。次に LicenseServerを使用する新しいマシンからの製品を全て登録します。この手順は移行後しご行うこともできます。
- 3. FlowForce Server サービスが作動中の場合、停止してくたさい。Linux、macOS、とWindowsのための命を参照してくたさい。
- 4. DATADIR を例えば、temp_data に名前を変更してけさい。
- 5. --datadir とてDATADIR を、--olddatadir とてDATADIR_BACKUP を提供してmigratedb コマドを 実行します。例:

Windows

```
FlowForceServer migratedb
    --datadir=C:\ProgramData\Altova\FlowForceServer2021\data
    --olddatadir=C:\transfer\backup_data
```

CentOS

```
sudo ./flowforceserver migratedb
--datadir=/var/opt/Altova/FlowForceServer2021/data
--olddatatdir=/home/chang/backups/data
```

6. (順番に) FlowForce Server とFlowForce Web Server サビスを開始します

4 配布済みの実行

データのスループナを向上し、基本的なフォールトレランスを提供するために、FlowForce Server インスタンスかクラスターとして実行さえるように構成します。これにより以下の利点が提供されます:

- バランスのロード
- 更に適格なリソース管理
- スケジュールされた管理
- サービスの中断のリスクの削減

バランスをロードする方法

ハードウェアの制限によりFlowForce Server が同時に実行される複数のショブインスタンスによい圧倒される場合、ワークロードを他の実 行中のFlowForce Server(しわゆる「ワーカー」)再分配することができます。「マスター」マシンと複数の「ワーカー」マシンによく構成され ているクラスターをセットアップし、クラスター内のライセンス供与済みのコアを活用することができます。

リーンなリソース管理方法

「マスター」として指定されているマンンの一つは継続的にジョブトリガーを監視し、状況に応じてワーカーにキューされているアイテム、おけよ 自身に割り当てます。例えば、マスターマシンをジョブインスタンスを処理しないようご任意で構成し、データ処理とは異なり、リソースを自由に し、リノースを FlowForce Service の継続的な準備のためこのみ使用することができます。

ワーカーの計画された管理方法

サービスの準備を邪魔することはく、「マスター」ではないの実行中のFlowForce Server インスタンスを再起動、おけよ 一時的にシャト ダウンすることができます。「マスター」は常に使用できることが期待されます。再起動、おけよ シャトダウンはサービスの準備を中断します。

サービスの中断の危険性を削減する方法

ハードウェアの故障、停電、接続されていないやトワークケーブルなどの場合、影響は使用中のマンンが「ワーカー」おけばマスター」により異なります

- マンか「ワーカー」の場合、ワーカー上の実行中のFlowForce ジョブは失われます。ですが、同じジョブの新規のインスタンスが 異なるワーカー(おけよ、構成されている場合はマスター)により「き継がれるけっか、FlowForce サービスの一般的な提供は失 われません。失敗を含むジョブの実行ステータスは、マスターイユポートされ、ジョブログ上で表示されるけっか、管理者は適切なアク ションを手動でとることができます。
- マシンが「マスター」の場合、サービスの供給は失われます。この場合、マスターが使用できない限り、新規のジョブインスタンスを開始 することはできません。

4.1 配布済みの実行に関連する用語

次の用語は配布済みの実行とランスのロードに関連する用語です。

サーバーインターフェイス

サーバーインスタンスは実行されている、ライセンス供与済みのFlowForce Server ことです。両方のサービス(FlowForce Web Server とFlowForce Server)はマシン上で実行されていることが想定されています。

ジョブインスタンス

ジョブインスタンスはジョブとは異なります。FlowForce ジョブをジョブ構成ページから構成すると、ジョブ構成を作成することができます。ジョブのようの定義済みのトリガーの必要条件が適用されると、ジョブの実行か開始されます。ジョブインスタンスはジョブと関連する実行キューにより 定義されているとおりクラスター内で再配布されます。ジョブインスタンスは常に単一のクラスターメンバー上で全体で作動します。

クラスター

クラスターはインスタンスが使用できない場合、ジョブを並列して実行、ませよ、ジョブを再配布するために通信する Flow Force Server の 複数のサービスインスタンスを表します。クラスターはつの「マスター」と複数の「ワーカー」によ、構成されています。

マスター

ジョブことトリガーされた条件を継続的に評価し、FlowForce サービスインターフェイスを提供する「マスター」はFlowForce Server イン スタンスです。マスターは同じクラスター内のフーカーマシンを考慮し、ジョブインスタンスを処理する、更にてませま、代わりに)、ジョブインスタンス を割り当てるように構成されている可能性があります。

ワーカー

ローカルジョブを実行する代わりにマスターインスタンスと通信するように構成されている FlowForce Server インスタンスです。ワーカーはマスターFlowForce Server か割り当てているジョブのみを実行することができます。

実行キュー

実行キューはジョブの「プロセッサー」です。これはジョブインスタンスかどのように実行されるかを管理します。実行するオーガコは、各ジョブはター ゲト実行キュー(書い当てられます。キューは(キュー(書い当てられているすべてのジョブの)ジョブインスタンスが一度に実行される数、実 行間の遅延を管理します。デフォルトで、キュー設定はジョブニ対してローカルですが、キューを複数のジョブによりシェアされるスタイドアロンオブ ジェクトとして定義することもできます。キューはジョブニ対してローカルであること、おけよ、複数のジョブで共有されることができます。複数のジョ ブが同じキュー(こ書い当てられている場合、実行のオーガロそのキュー(は共有されています。

更に、適切な(特権とは異なるコンテナーパーミッションを有する場合のみキューを確認、おけよ、キュービジョブを割り当てすることができま す。具体的には、キューを作成するけっとコーザーが「実行キューを定義する「特権を持つ必要があります、特権のしくみを参照してくけさい。 更に、(特権とは異なる)適切なパーミッションがある場合のみユーザーはキューを確認、おけよキュービジョブを割り当てることができます。 パー ミッションのしくみ</u>を参照してくけさい。これは、キュービジョブを割り当てできることを意味します。キューへのアクセスを制限するけは、キューか定 義されている個所にコンテナーをサビゲートし、コンテナーのパーミッションをロールauthenticated のけっかば「キューーアクセス無し」に変更 します。次に、パーミッション「キューー使用」を特定のロール、おけは、必要なユーザーに割り当てます。詳細に関しては、<u>houblic コンテナ</u> <u>ーへのアクセスを制限する</u>を参照してください。

4.2 「マスター」モードでの FlowForce Server の作動

ジョブこよトリガーされた条件を継続的に評価し、FlowForceサービスインターフェイスを提供する「マスター」はFlowForce Serverイン スタンスです。マスターは同じクラスター内のフーカーマシンを考慮し、ジョブインスタンスを処理する、更に、ませま、代わりに)、ジョブインスタンス を割り当てるように構成されている可能性があります。

インストール後すぐ、FlowForce Server インスタンスは(自身を含む)1つのマシンクラスターのマスターとしての役割を果たします。しかしな がらワークロードを引き受けるワーカーが存在しないサンタンフレイン配布されません。クラスターをセットアップするはよこのドキュメントで記 述されている通り追加のFlowForce Server インスタンスをインストールし、「ワーカー」モードに変換します。パランスをロードするクラスター は、マスターマンノはつれえ、少なくとも1つのマンンがワーカーとしての役割を果たしている場合のみセットアップする準備が整っていると考えられま す。

メモ
クラスター内には一つのマスターマシンのみか存在することができ、ワーカーの数量に
排限されていません。

スタイアロンFlowForce Server インスタンスとマスターインスタンス間の操作に差はありません。ジョブの構成とログの処理は同様に行われます。唯一の差は、マスターは同じクラスターからのアーカーと通信することです。クラスター管理ページ内では、セキューティーケンを確認していないがジョインする予定のアーカーを含む、マスターミジョインされたワーカーのノストを常にビューすることができます。このページから、このようなワーカーを確認するセキューティーケンを生成し、ワーカーを完全に削除することができます。詳細は、「ワーカー」モードにFlowForce Serverを変更すると「ワーカー」モードを停止する方法を参照してくたさい。

マスターマシンは継続されたサービスの提供、ワーカー(割り当て済みのジョブインスタンスのステータスの収集、結果のレポトの責任を担います。この理由のため、マスターマシンは、処理環境の需要に従い、ランスされていることが重要です。これを達成するためはは、マスターが生にサービスインターフェイスを提供する一方、ジョブをワーカーにより処理されるキュー(コンダイントすることができます。マスターは、ワーカーが存在しては、イベント内でも、自身でフークロードの処理の一部を行うように構成することができます。。配布された実行のセルトアップを参照してくたさい。

4.3 「ワーカー」モードに FlowForce Server を変更する

「ワーカー」モードにFlowForce Serverを変更することは、「マスター」 FlowForce Server インスタンスによりクエストされている通り ジョブの処理のためこのみにコノノーズを割り当てることを意味します。通常モードに切り替えない限り、ワーカーに変更されると、FlowForce Server はコーカルで構成されているトリガーとショブを実行することはできません。FlowForce Server インスタンスの「ワーカー」ステータス はWeb 管理インターフェイス上に常に表示されています。

下に表示される通り、クラスター管理ページからFlowForce Serverを常にフーカーノードに変換することができます。ワーカーモードを必要 としたは、場合、FlowForce Serverを停止し、通常モードに戻すことができます。「ワーカー」モードを停止するを参照してくけざい。

W indows 上では、インストール中にクラスター構成オプションの一部を指定することができます。Windows 上でのクラスターインストールを参照してくたさい。

前提条件

- FlowForce Server インスタンスがインストールされており、作動している必要かあります。同じ必要条件が「マスター」としての役割を果たす2番目のFlowForce Server インスタンスにも適用されます。
- ケラスターに関連したアグションを取る必要のある各マシン上で、FlowForce ユーザーアカウトは、「クラスターの保持」特権を必要とします。
 特権のしくみを参照してください。デフォルトでは、ルートユーザーアカウトにはこの特権が存在します。
- MapForce Server、StyleVision Server、RaptorXML Server、おは、RaptorXML+XBRL Server ライセンを 必要とするジョブをワーカーが作動している場合、これたのソールはワーカーインスタンスにインストールされており、ライセンス供与され ている必要があります。マスターインスタンスはこのようなジョブを実行しません(全てのジョブとキューがワークロードをワーカーに再配 布することを想定して) これらのソールはマスターにインストールされる必要はありません。
- マスターマシン上のセナア・プページを開き、マスターインスタンスへの接続が有効化されていることを確認し、バインドアドレスか設定されていることを確認してください。例

マスターインスタン	ス暗号済み接続	
有効化済み:		
バインドアドレス:	● 全てのインターフェイス (0.0.0.0) ∨ ○ その他:	ポート: 4645

セトアップページを開くを参照してくたさい。

実行中のFlowForce Server をワーカー」モードに変更する方法

- 1. ワーカーナンる Flow Force Server インスタンスコングオノレ、 Flow Force Server コングナノするを参照していさい。
- 2. 「管理」をクリックして、「クラスター」クリックし、クラスター管理インターフェイスにアクセスします。
- 3. 「マスターインスタンスにジョインするリクエスト」をクリックします。
- 4. マスターゴなるマシンのホスト名を入力します。
- 5. 任意で、ジョインルケエストを識別するためにカスタムテキストメッセージを入力します。(このサンプルでは、「ワーカーマシンからこんにちは!」)。

マスターインスタン	スをジョインするリクエスト			
マスターインスタンスをジョ	ョインするリクエスト			
ホスト名:	host.any.domain.net	ポート	4645	
コメント:	ワーカーマシンからこんにちは!			
マスターインスタンスを	ビジョインする最初のステップはマスターインスタンスへの	のリクエストへの送信です。以降、ジョインリクエストをレビューしマス	マターインスタンス上で承諾する	ることはできません。
マスターのジョインのり	クエストの送信キャンセル			

6. 「マスターのジョインのリクエストの送信」をクルクします。

バインドアドレスがマスターマシン上で正確に構成されており、ポートがファイアウォールによりブロックされていないにを確認して くたさい。上記の必要条件を参照してくたさい。

- マスターである Flow Force Server コングオンし、ケラスター管理インターフェイス こアクセスします。
 ワーカーマシンからのジョイン パフエスト エト 「 を検索し、「 リクエストの承認」をクリックします。

リクエスト	
ID: 8D2F22F15804A14C878E12B9881A6D1A コメント: ワーカマシンからこんにちは!	リクエストの承諾リクエストを無視する

9. ワーカーマシンからのリクエストの横の「トーケンの表示」をクリックします。このフーカーをクラスター(ニショインするため)こ必要な秘密 キーが表示されます。

ステータス:	オフライン	ワーカーを削除
ID:	8D2F22F15804A14C878E12B9881A6D1A	
トークン:	dxkFuStNURCUnXIthoosZKqZF0N6q8w9ZIs29SX2UNBCUjvu/ibFWNNp7wHk/K6a トークンのコピー トークンを非表示	

- 10. ワーカーマンソニ秘密キーを転送します。これを行う簡単な方法は、プラウザー内で両方のFlowForce Serve インスタンスを開 き、1つのウィドウから他のウィドウニピーし張り付けます。ませよキーを転送するために、電子メールませま他の方法を使用す ることができます。
- 11. ワーカーマンからクラスター管理インターフェイスにアクセスします。
- 12.「マスターインスタンスのジョインを完了する」をクリックします。

マスターがジョインリ	ウエストを承諾するとマスターインスタンスを完了します
マスターインスタンスのジョイ	>の完了
ホスト名:	host.any.domain.net
トークン:	マスターインスタンスクラスター管 理ページからトークンをここに張り 付けます .::
マスターインスタンスをジ	ョインするために2番目のステップはメンバートークンをこのインスタンスにコピ-
マスターのジョインを完了す	する キャンセル

- 13. マスターの名前を入力し、秘密キー(トークン)を提供されたテキストボックスに張り付け、「マスターのジョインを完了する」をクリックします。
- 成功した場合、ページ内に通知メッセージが表示されます。

Joined the master.	
Cluster management	
Worker	
Leave master instance and switch to Standard mode to configure and execute local jobs	
Leave Master Instance	

FlowForce Server はフーカーモードで、マスターマンノによるリクエストによりのみジョブを実行することができます。マンンをスタードアロンノー ドに切り替える必要かある場合は、「マスターインスタンスから移動する」をクリックします。「ワーカー」モードを停止する方法を参照してください。

4.4 Windows 上でのクラスターインストール

FlowForce Server をWindows にインストールする場合、ウィザードのステップのト つことり FlowForce Server のクラスターの状態を 定義することができます: スタードフロン、マスター、ワーカー。

Altova FlowForce Server 2020 Advanced Edition - インストーラー ウィザード			
ALTOVA®			
FlowForce クラスター構成 FlowForce クラスターをセットアップするにはここを構成してください ② スタンドアロン ③ マスター ③ ワーカー FlowForce サーバーはクラスターの一部ではありません			
< 戻る(B) 次へ(N) > キャンセル			

インストール後にクラスターをセットアップすることもできます。それ以外の場合、次のオプションから1つを選択してくたさい

- a. (デフォルト)複数のFlowForce Server インスタンスをクラスターとして実行しない場合、「スタンドアロン」を選択します。
- b. <u>マスター</u> インスタンスとしてこの Flow Force Server インスタンスを使用する場合「マスター」を選択します。このオプション は、マスター Flow Force Server インスタンスがリッスンするポート番号を指定することを必要としてます。このポートは 「Flow Force Server」と「Flow Force Web Server」サービスにより使用されているポートと異なることに注意してくだ さい。
- c. ワーカーインスタンスとしてこのFlowForce Server インスタンスを使用する場合、「ワーカー」を選択して下さい。

「マスター」を選択すると、セトアップページ内の「マスターインスタンス暗号化済み接続」がインストール中に指定された値に設定されます。 クラスターゴーカーを追加することができます。これを行うゴは、新規のFlowForce Server インスタンスを上に示されるようコフーカーとして 追加する、おけよ、既存のFlowForce Server インスタンスをワーカーモードに変換します。「ワーカー」モードにFlowForce Server を 変更するを参照してください。アプローチに関わらず、マスターインゴロインする前に、「ワーカー」モードにFlowForce Server を変更するで説 明されているとおり、各ワーカーのセキュリティトークを手動で確認する必要があることに注意してくたさい。

「ワーカー」を選択した場合、ルートユーザ(おけよ「ケラスターを管理する」特権を有するユーザー)として最初にエグインた後、ケラスター 管理ページにダイン外されます。ケラスター管理ページから、マスターイごコインすることをリクエストでき、「ワーカー」モードにFlowForce Server を変更する内て説明されている通り処理を完了することもできます。

4.5 配布された実行のセットアップ

配布済みの実行のコアコは実行キューの概念が存在します。

実行キューインショブの「プロセッサー」です。これはジョブインスタンスかどのように実行されるかを管理します。実行するオンタリコよ 各ジョブはター ゲト実行キューイ:書い当てられます。キューしよ(キューイ:書い当てられているすべてのジョブの)ジョブインスタンスが一度に実行される数、実 行間の遅延を管理します。デフォルトで、キュー設定はジョブニマトしてローカルですが、キューを複数のジョブによりシェアされるスタメドアロンオブ ジェケトとして定義することもできます。キューインショブに交けしてローカルであること、まりょし、複数のジョブで共有されることができます。複数のジョ ブが同じキューイニ書い当てられている場合、実行のすっかにそのキューイは共有されています。

更に、適切な(特権とは異なるコンテナー・トミッションを有する場合のみキューを確認、おけよ、キュービジョブを割り当てすることができます。具体的には、キューを作成するけのビューザーが「実行キューを定義する特権を持つ必要があります、特権のしくみを参照してくけさい。 更に、(特権とは異なる)適切ない・ミッションがある場合のみユーザーはキューを確認、おけはキュービジョブを割り当てることができます。 パー ミッションのしくみを参照してくけさい。これは、キュービジョブを割り当ててきることを意味します。キューへのアクセスを制限するけは、キューが定 義されている個所にコンテナーをナビゲートし、コンテナーの、トミッションをロールauthenticated のけっぱってキューーアクセス無し」に変更 します。次に、パーミッション「キューー使用」を特定のロール、おけよ、必要なユーザーに割り当てます。詳細に関しては、/public コンテナ ーへのアクセスを制限するを参照してください。

共有されたキューは単一のFlowForce マシン上、おけよクラスターとして作動する複数のFlowForce Server インスタンス上でサーバ 一のロードを管理する柔軟的なメカニズムを提供します。ロードの、ランスの構成は複数のステップのプロセスです:

- 1. 最初に、資格情報おゴンゴンどのFlowForce構成データン類似した専用のページを作成します。
- 2. それぞれのキューに、処理設定を定義します。例えば、マスターのみ、ワーカーのみで実行されるキューを構成します。例えば、デ フォルトでマスターとすべてのフーカーでキューが実行されるように構成することができまが、すべてのフーカーが使用できない場合、 キューはマスターにのみファールッグします。
- 3. 各ジョブの構成を編集し、ジョブを前に作成されたカスタムキューイニョル当てます。

キューの作成

キューをスタイアロンオブジェクトとして作成する

- 1. 「構成」をクリックして、キューを作成するコンテナーにナビゲートします。
- 2. 「作成」をクリックして、「キューの作成」をクリックします。

以下にキューを作成 /public				
‡ :	ユー名:			
‡ :	ューの説明:			
				.:
キュー設定				
(•			
	実行:	マスターまたはワーカー 🗸	1 🕨 🕒	۲
	実行間の最短時間:	0 秒		
	並列実行の最大数:	1 インスタンス		
(•			
保存				

- 3. キューの名前、そして任意で、詳細を入力します。すべての設定の参照に関しては、下の「キューの設定」を参照してくたさい。
- 4. [保存]をクルクします。

キューの設定

構成のために使用することのできる設定は下にリストされるとおりです。

キューの名前	キューを識別する名前を入力します。これは必須のフィールドで、スペースで開始し、終了する必要があります。ますは文字、数字、単一スペース、アンダースコア(″_ ″)、ダッシュ(″-″)および終止符(″.″)文字のみを含むことができます。 (ローカルではなく)キューカ文タ・ドアロンキューとして定義されている場合のみこのフィールドは適応可能です。	
キューの説明	オプションで、キューオブジェクトのための説明を入力します。	
	(ローカルではなく)キューカヌタイブロンキューとして定義されている場合のみこのフィールドは適応可能です。	
実行	このキューからのすべてのジョブインスタンスかどのようご実行されるかを指定します: • マスターまたはワーカー - このキューの - 部であるジョブインスタンスはマスターまけ」はフ ーカーマンン上で使用できるサーバーコアに従い実行されます。 • マスターのみ - ジョブインスタンスはマスターマシン上でのみ実行されます。 • ワーカーのみ - ジョブインスタンスは使用できるワーカー上で実行されますが、マスター 上では実行されません。	
実行間の最短時間	実行キューは現在アクティブなルールご従い割り当てられたワーカーの数量を乗算した「平行実行の最大数」設定により管理されている使用可能なスロットの数量である実行スロットを提供します。各スロットは連続的にジョブインスタンスを実行します。 「実行間の最短時間」設定は、次のジョブインスタンスをすく12ピックアップしないように、スロットを ジョブインスタンスの完了後、短い間、使用中とマークします。この機能により、この実行キューの ために最大のスループットを削減し、同じマンン上での他の実行キューと他の処理のためにCPU タイムを提供します。	
----------	---	
並列実行の最高数	このオプタンはキュー上で使用可能な実行スロナの数量を定義します。設定は現在のキュー 内で並列に実行される同じジョブのインスタンスの数量を決定し、各スロナトはインスタンスをシー ケンス順に実行します。しかしながら、並列で実行を許可するインスタンスの数量は使用可能 なマシンハノースを競合することに注意して下さい。この値の増加は、集中したこ1/0 オペレーショ ンを実行しない、おちは大量のCPU時間を必要としない軽量のジョブを処理するキューが受け 入れることが可能です。(重量のジョブインスタンスを一度に処理するナッロンデフォハトの設定 1 は最も保守的で、リノース集中型のジョブを処理するジョブに最適です。 このオプタンはまFlowForce Server により受け入れられる(Webサービスとしてジョブを公開 するケライアントからの、並列HTTPリケエストの最大数に影響を与えません。詳細に関しては FlowForce Server プールスレンドの再構成を参照してくたさい。	

キュー設定の複数の設定を定義することができ、 한 ポタンをクリックすることにより、それそれは異なる処理必要条件を持つことができます。

設定の特定のセナの優先順位(これを「ルーリ」と呼ぶことします)を変更するはよ「上に移動」
し おは「下に移動」
と か
アクトックします。例えば、マスターのみか使用できる場合、これらのケースのオメリコレールを定義することができ、マスターとフーカの両方が使用できる場合、他のレールを指定することができます。これにより、特定の場合でクラスターの状態により、キューのオメリニフォール・シクメカニズ
ムを有効化することができます。
キューを処理する場合、FlowForce Server は常にクラスターの状態をモニターし、ワーカーが使用できる
かを検知することができます。
ですから、複数のキュー設定ルールを定義すると、FlowForce Server は、定義された順番に、上から下に
評価し、「実行」に従い、少なくとも1つの割り与えられたクラスターメンバーを持つ最初のルールを取得します。」ます。

サンプルの通り、クラスターが一つのマスターと4つのフーカーマシンを含むセントアップを例にしましょう。キューの設定は下の通り定義されています

‡	ユー設定			
	•			
	実行:	ワーカーのみ	~	1 4 🕤 💼
	実行間の最短時間:	0 秒		
	並列実行の最大数:	4 インスタンス		
	•			
	実行:	マスターのみ	\sim	1. 6
	実行間の最短時間:	5 秒		
	並列実行の最大数:	1 インスタンス		
	•			

上記の構成か示す通り、FlowForce はクラスターの現在の状態によりキューを以下のように処理します。

- 全てのフーカーを使用できる場合、上のルールがマッチされ適用されます。具体的には、16 個までのジョブインスタンスの同時実行 か許可されています(各ワーカニマ扎て4 インスタンスの実行が可能です)。実行間の最短の時間は0 秒です。
- 3 人のフーカのみが使用できる場合、上のルールがマッチされ適用されます。具体的には、12 個までのジョブインスタンスの同時実行が許可されており、実行間の最短の時間は0 秒です。
- 使用できるワーカーか存在しない場合、2番目のルールがマッチされ、適用されます。具体的には、1つのインスタンスか許可され、 具体的には、1つまでのインスタンスの同時実行が許可されており、実行間の最短の時間は5秒です。

この種類の構成でも、ワーカが存在しない場合の実行が可能です。「マスターのみ」のルールは厳しいことに注意してくたさい(1つのインスタンスのみ、各実行間の遅ればお秒です)これは、すべてのフーカが失敗した場合でも、マスターから処理パワーを取りすぎることはおりません。

キューージョブを割り当てる方法

キューを構成すると、このキューイニョル当てる各ジョブの構成の編集を行う必要があります。「キューの設定」グループ内のション構成ページ内でキューの設定を見つけることができます。

キュー設定		
キューを使用したジョブの実行: ● 既存のキューを選択: ○ ローカルキューを定義: 	/public/Queue1	• 2
保存	名前 ◆ ⁽²⁾ Queue1 (²⁾ キュ−1	

メモ 「ローカルキューの定義」を選択すると、FlowForce Server はジョブをこのジョブのインスタンスであるランタイムでデフォルトの キューイ:指定されているローカルの設定と共に割り当てます。キューの設定の定義を参照してくたさい、ローカルキューイ」「掘れの処理を行用するためにキューイはスタインアロン(ジョブの外部)で作成される必要があります。

4.6 「ワーカー」モードを停止する方法

以前の通常のFlowForce Server 状態にフーカーマシンを変更する必要がある場合、ワーカーマンンのプラスター管理インターフェイスから変更することができます。

- 1. FlowForce ユーザーアカウトに「クラスターの管理」特権が存在することを確認してくたさい。特権のくみを参照してくたさい。
- 2. ワーカーマンンで、「管理」をクリックして、「クラスター」をクリックします。
- 3. 「マスターインスタンスから移動する」をクリックします。

これはFlowForce Server インスタンスを通常の作動モードに変更します。マスターにより明示的に消除されない限り、マスターに登録され続けます。この状態でクラスターに再登録する場合、マスターマシン上でこのフーカーのために秘密キーを生成することができます。マスターマシンからフーカーを完全に消除する場合、下の指示を参照してください。

「ワーカー」モードを停止する方法

マスターマシン上で、過去のマスターインスタンスペジョインをリクエストしたすべてのフーカーがクラスター管理ページ内で表示されています。これ 「コキセキュ」ティトークンを確認済みのフーカーと、確認していないワーカーの両方が含まれています。後者のカテゴリよ(ワーカーではなく)通常の状態に変更されたマシンを含んでいます。

ワーカーモードを停止することが、ワーカーを削除するとフーカーをワークモードのまま保留し、マスターイ接続することができななります。 マスターへの接続を再度可能にするこれ、上で説明されているとおり、ワーカーマシン上で「マスターインスタンスから移動」アクションを 実行します。

ワーカーをマスターインスタンスから削除する方法:

- 1. FlowForce ユーザーアカウトが クラスターの管理」特権を所有していることを確認してくたさい。特権のしくみを参照してくたさい。
- 2. マスターマンン上で、「管理」をクリックして、「クラスター」をクリックします。「ワーカー」セクション内でフーカーのノストが表示されます。

マス	9–		
>	インバー		
	ステータス: ID:	マスター 04E543CD37DB584B967C8EC4B4B8CBF0	
	ステータス: ID: トークン:	オフライン 8D2F22F15804A14C878E12B9881A6D1A トークンを表示	ワーカーを削除

- 1. 「ワーカーの削除」をクリックします。確認メッセージが表示されます。
- 2. 「確認して削除する」をクルクします。

5 ユーザーアクセスの管理

このセクションコはFlowForce Server 内のユーザーアクセス管理に適用することのできる手順と概念か含まれます。

5.1 ユーザーとロール

このセクションは以下のトピックを含んています

- <u>ユーザーの作成</u>
- <u>ロールの作成</u>
- <u>ドメインユーザーとロールをインポート</u>
- <u>デフォルトのユーザーとロール</u>
- <u>ユーザーとロールの名前の変更</u>
- <u>ユーザーコールを割り当てる</u>
- <u>ロールを他のロールに割り当てる</u>

5.1.1 ユーザーの作成

ユーザーとはFlowForce Server にコングオルジョンを構成、おけはMapForce おけはStyleVision 変換をデプロイレ、FlowForce Server を管理する個人を指します。ユーザーが使用することのできるアグションは以下には影響されません

- a)割り当てられたパーミッション おけは特権
- b) ユーザーがソンバーであるロールに書い当てられているパーミッションと特権

FlowForce Server ユーザーを追加する

- 1.「管理」をクルクします。そして、「ユーザー」をクルクします。
- 2. 「ユーザーの作成」をクリックします。
- 3. 必要なフィールドに情報を入力します。

<i>ユーザー名</i>	ユーザー名を入力します。以下の制約が適用されます: • フィーリドが空であってはなりません。 • ユーザー名の始めと終わりにスペースを使用することはできません。 • 許可されている文字:英数字、数字、アンダースコア、(_)、ダッシュ、 (-)、およびピ、リオド(.)。
ノなワード	ユーザーの、マワードを入力します。
<i>、 </i>	ユーザーの、アワードを再入力します。
次回ログインで/ &ワードを変 更する	このチェックボックスを選択すると、ユーザーは、マワードを次回ログインで変更するように問われます。

- 5. 「保存」をクルクします。

5.1.2 ロールの作成

ロールはビジネスニースをベースしたセキューティの強化を手助けする特権のセルです。標準的なロールをベースしたセキューティースは 最低2つのロールが含まれます:管理者と一般ユーザー。各ロールは、ロールに与えられた特権により定義されています。例えば、管理者は、 自身のおよび他のユーザーの、マワードを変更することができますが、一般ユーザーは自身の、マワードのみしか変更することができません。ユー ザーノコールを割り当て、必要に応じて、ユーザーのロールを取り消すことができます。

FlowForce Server ロールを追加する

- 1. 「管理」をクルクし、「ロール」をクルクします。
- 2. 「ロールの作成」をクリックします。
- 3. ロール名を入力します(例えば「Administrator」)。
- 4. 「特権」から、ロールに割り当てられる特権を選択します(使用することのできる特権に関しての詳細は、以下を参照してくたさい、 特権)。
- 5. 「保存」をクルクします。

5.1.3 ドメインユーザーとロールをインポート

FlowForce Server ユーザーの作成に加え、ドメインユーザーアカウトとグループをWindows Active Directory からインポートすることができます。

「ドメインユーザー」コログインを許可」設定がディノカリサービス設定内で有効化されている場合、Windows ドメインユーザーはア カウトがFlowForce Server データベースに明示的にインポートされていない場合でも、FlowForce Server にログオンすることができ ます。ドメインユーザーのFlowForce サーバーへのログオンを確実にするには、ユーザーのWindows アカウトか明示的に管理者によりイ ンポートされており、「ドメインユーザー」コログインを許可」チェックボックスがウリアされている状態で、ドメインユーザーをインポートします。

メモ ローカルズシンのアカウトはActive Directory の一部ではない可能性があり、FlowForce Server にんポトできません。

W indows ドメイン ユーザーアカウント を Flow Force Server にインポート する:

- 1. 「設定」から「デルノトリサービス」の下の「有効化」チェックボックスを選択し、デルノトリサービスの設定を変更する内で説明 される通い優先されるデルノトリサービスプロバイダーを構成します。
- 2.「管理」の下の「ユーザー」をクリックします。
- 3. 「ドメインユーザーのインポート」をクリックします。

ンテキスト: vie.altova.com (vie)		\sim
] 名前 ≑	説明	
🛛 💄 Abagael Aasen	user.5@maildomain.net	^
🛛 💄 Abagail Abadines	user.6@maildomain.net	
🛛 💄 Abahri Abazari	user.7@maildomain.net	
Abbas Abbatantuono	user.8@maildomain.net	
Abbe Abbate	user.9@maildomain.net	
💄 Abbey Abbie	user.10@maildomain.net	
Abbi Abbott	user.11@maildomain.net	
Abbie Abdalla	user.12@maildomain.net	~
	discritz(ginandomainine)	>

- 4. 適用できる場合、「コンテキスト」ドロップダウノノストから選択のドメインを選択します。
- インポトするユーザーアカウト名を入力します。部分的な検索は有効です、ですから、例えば、「ad」などの値を入力すると、アカウト「管理者」、「Admanager」、および「Admin」は、LDAP サーバーおけはActive Directory から抽出され、Webページダイアログで表示されます。Active Directory の場合、FlowForce Server は、単一の向内で複雑な検索条件を指定することのできるAmbiguous Name Resolution (ANR)検索のアルゴリズンを使用します。例えば、「ji sm」と入力することにより、Jim Smith とう名前の個人のアカウトを抽出することができます。Active Directory 内のAmbiguous Name Resolution (ご関する Microsoft ドキュントを参照してくたとい。)
- 6. インポートされる記録の横のチェックボックスをチェックし、「インポート」をクリックします。Active Directory から記録が抽出される まで待ってください。ドメインがローカルではない場合、待ち時間が増加する可能性があります。

ドメイングループを Flow Force Server にインポート する

- 1. 「管理」から「Roles」をクリックします。
- 2. 「ドメインロールのインポート」をクリックします。
- 3. 上のステップ 4-6 に従います。

以下も参照してくたさい

• デルケリサービス設定の変更

5.1.4 デフォルトのユーザーとロール

デフォルトのユーザー

FlowForce Server には、以下の定義済みのユーザーが存在します。

[≜] root	ユーザーは、最初の、トップレベルのFlowForce Server 管理者です。デフォルトでは、システム内の全ての、トミッションおよび特権を有します。
anonymous	これは、明示的コングインないユーザーのナダの特別なアナナントです。FlowForce Server 管理インターフェイスへの匿名のアクセスは、不可能ですが、として示されている特定の Webサービスのナダの匿名のアクセスを有効化することができます(以下を参照: <u>ジョブを</u> Webサービスとして表示する)。

特権を変更することはできますが、ビルトインユーザーを削除することはできません。

メモ ユーザー▲ root は、自身のパーミッションと特権を含む特権とパーミッションを全て変更することができます。 意図せず自身のシ ステムアクセスを失う可能性があるため、▲ root としてログインする場合は、慎重にコレートの特権を編集してくたさい、これが発生した場合、<u>ルレートパワワードの再設定</u>を参照してくたさい。

デフォルトのロール

FlowForce Server には、以下の定義済みのロールが存在します。

authenticated	このロールは 既存のユーザー名と ダワードの使用を認証されている全てのユーザーを含みます。 ユーザー 🎍 anonymous 以外の各 FlowForce Server ユーザーは このロールのメン バーです。 デフォルトでは、 このロールは、 自身のノ ダワードを設定する特権を有しています。
all	このコー・リは、ユー・ザー ^島 anonymous を含む全てのFlowForce Server ユー・ザを含み ます。デフォハト・では、 このコー・リルゴは特権は与えられていません。

ロール authenticated おけま all は ビルト インのため これらのロールをユーザー に割り当てる おけま ユーザーか 消除すること はできません。ビルトインロールのメンバーシップは自動的に FlowForce Server に い管理されています。新しい ローザーを追加する都度、FlowForce Server は、自動的に新しいローザー コール authenticated とロール all を割り当てます。

ビルトインのユーザーとロールの特権を変更する場合は、この変更が引き起こす影響を考慮してくたさい。現在与えられてい全ての特権のグローンジルビューを確認するコよ、特権レポートを使用してくたさい(以下を参照: 特権レポート)。

5.1.5 ユーザーとロールの名前の変更

ユーザー名を変更する

- 1.「管理」をクトックします。そして、「ユーザー」をクトックします。
- 2. 編集するレコードをクリックします。
- 3. 「ユーザー名」テキストボックスに新しい名前を入力し、「保存」をクリックします。

ノモ

- ユーザー名か変更されても、現在割り当てられているユーザーパワードは変更されません。
- (名前変更の特権を有すると前提して)ユーザー自身の名前を変更する場合、変更された名前は保存をクリックすると有効 しなり、ページの右上に表示されます。

ロールの名前を変更する

- 1. 「管理」をクリックし、「ロール」をクリックします。
- 2. 編集するレコードをクリックします。
- 3. 「ロール名」テキストボックスに新しいロール名を入力し、「保存」をクリックします。

ᄯ

- ロールの名前が変更されても、ロールのメンバーは変更されません。
- デフォルトロールである ²all と²authenticated は変更することができません。

5.1.6 ユーザーにロールを割り当てる

ユーザーに1つまたは複数のロールを割り当てる

- 1.「管理」をクリックします。そして、「ユーザー」をクリックします。
- 2. ユーザーのノストから、編集するレコードをクリックします。
- 3. 「利用可能なロール」から、ユーザーに割り当てられるロールを選択し、「割り当て」をクリックします。

ユーザーから1つまたは複数のロールを削除する

- 1. 「管理」をクリックします。そして、「ユーザー」をクリックします。
- 2. ユーザーのノストから、編集するレコードをクリックします。
- 3. 「ユーザー'くユーザー名>'に割り当てられたロール」から、ユーザーから削除するロールを選択し、「削除」をクリックします。

ロールを複数のユーザーに割り当てる

- 1.「管理」をクルクします。そして、「ロール」をクルクします。
- 2. ロールのノストから、編集するレコードをクリックします。
- 3. 「利用可能なユーザー/ロール」から、ロールを割り当てられるユーザーを選択し、「割り当て」をクトックします。

ロールを複数のユーザーから削除する

- 1.「管理」をクリックします。そして、「ロール」をクリックします。
- 2. ロールのノストから、編集するレコードをクリックします。
- 3. 「ロール'<ロール名>'のメンバー」からロールを削除するユーザーを選択し、「削除」をクリックします。

5.1.7 ロールを他のロールに割り当てる

ロールを他のロールに割り当てることにお所属する機関、おけよビジネスの階層をFlowForce Server 内でモデルすることができます。例 えば ² 社員 とうロールと ² マーケティング部署 とうロールを作成します。ロール² マーケティング部署 を² 社員 のメンバ ーナなるように割り当てます。これは ² 社員 に与えられた全ての特権とペーミッションが自動的に² マーケティング部署 のメンバーに よ継承されることを意味します。

ロールを他のロールに割り当てる

- 1.「管理」をクリックします。そして、「ロール」をクリックします。
- 2. ロールのリストから、他のロールを割り当てるロールをクリックします、例えば ロール ² マーケティング部署 (コール ² 社員 から特権を継承する場合 社員 (Employees) をクリックします)。
- 3. 「利用可能なユーザー/ロール」から、割り当てられるロールを選択し、「割り当て」をクリックします。

次も参照してくたさい

- <u>特権の動作方法</u>
- <u>۱²-ミッションのしくみ</u>

5.1.8 ルートパスワードの再設定

²root ユーザーアカウトの ペワードを忘れた場合、おけ紛失した場合、コマイドラインインターフェイスからデフォルトの値をルセルする ことができます(以下を参照 コマイ <u>resetpassword</u>)。

ルト・スワードをレナオる場合、FlowForce バイナルデータファイルを含む、FlowForce か動作しているオペレーティングシステムへの アクセスを有するとみなします。これは、FlowForce をインストールする場合、おけよ、新規 FlowForce バージョンまけはサーバーを手動 で移行する場合に必要なアクセスと同類のアクセスです。

パマワードをやかする場合、 Aroot ユーザーの特権は、 デフォルト値に戻されます(つまり、全ての特権が与えられます)。

ルートノダワードの地水は、 Lot ユーザーのみに影響し、他のFlowForce ユーザーコよ影響しません。

5.2 特権

このセクションは以下のトピックを含んでいます

- <u>特権の作動方法</u>
- <u>特権レポーの閲覧</u>

5.2.1 特権の作動方法

特権は、FlowForce Server 内でユーザーが行うことのできる内容を定義します、例えば、自身のパワフト、設定、ユーザーとロールの読み 取り、ジョブの停止、など)。特権は、パーミッションはエレテナーへのユーザーアクセスを管理し、特権はグロー・シルこFlowForce Server 全体に対して効果がある点でパーミッションと異ないます。次の簡単なルールは、特権とパーミッションの違いを理解する助けてないます。特権 はグロー・シルであり、パーミッションはローカルです。

パーミッションは類似して、特権は個々のユーザーとユールに割り当てることができます。ですから、ユーザーがFlowForce Server にエグオン すると、有効な特権はおい下により決定されます。

- a)直接割り当てられた特権
- b)ユーザがシンドーであるロールに割り当てられた特権

以下の特権がFlowForce Server で使用することができます。

実行キューの定義	ジョブ実行キューを作成し管理する権利を与えます。これコンショブに対してローカルなキューと、 ジョブ外部で定義されている外部キューか含まれます。外部キューゴ起市済みの実行と組み合わせて使用されます。配布済みの実行を参照してくたさい。
クラスターの保持	複数のFlowForce Server インスタンスをクラスターとして管理するアクションを実行する権利 を提供します。例えば、FlowForce Server の現在のサービスインスタンスを「ワーカー」に変 換できるように、ユーザーはこの特権を必要とします。 <u>配布済みの実行と、ランスのロード</u> を参照 してくたさい。
グロー・シレな設定の管理	この特権は、タイムゾーンとメールサー、一設定などの設定ページ内で使用することのできる FlowForce Server グロー、VL設定を変更する権利を与えます。 これは、 FlowForce Server 管理者の特権であり、管理者のみに与えられる特権です。
ユーザー、ロール、特権の管理	この特権は、次のデータを追加、編集、および削除する権利です: ・ ユーザー ・ ロール ・ 特権 ・ パスワード これは、FlowForce Server 管理者の特権であり、管理者のみに与えられる特権です。デ フォルトでは、ユーザー & root のみかこの特権を有しています。
セキュリティの上書き	この特権を持つユーザーは、セキューティの「書き込み」パーミッション無しにコンテナーを変更す ることができます。これによりFlowForce Server 管理者は、誤ってアクセス付加ロジェオニアク セスを取り戻すことができます。 これは、FlowForce Server 管理者の特権であり、管理者のみに与えられる特権です。デ フォリトでは、ユーザー ^{&} root のみかこの特権を有しています。

<i>ユーザーとロールの読み取り</i>	デフォルトでは、ユーザーは、自身のユーザーアカウトトとメンバーであるロールを閲覧することができます。この特権を有すると、ユーザーは、全ての既存のユーザーとロールを閲覧することができます。
	デフォルトでは、ユーザー A root のみかこの特権を有しています。
機密データの抽出	この特権は機密なデータの以下のカテゴレをプレーンなテキストとしてビューする特権を与えます
	 パスワード 証明書秘密キー OAuth 2.0 アクセストークン、更新トークン、およびクライアントの秘密.
	デフォルトでは、ユーザー 🚨 rootのみか特権を有しています。それ以外を設定する理由が存在 する以外、この特権は 🚨 root のみに予約されています。
ノなワードの設定	この特権は、ユーザーにユーザー自身の、ペワードを変更する権利を与えます。この特権を有さ ないユーザーはFlowForce Server 管理者が設定したノペワードを取得する必要が決ま す。
	デフォリト では ^{La} anonymous 以外の ^{La} authenticated ローリとすべてのユーザーア カウト かこの特権を有します。
· ジョブの停止	この特権は、ジョブを作成したユーザーは関わり無く、実行中のFlowForce Server ジョブを 停止する権利を与えます。
フィルター処理無しのログの表示	デンオルトでは、ユーザーは「読み取り」アクセスかある構成に関連したログエトリを閲覧する

ここいにきょ9。この特徴を有9 ると、ユーザーは、特定の構成に実理しよい うべくのエクエ を閲覧することができます。

継承

ユーザー(例えば、 Alethia Alonso に) 直接、 おさよ、特定のコール 例えば、 ⁴²マーケティング部課長)に割り当てることができま す。 長期的に 特権の管理を簡素化するかか、後者のアプローチが奨励されます。 例えば、 ユーザー は所属部署を変更、 おさよ、 入社、 退 社する可能性からいます。 この場合、 個々のユーザーの特権を管理することは、 非効率的なタスクインな可能性からいます。 特権を、 ユーザー ではなく、 ロールに割り当てることにより、 細部に及ぶ管理を簡素化することができ、 各ユーザーではなく、 各グループ、 おさま部署のビジネスニ ーズに 焦点を当てることができます。

ロールを他のロールに割り当てることにお所属する機関、おさよビジネスの階層をFlowForce Server 内でモデルすることができます。例 えば ² 社員 とうロールと ² マーケティング部署 とうロールを作成します。ロール² マーケティング部署 を² 社員 のメンバー インなるように割り当てます。これは ² 社員 に与えられた全ての特権とペーミッションが自動的に² マーケティング部署 のメンバーに よ、継承されることを意味します。

更に、² マーケティング部課長 ロールに² マーケティング部署 ロールのメンバー なるように割り当てることができます。この場合、² マーケティング部課長 ロールは、²マーケティング部署 と² 社員 ロールの双方から特権を継承します。新しいマネージャーが入社 すると、² Alethia Alonso が² マーケティング部課長 ロールに割り当てられていると、広いロールからの特権を継承すること」ないま す。

& 社員
& マーケティング部署
& マーケティング部課長
& Alethia Alonso

図か表示するようこ、 ^A Alethia Alonso は、パーミッションと特権 をロール A マーケティング部課長 から継承します。このロール、は A マーケティング部署 およびから特権を継承します。

新規にインストールされたFlowForce Server システムでは、デフォルトのユーザーとコール、ユーザーと特権のダイアグラムは以下のようになります。



図が表示するように、システム内の各ユーザーは、 ⁴ all ローリル内で定義された特権を継承します。 しかしながら、既存の ユーザー(この 場合、 ⁴ root)のかが、 ⁴ authenticated ローリル内で定義された特権を継承します。 FlowForce Server に新規のユーザーを 追加すると、これらのユーザーは ⁴ all と⁴ authenticated ローリルご割り当てられます(該当する場合、これらのローリル内で定義され た特権が与えられます)。



以下も参照してくたさい

- <u>デフォルトのユーザーとロール</u>
- <u>特権レポートの閲覧</u>

5.2.2 特権レポートの閲覧

As a FlowForce Server 管理者とて、特にユーザーとロールの数が増えると、各ロールおけはユーザーに与えられた特権を管理すること は難しく感じるかもしれません。現在ユーザーとロールに割り当てられている全ての特権の概要を素早く確認するために FlowForce Server はお以下のレポートが搭載されています。

- 特権レポート
- ユーザーレポートの特権

これらのレポートを閲覧するコよ「管理」をクルクし、「レポート」をクルクします。

特権レポート

このレポートは、FlowForce Server 特権をリストしています。それそれの特権に関して、特権を与えられたユーザー、ませま、ロールにより継承された特権を確認することができます。



特権レポート(サンプル)

ユーザーレポートによる特権

このレポートは、FlowForce Server ユーザーをリストします。それそれのユーザーに関して、現在割り当てられている特権、および、これらのユーザーが与えられている、ませは、継承している特権を確認することができます。

ユ	ーザーレ	ポートによる特権	
	プリンシバル	特権	与えられた、またはプリンシパルにより継承。
	& Manager	ユーザーとロールの読み取り	& Manager に与えられている
	& all	独自のパスワードをセット	🤽 📶 に与えられている
	anonymous	独自のパスワードをセット	<u> 組</u> から継承
	& root	定義の実行のキュー	💄 root に与えられている
		クラスターの管理	💄 root に与えられている
		グローバル設定の管理	💄 root に与えられている
		ユーザー、ロール、特権の管理	💄 root に与えられている
		セキュリティの上書き	💄 root に与えられている
		ユーザーとロールの読み取り	💄 root に与えられている
		独自のパスワードをセット	💄 root に与えられている
		ジョブの停止	💄 root に与えられている
		フィルター処理なしのログの表示	🙎 root に与えられている

ユーザーレポートによる特権 (サンプル)

5.3 パーミッションとコンテナー

このセクションは以下のトピックを含んています

- パーミッションのしくみ
- <u>コンテナーを理解する</u>
- ユンテナーの作成、名前変更、および移動
- ユンテナーのペーミッションの閲覧
- ユテナーのトーミッションの変更
- <u>/public コンテナーへのアクセスの制限</u>

5.3.1 パーミッションの動作方法

パーミッションは、ユーザーのユンテナーへのアクセスを管理します。特権と同様、<u>パーミッション</u>は、ユーザーとユールに与えられることができます。ですから、ユーザーがロールの一部である場合、ロールに与えられたパーミッションは、ユーザーにも自動的に適用されます。

デフォルトで、コンテナーに設定されたパトミッションは、親コンテナーから継承されます。例えば、コンテナーA には、子コンテナーB かあると します。コンテナーA にアクセスするパーミッションを持つユーザーは、コンテナーB にもアクセスすることのできるデフォルトのパトミッションを持つ ことったります。しかしなから、管理者は、コンテナーの階層の全てのレベルで、全てのユーザーおよびロールのパトミッションを再定義することが できます。

FlowForce は ユーザーがエテナーと作業するとエテナー トミッションをチェックします。例えば ユーザーは必要とされる トミッションを 与えられている場合のみコンテナーのエンテンルを閲覧、おけよ 変更することができます。ジョブの実行で トミッションは評価されない ヤムか パ ーミッションの変更は既存のジョブコム適用されません。

各 FlowForce Server コンテナーのために、以下のトミッションの型を設定することができます。

コンテナー

「コンテナー」のペーミッションはユーザーが現在のコンテナー内のオブジェクトと作業できる項目を定義します。

継承	ユーザーに親コンテナー上で定義されているとおとのコンテナーにアクセスする同じ権利を提供します。	
読み取り	ユーザーにコンテナーのコンテンツをリストする権利を提供します。	
読み取り、書き込み	コンテナーのコンテンンをリストし、コンテナー内でオブジェクトを作成し削除する権利をユーザーに提供します。	
	メモ 新規構成オブシェケトの作成に成功するコよ おっよ、既存のオブシェケトを削除するコよ ユーザーはコン テナーの読み取り、書き込み パーミッションと構成の読み取り、書き込み パーミッションを与えられてしる必要 かあります。	
アクセス無し	ユーザーニンテナーーに入力する権利を拒否します、更に具体的には、コンテナーはユーザーに対して無効化済み と表示されます)。	

構成

「構成」パーミッションはユーザーが現在のエレテナー内で構成オブジェクトと作業できる項目(具体的には、ジョブと資格情報)を定義しています。

継承	ユーザーに親コンテナー上で定義されているとお、構成オブジェクトに関連した同じ権利を提供します。	
読み取り	コンテナー内で構成オブジェクトについての情報をビューする権利をユーザーは提供します(実行ステップませま)ョ ブのトリガーなど)。	
読み取り、書き込み	このエンテナー内で構成オブジェクトを変更する権利をユーザーは提供します(例えば、ジョブのトリガーの編集など)。	
	メモ 新規構成オブジェクトの作成に成功するコよ おっよ 既存のオブジェクトを削除するコよ ユーザーはコン テナーの読み取り、書き込み パーミッションと構成の読み取り、書き込み パーミッションを与えられている必要 があります。	
アクセス無し	コンテナー内で構成オブジェクトについての情報をビューする権利をユーザーマ対して否定します(更に具体的に は、構成オブジェクトはユーザーマ対して無効化済みと表示されます)。	

資格情報

このパーミッションはユーザーがこのコンテナー内で定義されている資格情報をどのように使用できるかを定義します。

継承	親コンテナー上で定義されているようニューザーに資格情報に関連した同じ権利を提供します。	
使用	このコンテナー内で定義されている資格情報を再利用する権利をユーザーは提供します。	
アクセス無しこのコンテナー内で定義されている資格情報を再利用する権利をユーザー対して否定します。		

+1-

このトミッションはユーザーがこのコンテナー内で定義されているキューどのように使用するか定義します。

継承	親コンテナー上で定義されているようニューザーゴ同じキュー権利を提供します。	
使用	ユーザーがションをこのコンテナー内で定義されているキューに書り当てる権利を提供します。	
アクセス無し ユーザーがショブをこのエンテナー内で定義されているキューイン書い当てる権利を拒否します。		

サービス

「サービス」パーミッションはHTTP リクエストィンターフェイスを介してWeb サービスとして公開されているジョブへのアクセスを定義します。 更に、ジョブがAS2 サービスを公開する場合、このパーミッション管理はジョブにより公開されているAS2 サービスへのアクセスをエストロール します。AS2 サービスの受信を参照してください。

継承	親コンテナー上で定義されているようニューザーニサービスに関連した同じ権利を提供します。
	ユーザービスにアクセスする権利を提供し、リクエストィンターフェイスからジョブを実行する権利をユーザーに 提供します。

	 メモ サービスパトミッションチェックは、エンテナー階層チェックをスキップします。このため、使用パトミッションが 与えられている場合、対応するジョブが定義されているエンテナへの読み取りアクセスを持たずユーザーが サービスを使用できる場合があります。 使用パトミッションをユーザー ▲ anonymous に与えると、サービスは、ブリックで使用することがで き、認証を必要としません。 	
アクセス無し	Webサービスとしてジョブニアクセス権利をユーザー(ネオして否定します。	

関数

ジョブ、証明書、他の構成データコルえ、コンテナーゴは関数か含まれる場合があります。これらゴは、FlowForce にデプロイ済みの FlowForce 関数、RaptorXML 関数、およびMapForce マルング、おゴは、StyleVision 変換も含まれます。

FlowForce ユーザーかジョブを作成すると、ジョブ内の実行ステップの一部は同じコンテナー、おけよ、異なるコンテナーから関数を参照する 場合かあります。「関数」 パーミッションは、ユーザーか関数をパーミッションが定義されているコンテナーから呼び出す、おけよ、参照できるかを 定義します。

例えば、「Restricted」と呼ばれる FlowForce コンテナーは複数のMapForce マンングを管理者がデプロイルと仮定します。管理 者は「関数」パーミッションを変更することにより、ユーザーがCのエンテナー内の関数を参照するかを決定できます。具体的には、コンテナー 「制限されている」上で「関数 - 使用」パーミッションを持つユーザーおけはコールはこのエンテナーから関数を参照することができます(例、 実行ステップを作成する際にドロップダウンリストから選択します)。他方、「関数 - アクセス無し」パーミッションを持つユーザー、おけよ ロー ルば制限されている」コンテナーから関数を選択することができません。

ジョブ内で関数が既に使用された後に、管理者がユーザーの関数へのアクセスを取り消すと、これらのユーザーはショブを実行することができなないます。ジョブ構成ページは、この場合、「選択されている関数を使用する、トミッションがありません」というテキスト共にメッセージを表示します。

継承	親エンテナー上で定義されているようニューザーに関数に関連した同じ権利を提供します。	
使用	ユーザーニンテナー内で定義されている関数を呼び出す、参照する)権利を提供します。	
アクセス無し	ユーザーニンテナー内で定義されている関数を呼び出す、参照する)権利を否定します。	

証明書

この トラッションはユーザーか現在のエンテナーからデンタルセキューティ証明書にアクセスする方法を定義します。詳細に関しては、<u>AS2 証</u> 明書の構成 を参照してくたさい。

継承	親コンテナー上で定義されているようにユーザーに同じ権利を提供します。	
使用	ユーザーニンテナー内で定義されている証明書を使用(参照)する権利を提供します。	
アクセス無し	ユーザーコンテナー内で定義されている証明書を使用(参照)する権利を拒否します。	

AS2 パートナー

この トラッションは どのようニューザーが現在のコンテナー内で定義されている AS2 パトナーオブジェクト にアクセスてきるかを定義しています。詳細に関しては AS2 証明書の構成 を参照してくたさい。

継承	親コンテナー上で定義されているようにユーザーに同じ権利を提供します。	
使用	ユーザーニンテナー内で定義されているAS2 パートナーオブジェクトを使用 (参照) する権利を提供します。	
アクセス無し	ユーザーニンテナー内で定義されているAS2 パートナーオブジェクトを使用(参照)する権利を否定します。	

リソース

このパーミッションはどのようにユーザーがこのコンテナーで定義済みのリソースの使用できるかを定義しています。

継承	親コンテナー上で定義されているようニューザーゴ同じリノースは関連する権利を提供します。	
使用	ユーザーニのコンテナー内で定義されているハノースを使用(参照)する権利を提供します。	
アクセス無し ユーザーにこのコンテナー内で定義されているリソースを使用(参照)する権利を否定します。		

セキュリティ

セキュリティノトミッションは現在のコンテナー内で定義されている子のコンテナーにアクセスするトミッションを管理します。

デフォルトでは、ユーザーは読み取り専用パーミッション(すなオタ、割り与えられているパーミッション、おけよ、メンバーであるローリのみ か許可されています。しかしなから、*読み取りユーザーとロール特権*を持つユーザーは、すべてのパーミッションエイリを読み取ること ができます。

継承	親コンテナー上で定義されているようにユーザーにセキューティー、関連した同じ権利を提供します。
読み取り セキュリ ティ	コンテナーの子のパーミッションをビューする権利をユーザーに提供します。
読み取りと書き込み セキュリティ	コンテナーの子のパーミッションを変更する権利をユーザーは提供します。
アクセス無し	コンテナーの子のペーミッションをビューする権利をユーザーに否定します。

5.3.2 コンテナーを理解する

名前か意味するように、エンテナーオン・ゲージされたデータです。FlowForce Server では、エンテナーは、オペレーシングシステム上のフォ ルダーと大志かに比較することができます。エンテナーは以下を含むことができます:ジョブ、資格情報、関数と他のエンテナー。エンテナーでパ <u>ーミッション</u>を設定することに以、エンテナー内のデータを閲覧およびデータニアクセスする権利を管理することができます。データをエンテナー別 に整理し、対応するパーミッションを各エンテナーのためにセットアップすることは、は、セキュリティ対策となるでしょう。

FlowForce Server の一番上のエンテナーはルト(/) エンテナーです。デフォルトでは、ルートコンテナーは次の定義済みの FlowForce Server コンテナーを含んでいます。

/public	 /public コンテナーは、FlowForce ユーザーがショブと資格情報を作成することのできるデフォルトのロケーションです。デフォルトの状態は空で、FlowForce のユーザー全てかアクセスすることができます。 /public コンテナー は次の場合デフォルトのロケーションとして使用されます: マンピングを MapForce から FlowForce Server にデプロイする場合。 Style Vision からの変換を FlowForce Server にデプロイする場合。 しかしながら、マンビングまけは変換を必要であれば、異なるコンテナーにデプロイすることができます。
/RaptorXML	このエンテナーは、RaptorXML Server かライセンスされている場合、使用することがで きます。このエンテナーは、RaptorXML Server に固有の検証と他の関数を保管しま す。
/system	system コンテナーは、FlowForce Server システム関数を含んでいます。このコンテ ナーを変更することは、奨励されません。

クトックして、コンテナーのコンテンンを確認し、Web管理インターフェイスからコンテナーをナビゲートすることができます。次のスクレーンショナは複数の構成オブジェクトを含むサンプル/publicコンテナーを表しています。

Home Configuration Log Administration Help						
Type here to search Search Recursive						
Name	Туре	Date modifi Modified b	ny Next run			
Example Job	job	2017-09-18 1 root	View log			
🔲 🙀 Example Certificate with Private Key	certificate + private	2017-09-18 1 root				
Example Certificate	certificate	2017-09-18 1 root				
my.credentials	credential	2017-09-18 1 root				
□ 🖳 🛃 Example AS2 Partner	AS2 partner	2017-09-18 1 root				
Example Mapping.mapping	MapForce mapping	2017-09-18 1 root				
Create Import Objects Move or Rename Selected Obj	ects Delete Selec	cted Objects Expo	rt Selected Objects Permissions			

サンプルFlowForce コンテナー

階層内のコンテナーコー実るコはページ上の階層リンクナビゲーションを使用してください。

(*再帰的 チェックボックスかチェックされている*場合)子オブジェクトを含む現在のエンテナー内、ませよ(*再帰的 チェックボックス のチェックが解*除されている場合)現在のエンテナー内のみでオブジェクトを検索することができます。

コンテナーは、ジョブ、デプロイされたMapForceマングませはStyleVision変換、関数、資格情報などを含みます。コンテナーを開くとオブジェクトに関する次の情報を確認することができます。

プロパティ	説明
名前	ファイルシステムのオブジェクトの名前を指定します。新規オブジェクトを作成する場合、使用されている名前 を使用することしてきない。点に注意してくたさい。
型	(資格情報、ジョブ、封当よ、関数などの)オブシェクトの型を定義します。オブシェクトの型を対応するアイコン で識別することができます:

プロパティ	説明	
	•	資格情報
	Î	(ビルトイン関数、MapForce マナングとStyleVision 変換を含む関数
	8	ジョブ
		コンテナー
	ñ	不足する構成オブシェクト。解決されてしては、依存関係を持つFlowForce Server データを トしようと試みるとこのアイコンが表示されます。 <u>不足する依存関係の処理方法</u> を参照してくたさ
		証明書、AS2 統合を参照してくたさい。
		(秘密キー付き)証明書、AS2 統合を参照してくたさい。
		AS2 パートナー(<u>AS2 統合</u> を参照)
変更された日付	オジェクト	か最後に作成された、または、最後に編集された日時を指定します
以下により変更されまし た	ボジェクト	を変更したユーザーの名前を指定します。
次の実行	トリガーに。 に、次回の	や実行されることが計画されているジョブのために、この列は、ジョブ設定内で定義されているよう 実行の日時を指定します。
	ジョブのオ <i>ะ</i> め す。	は、このガダンを使用することことり、対応するジョブの実行ログへ素早くアクセスすることができま

<u>パーミッション</u>を有すると想定し、(例えば、各部署にコンテナーをそれそれ作成する場合などカスタムFlowForce サーバーデータを保 管するナタリニ複数の追加コンテナーを作成することができます。おけよ、認証されたユーザーが使用することのできる、/public コンテナー内に データを保管することもできます。必要に応じて、/public コンテナーへのアクセスを制限することも可能です(以下を参照: <u>/public コンテナ</u> ーへのアクセスの制限)。

相対するペーミッションを有する箇所でコンテナーの移動、名前の変更、および削除を行うことができます。

5.3.3 コンテナーの作成、名前変更、および移動

(メンバーであるロールが) コンテナー/読み取り、書き込みパーミッションを有する場合、コンテナーの作成、名前変更、および移動することができます。(以下も参照してくたさい、パーミッションのしくみ)。

メモ FlowForce Server によデフォルで与えられた、RaptorXMLと/system コンテナーの内容を変更することは奨励されません。

コンテナーの作成方法:

1. 「構成」をクルクします。

- 2. 新規コンテナーを作成するために、既存のコンテナーをクリックします。階層のトップのレベルにコンテナーを作成する場合は、このボタンを無視してくたさい。
- 3. ページの左下にある「コンテナーの作成」ボタンをクリックします。

ホーム 構成 ログ 管理 ヘルプ	
□ 名前	
RaptorXML	
🗆 🛅 public	
🗆 🛅 system	
作成 ▼ オブジェクトのインポート	選択されたオブジェクトの移動または名前の変更をする
コンテナーの作成 ジョブの作成 資格情報の作成	Altova Flo

- 4. コンテナー名を入力してくたさい。以下の名前に関する制約が適用されます。
 - フィールドが空であってはなりません。
 - 空白文字で開始、おけお終了することはできません。
 - 英数字、数字、単一の空白、アンダースコア(_)、ダッシュ(-)、およびピリオト(.)文字を使用することができます。
- 5. 「保存」をクリックします。

コンテナーの名前を変更する

- 1. 「構成」、をクリックし、名前を変更するコンテナーに移動します。
- 2. ページの左下にある「選択されたオブジェクトの移動または名前を変更する」ボタンをクルクします。

別のコンラ	テナーに移動または名前を変更する	×
名前: コンテナー:	Copy to Prod	
	名前 t	
	system	
名前の変	更 キャンセル	

3. 名前ボックスにコンテナー名を入力し、「名前の変更」をクリックします。

コンテナーの移動方法:

- 1. 「構成」、をクックし、移動するコンテナーに移動します。 2. コンテナーの横のチェックボックスを選択し、「選択されたオブジェクトを移動または名前を変更する」をクックします。

別のコンき	テナーに移動または名前を変更する	×
名前: コンテナー:	Copy to Prod	
	名前 t	
	📄 system	
名前の変	更 キャンセル	

- 3. コンテナーの定義を以下をおこない選択します:
 - コンテナーテキストボックス内に ひを入力します。
- 4. 任意で、名前ボックス内にコンテナーの新しい名前を入力します。
- 5. 「移動」をクルクします。

複数のコンテナーを移動する

• 横のチェックボックスをクリックし、単一のコンテナーの移動と同じロジックを使用してくたさい。

コンテナー内の全てのオブジェクトを選択、または削除する

• 一番上のチェックボックスをクリックします。

5.3.4 コンテナーパーミッションの閲覧

関連するパーミッションを有する箇所で、コンテナーのパーミッションを閲覧することができます(以下も参照してください、<u>パーミッションのしく</u> <u>み</u>)。デフォルトでは、コンテナーに対するユーザー自身のパーミッションを閲覧することができます。ロールのメンバーである場合、ユーザーがメン バーであるロールに与えられているパーミッションを確認することができます。 ユーザーとロールの読み取りの特権を有する場合、コンテナーに 対しての他のユーザーとロールのパーミッションを確認することができます。

コンテナーのパーミッションを閲覧する方法:

- 1. 「構成」をクリックします。
- 2. 以下の方のつを行ってください
 - コンテナー記録の横の「パーミッション」ボタンをクトックします。
 - コンテナーを入力し、ページの右下にある「パーミッション」ボタンをクリックします。

	3 オブジェクトを削除しました。		
		検索語の入力	検索 ☑ 再帰
名前	型 ◆	日付は変更されま 以下により変更され 次の実行	
RaptorXML	コンテナー		パーミッション
🛛 🛅 public	コンテナー		パーミッション
🗏 🛅 system	コンテナー		パーミッション

ユーザートロール名 列は確認する権利を有する ペミッションを持つユーザーとロールを表示します。 ノ ペミッション 列は コンテナー つえし て特定のユーザー おけはロールが使用できる ペミッションの型を表示します。 例えば、下のイメージはルート(/)コンテナーのための認証済みのロール につけて使用することのできるデフォルトの ペミッションを表示しています。

/のためのパーミッション		
ユーザーまたはロール名 📚		パーミッション
A authenticated	コンテナー: 構成: 資格情報: キュー: サービス: 関数: 証明書: AS2パートナー: セキュリティ:	読み取り 読み取り 使用 使用 使用 使用 使用 使用 読み取り
/のためのパーミッション		
ユーザーまたはロール名 📚		パーミッション
authenticated	コンテナー: 構成: 資格情報: キュー: サービス: 関数: 証明書: AS2 パートナー: セキュリティ:	 読み取り 読み取り 使用 使用 使用 使用 使用 使用 該み取り

各パーミッションの空の詳細は見しては、以下を参照してくたさい、パーミッションのしくみ。

5.3.5 コンテナーパーミッションの設定

コンテナーのトミッションを以下の条件を満たす場合変更することができます

- ユーザー(おコよンバーであるローリ)が セキュリティ/セキュリティの読み取りと書き込み パーミッションを変更する箇所に相対 する親エンテナーで有する場合。例えば、「Marketing」のエンテナーの子である、エンテナー「ジョブ」のパーミッションを変更する 場合、エンテナー「Marketing」のパーミッション セキュリティ / セキュリティの読み取りと書き込みを有する必要があります。コン テナー(以下を参照)パーミッションのくみ)。
- ユーザー(おけはメンバーであるロール)がセキュリティの上書きの特権を与えられている(以下を参照:特権の作動方法)。

コンテナーのパーミッションの変更:

- 1. 「構成」をクリックします。
- 2. 以下のちのつを行ってくたさい
 - エテナー記録の横の「パーミッション」ボタンをクリックします。
 - コテナーを入力し、ページの右下にある「パーミッション」ボタンをクリックします。

ホーム 構成 ログ 管理 ヘルブ			
3 オブジェク	トを削除しました。		
		検索語の入力	検索 ☑ 再帰的
2 名前	型 \$	日付は変更されま 以下により変更され 次の実行	
RaptorXML	コンテナー		パーミッション
🔲 🛅 public	コンテナー		パーミッション
🔲 🛅 system	コンテナー		パーミッション
作成 ▼ オブジェクトのインボート 選択されたオブジェクトの移動はたは名前の変更をする 選択されたオブジェクトの移動はたは名前の変更をする 選択されたオブジェクトの移動はたは名前の変更をする	れたオブジェクトの剤除	営択したオブジェクトをエクスポート	パーミッション

- 3. 以下の方の1つを行ってくたさい
 - リストされるユーザーとコールのペーミッションを変更する場合、相対するユーザーおけココールの横の「変更」ポタンをクリックします。
 - リストされていないユーザーとロールントミッションを追加する場合、「パーミッションの追加」をクリックします。
- 4. 「パーミッションの編集」セクション内で、変更するパーミッションを有するユーザーを検索し、横のチェックボックスを選択します。 FlowForce Server 内で作成されたユーザーを検索、おけま Directory Service が有効化されている場合、ドメインユーザ ーを検索します。 FlowForce Server ペドメインユーザーをインポートする方法に関しては、ドメインユーザーとロールのインポー トを参照してください。

検索: FlowForce Server ──────────────────────────────────	
【 名前 ◆ 説明 構成:	継承
L & all FlowForce Server サービ	ごス: 継承 ~ フルアクセス
L & Alethia Alonso FlowForce Server 資格信	報: 継承 ~ アクセスなし
+1-	-: 継承 ~
開数:	総承
証明書	. 継承 ∨
Δ\$2./	(ートナー: 継承 🗸

- 5. 必要に応じて、パーミッションの相対するグループを変更してくたさい。各パーミッションの空の詳細に関しては、以下を参照してく たさい、パーミッションのしくみ。パーミッションの型を全て1度のカリックで変更する場合、「継承」、「フルアクセス」、および「アク セスなし」ボタンを使用してくたさい。
- 6. 「変更の保存」をクリックします。

5.3.6 /public コンテナーへのアクセスの制限

(トップレベルのルートコンテナーの下に存在する) /public コンテナーは、デフォルトでは、FlowForce Server 内で使用することができます。 定義済みの パーミッションなしで、 FlowForce Server すべてのユーザーが全ての場所から FlowForce Server にアクセスレ、デー ダを保管する さかの場所として、使用されます。 ですから、デフォルトでは、 /public コンテナーイコお以下の パーミッションがおます。

Permissions for /public

User or Role name 🗢		Permissions		
& authenticated	Container: Configuration: Credential: Queue: Service: Function: Security:	Read, Write Read, Write Use Use Use Read	inherited from 🛅 /	Change
🚨 root	Container: Configuration: Credential: Queue: Service: Function: Security:	Read, Write Read, Write Use Use Use Use Read, Write	inherited from & authenticated inherited from /	Change

/publicのためのパーミッション

ユーサーまたはロール名 🍨		Л	ーミッション	
authenticated	コンテナー: 構成: 資格情報: キュー: サービス: 関数: 証明書: AS2 パートナー: セキュリティ:	読み取り,書き込み 読み取り,書き込み 使用 使用 使用 使用 使用 読み取り	以下から継承 🎦 /	変更
🚨 root	コンテナー: 構成: 資格情報: キュー: サービス: 関数: 証明書: AS2 パートナー: セキュリティ:	読み取り,書き込み 読み取り,書き込み 使用 使用 使用 使用 使用 読み取り,書き込み	以下から継承 & authenticated 以下から継承 & authenticated	変更

/public בידד-סדאור מיר ביישיבי

これは、デフォルトでは、ユーザー名と、スワードを持つFlowForce Server ユーザー(つまり、 & authenticated ロールのメンバーであるユーザー)は以下を行うことができます:

• /public コンテナー内のオブシェクト(具体的には、ジョブ、資格情報、おけは他のエンテナー)を追加、変更、および削除することができます。

- /public コンテナー内で使用することのできる資格情報を再利用することができます。
- ジョブはWeb サービスとして使用できるように構成されていると想定して、/public コンテナー内に存在するジョブをWeb サービスとしてアクセスすることができます。
- /public コンテナー内で使用することのできる関数を参照することができます。
- /public コンテナー・割り当てられているいトミッションを読み取ることができます。
- メモ これらのパーミッションは/public コンテナーの子であるコンテナーにより継承されるかもしれません。通常、新規のコンテナーは親コン テナーのパーミッションを継承します。しかしなから、パーミッションは、▲ root ユーザー、ませよ、対応する特権を持つ他のユーザ ーにより、オーバーライドされる可能性があります。

/public コンテナーへのアクセスを必要に応じて制限することができます。しかしなから、このドキュメトに含まれるジョブの構成の別は、public コンテナーが存在することを想定していることに意してください。

/public コンテナーへのアクセスの制限:

- 1. ⁴authenticated ロールからこのエンテナーのドミッションを取り消します(以下を参照 <u>コンテナーパーミッションの設定</u>)。
- 2. 新しいロールを作成し、このロールを/public コンテナーヘアクセスするためコントミッションが必要な全てのユーザーに割り当てます (以下を参照: ロールの作成とユーザー(コールを割り当てる)。
- 3. 新規のロールはよ必要とされるパーミッションのみを割り当ててくたさい(以下を参照してくたさい) ユンテナーパーミッションの設定)。

5.4 パスワードポリシー

このセクションは以下のトピックを含んています

- パマワードポパンーの動作方法
- パマフードポパーの作成と害り当て

5.4.1 パスワードポリシーの動作方法

FlowForce Servers は、管理者が複雑なユーザー ペワードを管理する ペワードポリシーを使用します。 パワードポリシーとは、有効になるためにユーザー ペワードが必要とする最低条件です (例えば、最低 N 文字のなど)。

パマワードポパー内で定義することのできる、パマワードの複雑度のルールは、以下の通りです。

- パマワードの合計の長さの最小数(つまり、パマワードは、有効であるためには、少なくともN文字の長さである必要があります)。
- パマワードか含む必要かある英字の最小数
- パマワードか含む必要かある数字の最小数

(ユーザー、ロール、特権の管理の特権があると想定して)必要に応じて、パマワードポパシーを必要な数定義することができます。パマフードポパシーを定義すると、FlowForce ユーザーに割り当てることができます。ユーザーアカウストは、1度に1つのパマワードポパシーを有することができます。

ユーザーが、アワードの変更をリクエストすると、システムは、ユーザーの、アワードポリシー内で定義されたをチェックします。システムは、新しい、パスワードがユーザーのフードポリシー内で定義された。アスワードの複雑度を満たすかをチェックします。パスワードが、アワードポリシー内で定義 された複雑度の条件を満たされ、場合は、パスワードの変更は否定され、システムは関連したメッセージを表示します。

管理者がユーザーの、マワードを変更すると、FlowForce Server は、ノマワードポリシーを強制しません。おこ、ノマワードポリシーカ変更 されても、既存の、マワードは影響を受けることはありません。後者の場合、ユーザーカ既存の、マワードを変更しようと試みると、ノマワードポ リシーカ強制されます。

デフォル・では、FlowForce Server は、パワードの複雑度を強制しない空の、パワードポルシーを含む、パワードを含んでいます。 FlowForce Serverは、明示的にデフォルトの、パワードポルシーを、カスタム化されたパパワードポルシーを有さないユーザーアナウナトに書い 当てます。、デフォルトの、パワードポルシーを変更することはできません。

以下も参照してくたさい

• パマワードポパシーの作成と割り当て

5.4.2 パスワードポリシーの作成と割り当て

新規パマワードポルシーの作成:

- 1.「管理」をクリックします、「パマワードポリシー」をクリックします。
- 2. 「ポリシーの作成」をクリックします。

パスワード ポリシーの作成						
ポリシー名	: October					
パスワード ポリシー						
合計の長る	き: 少なくとも以下の数量が必要です	8	文字 💼			
英字:	少なくとも以下の数量が必要です	3	英字 💼			
数字:	少なくとも以下の数量が必要です	3	数字 💼			
保存						

3. 必要とされる なワードポパシールールを入力し、「保存」をクトックします。現在のユーザーのノストが定義されたポパシー下で使用することができます。新規ポパシーイニ割り当てられるユーザー記録をクトックし、「割り当て」をクトックします。

メンバー							
利	用可能なユーザー		ポリシーのメンバー '新規パスワードポリシー'				
	名前 💲	現在のポリシー	□ 名前				
	💄 Alethia Alonso	ទ デフォルトのポリシー					
	& Klaus Mauer	💬 デフォルトのポリシー					
	& Natsuo Shinohara	🖙 デフォルトのポリシー	·				
	& default for new user	🖙 デフォルトのポリシー					
	🚨 root	🖙 デフォルトのポリシー	<<				
	▲ 新しいユーザーのデフォルト	🖙 デフォルトのポリシー					

次も参照してくたさい

パマワードポシーの動作方法

6 ジョブの構成

このセクタンは、FlowForce Server ジョブの構成に適用することのできる手続きとこンセプトを含んできます。

6.1 資格情報

資格情報オブジェケトは、認証に関する情報を保管します。これはFlowForce Server ジョブが作動する箇所でのオペレーティングシステム上のユーザーアカウトに関連するユーザー名と「スワードの組み合わせですがHTTP おけはFTP 資格情報 おけはOAuth セキュリティ の詳細のセナーであることもできます。

証明書には以下の種類が存在します

- パマワード (ユーザー名と)マワードの組み合わせ)
- OAuth 2.0
- SSH +--

型パマワードの証明書は各ジョブで必要とされています。特定のオペンーションシステムユーザーとしてショブを実行することを可能してます。 具体的には、FlowForce Server 内でジョブを作成すると、ジョブを実行するユーザーアカウトの資格情報を適用する必要があります。 オペンーティングシステム上でユーザーアカウトに十分な権限が存在してい場合、ジョブの実行を成功することはできません。パマワード資格 情報は、FTP サーバーへの認証が必要な内蔵のFTP 関数を呼び出すためこ必要です。ファイルウォッチトリガーもパワード資格情報を 必要とします。

型 OAuth 2.0 の資格情報はOAuth 2.0 認証を必要とする個所でのWeb サービスを呼び出すジョブで必要とされます。

FlowForce Server 内では、資格情報をローカルの資格情報として参照される新規のジョブを作成する都度、おけよ、スタイアロン (再利用可能な)資格情報オブジェクトとして定義することができます。後者の場合、ジョブを作成する際に再度入力する代わりに定義済み の資格情報を参照することができます。変更する都度1つの場所で更新することができるけぬ、スタイアロン資格情報は便利です。すなわ ち、変更はスタイアロン資格情報を更新する際には、その資格情報の参照を使用するすべてのショブに影響を与えることを意味します。

OAuth 2.0 資格情報は(ローカルではな)、スタイドアロンとして定義し、その後に必要とされる個所でショブにより参照されることができます。

SSH キーはSFTP のために有効な資格情報の種類です。この種類の主なプレシッ タレはコニーグなキーのペアの使用法をベースはしています ます: 公開キーはメッセージを暗号化し、サーバーがこれを受け取り、秘密キーはこのメッセージの暗号化を解除します。SSH キーの資格情報を作成するコよ 参照をクリックして(下のスクノージンョナト参照) SSH キーを選択します。このファイルはPEM フォーマナのDSA または、RSA キーゴムります。 必要な場合は、パスレーズを提供します。

資格情報はSFTP 接続を認証するためコ使用されます。詳細に関しては、/system/sftp セクションを参照してくたさい。

下のスクレーンショナはRSAキーが資格情報セクションに既につポートされている失敗を表示しています。

Credential					
Credential type: 🔘	Password OAuth 2.0 💿 SSH Key				
User name:	test				
Public key:	ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDJSVSgvxD6/gJx1r3xKH9B 4oXB3dxjqpQCFC93AKRAIrVuqCWoGDFgQzCnMar75fpNqPqqWKWshn1l0z Ic35b2TsiZBSFuZGI+B8aem3liiSVS7nFyD4w4M8s5F1aNdiMKNWr5v+Qh8E2 654ajleKW7tGtaSVF/v7DYIFW7Ibu6NNS77c3AGLjFhgOn8Uze8YLAIH5ePT46 zKUeL143gXMoxItam5ODHPqHgXW9OtJIGkc6bl2unGquAoZbUy1jq6QApR Nb4F9UrncaculxQFwPYmE3NNwNj8fUwmxmVWZgTFrqx9vVdZ7knsdFVJGm qkHUt13gG7EJegDP7j5+nDHxG7veMROCaBqth+aRiV+7Ltr6F4LoLAFJ3ZQE M2032nQyyEwnNMfMjsOaDyBNyKSm4vs0ApFH/GFfa8PjV867wAE/IGd76Td q0ApLPGOKwIQTPBcZhWALuHAKINuBTNL+hIteTnle1PwtrDDj/anWUzvjKYu MWcqQr/zIIImC8E=				
Key type:	RSA				
Key length:	3072 bits				
Key imported file:	test_rsa.pem				
Import from file:	Browse Select file with DSA or RSA private key in PEM format with or without passphrase.				
Import passphrase:	•				
Allow usage for HT	TP:				
Allow usage for FTP):				
Allow usage for SSH/SFTP:					
Allow usage for job	execution:				

灹

- 関連するパーミッションか与えられている場合、ジョブからの資格情報をユーザーは参照することができます。資格情報を特定のエンテナーからユーザーおけはロールにアクセス可能にするけは、管理者がユーザーおけはロールに資格情報 使用パーミッションを許可する必要があります。(パーミッションのしくみを参照してくたさい)。
- オペレーティングシステムのログイン関数にクリアテキスト・パマワードが送信される必要があるため、FlowForce Server デー タベース内で暗号化を基に戻せる状態で保存することのできる、パワードとして保管されます。管理者はFlowForce Server のデータベースファイルへのアクセスが制限されていることを確認してくたさい。FlowForce Server アプリケーション データを参照してくたさい。

FlowForce Server ジョブとてマナングを実行するすめに、MapForce とMapForce Server にライセンを供与した場合、デザイン時にFlowForce Server 内のみがますではなく、MapForce 内に資格情報オブジェクトを作成することができます。MapForce 内から FlowForce Server に作成された資格情報をオプションで所属するマナングとして、おまよ、個別のオブジェクトとしてデプロイすることができます。 きます。ユーザー名およびがマワードなどの機敏なデータを保管するためにデプロイされた資格情報は必要ありません(これは任意により行うことができます)。 MapForce 内で資格情報を作成し、FlowForce Server にデプロイする詳細に関しては、MapForce ドキュメトを参照してくたさい (<u>https://www.altova.com/ja/documentation</u>)。FlowForce Server 内での資格情報の作成及び使用方法に関しては<u>資格情報の定義</u> と<u>ショブからの資格情報の参照</u>.を参照してくたさい。マノピングショブ内で資格情報を設定、おけよ オー・・・ライドするけよ マッ ピング関数内の資格情報を参照してくたさい。

6.1.1 資格情報の定義

複数のショブ全体で再利用することのできるスタイドアロンオブジェクトとして資格情報を定義することができます。

必要条件

• コンテナー-読み取り、書き込みおよび構成-読み取り、書き込みパーミッションか資格情報を保管するコンテナーで与えられていることを確認してくたさい。

資格情報の定義

資格情報オブジェクトの作成方法:

- 1. 資格情報を保管するコンテナーに移動します。
- 2. 「作成 | 資格情報の作成」をクルクします。

Credential name: Credential description:		my.credentials
C	redential	
	Credential type: Page 	ssword O Auth 2.0 O SSH Key
	User name: admin	
	Password:	show
	Allow usage for HTTP:	
	Allow usage for FTP:	
	Allow usage for SSH/SF	-TP: 🗹
	Allow usage for job exe	ecution: 🔽

資格情報フィールドは以下の通りです。

資格情報名 必須のフィールドです。FlowForce Server 内で資格情報が識別される名前です。	
---	--
資格情報の説明	資格情報についての情報を提供する追加の説明です。
------------------------------	---
資格情報の型	資格情報の種類(パスワード、 OAuth 2.0, おけよSSH キー)を選択します。 OAuth 2.0 に関する詳細は <u>OAuth 2.0 資格情報</u> を参照してくたさい。
<i>ユーザー名</i>	必須のフィールドです。ユーザーの名前はこの資格情報に関連しています。例えば、Windows オペレーティングシステム上でユーザーアカウトを識別するために資格情報が使用される場合、 Windows ユーザーアカウト名を入力してくたさい、Windows ドメイン内でユーザー名を指 定する場合、username@domainのフォームを使用します。
	HTTP おさまFTP のためご資格情報が使用される場合、これはHTTP おさまFTP ユーザ 一名である場合がみます、以下を参照)。
<i>ヽ��ワ</i> ー / ゛	資格情報のパマワードを指定します。使用される個所で、マワード無しのユーザー名のみか必要とされる場合、パマワードは空の文字列の場合かあります。
HTTP のための使用を許可する	基本のHTTP 認証を必要とする資格情報がWeb サービスを呼び出すジョブ内で参照されて いる場合このチェックボックスを選択します。
FTP のための使用を許可する	<u>/system/ftp</u> 関数を使用し FTP サーバー 接続するショブ内で 資格情報が参照される場合このチェックボックスを選択します。
SSH/SFTP のための使用を許可 する	/system/sftp/connect 関数を使用し FTP サーバー 法続するジョブ内で 資格情報か参照される場合このチェックボックスを選択します。
ジョブ実行のための使用を許可する	資格方法がオペレーティングシステムユーザーアカウトを識別する場合このチェックボックスを選択 します。実行に成功するために、資格情報を使用するジョブではすべてこの使用が有効化されて います。
	資格情報により識別されるユーザーアカウトカオペレーティングシステム上で十分な権限を有していることを確認してくたさい。例えば、ディレオリに書き込むジョブ内で資格情報か参照されている場合、ユーザーアカウトはそのディレオトリに書き込む権限を有している必要かあります。

6.1.2 OAuth 2.0 資格情報

型 パスワード の資格情報に加え、OAuth 2.0 認証詳細である資格情報オブジェクトを作成することができます。OAuth 2.0認証が必要とれるWeb サービスを呼び出す FlowForce Server ジョブ内のOAuth 2.0 資格情報を使用することができます。

ノタワード資格情報と同様の用法でOAuth 資格情報を作成することができます。<u>資格情報の定義</u>を参照してくたさい。他の FlowForce Server オブジェケト同様、対応するノトミッションが存在する場合のチューザーはOAuth 資格情報をビューしアクセスするこ とができます。ノトミッションのしくみを参照してくたさい。

OAuth 2.0 資格情報オブジェクトに関するフィールドは下にリストされています。これらの値を取得するコは(例えば、Google API Console、Facebook API、Bitbucket API などの) Web サービスプロバイダーに登録する必要がおます

リダイレクト URI	認証サーイ・がFlowForce Server(トーケノおコはエラー)に送信するURI を指定します。 このフィールドはFlowForce Server により自動的に記入されます。
	FlowForce Server が自動化フローを開始するURIを指定します。Web サービスプロ・イ ダーを使用して登録後にこの値を取得することができます。
トーケンエンドポイント	FlowForce Server がークンフローを開始するURIを指定します。Web サービスプロ・イ ダーを使用して登録後にこの値を取得することができます。

クライアント ID	クライア・トアプリケーションのID です(この場合、FlowForce Server)。Web サービスプロ ノンイダーを使用して登録後にこの植を取得することができます。
クライアントの秘密	クライア・トアプリケーションに関連する秘密です。Webサービスプロ・イダーを使用して登録後 にこの値を取得することができます。
<i>スコ</i> ープ	プロ・イダーにより必要とされる場合のケライア・トアプレケーションのスコープです。Webサービス プロ・イダーを使用して登録後にこの値を取得することができます。
トークンエンドポイント認証	OAuth 2.0 認証サーバーの多くはPOST リクエスト ヘッダーー 認証の詳細が送信されること を必要とします。これもデフォルトでドロップダウンリストから選択された値です。
	OAuth 2.0 認証サー・トーの一部はPOST リクエストのボディ内のみて認証の詳細を受け入れる場合があります。このような認証サー・トーのために POST リクエストボディ内 でドロップダウノノストから値を選択してくたさい。
アクセストークン	認証サー・イニと版されるアクセストークンです。リソースサー・・・がアクセストークンが正確で有 効であると判断した場合のみFlowForce Server ジョブは成功します。
	OAuth 資格情報を作成し、最初にこの値を手動で取得する場合、(「更新トーケン」以外のその他すべてのフィールドを記入し「認証して保存」をクルクします。
	Webサービスプロ・イダーにより設定される期間が過ぎるとこのトーケンの期限は切れます。トーケンの期限が過ぎると、FlowForce Server は「更新トーケン」値を使用して新規のトーケンを認証サー・ドーコンケエストします。
更新トークン	認証サーバーによし返される更新トークンです。「アクセストークン」の期限が切れると必要になります(下を参照)。稀なケースとしてアクセストークンの期限が切れない場合、これは必要おません。

「以下のための使用方法の許可」チェックボックスはOAuth 2.0 だけではなく FlowForce, 内のすべての資格情報に適用されます。パスワードの資格情報のために説明されている同様の意味を持ちます。HTTP のために使用する OAuth 2.0 資格情報のために「HTTP のための使用を許可する」チェックボックスが選択されていることを確認してくたさい。それ以外の場合、ジョブはランタイムエラーと共に失敗します:「資格情報は必要とされる使用方法をサポートしません」(このメッセージまけは FlowForce ログ内で表示されているようなテキストが表示されます)。

6.1.3 ジョブから資格情報を参照する方法

資格情報の定義で説明される通りご資格情報オブジェクトが作成されると、資格情報が必要とされる多種のエレテキストで参照することができます。例

- FlowForce Server か作動する箇所でオペレーティングシステム上のユーザーアカウトを識別するパスワード 資格情報か作成されます。この資格情報を複数のジョブから後に参照することができます。この例は下で説明されています。
- FTP ユーザー名と マワードを識別する パマワード 資格情報が作成されます。。
 FTP 関数を呼び出すジョブからこのような資格 情報を参照することができます。
- OAuth 2.0 資格情報が作成されます。 OAuth 2.0 認証を必要とするWeb サービスを呼び出すジョブ内でこの資格情報を 参照する可能性があります。

次のサンプルは、FlowForce Server が作動する箇所でオペレーティングシステム上のユーザーアカウトを識別する マワード資格情報を 参照する必要がある共通のケースを説明しています:

- 1. 型パワードの資格情報を資格情報の定義内で説明される通りに作成します。
- 2. 新規のジョブを作成、ませよ、既存のジョブを編集します。

3. 「資格情報」で「既存の資格情報を選択」をクリックし、定義済みの資格情報レコードを参照します。

沙貝	發格情報		
	資格情報によりジョブを実行:	● 既存の資格情報を選択: 〇 ローカルの資格情報を定義:	/ O public
7	ニュー設定		名前◆
	実行間の最短時間: 0 秒 並列実行の最大数: 1 イン	マスタンス	See <u>my.credentials</u>

資格情報レコードを含むショブが存在する場合、資格情報オブジェクト自身として参照することができます。例

資格情報		
資格情報によりジョブを実行:	◎ 既存の資格情報を選択:	/public/JobWithLocalCredentials
	◎ ローカルの資格情報を定義:	/ O public
キュー設定		名前 🕈
実行間の最短時間: 0 秒		JobWithLocalCredentials
	.7 41.7	🖙 my.credentials
	///////////////////////////////////////	

この場合、(ローカル資格情報を持つ)埋め込まれたショブの資格情報はメインショブの資格情報として使用されます。 資格情報はレンカされ ており、コピーされていません。 ローカルで定義されている資格情報を埋め込まれているショブで変更すると、 メインショブに伝達されます。

6.2 ジョブの作成

このトピックでは、FlowForce Server Web 管理インターフェイスから、ジョブを作成する方法について説明されています。一般的な手順は、使用することのできる設定とジョブの構成をよりよく理解するためのたのです。しかしなから、リストされている全てのステップおよび設定が、実際のシナリオに適用できるとお限りません。定義として、ジョブは柔軟性を持つプロセスであり、1つの結果を得るために、幾通りにの方法かあります。。ジョブとの作業により行うことのできる多種のアイディアを得るためにこのドキュメトロは他のジョブの構成の例が含まれています。

必要条件

- ジョブが作成されるエンテナーのために、以下のペーミッションか与えられていることを確認してくたさい(:ニンテナーのパーミッションの規以下を参照覧)。
 - コンテナー-読み取り、書き込み
 - 構成 読み取り、書き込み

ビナ

- ジョブを作成する前に、ジョブが実行されるのオペレーティングシステムユーザーアカウト資格情報を保管することが奨励されています(以下を参照)
- ジョブを作成する際、ジョブ間、おさは、ステップ間で値を、欠することを想定している場合、一般的なエラーを避けるために、
 FlowForce 式を参照してください。

ジョブを作成する

- 1. 「構成」をクリックし、ジョブを作成するコンテナーに移動します。
- 2. 「作成」をクトックし、「ジョブの作成」をクトックします。
- 3. ジョブ名、および、任意でショブの説明を入力します。
- 4. ジョブカランタイムコ値をパマする場合、必要とされるジョブ入カパラメーターを作成します(以下を参照: 入力パラメーターの管理)。
- 5. 実行ステップで、ジョブのステップを追加します(以下を参照: ステップの管理)。ジョブコは少なくとも1つのステップが必要です。
- 6. ジョブの最後のステップか結果を返すとまた、結果を他のジョブで使用する場合、実行の結果から戻り値の型を選択します。(以下を参照: ジョブの戻り値の型を宣言する)。
- 7. FlowForce Server で返された結果をキャッシュする場合、キャッシュの優先順位を指定してくたさい(以下を参照: <u>ジョブの結</u> <u>果をキャッシュする</u>)。
- 8. トリガーから、ジョブを実行するトリガーを追加します(以下を参照: トリガーの管理)。ジョブがWebサービスとして公開される場合、トリガーを追加する必要はありません。
- 9. ジョブがWebサービスとして公開される場合、「このジョブをHTTPを介して利用可能にする」チェックボックスをクリックします (以下を参照: ジョブをWebサービスとして表示する)。
- 10. 資格情報から、既存の資格情報の記録、おけよ ローカルの資格情報を指定します。(以下を参照: 資格情報)。
- 11. ジョブが返した結果を他のジョブで使用する場合、おけよ、Webサービスとして公開する場合、ジョブのキャッシュ設定を定義してください(以下を参照: ジョブの結果をキャッシュする)。
- 12. 任意で、ジョブのキュー設定を定義します(以下を参照: キュー設定の定義)。
- 13.「保存」をクリックします。FlowForce Server は、入力された情報を検証し、必要であれば、注目が必要なフィールドを赤で、 イライトします。

6.3 ジョブの複製

既存のジョブのコピーを必要であれば作成することができます。コピーされすジョブは(実行ステップ、トリガー、キャッシュの設定など)元のジョ ブと同じ性質を有します。ジョブの複製により時間を節約することができます。例:既存のジョブをテンプレートとして素早くジョブを作成する場 合など。

既存のジョブのコピー

- 1. 既存のショブを開き、そして、ページの下にある、「名前を付けて保存」ボタンをクルクします。
- 2. 新規ジョブの名前を入力して「名前を付けて保存」ボタンをクトックします。

名前を付けて保存	
ジョブ名: CopyInvoices2	
名前を付けて保存キャンセル	

- メモ 既存のジョブの資格情報がショブ内で、ローカルに定義されている場合、FlowForce は、セキュリティ上の理由から、もう一度パス ワードを入力するようにおいます。スタイドアロン資格情報として資格情報が定義されている場合、このステップは、必要ありません。スタイドアロンとローカルの資格情報の違いに関する詳細は、以下を参照してくたさい、資格情報。
- メモ ジョブを複製する際に、特定のショブエレポーネントか競合を引き起こす場合、FlowForce は、エラーを表示し、ジョブの複製を行いません。例えば、Webサービスを含むショブを複製しようと試みると、サービスは既に元のショブには使用中のため、複製することはできません。この場合、WebサービスのURLを変更するか、おけよ、完全に削除してくたさい。

複製されたショブは、既存のショブとして同じコンテナーに保存されます。異なるコンテナーに移動する場合は、2つのステップから構成されるオペレーションとして行うことができます:

- 1. 最初に、既存のジョブを上記のように新しい名前で保存します。ページを移動することなく同じジョブのエピーを複数作成することもできます。
- 2. 親コンテナーページン移動し、1つまけは複数のジョブを選択し、「選択されたオブジェクトを移動」をクルクします。

6.4 入力パラメーターの管理

FlowForce Server ジョブのエテキストで、入カハラメーターは、どのジョブが特定の方法で実行されるかを表したジョブに与えられた任意の精報を意味します。ジョブの入カハラメーターは、関数引数にプログラム言語の観点から類似しています。、入カハラメーターは多種の型(例え)ば、ファイルはだはディレクトリルファレンス、テキスト、番号、ブール値、その他)であることができます。一部のジョブの型に対しては、FlowForce Server は、入カハラメーターを自動的に作成します。

- 灹
 - FlowForce Server は、自動的にファイルシステムおけるHTTPトリガーを使用するショブにtriggerfile とら入カパ ラメーターを追加します(トリガーの管理も参照してください)。triggerfile パラメーターはトリガーをアクティブ化する、削除してはならない、ファイル名を含んでいます。
 - ジョブが デメーターを使用し Web サービスとして公開されている場合、FlowForce Server はリケエスト内のアプリケーションの呼び出しにより なされる デメーターを期待します(ジョブをWeb サービスとして表示するも参照してくたさい)。

入力パラメーターは以下のフィールドと構成されています。

名前	必須フィールドです。入力/ デメーターの名前を指定します。ジョブの実行ステップ、から後にこの / デメーターを参照する可能性があり、詳細を記した名前をできるけっ使用してください。 入力/ デメーター名は文字で開始し以下の文字のみを含むことができます: a-z、A-Z、O-9、お よび'_'。
型	必須フィールドです。以下のうちの1つてある入力パラメーターのデータ型を指定します: 文字列 ファイルとしての文字列 ディレクトリとしての文字列 ファイルとしての文字列まざはディレクトリ ストリーム 数値 ブーリ値 期待されるパラメーターの値のデータ型に従いデータ型を選択してくたさい。
	任意のフィールド。パラメーターのデフォルトの値を指定します。この値はランタイムごうブの呼び 出し元により値が指定されない場合使用されます。
説明	任意のフィールド。パラメーターの目的を説明します。現在のジョブを他のジョブの実行ステップとて使用する場合、この説明は、パラメーター名の横に表示されます。

以下のがたを、ラメーターを管理するために使用してくたさい。

•	ノデメーターを追加します。
	ノ・ラメーターを削除します。
(b)	パラメーターを複製します。

1	パラメーターを上下に移動します。
•	前の削除アケンョンを元に戻します。

6.5 ステップの管理

ステップは、FlowForce Server ジョブが実際に行う内容を定義します、例えば、ファイルの判除、MapForce マピングの実行、おさま 電子メールの送信など)。最も簡単な形式としては、ステップは失敗おされな功した成果を持つオペレーションで、実行されるために関数 必要とします(以下を参照:<u>実行ステップ</u>)。しかしながら、ループとして他のステップを条件付きで実行する方法を与えるステップも存在しま す(以下を参照:<u>選択ステップ</u>、エラー/成功の処理ステップ およびForEach ステップ)。同じジョブ内で必要な数だけステップを作成 し、実行される順序を設定することができます。

ステップを追加するコよ、ステップの型に対応するポタンをクリックします。

実行ステップ			
新規実行ステップ	新規チョイス ステップ	新規 For-each ステップ	新規エラー/成功処理ステップ

次のステップの型を作成することができます。

実行ステップ	型「実行ステップ」のステップにと特定のFlowForce 関数を実行することができます。 FlowForce Server にと提供されるMapForce マメニングをデブロイする おけま変換を 行う、他のショブのステップを実行する <u>ビリトィン関数</u> を含む関数を使用することができます。	
選択ステップ	型「選択ステップ」のステップにより、どの条件下で他のジョブステップが実行されるかを定義する ことができます。選択ステップには以下の構造があります:	
	When {some expression} Execute (some step) Otherwise Execute (some other step)	
	各「選択ステップ」の下では、他の選択ステップサブ条件)をネストすることができます。例	
	When {expression} When {expression} Execute (step) Otherwise Execute (step) Otherwise Execute (step)	
	定義することのできる条件ステップの数量に制限はありません。	
	When / Otherwise ペアの中では FlowForce Server は true である条件のみを 実行します。他の条件は無視されます。	
For-Each ステップ	型「For-Each ステップ」のステップにより、シーケンス中を反復し、実行ステップを無制限に繰り返すことができます(例えば、ディレクトリ内のファイルのリスト)。「for-each」ステップの構成 は以下の通りです:	
	For each item in sequence {sequence expression} Execute (step)	
	FlowForce はシーケンス式の全てのアイテムをループして完了するまでステップを実行します。	

エラー/成功の処理ステップ	ジョブのステップに失敗すると、FlowForce Server は、ジョブを中断します。型「エラー/成功の処理ステップ」のステップを使用して、ジョブを完全に完了する前に、クノーシアップアグションを
	実行します(以下を参照: <u>ステップのエラーの処理</u>)。

ステップを追加するコよ、ステップの型に対応するポタンをクリックします。

実行ステップ			
新規実行ステップ	新規チョイス ステップ	新規 For-each ステップ	新規エラー/成功処理ステップ

以下の状とをステップを管理するために使用してください。

•	ステップを追加する。
	ステップを削除する。
•	ステップを複製する。
t	上下に移動します。
•	前の削除アクションを元に戻す。

6.6 順番にステップを処理する

FlowForce ジョブを最も簡単な方法で作動するコは「実行ステップ」のシングルステップとして追加することです。このようなステップは、 FlowForce ビルトイン関数、シェルロマイ、MapForce マメピング、StyleVision 変換、RaptorXML 関数、おゴはスクリプトを呼び 出すためコ使用することができます。

複数の実行ステップをジョブに追加する場合、FlowForce は、(一番上の)最初から最後のステップを順番に処理します。このルールは、ステップ内のサブステップにも適用されます。下の図はこのンナリを表しています。



順次処理

デフォルトでは、FlowForce にエラーが発生した場合、エラーが発生した箇所で、処理は停止し、その後のステップは、実行されません。



順次処理エラー

ステップが失敗しても、ジョブ全体の実行を停止しない場合があります。この場合、ジョブの実行が停止されないように実行ステップと呼ばれる関数を構成します。その後のステップ全ては、ステップが失敗しても、通常通い処理されます。 次の関数を呼び出すステップのみにこの振る舞いを与えることができることに注意してくたさい

- <u>/system/filesystem</u>の下の全ての関数。
- /system/ftpの下の全ての関数。
- システム/system/mail 関数
- ①/system/shell/commandline 関数。

詳細に関しては、各関数(特に「エラー時はは中断」、デメーター)の説明を参照してください。

順次処理は、FlowForce ジョブを処理するための1つの方法で、ニーズに常に応えるとは限りません。高度な処理に関しては、以下を参照してくたさい

- <u>条件付きでステップを処理</u>
- <u>ステップのエラーの処理</u>

6.7 条件付でステップを処理

条件に従いステップを処理するには、型「選択」のステップを追加します。条件に従いステップを処理するには、型のステップを追加します。このようステップは次の2つの部分から構成されています。「When (の場合)」部分とOtherwise」部分。



条件付き処理を含むジョブの抽象的な表現

上のイメージで表示されている通り、「When」パトト「Otherwise」パトトは、実行ステップを必要とます。また、「When」パトトは 式を必要とます。この式は、ジョブが実行される際、2つのパトトのうちどのパトを実行するかを決定します。具体的には、「When」式 がブール値のtrueを評価すると、ステップCが実行されます。それ以外の場合、ステップDが実行されます。 抽象的なジョブの処理 のシーケンスは、以下のとおりです(実行エラーが存在しないと前提して):

- 1. ステップA
- 2. ステップB
- 3. ステプ C おは 式にり D
- 4. ステップE
- 5. ステップ F

例えば、下に示されている通り、FlowForceは式「fileexists」がブール値のtrue を評価すると、ビルトインdelete 関数を実行します。

選択			
When	fileexists		
•			
4	関数の実行	/system/filesystem/de	elete
	バラメーター:	バス:	c:\temp\myfile.xml
		エラー時には中断:	+
		作業用ディレクトリ:	•

「fileexists」は、前のステップでませる人力、デメーターとして宣言されている必要があります。例

ステップの結果を割り当て fileexists

それ以外の場合、構文検証エラーを表示します

When *fileexists*

FlowForce 式に関する概要に関しては FlowForce 式を参照してください。FlowForce ジョブ内での条件付きの処理の例に関しては、以下を参照してくたさい。例、ノンカ存在するか確認する。

0

6.8 ステップのエラーの処理

ジョブのステップに失敗すると、ジョブは失敗したとみなされます。(ログ、おけよ、電子メールの送信などのためはつジョブを完全に終了する前 に、クリーンアップのアクションを行うには、型「エラー/成功の処理」を作成することができます。「保護されたブロック」として参照されるステップを 処理するエラー/成功により、ステップの実行を保護することができます。次の図には、保護されたブロックの構成を表示しています。

保護されたブロック
ステップA
ステップ
エラー時
ス ምップ ^с
成功時
ステップ D
常に
ステップ E

保護されたブロックの構造

保護されたブロック内の1 つのステップに失敗すると、エラー・ンドラーは、ジョブか終了される前に何か起こるかを管理します。エラー・ンドラー は以下の内の1 つです:

- 成功時(保護されたブロック内の全てのステップに成功する場合に実行されます)。
- エラー時(保護されたブロック内の)つのステップに成功する場合に実行されます)。
- 再試行時(この)ンドラー保護されたブロックのカウトの再試行オプションが0よりしたきな値を設定されている場合に作動します。保護されたブロック内のステップが失敗すると再試行が行われます)。詳細に関しては下の「エラー時に再試行」セグションを参照してください。
- 常に(保護されたブロック内のステップの成功に関わらず実行されます)。

保護されたブロックが実行を完了すると、FlowForce Server は、出力をベースユナた定義されたッシドラーを実行します。例えば、上に表示される図では、保護されたブロックは、ステップAとステップBです。また、エラー処理の論理は以下の通りです:

- A か失敗すると、A、C およびE か実行されます。
- B か失敗すると、A、B、C およびE か実行されます。
- A とB に成功すると、A、B、D およびE か実行されます。

実際には、処理する各ジョブのためいコンすべての、レドラーの型を定義することはできますが、必要はありません。最も一般的なシナリオは 「エラー時」と「常に」、ハンドラーのみを定義することです。例えば、下のイメージは、「エラー時」と「常に」を使用した単純型の保護さ れたブロックを表示しています。

+							
4	関数の	実行 /sy	stem/sh	ell/commandlin	e		
	パラメ	-タ-:	コマン	۲ ۲ :	script.cmd		
			エラー	時には中断:			
			作業用	ディレクトリ:	C:\sc	ript	
_	フテッ		を割り	47 output		×	以下として結用
+	~/ 9	ノリカ相来	28178			<u>^</u>	以下として相来
4	エラー	/成功処理	Eにて実	行			
	+						
	エラー	·時		✓ do			
	+						
	-	関数の実	€行 ∕sy	stem/mail/send			
		パラメーター:	ラメーター:	差出人:		flowforce@localhost	
			宛先:		to@exampl	e.org	
				件名:		The job inst	ance{instance-id()}has failed
				本文:		+	
				添付:		+	
				エラー時には中	中断:	+	
	=	ステッフ	プの結果	を割り当て 名詞	前		以下としてブール値
	•						
	常に			✓ do			
	+						
	4	関数の実	€行 ∕sy	stem/shell/com	mandl	ine	
		パラメー	-タ:	コマンド:		log_mess	age.cmd
				エラー時には中	中断:	+	
				作業用ディレク	עלל	C:\script	

最初のステップは、<u>\system\shell\commandline</u>関数を呼び出して、C:\scripts ディレオトリからスクリプトを実行します。この ステップの実行は2つの、レベラーにお保護されています:「エラー時」と「常に」。「エラー時」、ハベトラーは最初のステップに失敗するよ リガーされます。具体的には、最初のステップに失敗すると、エラー処理ステップが件名ライン内に失敗したごもブイレスタンスのIDを含む電 子メーリを送信します。「常に」、ハベトラーは最初のステップの成功に関わらず無条件で実行されます。この、ハベトラーはメッセージをthe C: \scripts ディレクトリカらのスクリプトを実行してログします。上記に類似したサンプルに関しては、エラー処理をジョブに追加するを参照してくたさい。

6.8.1 エラー時に再試行する

実行が何らかの理由で失敗した場合、ステップをもう一度実行する必要かある場合かあります。例えば、ジョグがオンラインファイルからのデータ を入力として取る場合にや、トワークがダウス、ままは、サービスがジョブが作動する時に使用できないサースかあります。エラーの場合に再試行 を可能にするかかに下していたのでを Server は 再試行のカウト オプシュンを提供します。デフォルト ではオプションは0 です。これは保護さ れたブロックが一度のみ実行され、再試行が行われないことを意味します。0 よりたきな値にこのオプションを設定すると保護されたブロック 全体の実行が可能な限り試みられます。各再試行で FlowForce し前の結果を破棄し、保護されたブロックか最初から実行されます。再 試行の最大数に到達しても、保護されたブロックがエラーに遭遇する場合、ジョブは失敗したものとしてみなされます。エラーイお外部のブロック に伝達され、再試行の最大数に到達し、保護されたブロックの実行が失敗したことがレポートされます。

次の例を考慮してくたさい

保護されたブロック
ステップ A
ステップ B
エラー時
ステップ c
再試行時 ステップ D
再試行時
ステップ E
常に
ステップ F

上の図は2つの保護されたステップには構成されるジョブについて説明しています(具体的ははエラー処理はステップAとBにとって有益です)。「再試行カウントの設定」オプションは21記定されています(これは最初の実行が対功したい場合、2度再試行が行われることを意味します)。

ジョブカ最初に実行されると、B ステップか失敗すると仮定します。これはステップA、B、F か実行されることを意味します。再試行か2回残っているため、エラーはおき報告されていません。

2回目のトライで、B かもう一度失敗します。今回はステップA、B、D とF か実行されます。唯一の違いば 再試行時」 ブロックも実行されていることです。

最後に3番目(そして最後)の再試行ではBIコモラーは発生しません。今度はステップA、B、D、E、およびFか実行されます。ジョブの結果は成功としてみなされます。

ステップBが度目のトライで失敗すると、ステップA、B、C、D、およびFが実行されるはずでした。ジョブの結果は失敗としてみなされます。

再試行が実行される都度 (A とB の両方ではなく)ステップB のみを実行する必要がある場合、ステップB を保護されたブロックで囲み、 再試行のカウント オブンョンをこのレベルのみに設定します。外部の保護されたブロックは標準の再試行カウント = 0 を持つことができます。

FlowForce ログ内でしポートされるジョブインスタンスの再試行の数値。更にランタイムでの再試行の数値を取得し処理する必要かある場合、<u>retry-count</u>式関数を呼び出します。最も内部の保護されたブロックのために発生した再試行の数値を返します。

エラーが発生した場合複数回トライされる実行ステップを追加する方法:

1. ジョブ構成ページで新規のエラー/成功処理ステップをクトックします。

E	Execution Steps							
	new Execution step	new Choose step	new For-each step	new Error/Success handling step	new Postpone step			

2. 「エラー/成功処理を使用して実行」のセグションに再トライの回数を入力します。

•						
	$\ensuremath{Execute}$ with error/success handling - on error, retry	3	times	(†) U	۲	

3. 「エラー/成功処理の実行」セクションの下の ・ ボタンをクリックしてエラーの場合再トライされる実際の実行ステップを追加しま す。下で説明されているサンプルではFTP サーバー こアイルをアップロードするショブはエラーの場合3回再トライされています。こ れは現在のショブの外部で定義されナジョブで、関数のようゴ呼び出されます。呼び出されナジョブの実装の詳細はここでは関連性 がありません。他の実行ステップは同様に再トライされることができます。

+	
	Execute with error/success handling - on error, retry 3 times
	•
	Execute function /public/UploadFileToFTP
	= Assign this step's result to name
	•
	On error 🔹 do
	•
	new error/success handler

4. 必須の「エラー時」セクションの下の や ボタンをクトックしてエラーの場合呼び出される実際の実行ステップを追加します。このサン プルでは、 log ファイルへ書き込むジョブはエラーの場合呼び出されます。 関数のようご呼び出される外部ジョブで、実装の詳細は 表示されません。 エラーログの基本情報は、ジョブニエラー処理を追加するを参照してください。

+		
	Execute with error/success handling - on error, retry 3 times	
	•	
	Execute function /public/UploadFileToFTP	▼ 🗗
	= Assign this step's result to name	
	\bullet	
	On error 🔹 do	
	•	
	Execute function /public/LogError	▼ 🗗
	= Assign this step's result to name	
	•	
	new error/success handler	

5. オプションで「成功時」「再トライ時」「常に」を追加します。このような追加ハンドラー型を構成するコは新規のエラー/成功ハンド ラーをクリックして、リストからハンドラー型を選択し ・ をクリックして必要なステップを対応するハンドラーの下に追加します。

6.8.2 実行する際の注意点

保護されたブロックは複数のステップを処理する場合かがます。複数のステップを処理するはよ、保護されたブロック内の他のステップの後に 追加します。構造は関しては、保護されているブロック内のステップは、標準の処理されていないステップと同様です(例えば、関数を実行、 式の埋め込み、ループを作成することなどができます)。一部の場合、保護されたブロック内のステップは以下で説明される通り特別な処理を 必要とする場合かがます。

例外ハンドラーは複数の例外ステップを含むことができる点を最初に考慮してくたさい。例えば、1つのステップはファイルを生成し、他のファイ ルは生成されたファイルを変換し、3番目のファイルは変換されたファイルを電子メールとして送信します。これは有効な構成です。また、複数 の実行ステップを持つハンドラーは複雑性を追加し、エラーハンドラー内でエラーが発生する可能性があるため、慎重に扱われる必要があります。

同じハンドラー内に複数のステップが存在する場合、全てか実行されるまで、シーケンス順に、おけよ、ステップが失敗するまで実行されます。ステップの失敗後はステップは実行されません。しかし、ハンドラーに失敗すると、存在する場合外側のハンドラーにより出力が処理されます。

この点に対応するオータコニハンドラー内のステップの数を制限し、エラーの可能性を制限する必要があるかもしれません。ハンドラー内部のステップが少ないようが、ハンドラーの実行が完了する可能性は高いです。後続するステップが前のステップに依存する場合、例えば、このステップのみのオータンニンデーができます。

更に留意する点は例外ハンドラーから保護されたブロック内の結果を参照することができない点です。理由は、ステップか失敗すると、保護済 みのブロックは、定義されていないもの」なり、定義されていない、結果を処理することは不可能がからです。 このため、例外ハンドラーを持つ保護されたブロック内に例外ステップが置かれると、例外ハンドラー内の実行ステップの結果にアクセスすることは不可能にないます。ハンドラーが「成功時」、「エラー時」ませは「常に」であることは関係ありません。

保護されたブロック内のステップのためにのみ制限を適用することができます。この点をよりよく理解するために、以下のサンプルジョブをよく考慮してくたさい

実行	ステップ						
+							
	関数の実行 /system/compute						
	パラメーター: 条件式: 'I am step 1'						
=	ステップの結果を割り当て result1 × 以下としてTO						
+							
	エラー/成功処理にて実行						
	•						
	▲ 関数の実行 /system/compute						
	パラメーター: 条件式: 'I am step 2'						
	= ステップの結果を割り当て result2 × 以下としてTO						
	◆ ▲ 関数の実行 /system/compute						
	パラメーター: 条件式: 'I am step3'						
	= ステップの結果を割り当て result3 × 以下としてTO						
	•						
	エラー時 > do						
	•						
	▲ 関数の実行 /system/compute						
	パラメーター: 条件式: 'I am error handler step'						
	= ステップの結果を割り当て result_handler 以下としてTO						

上記で説明されているジョブでは、各ステップに結果が存在します。例えば、最初のステップには、result1 が、2番目のステップには result2 が存在します。上記のジョブでは、各ステップに結果が存在します。これらの結果の値にアクセスする必要がある場合は、以下の点 に注意してくたさい

ステップ1は保護されたブロックの外部であるため、後続する他の捨てべのステップにより結果にアクセスすることができます。具体的には、result1にステップ2 おけはステップ3、およびエラー・ハドラーからアクセスすることができます。

- ステップ2と3はいドラーからではよく同じ保護されたブロックの後続するステップからのみ結果にアクセスすることのできる保護されたブロックの内部に存在します。すなすち、result2はステップ3からアクセスすることが可能ですが、エラー・いドラーからアクセスすることはできません。
- 最後に、ハンドラー内に存在し、ハンドラーの唯一および最後のステップであるため、エラー・ンドラーの結果である result_handler には他のステップからアクセスすることはできません。後続するステップが存在する場合、result_handler 結果を消費することができます。

上記に留意して、保護されたデザインは結果の可視性を構成しています。実際のノリューションは場合により異なります。例えば、ファイル名を作成するために「エラー時」ブロック内のステップの結果が重要な場合、自身の保護されたブロック(ネストされ保護されたブロック)の内 部に含むことができ、failed-step 関数を呼び出して、エラーを含む出力を取得し、ファイルを作成することができます。エラーを引き起こしたステップを識別することはできませんが、この関数は、エラーが発生した場合エラー情報を含むresult型を返します。result はンエル コマインを作動する抽象的な結果を表し、MapForce 変換、おけよ、StyleVision スタイルシートは期待する出力を作成する場合だけ ではなく、エラーが発生した場合でも処理されることができます。これは、サンプルによい理解することができ、エラー処理をジョブに追加するで詳細に説明されています。

6.9 延期されたステップ

エラーか発生しないことを前提として全ての処理ステップが完了した後のみらこFlowForce ジョブは結果を返します。実行されているステップが存在する限り、ジョブは結果を返す前に完了するまで待つ必要があります。ジョブがWebサービスとして公開されることは、HTTPトラン ザブションがジョブの実行中に開かれたままであることを意味します。これは処理されたデータの量により異なり、数分、おけは数時間かかる場合があります。

このようなケースを正常に処理する」コは 延期されたこステップと呼ばれる FlowForce の特別な型を使用することができます。処理はジョブが 結果を返した後のみに発生するため、延期されたステップはジョブが結果をすくに返すことを回避しません。(このため「延期された」ステップと 呼ばれます)。延期されたステップがジョブに含まれる場合、FlowForce は最初に延期されたステップではない全てのステップを処理し、延 期されたステップの処理を行います。

延期されたステップを持つジョブは結果を早く返すかもしれませんが、ジョブは延期されたステップの全ての実行か完了するまで実行中とみなされます。全ての延期されたステップの処理に成功した場合のみジョブが成功したものと考えられます。

延期されたステップをジョブ内でステップが許可されている個所で作成することができます。これを行うには、新規のジョブを作成するには既存の ジョブを開き、「実行ステップ」の下の「新規の延期ステップ」ボタンをクリックします。

実行ステップ				
新規の実行ステップ	新規の選択ステップ	新規の For-each ステップ	新規エラー/成功処理ステップ	新規の延期ステップ

これは新規の「延期された」シーケンスをジョブ構成ページに追加します

+				
	延期	 • • 	۲	۲
	•			

「延期された」シーケンス内の 한 をクトックして延期される実際のステップを作成します。 以下のサンプルのように延期されたステップがお互い しな存する場合、 複数の延期されたステップを作成することができます。 延期されナシーケンスをネスト することができますが、 その場合、 実行 の順序は重要しています。 下の「 高度なシナリオ」を参照してください。

脚注:

- 延期されナシーケンスは他のショブと同様に現在のショブ内にあり、ジョブと同じ構造を持つことかできます。例えば、ループ(おけよ 部分的なループ)おけは(保護済みの) エラーの処理ブロックを含むことができます。
- ジョブは複数の関連しない延期されたシーケンスを含む場合かあります。これはエラーの処理のために重要な場合かあります。エラー か延期されたシーケンス内で発生する場合、他に場ぐ響を受けません。詳細に関しては以下の「エラーの処理」セクションを参照して ください。
- 全ての延期されナシーケンスはすべての延期されていないステップの後に上から下に実行されます。特別なケースのために下の「高度なシナリオ」を参照してくたさい。
- メインのジョブか結果を返した後実行される呼び出されたサブショブ内でステップを延期することは可能です。

例: 延期されたステップ

次のサンプルは延期されたステップの使用例について説明しています。Webサービスとして公開されており、ブラウザーを含むクライアントをいっても呼び出せるため、このジョブはよい例です。このようなジョブの詳細は<u>Webサービスとしてジョブを公開する</u>を参照してくたさい。

実行	スティ	ップ		
	延期			
	•	関数の実行 /sy	stem/shell/commandlin	e v
		パラメーター:	コマンド: エラー時には中断: 作業用ディレクトリ:	dir"C:\Program Files" /S + +
	=	ステップの結果	を割り当て output	× 以下として結果
		関数の実行 /sy	/stem/filesystem/copy	• 2
		パラメーター:	ソース: ターゲット: 上書き: エラー時には中断: 作業用ディレクトリ:	{as-file(stdout(output)} C:\FlowForce\Postponed\output.txt + + +
	-	ステップの結果	を割り当て名前	以下としてブール値
•				
1	関数の パラン	D実行 /system/co メーター: 条件式	ompute task has been subm	nitted successfully'
=	ステッ	ノプの結果を割り	当て名前	以下としてTO

ステップAは大きなシステムディレクトリ内のすべてのディレクトリとファイルを再帰的にリストする時間がかかるシェルロマンドを実行します。この 理由からステップAは延期された」シーケンスの内側に置かれています。ステップBはAに以生成された標準の出力をファイルに書き込み ます(stdout)。ステップBはAに以生成される出力に以異なり、延期されたシーケンスの一部となる必要があります。

ステップ C はサービスのコーラーにタスクの送信が成功したことを通知します。このステップはショブが実行されると最初に実行されますが、意図的に最後のものとして追加されます。

具体的にはWebサービスが呼び出されると、上のステップが以下のシーケンスを実行します:C*、A、B。理由はAとBは延期されたステップでCが最初に実行されるからです。アスタリスケマークはショブが結果に戻る場所を示しています。

この構成の利点はステップ C の実行後ジョブがすくに結果を返し、HTTP トランザブションが終了できるけっか、他のリスエストのけっかにサーバ ーリハースを空けることができるからです。ジョブの結果を返した後、FlowForce は延期さえたステップ A とB を通常のシーケンスで実行します。

確認として、プラウザー内でショブを呼び出すとショブメッセージ「タスクの送信に成功しました」」が遅延なく表示される一方、output.txt ファイ ルが作成されるまでショブは実行を継続します(リストされたディンケーリのサイズにはにれば数秒から数分かかります)。A おけまB 失敗しな し場合、出力ファイルは C:\FlowForce\Postponed\output.txt で作成されます。 メモ このサンプルでは、具体的には文字列「タスクの送信に成功しました」Webサービスをテストする際にプラウザーに送信される結果を生成するかがステップCliおうで最後に実行されます。Cを上に移動する場合でも、最初に実行され、延期されたステップ Bliおうびの最後に実行されるステップのままです。しかしながら、これはぶうびの結果を変更し、プラウザーはIIIに類似した空の出力を表示する可能性があります。FlowForceジョブの結果は常に最後に実行されたステップの結果であるからです。延期されたステップは値を返す必要はありませんが、代わりにからのシーケンスを生成します。

高度なシナオ

ジョブは複数の延期されたステップを含みますが、前後に配置されている、おとは延期されたステップの複数のシーケンスである必要におりません。例えば、ジョブは以下の順序で以下のステップには、構成されているとします

A postpone B C **D を延期**

上のステップは以下の順序で実行されます:A、C*、B、D。確認できるように、延期されていないステップが最初に実行され、その後に延期されたステップが実行されます。アスタリスクマークはジョブが結果に戻る場所を示しています。

条件(choose ステップ)内で延期されたステップを追加することもできます。この場合、延期されたステップは対応する "when" おけよ "otherwise" ブランチが実行されると実行されます。

when expression=true
{
 postpone A
 B
 C
}
otherwise
{
 postpone D
 E
 F
}

上記の曖昧なジョブの場合、式がtrue を評価すると、ステップは以下の順序で実行されます: B、C*、A。それ以外の場合実行順序は 以下の通りないます: E、F*、D。アスタリスクはジョブが結果を返すことを示しています。

for each ステップ内では ループの一部であっすとおりループと同じ順序で延期されたステップは延期されていない ステップとして処理されます。



例えば、ループか8回実行されると仮定すると上記のステップは以下の順序で実行されます: A1、A2、A3*、B1、B2、B3、箇所:

- 数値はレープの番号を示しています
- アスタリスクはショブが結果を返すことを示しています。

他の延期されたステップ内で延期されたステップをネストすることもできます。この場合、ルールは以下の通りです

- 同じ深さの外側のステップか最初に処理されます
- ネストされた延期されたステップは親シーケンスが完了した後のみに実行されます。

例えば ジョブは次のネストされているステップによく構成されていると仮定します



上のルールに従うとステップは以下の順序で実行されます:A、G、N*、B、D、F、C、E、H、K、M、J、L。アスタリスクマーグはションが結果 に戻る場所を示しています。

上のように高度な構成を作成しテストする必要がある場合、FlowForce ログからステップの実行ステップをトラックすることが可能です。

エラーの処理

以前に述べられているととおり、延期されたステップは延期されたシーケンスに属しています。ジョブ内の複数の箇所で複数の延期されたシーケンスの含まれている場合かあります。エラーか発生すると、一般的なルールは以下の通りです:

- 延期されたステップがエラーに遭遇すると、同じ延期されたシーケンス内の次のステップと共にキャンセルされ、エラーがログされます。これは現在のシーケンスのみに影響を与えます。同じジョブ内に他の延期されたシーケンスが存在する場合、実行は継続されます。
- 保護されたブロック内の延期されたステップがエラートご遭遇すると、通常これはブそのブロックの一部であるすべての延期されたステップをキャンセルします。保護されたブロックはエラーが発生した場合複数回再試行するように構成されていることを考慮すると、これにはロールレックの役割を果たし必要です。エラー時に再試行するを参照してくたさい。

上記の最初のルールの作動を説明するココお以下のサンプルを確認してくたさい

実行	「ステ	ップ	
+	延期		
A	•	関数の実行 /system/compute パラメーター: 条件式: as-file(stream-from-string('The quick brown fox'))	
	=	ステップの結果を割り当て output 以下としてTO	
В	•	関数の実行 /system/filesystem/copy {output} for ソース と C:\FlowForce\Postponed\demo.txt for ターゲットと。	. ◄ 🖻
	延期		
C	•	関数の実行 /system/compute 'All postponed steps have finished' for 条件式と。	• 2
D	関数の 'Job h	実行 /system/compute Tas been submitted' for 条件式と。	• 2

上記のジョブでは唯一の延期されていない ステップのためステップ D か最初に実行されます。

ステップD 以外にも2つの延期されナシーケンスが存在します:最初のシーケンスリコはAとB か含まれており、2番目のシーケンスリコはステップC か含まれています。ステップA か失敗したと仮定します。この場合、FlowForce Server は延期されナシーケンス全体をキャンセルし、ステップB もキャンセルします。ステップC を含むシーケンス よ影響を受けず、実行を継続します。

6.10 ジョブの戻り値の型を宣言する

ジョブの結果を他のジョブで使用する場合、ジョブの結果をキャッシュする場合、ジョブの戻り型を宣言する必要かあるかもしれません。戻り植の型を宣言する場合は、以下の点を考慮してくたさい

- 戻り値の型は、結果を実際に返すジョブのためコは意味があります。
- ジョブの結果をキャッシュする場合、戻り値の型を宣言する必要があります。
- ジョブの戻り値の型は、ジョブ内の最後のステップの同じデータ型である必要かあります。ジョブの最後のステップが、ジョブのために宣言されたものと異なるデータの型を返すと、FlowForce Server は、「型と結果は一致しません」に類似したエラーを返します。このような型の一致に関するエラーが生じると、式関数を使用して、ジョブ内の最後ステップのデータ型をジョブの戻り値の型として宣言されたデータ型に変更します(以下を参照: データ型をステップで処理する)。

ジョブの戻り値の型を宣言します

- 1. 新規ジョブを作成、おけば既存のファイルを編集のために開きます。
- 2. 実行の結果セクション内の戻り値の型のを選択します。

実行結果	
戻り値の型を以下と宣言する: ストリーム ▼	

使用することのできる戻り値の型は、以下のとおりです

<i>無視'破棄</i>	これはデフォルトの戻り値の型です。これは、ジョブの結果が存在すると想定して、 FlowForce Server にジョブの結果を無視、おけよ、破棄するようこ命令します。ジョ ブか結果を返ざみ、場合、おけよ、返された結果を処理する必要がない場合このオプショ ンを選択してくたさい。
文字列	ジョブによし返されるデータが、テキスト値(文字列)を示すことを指定します。
<i>X</i> + <i>Y</i>	ジョブによし返されるデータが、ストリームを示すことを指定します。
数值	ジョブによし返されるデータが、数値を示すことを指定します。
ブール値	ジョブによし返されるデータが、ブール値であることを指定します。

6.11 データ型をステップで処理する

結果をステップから他のステップ、おけよ ジョブコ なすると、ステップの結果のデータ型を変更する必要かある可能性かあります。 例えば、ディレノトリのコンテンンをリストするステップを作成すると、戻り値の型か結果にないます。 この結果を特定の方法で処理するには、(例えば、電子メールに送信する、おけよファイルに書き込む場合)結果をストリームおけま文字列に変換する必要があります。

ステップの戻り値の型(ステップ1と呼ぶことします)を以下のように変更することができます:

- 1. ステップ1の「このステップの結果を割り当てる」フィールドに値を入力します。ステップの結果にアクセスするために、この値を参照する必要があります。
- 2. ステップ1の後に新規の実行ステップを追加します(ステップ2と呼びます)。
- 3. ①system/compute ビルト イン 関数を 関数 ステップ 2 の 実行 関数 とて 設定します。
- 4. ステップ2の式フィールド内では、ステップ1 により返された値を必要とされるデータ型に変換する式を入力します。例えば、ステップ1 に返された値が、出力を返すコマイドラインの結果で、これをストリームに変換する場合は、以下を使用してくたさい

stdout(result)

結果は、ステップ1には返された値に付けられる名前です。

使用される式は、達成を目的とする結果により異なります。FlowForce Server は、演算子と組み合わせ、ステップ内で使用する有益な式を作成することのできる。多種の式関数を提供します。使用することのできるオプションを理解するコよ、以下を参照してください、ステップ結果関数。

例

デルノトリのコンテンパとリストする実行ステップを以下のように作成したと想定しますよ、以下の通りです

実行	ステップ			
4	関数の実行	/system/shell/comma	ndline	
	バラメーター:	コマンド:	dir	
		エラー時には中断:	+	
		作業用ディレクトリ:	c:\	
=	ステップの 結界	そまり当て 名前		としてresult

この段階では、ステップの出力は、結果です。(例えば、ディレクトリコンテンンをファイルに書き込む、おけよコンテンンを電子メールとして送信するなどこの結果を他のステップで使用するけよ、結果ではなく、型ストリームの値が必要にないます。

「ストリーム」データ型にステップをリストするディレクトリの出力を変換する

1. ステップ1 の結果を as Step1Output とて宣言します。これを行うには、「このステップの結果を割り当てる」フィールド内 にStep1Output を入力します。

- 2. ①system/compute 関数を呼び出す新規の実行ステップを追加します。
- 3. パラメーターに、次の式を入力します: stdout (Step1Output)、step1output がステップ1の結果である箇所.

実行	ステップ	
+		
4	関数の実行 /system/shell/commandline	▼ 🛃
	バラメーター: コマンド: dir エラー時には中断: 作業用ディレクトリ: c\	
=	ステップの結果を割り当て Step1Output として result	
4	関数の実行 /system/compute	• 🛃
	バラメーター: 条件式: stdout (Step1Output)	
=	ステップの結果を割り当て名前 としてTO	
新規	の実行ステップ 新規の選択ステップ 新規の For-each ステップ 新規エラー/成功処理ステップ	

この段階では、ジョブ出力データの型は、意図としたとおり、ストリームです。ステップの2番目の結果を電子メールに送信する新規の実行ステップを作成することができます。

電子メールにディレクトリのリストを送信する

- 1. stdout (Step10utput) から content (stdout (Step10utput)) にステップの式を変更します。これを行うことに り、ストリームの値を文字列の値に変換することができます。
- 2. ステップの結果をStep2Outputと宣言します。これを行うには「このステップの結果を割り当てる」フィールド内に Step2Outputを入力します。
- 3. 新規の実行ステップ(ステップ3)を次の設定と共に追加します(<u>電子メールサーバー設定</u>はビルトイン^① /system/mail/send 関数を使用する前に構成されなければならない点に注意してください):

実行関数	次を参照します: ①/system/mail/send 関数。
差出人	電子メールの差出人を入力します。
宛先	電子メールの宛先を入力します。
件名	電子メールの件名を入力します。 例えば、「Directory Listing」。
Body	Set to ト をクリックし、Step2Ouput を選択します。

この段階では、実行ステップは以下のようことります

	/system/shell/comm	andline		▼ 🛃				1	t
バラメーター	: コマンド: エラー時には中断: 佐業田ギィックロリ	dir +			文字列 (必須)	セットする値 🕨			
ステップの紙	「F来用」 10 JF 9. 調果を割り当て Step10	utput	としてresult		大子列(パラジョ.	7) ゼット9る旭 🖡			
関数の実行	/system/compute			• 2			()	•)
バラメーター	: 条件式: stdout(Step1Output)			条件式T0 (必須) セットする	値 ▶			
ステップの 編 関数の 実行	s果を割り当て Steo2C	utput	としてT0	• @				(†) (I	L
バラメーター	: 差出人:	somone@exam	iple.org		文字列 (必須)	セットする値 🕨			
	宛先:	somone@exam	iple.org		文字列 (必須)	セットする値 🕨			
	件名:	Directory listing	J		文字列 (必須)	セットする値 🕨			
		{Step2OUtput}			文字列 (オプション	イ) セットする値 🕨			
	本文:				.4				
	本文: 流付:	+			.4				

4. ジョンを1度おけまに期的に実行するタイマートリガーを作成します(例えば、毎60分)。

۲	リガー	
	実行	毎日 🔄 毎 1 日
	繰り返し	毎 60 分 🥥 全日、または 🔘 開始時間 🛽 08:00:00 終了時間 🕒 20:00:00 💼
	開始:	 2016-07-20 08:00:00
	有効期限:	•
	タイムゾーン:	Europe/Berlin
	☑ 有効化る	されました
	新しいタイマ	ー 新規ファイル システム トリガー 新規 HTTP トリガー

5. 「保存」をクルクします。

6.12 Windows ネットワークパスの参照

ジョブを作成する際、通常 FlowForce Server が作動するマシン上で、おけよ、やトワーク上のファイル ジを参照します。マップされたネットワーケドライブなどの、Windows ネ・トワーク・ジを参照する場合、Universal Naming Convention (UNC)構文を使用します。これは、ドライブ文字は、システムに対してグロー・ジレではないため、おっ、各ログインセッションは自身のドライブ文字に書い当てられているため、これはドライブ文字がシステムに対してグロー・ジレではなく、ログオンセッションが自身のドライブ文字に書い当てられているため、

UNC 構文は以下のフォームをとます

\\server\sharedfolder\filepath

該当する箇所

- server が(DNS によに定義されるマネトワーク内のサードー名を参照する箇所.
- sharedfolder が管理者にお定義されるラベルを参照する箇所例えば、admin\$が通常オペレーティングインストールのルートディレクトリである箇所)。
- filepath がエアの下のサブディノトリを参照する箇所。

6.13 トリガーの管理

ジョブを作成する際、トリガーとしてジョブをトリガーする条件(おけは基準)を指定する必要があります。 FlowForce Server は、定義されたリガーを監視し、トリガーの条件が満たされるとショブを実行します。

同じジョブのために、複数のトリガーを作成することができます、また、定義されたトリガーを有効化、または、無効化することもできます。有効化されたトリガーが実行されると、FlowForce Server は、ジョブの全てのステップを実行します。

ノラメーターを持つジョブ内出トリガーを使用する場合、全ての、ラメーターはデフォルトの値を必要とします。それ以外の場合、ジョブは実行 されません。

以下の種類のトリガーを作成することができます。

- タイマートリガー
- ファイルシステムトリガー
- HTTP トリガー

タイムトリガーを使用することにより、ジョブが特定の時刻に(毎日、毎週、おとは週の特定の日に、おとは他の日に実行されるなどの)柔軟 性に富んだ再帰オプションを使用して、特定の時間の間隔に開始される計画を立てることができます。

ファイルシステムトリガーを使用することにより、新しく追加されたファイルおけま変更されたファイルなど、ファイルおけまディレクトリの変更を監視 することができます、削除されたファイルは監視できないことに注意してください。(例えば、60秒ごとて実行など)変更の収集の間隔を柔軟 し構成することも可能です。また、任意でトリガーの開始と終了の日付を設定することもできます。ワイルドカードを使用して、ディレクトリの特 定のファイルをフィルターすることも可能です。

HTTP トリガーを使用することにより、URI (Uniform Resource Identifier)の変更を監視することができます。特に、Last-Modified と Content-MD5 HTTP ヘッダーフィーリドの変更を収集することができます。(例えば、60 秒ごとこ実行など)変更の収集 の間隔を柔軟に構成することも可能です。また、任意でトリガーの開始と終了の日付を設定することもできます。

トリガーを追加するコよトリガーの型に対応したがみをクリックします。



下にしてされているとしが一の横にあるためを使用していが一を管理してください。

	トリガーの削除
6	トリガーの複製
•	前の削除アクションを元に戻す

トリガーフィーリドの一部はは 👥 と 🔍 ポタンか横に表示されています (例えば、タイマートリガーの開始日)。 これらのポタンを使用して トリガーフィーリドの値を設定おけおクリアします。 値は ページに表示されると設定済みとみなされます。 例えば 下のイメージでは、「繰り返 し」の値は設定されていませんが、「開始」の値は、「2016-03-29 00:00」1120定されています:

トリガー	
実行	毎日 毎 1 日
繰り返し	+
開始:	 2016-06-30 < 00:00:00

トリガーの値が効果を発揮するコよ、ジョブの保存が必要なことに注意してくたさい。

以下も参照してくたさい

- <u>タイマートリガー</u>
- <u>ファイルシステムトリガー</u>
- <u>HTTP トリガー</u>

6.13.1 タイマートリガー

タイムトリガーを使用することにより、ジョブが特定の時刻に(毎日、毎週、おけは週の特定の日に、おけは他の日に実行されるなどの)柔軟性に富んた再帰オプションを使用して、特定の時間の間隔に開始される計画を立てることができます。

下のスクノーンショナーはタイマートリガーのサンプルを表示しています。
1.01.18

۲	リカー		
	実行	毎日 🖌 毎 1 日	
	繰り返し	毎 60 分 ⓒ 全日、または ()から 🔮 08:00:00 まで 🔮 20:00:00 🍵	
	開始:	 2016-08-10 12:00:00 	
	有効期限:	 2016-08-16 11:00:00 	
	タイムゾーン:	Europe/Berlin	
	📝 有効化さ	れました	
	新規トリガー	新規ファイル システム トリガー 新規 HTTP トリガー	

タイマートリガーココントの構成があります。

タイムトリガーの例

	• 月の週にある日で	
繰り返し	トリガーの繰り返しオプションを定義します。 実行ドロップダウンリスト内出指定された日数でイベント か繰り返し実行されます(前のフィー)いドを参照)。	
	毎フィールドは繰り返しの間隔を分数で定義します。	
	「から」どまで」のフィールドは繰り返されるイベントの時間の範囲を定義します。	
開始	トリガーの開始日時を定義します。	
	「1度」か実行ドロップダウンリストから選択されている場合、開始の日時は入力必須の項目です。	
	日付フィールドをクトックすると、開始日を選択することのできるポップアップカレンダーか開かれます。	
有効期限	トリガーの日時の有効期限を定義します。	
タイムゾーン	開始および有効期限の日時に適用することのできるタイムゾーンを定義します。デフォルトのタイムゾーンがサーバー管理設定内で定義されます(以下を参照 デフォルトのタイムゾーンの設定)。	
有効化	有効化されたチェックボックスにより、トリガーを有効化、おけよ、無効化することができます。このオプションは、新しいジョブを作成し、テストする際に運利です。	

6.13.2 ファイルシステムトリガー

ファイルシステムトリガーを使用することにより、新しく追加されたファイルおけま変更されたファイルなど、ファイルおけまディレオトリの変更を監視 することができます(削除されたファイルは監視できないにとい注意してくたさい)。(例えば、60 秒ごとに実行など)変更の収集の間隔を柔軟 に構成することも可能です。また、任意でトリガーの開始と終了の日付を設定することもできます。ワイルドカードを使用して、ディレオトリの特 定のファイルをフィルターすることも可能です。

下のスクレーンショントはファイルシステムトリガーの例を表しています。

トリガー		
チェック 開始: 有効期限: タイムゾーン ☑ が有効化	変更の日付 ファイルまたはディレクトリ: ・ ・ Europe/Berlin ▼ されました	ポーリング間隔: 60 秒。解決するまで 0 秒待つ。

サンプル ファイルシステムトリガー

ファイルシステムトリガーコお以下の構造があります。

トリガーがURI をどのようこポーリするかを指定します。有効なオプション	
新規作成	新規のファイルませまディレケト」が指定されたディレケトリに追加されると、トレガ ーは実行されます。サーバーのロード量に関しては、このオプションは最も少な

		いサーバーのリースを必要とます。	
		新規のトリガーが追加されジョブが保存されると、そのディレクトリ内の既存の ファイルは新規に作成されたものと考慮され、それぞれのためにジョブが実行さ れます。	
		ファイルが削除され、後に追加されるどうブはう一度実行されます(これは 「ポーレク間隔」秒か既に削除から経過した場合のみ実行されます)。ファイ ルの名前か変更された場合でもこのトリガーは実行されます。	
		ポールされたディレクトリめ変更されない場合このトリガーは実行されません。このような振る舞いめ必要な場合、次のオフション、「変更された日付」)を参照してくたさい。	
	変更された日付	トリガーは、指定されたファイルの最後の変更タイムスタンプをチェックします。日 付か変更されている場合、おさよ新し、ワァイルが追加されている場合、おこ はファイルの名前か変更された場合トリガーは実行されます。	
		このオプタンゴ前のものともサードーから若干多くのリノースを取ります。	
	コンテンツ	このオプションは特定のファイルのナメのハッシュコードを計算し保管します。ポー リング間隔が経過すると、ハッシュコードは、再計算され、保管されている値と 比較されます。差異が存在する場合、トリガーが実行されます。これはサーバ 一大量のコードを課すこと」なる点に注意してくたさい。	
		日付か変更されている場合、おけよ、新しいファイルが追加されている場合、ま たはファイルの名前か変更された場合トリガーは実行されます。	
ポールング間隔	デルクトリがポールされる間 す。	隔を秒数で指定します。デフォルトの直は60 秒です。最小の直は1 秒で	
N 秒待つ	次のショブを開始するまでの	待ち時間を秒数で定義します。	
開始	トリガーの開始日時を定義します。これは任意のフィーリドです。		
	日付フィールドをクリックすると、開始日を選択することのできるポップアップカレンダーか開かれます。		
有効期限	トリガーの日時の有効期限を定義します。		
タイムゾーン	開始および有効期限の日時に適用することのできるタイムゾーンを定義します。デフォルトのタイムゾーンがサ ー、・管理設定内で定義されます(以下を参照:デフォルトのタイムゾーンの設定)。		
有効化	有効化されたチェッンボックスにより、トリガーを有効化、おけよ、無効化することができます。このオプションは、新しいジョブを作成し、テストする際に便利です。		

Triggerfile パラメーター

ファイルシステムませまHTTPトリガーを作成する場合、FlowForce Server は自動的にtriggerfile 入力パラメーターをジョブに追加 します。ジョブが実行されると、FlowForce Server はパラメータを次に設定します:

- a) ジョブをトリガーしたファイル(ファイルシステムトリガーの場合)。
- b) 収集されたURI (HTTP トリガーの場合)のダウレロードコンテンクを含む一時ファイルの名前。

ジョブ入力パラメーター			
◆名前: triggerfile	型: 文字列	▼ デフォルト:	
•			

triggerfile、ラメーターの値を次のステップ内の入力値とていなすることができます。このようにして、トリガーするファイルを必要に応じて使用、おとは、処理することができます。

デンォルトではtriggerfile パラメーターはトリガーするファイルの絶対パンを含んでいます。パンの一部を抽出するコはファイルパン式関数を使用します。

triggerfile、デメーターを使用するジョブの例に関しては、ディレオリポーレグジョブを作成するを参照してくたさい。

6.13.3 HTTP トリガー

HTTP トリガーを使用することには、URI (Uniform Resource Identifier)の変更を監視することができます。特に、Last-Modified と Content-MD5 HTTP ヘッダーフィーリドの変更を収集することができます。(例えば、60 秒ごとに実行など)変更の収集 の間隔を柔軟に構成することも可能です。また、任意でトリガーの開始と終了の日付を設定することもできます。

下のスクレーンショナはHTTPトリガーの例を表しています。

トリガー			
チェック 開始: 有効期限: タイムゾーン:	□ンテンツ ∨ URI: http://example.org + + Europe/Berlin ▼	ポーリングの間隔: 60 秒。解決するまで 0 秒待つ。	1
☑ が有効化	されました		

サンプルHTTPトリガー

HTTP トリガーココントの構造があります。

	トリガーがURI をどのようこポーリするかを指定します。有効なオプション	
	HTTP ヘッダー日付	最後に変更されたHTTP ヘッダーをチェックするようにシステムに命令します。最後に変更 HTTP ヘッダーが不足している場合、コンテンン・MD5 ヘッ ダーのみがチェックされます(次のオプションを参照してくたさい)。
	コンテンツ	コンテンチMD5 HTTP ヘッダーをチェックするようこシステムに命令する。こ れはシッセージ統合チェックとして使用されてしる128ビット「ダイジェスト」 です。ポートング間隔が、なされた後、MD5 ヘッダーカ変更されるとファイル がトリガーされます。ヘッダーがサーバーには提供されていない場合、コンテ ンソは取得され、ろーけるで、ッシュ化されます。
ポーシング間隔	URI がポールされる間隔を秒で指定します。	

解決するまでN 秒待つ	次のションを開始するまでの待ち時間を秒数で定義します。	
開始	トリガーの開始日時を定義します。これは任意のフィーリドです。	
	日付フィールドをクリックすると、開始日を選択することのできるポップアップカレンダーか開かれます。	
有効期限	トリガーの日時の有効期限を定義します。	
タイムゾーン	開始および有効期限の日時に適用することのできるタイムゾーンを定義します。デフォルトのタイムゾーンが ー、、一管理設定内で定義されます(以下を参照: デフォルトのタイムゾーンの設定)。	
有効化	有効化されたチェックボックスにより、トリガーを有効化、ませよ、無効化することができます。このオプション は、新しいジョブを作成し、テストする際に運利です。	

Triggerfile パラメーター

ファイルシステムませまHTTPトリガーを作成する場合、FlowForce Server は自動的にtriggerfile 入力パラメーターをジョブに追加 します。ジョブが実行されると、FlowForce Server は、ラメータを次に設定します:

- a) ジョブをトリガーしたファイル(ファイルシステムトリガーの場合)。
- b) 収集されたURI (HTTP トリガーの場合)のダウノロードコンテンンを含む一時ファイルの名前。

ジョブ入力パラメーター			
ب ج غند triggerfile	म्॥. उंग्रंग्रेग	.	
•2 mggernie		エーテンオルト:	

triggerfile、ラメーターの値を次のステップ内の入力値とていなすることができます。このようにして、トリガーするファイルを必要に応じて使用、おとは、処理することができます。

デフォルトではtriggerfile / デメーターはトリガーするファイルの絶対/ ひを含んでいます。/ ひの一部を抽出するコはファイル/ や式関数を使用します。

triggerfile、デメーターを使用するジョブの例に関しては、ディレカリポーレグジョブを作成するを参照してください。
6.14 ジョブを Web サービスとして公開する

FlowForce Server ジョブをWebサービスとして公開することができます。Webサービスとして公開されるジョブはプログラング的にアクセスできることを基本的に意味します。テストとディッグのために、このようなジョブをブラウザーから呼び出すこともできます。

Webサービスとしてジョブを公開する

- 1. ジョブを作成する、おけよ、既存のショブを編集のために閉きます。
- 2. 「HTTP を介してこのジョブを利用可能にする」チェックボックスを選択します。

Service

Make this job available via HTTP at URL http://<*FlowForce server*>/service/ CheckPathService HTTP(S) service V

3. URL テキストボックス内にWeb サービスの名前を入力します。

ジョブを保存するとFlowForce Server はHTTP GET とPOST リクエストを指定されたURL で受け入れるリスナーサービスを開始します。サービスとして公開されるショブは、FlowForce サーバーが作動している間はアクティブです。

メモ <u>セルアップページ</u>からFlowForce Server サービスのオスト名 フィールドを設定すると Web サービスの呼び出し [▶] ボタン が使用できるようこなりますこのボタンをクリックすると新規のブラウザーのウィンドウ内で Web サービスが呼び出されます。 FlowForce Server のナメのオスト名を構成していない場合、をブラウザーのアドレスパー内にURL を手動で入力して Web サービスを呼び出すことができます。

Webサービスか呼び出されると、FlowForce Server は指定されたショブの実行ステップを実行し、次を返します:

- ジョブか結果ファイルを生成する場合、最後のステップの最初の結果ファイル。
- 結果ファイルが生成されない場合、最後のステップの標準出力(コマイドライン出力と作業する場合などが当てはまします)。

結果に従ったコンテンツ型 ヘッダーカ表示され、HTTP 200 ステータスと共に有効な結果が返されます。コンテンツ型 ヘッダーは実際の結 果により異なります。MapForce マンピングは XML 出力が存在する場合、おけよ、テキスト出力のためのtext/plain が存在する場 合、text/xml で結果が表示されます。他の関数の標準の出力もtext/plain で返されます。結果は修飾記号なしてレスポンスボ ディとして返されます。

実行エラーはジェネリック名エラーチッセージと共にHTTP 5xx ステータスとして報告されます。詳細に関しては、FlowForce Server ログ をチェックしてください(ジョブログの確認を参照)。

このようなジョブの構成を説明するステップパイステップのサンプルゴ関してはWebサービスとしてジョブを公開するを参照してくたさい。

全てのショブステップが実行される前にFlowForce か結果を返すように構成することは可能です。サービスとして呼び出されたショブが時間 を要す場合場合この機能は特に役に立ちます。早い結果は処理のためにFlowForce Server により受け入れられた呼出元により確認と して扱われます。詳細に関しては<u>延期されたステップ</u>を参照してくたさい。

現在構成しているすべてのFlowForce Server Web サービスを確認する

以下の内のこのを行ってくたさい

- FlowForce Web 管理インターフェイス内でホームに移動し全てのアクティブなトリガーとサービスを表示するをクトックします。アクティブなトリガーとサービスも参照してください。
- 以下のURL に以下のブラウザーからアクセスします: http://[FlowForceServer][ServerPort]/service/*
- メモ URL 内では [FlowForceServer] と [ServerPort] は FlowForce Server がリンスンするやトワーケアドレスとポートを参照します。デフォルトでは、「同じマシンからアクセスすると想定し、FlowForce Server は <u>http://localhost:4646</u> で作動します。サーバー名とポートは管理ページで定義されているとおりです。<u>ネルワーク設定の定義</u>を参照してくたさい。

Web サービスパラメーター

Webサービスとしてジョブを公開すると、全てのジョブの、ラメーターは自動的にサービスの、ラメーターしています。 ジョブの、ラメーターにデフォ ルトが存在してい場合、サービスを呼び出す際にデフォルトを与える必要があります。 リクエスト内で与えられていない場合、任意の、ラメータ ーはデフォルトの値を取ります。

サービスが呼び出されると、FlowForce Server は、リクエスト内で与えられオン ウメーターをジョブ内で定義されオン ウメーターマ対して検証 します。 パラメーターの検証に失敗すると、FlowForce Server は、5xx HTTP ステータスを返します。 この場合デバッグとテストのオンガニ FlowForce Server はHTML パウメーターフォームも表示します。 デバッグとテストのオンガニHTML フォームを使用して、パウメーターを手 動で入力することができます。 型 ストリームの、ウメーターのオンガニ、ページ内でブラウザーボタンが使用できるようしなり、パウメーターか必要と するよう こファイルをアップロードするオンガニ使用することができます。

テスト HTML フォームを条件無しに表示するコよリクエスト内I ビルドインの(値を持つ)パラメーター showform を与えます。

ノウメーターを持つFlowForce Web サービスを呼び出すコよ次のオプノョンの一つを使用することができます。

- 1. 文字列、およ、数値などの簡単型の・ラメータコよ GET およ POST リクエストのURL 内に提供することができます。例えば<u>Webサービスとしでショブを公開する</u>を参照してくたさい。
- 2. POST の場合は、更にmultipart/form-data おはapplication/x-www-form-urlencoded とつ デ ータを提供することができます。FlowForce 内の型「stream」が デメーターの場合、(FlowForce か期待するため) multipart/form-data とてクライアトか提供する必要がおます。このようよ デメータのために、 ブランザーは、対応する パテメーターの横に「参照」 ポタンを表示する HTML フォームをテストします。
- 3. ケライアントはPOST リクエストのボディ内の任意のコンテンンを含むことができます(ハラメーターとしてではなく、HTTP リクエストのボディとしてポストされている JSON おけはXML などのコンテンンを具体的に示しています)。これを可能にするけよ FlowForce ジョブは型 stream の single ハラメータを含んている必要があります。追加の非 stream ハラメータを必要とする 場合、これらは POST URL 内に提供される必要があります。しかしながら、FlowForce 内で一つの型 stream の・ラメータが 定義される必要があります。これらの条件が満たされると、stream ハラメーターのナッグにデータとしてリクエストボディか扱われます。 他の構成は必要ありません。例えば、JSON を FlowForce Web サービスへポストする方法を参照してくたさい。

Web サービスの認証

デフォルトでは、FlowForce Server はHTTP ベーシック認証を使用して、Web サービスを呼び出すクライアントを認証します。ユーザー の資格方法はFlowForce Server ユーザーデータベースに対してチェックされます(FlowForce Server Web 管理インターフェイスに ログオンするオーガー使用されるユーザー名と、ペワード).

Authenticatio	n Required X
?	http://127.0.0.1:4646 is requesting your username and password. The site says: "Altova FlowForce"
User Name:	john_doe
Password:	•••••
	OK Cancel

資格情報無しでサービスが使用できるようにするコよ デフォルトの ^Lanonymous ユーザーに*サービスの使用パーミッションを*与えます (<u>パーミッションのしくみ</u>を参照してくたさい)。

サービスが匿名の使用を許可する場合 HTTP 資格情報を提供することができます。資格情報はFlowForce Server のユーザーデータ ベースに対してチェックされ、サービス実行ファイルは、匿名ユーザーの代わりに認証済みのユーザーの属性にされます。

無効な資格情報を提供すると、401 のHTTP 状態をインターフェイスは返します。サービスの使用はこのサービスの匿名のユーザーに与え られません。リクエストィンターフェイスは、401 のHTTP 状態を返します。

有効な資格情報が与えられ、このサービス上でサービスの使用パーミッションの認証されていないコーザーに与えられていない場合、リクエストインターフェイスはHTTP 4xx 失敗状態を返します。存在しないサービスにアクセスしようとすると、HTTP 4xx 失敗状態が返されます。

HTTP ベーシック認証しかえ、任意で、Windows ドメインの認証を構成することもできます。構成方法に関しては、ディレクトリサービスの 変更設定の変更を参照してください。Windows ドメイン認証が構成されると、有効なユーザー名と なワードを提供することに以、ユーザ ーはFlowForce Server により公開される Web サービス こアクセスできるよう てよります。重要な点は、ユーザー名がプレフィックス NT/を 含む必要があり、ドメイン名を除外する必要があることです。例 NT/john.doe@my.domain.com.

Authentication	n Required X	
?	http://127.0.0.1:4646 is requesting your username and password. The site says: "Altova FlowForce"	
User Name:	NT/john.doe@my.domain.com	
Password:	•••••	
	OK Cancel	

キューの設定

サービスURL はいガーの特別な型で、このためは接続か構築されると同じキューの制約の対象となります。<u>キューの設定の定義</u>を参照してくたさい。

HTTP リクエストボディの最大サイズを構成する方法

大体 100 MB くらいのHTTP リクエストボディの最大サイズを確立する FlowForce Server 内に存在するデフォルトのミット。コーラー がHTTP リクエストをWeb サービスとして公開されている FlowForce ジョブコポストし、HTTP リクエストボディがミットを超すと FlowForce Server は以下のテキストを含むエラーを返す場合があります: リクエストと共に送信されたエンティティか許可されている最大バイトを超える場合。

大きなサイズのリクエストを受け入れる方法

- 1. <u>flowforce.ini</u> ファイルをテキストエディターで開きます。
- 2. オプロンmax_request_body_size を[Listen] おけは[ListenSSL] セクションに追加して、許可される、作物を最大に設定します。

例えば500 MB の最大サイズを有効化するコはflowforce.ini ファイルは以下のよう」ないます:

```
[Listen]
active=1
host=0.0.0.0
port=4646
hostname=somehost.example.org
max request body size=500000000
```

.ini ファイルに関する詳細は構成ファイルレファレンスを参照してくたさい。

Flow Force Server プールスレボの再構成の方法

大量の並列 HTTP サービスリクエスト (例えば 1度に20 ませま以上)を期待する場合サーバーをプールスレンドの若干大き、数量に再構成することが可能です。

- 1. flowforce.ini ファイルをテキストエディターで開きます。
- 2. オプタン thread_pool を.ini ファイルの[Listen] おま[ListenSSL] セクタイン追加し 20 以上の値に設定します。
- 3. サービスを再起動します。
- メモ (多くのプールスレイドを必要としな) FlowForce Web Server のナダと(異なるポート上の) 他のノスエストのナダン酒別の [Listen] セクションを2つ持つことはよい案です。それ以外の場合、FlowForce Web Server はプールスレイドのナダの他の HTTP リクエストと競合します。

6.15 ジョブの結果をキャッシュする

キャッシュ」ことリサー・・・のロードとショブのレスポレスの時間を削減することができます。完了するまで「こ時間がかかり、サー・・・のロード量が大きななくことがある、Webサービスとして公開される、おりまデータ集中型のショブにとって、キャッシュ」は特に役に立ちます。

ジョブの結果のキャシシュとは、FlowForce Server が、内部のレポネリコジョブの結果を用意し保管(つまり、キャシシュ)することを基本 的に意味します。ジョブコ・ラメータがーが存在する場合、システムは構成できる上限まで、各パラメーターの組み合わせのためにキャシシュエン トリを作成します。(ユンシューマーとされる)他のジョブからキャシシュを持つジョブが呼び出された場合、FlowForce Server は、(ジョブを再 度実行する代わりにコンシューマーム対する結果を返し、レスポンスの時間を削減します。

サーバーへのデータのコードの量の、ランスを保つコよ、一方でキャッシュか古くならないように注意し、もう一方で、使用する環境とショブにより キャッシュを指定します。この、ランスを達成するために、FlowForce Server に搭載されるキャッシュの設定に慣れ親した、必要につ応じて キャッシュを使用してください。

キャッシュされたショブの結果と作業する場合、以下の点に留意してください

- 結果がキャッシュされるジョブによるデータ型を宣言する必要があります(以下を参照 ジョブの戻り値の型を宣言する)。
- 結果がキャッシュされるジョブとコンシューマージョブは、同じ資格情報を使用する必要があります。資格情報が異なると、ジョブは、 キャッシュか定義されていないものとして実行されます。
- キャッシュされたショブの構成を変更すると、既存のキャッシュデータは無効」ています、フラッシュされます)。

結果	のキャッシング:	
v	結果をキャッシュする	このジョブが他のジョブより呼び出されるとキャッシュは使用されます。
	 キャッシュされた結。 これを呼び出し、ま。 "コンシューマーに。 ジョブ入力パラメージ 	果を作成または更新する時間のトリガーを追加する、またば コンシューマーによって開始済み" オプションを確認する。 たキャッシュから利益を得るジョブを作成する。 こって開始済み" オプションが選択されている場合、キャッシュの更新または削除タイマーを追加しキャッシュ エントリが古くなることを回避します。 マーが コンシューマーによって開始済み" オプションを設定している場合、" 最大キャッシュ エントリ数" は入力パラメーターの可能なパリエーションを期待できます。
	🔲 コンシューマーにより開	始済み
	最大キャッシュ エントリ数:	1
	HTTP から使用可能な 📃	新しいキャッシュ コンシューマー ジョブ/public/db2osm.cached を自動作成する
	新規キャッシュ更新タイマ、	- 新規キャッシュ タイマーの削除
+1	ッシュの設定	

使用することのできるキャッシュの設定は、以下のとおりです。

結果をキャッシュする	ジョブの結果をキャッシュする場合このドックスを選択します。このドックスをチェックすることにより、 現在のジョブの <u>コンシューマー</u> に、ジョブを実行する代わりに、キャッシュされた結果を読み取るようこ命令します。 定義された・リガーカ実行されたさめ、おけよジョブのWebサービスが呼び出されたさか、(コン シューマーによりではなく)現在のジョブが直接実行されている場合、FlowForce Server は、キャッシュを更新します(おけよ、ジョブ・ウメーターキャッシュ内で見つからない場合、与えら れたっ、ウメーターの組み合わせをベースにて新規のキャッシュエントリを作成します)。
コンシューマートことり開始される キャッシュ	このオプションが有効化されていると、現在のショブを呼び出す全てのショブ(つまり、 <u>コンシューマー</u> ジョブ)は、キャッシュか既存ですよ、場合、キャッシュを作成します。それ以外の場合、トリガーと Webサービスの呼び出しのみがキャッシュを作成します。
最大キャッシュエントリ数	このオプションは、各ジョブのために作成されるジョブの結果のキャッシュを制限します。 ジョブンパ ラメーターが存在する場合、(組み合わせは無限と想定して)このオプションを全ての可能ない ラ メーターの組み合わせに設定されることが奨励されます。

<i>キャッシュコンシューマージョブを自動 的に作成する</i>	キャジュエシューマージョブは、HTTP アドレスで指定されるWebサービスです。エシューマーWebサービスは、結果がキャジュされるジョブのキャジュを取得し管理する便利な方法としての役割を果たします。呼び出されると、エシューマージョブは、最初にキャッシュされたインのジョブの結果を使用しようを読みます。キャジュされた結果が存在せず、「エンシューマーにより開始」オプシューマーにより開始」オプシュンが開始」オプションが、「コンシューマーは、メインジョブには返される実際の結果を取得します。キャジュされた結果が存在せず、「コンシューマーにより開始」オプションが有効化されていると、エシューマーは、メインジョブには返された実際の結果を取得し、キャッシュも作成します。
キャッシュタイマーの更新	 キャッシュタイマーの更新は、システムかどの頻度で現在のジョブをキャッシュするか管理します。現在キャッシュされているすべての、デメーターの組み合わせか更新されます。 実行 更新 ● 毎日 ● 毎 1 日 繰り返し ● 時間: ● 15:57:00 ● タイムゾーン: Europe/Berlin ● す 効化されました 新規キャッシュ更新タイマー 新規キャッシュ タイマーの削除
<i>キャッシュタイマーの判除</i>	キャッシュタイマーの削除は、システムが現在のジョブのキャッシュをどのように削除するかを管理します。 実行 完全削除 毎日 ● 1 日 繰り返し ・
キャッシュの保存と更新	ボタをクトックして、キャッシュを手動で保存および更新します。

キャッシュタイマーの更新とキャッシュタイマーの削除は、「削除」(
) ポタンとクリックすることにより、他のトリガーと同様「削除することが できます。おこ、「削除を元に戻す」(
) 操作が両方のメニューーご対して、保存、おけよページの再ロードが適用されるまで使用すること ができます。「複製」 ポタン(
) により、現在のトリガーと同じ設定を持つトリガーを作成することができます。 以下も参照してくたさい

• ジョブの結果をキャッシュする(例)

6.16 キュー設定の定義

キューの設定によりサーバーリソースの使用をさらて柔軟的に管理できます。例えば、キュー構成により一度に並列に作動するジョブインスタンスの数を制限することができます。

実行キューはジョブの「プロセッサー」です。これはジョブインスタンスかどのようご実行されるかを管理します。実行するためはは、各ジョブはター ゲト実行キュー(書叫当てられます。キューは(キュー(書叫当てられているすべてのジョブの)ジョブインスタンスが一度に実行される数、実 行間の遅延を管理します。デフォルトで、キュー設定はジョブニメしてローカルですが、キューを複数のジョブによりシェアされるスタ・ドアロンオブ ジェケトとして定義することもできます。キューはジョブニメしてローカルであること、おけよ、複数のジョブで共有されることができます。複数のジョ ブが同じキュー(書叫当てられている場合、実行のためにそのキュー)は共有されています。

更に、適切な(特権とは異なるコンテナー・トミッションを有する場合のみキューを確認、おけよ キュービジョブを割り当てすることができま す。具体的には、キューを作成するけなにユーザーが「実行キューを定義する「特権を持つ必要があります、特権のしくみを参照してくたさい。 更に、(特権とは異なる)適切なパーミッションがある場合のみユーザーはキューを確認、おけよキュービジョブを割り当てることができます。 パー ミッションのしくみを参照してくたさい。これは、キュービジョブを割り当てできることを意味します。キューへのアクセスを制限するには、キューか定 義されている個所にコンテナーをナビゲートし、コンテナーのパーミッションをロールauthenticated のけっな」「キュー アクセス無し」に変更 します。次に、パーミッション「キューー使用」を特定のロール、おけよ、必要なユーザー「割り当てます。詳細に関しては、<u>Joublic コンテナ</u> ーへのアクセスを制限するを参照してくたさい。

スタイアロンキューの作成

キューをスタイアロンオブシェクトとして作成する

- 1. 「構成」をクリックして、キューを作成するコンテナーインナビゲートします。
- 2. 「作成」をクリンクして、「キューの作成」をクリンクします。

以下にキューを作成 /public	
キュー名: キューの説明:	
キュー設定	
•	
実行: マスターまたはワーカー >	1 4 🕤 💼
実行間の最短時間: 0 秒	
並列実行の最大数: 1 インスタンス	
÷	
保存	

3. キューの名前、そして任意で、詳細を入力します。すべての設定の参照に関しては、下の「キューの設定」を参照してくたさい。

4. [保存]をクルクします。

ローカルキューの定義

スタイドアロンキューの作成の代わりに、ジョブ内でローカルにキュー設定を定義することができます。これを行うには ローカルキューの定義 オプ ションをジョブ構成ページから選択し、キューの設定を指定してくたさい。 下のイメージはデフォルトのキュー設定を表示しています。

キュー設定			
キューを使用したジョブ	の実行: ○ 既存のキューを選択: ◉ ローカルキューを定義:	実行間の最短時間: 並列実行の最大数:	0 秒 1 インスタンス

既存のキューの選択 オプションを選択すると、以前に定義されているスタンドプロン、外部キューを指定する必要が決ます。実行間の最短時間と並列実行の最高数 設定のレファレンスに関しては、下の「キュー設定」のセグションを参照してくたさい。

キューの設定

構成のために使用することのできる設定は下にリストされるとおりです。

キューの名前	キューを識別する名前を入力します。これは必須のフィールドで、スペースで開始し、終了する必要があります。おけは文字、数字、単一スペース、アンダースコア(″_ ″)、ダッシュ(″-″)および終止符(″.″)文字のみを含むことができます。
キューの説明	オプションで、キューオブジェクトのナックの説明を入力します。
	(ローカルではなく)キューカダダボアロンキューとして定義されている場合のみこのフィールドは適応可能です。
実行	このキューからのすべてのジョブインスタンスかどのようこ実行されるかを指定します: • マスターませまして、カー・このキューの一部であるジョブインスタンスはマスターませまして -カーマンン上で使用できるサーバーコアム従い実行されます。 • マスターのみ - ジョブインスタンスはマスターマンン上でのみ実行されます。 • ワーカーのみ - ジョブインスタンスは使用できるワーカー上で実行されますが、マスター 上では実行されません。
実行間の最短時間	実行キューは現在アクティブなルールご従い割り当てられたフーカーの数量を乗算した「平行実行の最大数」設定により管理されている使用可能なスロットの数量である実行スロットを提供します。各スロットは連続的にジョブインスタンスを実行します。 「実行間の最短時間」設定は、次のジョブインスタンスをすく11とックアップしないように、スロットをジョブインスタンスの売了後、短い間、使用中とマークします。この機能により、この実行キューのために最大のスループットを削減し、同じマンン上での他の実行キューと他の処理のためにCPUタイムを提供します。

並列実行の最高数	このオプタンはキュー上で使用可能な実行スロナの数量を定義します。設定は現在のキュー 内で並列に実行される同じジョブのクンスタンスの数量を決定し、各スロナトはインスタンスをシー ケンス順に実行します。しかしながら、並列で実行を許可するインスタンスの数量は使用可能 なマシンリノースを競合することに注意して下さい。この値の増加は、集中したこ1/0 オペレーショ ンを実行しない、おさま大量のCPU時間を必要とした。軽量のジョブを処理するキューが受け 入れることが可能です。(重量のジョブインスタンスを一度に処理するさなはコデンオパトの設定 1 は最も保守的で、リノース集中型のジョブを処理するジョブに最適です。
	このオプタンはFlowForce Server によし受け入れられる Web サービスとしてショブを公開 するケライアントからの並列 HTTP リクエストの最大数に影響を与えません。詳細に関しては FlowForce Server プールスレッドの再構成を参照してください。

キュー設定の複数の設定を定義することができ、
・
・
・
ボタンをクリックすることにより、それぞれは異なる処理必要条件を持つことができます。
このようなセナアップに関する詳細は配布された実行のセナアップを参照してくたさい。

6.17 Windows ネットワークパスの参照

ジョブを作成する際、通常 FlowForce Server が作動するマシン上で、おけよ、やトワーク上のファイル ジを参照します。マップされたネットワーケドライブなどの、Windows ネ・トワーク・ジを参照する場合、Universal Naming Convention (UNC)構文を使用します。これは、ドライブ文字は、システムコントしてグロー・ジレビオは、オンタ、おっ、各ログインセッションは自身のドライブ文字に書い当てられているオンタ これはドライブ文字がシステムコントしてグロー・ジレビオなく、ログオンセッションが自身のドライブ文字に書い当てられているオンタ

UNC 構文は以下のフォームをとます:

\\server\sharedfolder\filepath

該当する箇所

- server が(DNS に以定義される)ネトワーク内のサード-名を参照する箇所.
- sharedfolder が管理者にお定義されるラベルを参照する箇所例えば、admins が通常オペレーティングインストールのルートディレクトリである箇所)。
- filepath がエアの下のサブディノト・を参照する箇所。

6.18 FTP サポート

FlowForce Server 内でショブを構成する場合、FTP サーバーは接続する必要がある場合があります。例えば、ジョブがFTP サーバーか らファイルを抽出する必要がある場合、FTP サーバー上のディレクトリのエレテンンをリスト、おけよ、FTP を介してファイルを削除する必要が ある場合があります。上記のようよオペレーションを有効化するは、FlowForce Server は以下のプロトコルを使用して FTP サーバーへの 接続のすっかのサポートを提供します: FTP、FTPS、SFTP。

- FTP ファイル転送プロトコル(FTP)はFlowForce Serverとモト FTP サーバー間の転送を可能はします。プロトコルは TCP ポート 21 に割り当てられています。FTP プロトコルはファイル転送のすめの安全なモードのすめにデザインされていません。デ ータを安全に転送するコはFTPS おけはSFTP を以下で説明されているように使用してくけざい。
- FTPS プレーンFTPの使用の代替法は安全なTLS/SSL チャンネルを介してFTPを参照するFTPセキュア(FTPS)プロトコルです。FlowForce Server はFTPの使用をTLS/SSL 明示的な暗号化を使用したFTP とて既知の <u>RFC</u> 4217 に従いサポートしています。FTP おけまFTPSを介して接続を構築するけは/system/ftp コンテナー内で使用可能な 関数を使用してください。
- メモ with TLS/SSL 明示的な暗号化を使用したFTP は非奨励と考えられサポートされていません。
 - SFTP 安全な FTP (SFTP) はデータを Secure Shell (SSH) を使用して安全に変換するメンドです。SSH を使用したFTP はプレーン FTP と比較して大幅に異なる作動をします。このため、FlowForce Server Advanced Edition はプロトコルの具体的な使用を可能にする個別のビルトへ、関数を提供します。SFTP を介して接続を構築するには、<u>system/sftp</u>コンテナー内で使用可能な関数を使用してくたさい。
- メモ SFTP サポトはFlowForce Server Advanced Edition でのみ使用することができます。

6.19 ジョブのインポートとエクスポート

ジョブとデプロイされたMapForce マピングとStyleVision 変換を含む他の構成オブジェクトをFlowForce Server からんポートすることができます:

- 他の作動中のFlowForce Server インスタンスへ(オンライン エクスポート)
- ファイルへ(オフラインエクスポート)

他の作動中のFlowForce Server インスタンスにオブジェクトをエクスポートする場合、エクスポートされたオブジェクトは、サーバーのWeb 管理インターフェイスですく使用できるようになります。

オブジェクトをファイルレニケスポートする場合、FlowForce Server は、選択されたオブジェクトとそのな存関係を含む.zip アーカイブを作成します。zip アーカイブは、エケスポートのオペレーションが実行された日時に従い、名前がストンれます。名づけの変換は以下の通りです: export_YYYYMMDDThhmmss。例えば、2015-2021年の8月6日の10:51:33 にエケスポートされたファイルは、 export_2015-20210806105133.zip. と名づたわます。

(インポートされたオブジェクトが保存先に存在せず、おこ、上書き可能と想定して)次にzip アーカイブを同じ FlowForce Server インスタ ンスにインポートすることもできます おけま地のインスタンス人ポートすることができます。

構成データのエクスポート

コンテナー内で特定の記録、おけま、コンテナー全体をエクスポートすることができます。とちらの場合でも、FlowForce は、エクスポート前に 記録のリストをレビューすることのできるダイアログボックスを表示します。コンテナー全体をエクスポートするように選択した場合は、例えば、こ のダイアログボックスは、選択されたコンテナーのすべての子の記録を表示します(ジョブまけは資格情報)。

オブジェクトをエクスポートする前に、FlowForce は、依存関係を持つ全てのオブジェクト、おけはエクスポートするオブジェクトには参照されるオブジェクトについて別のページで通知します。これにより、見逃された依存関係を一目で確認することができます。オブジェクトを実行中の FlowForce Server にエクスポートする場合、エクスポート先のサーバー上に各オブジェクトが存在するかを確認することができます。

セキュリティ上の理由から、FlowForce Server は以下に挙げれる重要なデータのカテューをエクスポートしません

- ジョブ内でローカルに定義されたパマワード
- 「スタンドアロン」資格情報の記録とて使用される、スワード
- (<u>/system/ftp</u> 関数などのシステム関数と共に保管される タワード)
- OAuth 2.0 ケライアトの秘密、認証トーケンと更新トーケン
- 証明書+秘密キーの組み合わせ内の秘密キー

機密なデータのこれらのカテゴリ全てをエクスポートするオンタン、エクスポート中にオプション「機密な情報のエクスポート」を選択します。このチェックボックスを選択すると、エクスポートされたアーカイブでは機密なデータがプレーシなテキスト書式で含まれることに注意してく ださい。

このチェックボックスを選択しない場合、機密なデータはエクスポートされません。FlowForce Server にデータをインポートする場合、 それぞれ個別のレコードを上書きすること、おけはスキップすることを選択できます。上書きすることを選択した場合、機密なデータが空の 値により置き換えられます。具体的には、OAuth 2.0 資格情報の場合、、パタワードは空白です。証明書の場合、証明書には秘 密キーが存在しません。資格情報の場合、クライアントの秘密、アクセストークン、および更新トークノはすべて空白です。

作動中のFlowForce Server インスタンスヘジョブをエクスポート する

1. 「構成」をクトクレ、エクスポートする記録を選択します。コンテナー内の特定の記録を選択、ませよ、コンテナー全体を選択する ことができます。 2. 「選択されたオブジェクトのエクスポート」をクリックします。

選択されたオブジェクトのエクスポート	×
▲ 名前 ✓ ③ db2osm.job ✓ ③ my.credentials ✓ ③ AirportStatus.mapping	
● サーバーにエクスポート	
サーバー: 10.100.61.11 ポート: 8082	
ユーザー: root	
パスワード: eeeeeee SSLの使用: 🗌	
○ ファイルにエクスポート	
エクスポートキャンセル	

- 1. ダイアログボックスから、「サーバートニエクスポート」をクリックし、FlowForce Server のホスト名と保存先、および作動するポートを入力します。
- 2. FlowForce Server インスタンスの保存先にユーザー名と、ペワードを入力します。そして、「エクスポート」をクトックします。 FlowForce は、依存関係を確認し、エクスポートから除外を選択することのできるページにエクスポートされる全ての記録を表示 します。

Export			
Name	Type 🗢	Dependencies	Remote Server
/public/my.credentials	credential	Referenced by:	Exists on remote server
/public/ConvertProducts.mapping	MapForce mapping	Referenced by:	Will be created on remote server
🗹 🔕 /public/ConvertProducts.job	job	Depends on:	Will be created on remote server
□	function	Referenced by:	Exists on remote server
Export options			
Export sensitive data:			

黄色し背景を持つ記録は、エクスポートされる記録です。黄色し背景色を持たむ、記録は、横のチェックボックスのチェックを解除することにより、エクスポートから除外した記録です。 最後に、 グレーの背景色を持つ記録は、 ビルトインシステム関数上の依存 関係を表しており、これらを操作することはできません

「リモトサイド」列は、「リモトサイドーで存在、おけよ存在していない可能性」を表示しています。.zipアーカイブ FlowForce に再インポトする再に保存先ファイルに依存関係が存在する場合、エクスポートかられらの記録を削除することが できます。それ以外の場合、依存関係無しにエクスポートする場合、そして依存関係が保存先で存在してい場合、ジョブは失敗 する可能性が大きいです。成功した例も参照してくたさい。

5. エクスポートするオブジェクトをクリックして選択し、「エクスポートの開始」をクリックします。

ジョブをファイルニエクスポートする

- 1. 「構成」をクトックし、およびエクスポートする記録を選択します。コンテナー内の特定の記録を選択、おさよ、コンテナー全体を選択することができます。
- 2. 「選択されたオブジェクトのエクスポート」をクトックします。

Name			
	lentials		
	Products.mapping		
	Products.iob		
-			
○ Export to ser	ver		
O Export to ser Server:	ver	Port: 0	
O Export to ser Server: User:	ver	Port: 0	
O Export to ser Server: User: Password:	ver	Port: 0	
C Export to ser Server: User: Password:	ver	Port: 0 Use SSL:	

3. ダイアログボックスから、「ファイルニエクスポート」をクトックし、「エクスポート」をクトックします。FlowForce は、依存関係を確認、または、エクスポートから除外することのできるページで全ての記録を表示します。

Export							
	Name	Type 🗢	Dependencies	Remote Server			
	🧽 /public/my.credentials	credential	Referenced by:	Might or might not exist on remote server			
	/public/ConvertProducts.mapping	MapForce mapping	Referenced by:	Might or might not exist on remote server			
	log /public/ConvertProducts.job	job	Depends on:	Might or might not exist on remote server			
	/system/filesystem/copy	function	Referenced by:	Might or might not exist on remote server			
Ex	Export options						
	Export sensitive data: 🗌						

黄色し背景を持つ記録は、エクスポートされる記録です。黄色し背景色を持たなし記録は、横のチェックボックスのチェックを解除することにより、エクスポートから除外した記録です。最後に、グレーの背景色を持つ記録は、ビルトインシステム関数上の依存関係を表しており、これらを操作することはできません。

作動中のFlowForce Server インスタンスでは無くファイルをエクスポートしているける、エクスポートされけオブジェイトが保存先 で存在するかを決定することはできません。この理由から、「リモートサーバー」列は、「リモートサーバーに存在、おけよ存在して しない可能性」を表示しています。.zip アーカイブ FlowForce に再インポートする再に保存先ファイルに依存関係が存在する 場合、エクスポートからこれらの記録を削除することができます。それ以外の場合、依存関係が不足している箇所でデータのインポ ートを試みると、「操作に失敗しました」パンが存在しません」などのエラーが表示され、インポートは失敗します。以下を参照してく たさい、成功した例。

4. 「ファイルへエクスポートを開始する」をクリックします。使用中のブラウザー設定により、.zip アーカイブをローカルディレクトリに保存するようにプロンプトされるかもしれません。おけよ、構成済みの保存先ディレクトリに自動的に、ブラウザーが自動的に保存する可能性もあります。

構成データのインポート

ファイルからジョブをインポートします

1. 「構成」をクリックし、「オブジェクトのインポート」をクリックします。

オブジェクトのインポート	×
ファイル: 参照 ファイルが選択されていません。	
インポート キャンセル	

ダイアログボックスから「参照」をクリックし、FlowForce Server からエクスポートされた、ソース.zip アーカイブを選択します。
 「インポート」をクリックします。FlowForce Server は、インポートされた記録を個別のページコな存関係と共に表示します。
 「現状」列は、「インポート」ボタッカウリックされた後の各記録ログのお目記録ログの情報は関して通知します。

Import					
	Name	Type 🗢	Dependencies	Current State	
V	☞ /public/my.credentials	credential	Referenced by:	Will be overwritten	
	🗑 /public/Cleanup	job	Depends on:	Will be created	
	() /system/maintenance/cleanup-files	function	Referenced by:	Exists	
l	nport Cancel				

3. 「インポート」をクリックします。

6.19.1 構成データのエクスポート

コンテナー内の特定のレコード、おけはコンテナー全体をエクスポートすることができます。どちらの場合でもFlowForce はエクスポート前のレコ ードのノストをレビューすることができるダイアログボックスを表示します。エクスポートのナッカニコンテナー全体を選択するお、このダイアログボック スは選択されたニンテナー(ジョブまけは資格情報)のすべての子レコードを表示します。

オブジェクトのエクスポート前にFlowForce はエクスポートを希望するオブジェクトに拡存する(おけよ 参照されている)すべてのオブジェクト に関して個別のページで通知します。これは一目で不足する依存関係を確認する助けけています。作動する FlowForce Server にオブ ジェクトをエクスポートする場合、各オブジェクトが目的先のサーバーに既存しているかを確認できます。 デフォルトではFlowForce Server は機密データの以下のカテゴレをエクスポートしません

- ジョブ内でローカルに定義されている タワード
- 「スタイアロン」資格情報レコードとて使用可能なパワード
- (/system/ftp 関数)などのシステム関数に保管されている マワード
- OAuth 2.0 クライアント秘密、トーケンの承認とーケンの更新
- 証明書+秘密キーのペア内の秘密キー

機密データのこれらのカテゴレをインポートするコよ、エクスポート中に「機密なデータのエクスポート」を選択してくたさい。チェックボックスを選択すると、エクスポートされたアーカイブココンレーシテキストフォームの機密データが含まれることに注意してくたさい。

チェックボックスを選択しないと、機密なデータはエクスポートされません。FlowForce Server データをインポートし直す場合、各レコードを上書きする、まけはスキップするオプションがあります。上書きを選択する場合、既存の機密データは空の値と置き換えられます。具体的には、資格情報の場合、パマワードは空のままです。証明書の場合、証明書には秘密キーは存在しません。OAuth 2.0 資格情報の場合、クライアートの秘密、アクセストークン、および更新トークノは空にないます。

作動中のFlowForce Server インスタンスに構成データをエクスポートする方法:

- 1. 「構成」、をクリックし、エクスポートするレコードを選択します。コンテナー内の特定のレコード、おけコンテナー全体を選択することができます。
- 2. 「選択されたオブジェクトのエクスポート」をクリックします。

選択されたオブジェクトのエクスポート	×
▲前 ✓ 	
● サーバーにエクスポート	
サーバー: 10.100.61.11 ポート: 8082	
ユーザー: root	
パスワード: •••••• SSLの使用: 🗌	
○ ファイルにエクスポート	
	_
エクスポートキャンセル	

- 3. ダイアログボックス内で「サーバーニエクスポート」をクリックして、目的 FlowForce Server のオスト名と作動先のポートを入力 します
- 4. 目的 FlowForce Server インスタンスにユーザー名と マワードを入力し、「エクスポート」をクリックします。FlowForce はな 存関係を確認し、エクスポートから削除することのできるページにエクスポートするすべてのレコードを表示します。

Export							
Name	Type 🗢	Dependencies	Remote Server				
D 🕞 /public/my.credentials	credential	Referenced by:	Exists on remote server				
D /public/ConvertProducts.mapping	MapForce mapping	Referenced by:	Will be created on remote server				
D 🔊 /public/ConvertProducts.job	job	Depends on:	Will be created on remote server				
System/filesystem/copy	function	Referenced by:	Exists on remote server				
Export options	Export options						
Export sensitive data:							

黄色い背景色のレコードはエクスポート済みのレコードです。黄色い背景色が無いレコードはチェックボックスの横の選択がアフト ているエクスポートに含まれていないレコードです。最後に、灰色表示されているレコードはビルドインシステム関数上の依存関係を 示しています。そのためこれらに交近してのアグションを取ることはできません。

「リモートサーバー」列はファイルが目的先で存在するかを示しています。目的先に依存関係が既存の場合、このようよしコードをエクスポートから除外することができます。それ以外の場合、依存関係無しでエクスポートし、依存関係が目的先に存在しない場合 はこのようなジョブは失敗する可能性があります。成功した例を参照してくたさい。

上記のとおり「機密な情報のエクスポート」オプロインよりペワード 証明書秘密キー、とOauth 機密データをエクスポート パッケージスオプロンで含むことができます。安全上の理由から、FlowForce Server からプレーシテキストでこのような機密デー ダを転送する必要がある場合以外、このチェックボックスを選択することは奨励されません。

エクスポートされているリストカ型「資格情報」を含んでおり、ターゲトカジャンション2019r3 おけばな存の場合、チェックボックス FlowForce 2019r3 または以前のために資格情報をダウングレードするが選択されている必要かがます。そのリース 後、新規の資格情報レコードに FlowForce の古し シャンションと互換性を持たせるために資格情報か新規の「使用の許可」オ プランを取得します。「使用を許可」オプションの詳細に関しては<u>資格情報の定義</u>を参照してくたさい。

5. エクスポートするオブジェクトをクリックし「エクスポートを開始」をクリックします。

ジョブをファイルにエクスポートします

- 1. 「構成」、をクリックし、エクスポートするレコードを選択します。コンテナー内の特定のレコード、おけはコンテナー全体を選択することができます。
- 2. 「選択されたオブジェクトのエクスポート」をクリックします。

Name				
mv.cred	entials			
ConvertProducts.mapping				
0				
○ Export to serv	er			
O Export to serv Server:	er	Port: 0		
O Export to serv Server: User:	er	Port: 0		
O Export to serv Server: User: Password:	er	Port: 0		
O Export to serv Server: User: Password:	er	Port: 0 Use SSL:		

3. ダイアログボックス内で「ファイルニエクスポート」をクトックして「エクスポート」をクトックします。FlowForce は依存関係を確認 し、エクスポートが削除することのできるページ、ニエクスポートするすべてのレコードを表示します。

Export					
	Name	Type 🗢	Dependencies	Remote Server	
	☞ /public/my.credentials	credential	Referenced by: /public/ConvertProducts.job	Might or might not exist on remote server	
	/public/ConvertProducts.mapping	MapForce mapping	Referenced by:	Might or might not exist on remote server	
	log /public/ConvertProducts.job	job	Depends on:	Might or might not exist on remote server	
	🗊 /system/filesystem/copy	function	Referenced by:	Might or might not exist on remote server	
Ex	port options				
	Export sensitive data: 🗌				

黄色い背景色のレコードはエクスポート済みのレコードです。黄色い背景色が無いレコードはチェックボックスの横の選択がアフされているエクスポートに含まれていないレコードです。最後に、灰色表示されているレコードにざいドインシステム関数上の依存関係を示しています。そのためこれらございてのアグションを取ることはできません。

ファイルニエクスポートし、FlowForce Server インスタンを実行していないサムタ、エクスポートされオオブシェクトが目的先に存在すると決定することはできません。この理由から、「リモートサーバー」列ば「リモートサーバー」存在、おけよ存在しない」を表示する可能性があります。.zip アーカイブを FlowForce にインポートし直す場合、依存関係が目的先に存在する場合、このようなレコードをエクスポートから除外することができます。不確かな場合にはすべての依存関係をエクスポートしてくたさい。それ以外の場合、依存関係が不足している個所からデータをインポートしようとすると、インポートは「操作に失敗しました」などのメッセージと共に失敗します。ノアムは存在しません。」作業済みのサンプルを参照してくたさい。

4. 「ファイルへのエクスポートを開始する」をクリックします。ブラウザー設定により、.zip アーカイブをローカルディレクトリニ保存するようこプロンプトされる、ませるエブラウザーが自動的に構成済みの目的ディレクトリニ保存する可能性があります。

6.19.2 機密な情報を含む、または含まないオプション

FlowForce Server からデータをエクスポートする場合、エクスポートから特定のオブシェクトを除外することができます。以下のサンプルはつのアプローチ間の違いを説明しています。

サンプル1: 機密なデータを除外する

同じコンテナー(「my.credentials」)内の資格情報レコードを参照するショブ(「AddNumbers」)か存在すると仮定します。

(二) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2	
🧾 🥯 my.credentials 算俗情報	股
AddNumbers ジョブ	

「機密なデータをエクスポート」オプションを選択せずにファイルに(.zip アーカイブ)に両方のオブジェクトをインポートする場合、以下の状態が発生します:

- 1. ジョブがエクスポートされます。
- 2. 証明書レコードが、アワード無しでエクスポートされます。

.zip アーカイブを2つのオブジェクトが存在しないFlowForce Server 環境にインポートする場合、両方のオブジェクトが作成されます。証明書レコードに関連したパマワードは空になります。

ターゲト環境内にオブジェクトが既存の場合、上書き、おけよ、対応するチェックボックスをクリアして、スキップすることができます

インポート						
	名前	型 🕈	依存間係	現状		
	©≕ /public/Jobs/my.credentials	資格情報	参照元: 🔕 /public/Jobs/AddNumbers	上書きされます		
	🗑 /public/Jobs/AddNumbers	ジョブ	依存先: ◎ /public/Jobs/my.credentials ① /system/compute-string	上書きされます		

レコードを上書きすることを選択した場合、以下の状態が発生します

- 1. FlowForce Server 内に既存のショブは .zip アーカイブからのショブにより上書きされます。
- 2. FlowForce Server 内の既存の証明書レコードは.zip アーカイブからのたのにより上書きされますが、目的先の マワードは空のままです。

資格証明を上書きした場合、既存の資格証明はそのままで保管されます。

サンプル2:機密なデータを含む

上に示されるように2つの同じレコードをエクスポートし、「機密なデータのエクスポート」オプションをエクスポート中に選択するとします。この場合以下が発生します:この場合、以下が発生します:

- 1. ジョブはエクスポートされます。
- 2. 資格情報レコードはペワードを含みエクスポートされます。

.zip アーカイブを2つのオブジェクトが存在しないFlowForce Server 環境にインポートする場合、両方のオブジェクトが作成されます。資格情報レコードに関連する、なワードは.zip ファイルのからのたのになります。

ターゲト環境にオブジェクトが存在する場合、これらを上書きませる対応するチェックボックスをクリアしてスキップすることができます。レコードを 上書きする場合、以下が発生します:

- 1. FlowForce Server 内に既存のショブは、.zip アーカイブからのショブにより上書きされます。
- 2. FlowForce Server 内で既存の資格情報レコードは.zip アーカイブからのレコードにより上書きされます。エクスポート先の スワードも.zip アーカイブからのためにより上書きされます。

資格証明を上書きした場合、既存の資格証明はそのままで保管されます。

6.19.3 構成データのインポート

以前にインポート済みのアーカイブをインポートする方法:

1.「管理」をクリックし「オブジェクトのインポート」をクリックします。

オブジェクトのインポート	×
ファイル: 参照 ファイルが選択されていません。	
インボート キャンセル	

ダイアログボックスで「参照」をクリックし、FlowForce Server からんポート済みのノース.zip アーカイブを選択します。
 「インポート」をクリックします。FlowForce Server は個別のページにんポートされるレコードビルドを依存関係と共にを表示します。「現在の状態」列はインポートボタンをクリッグ後に各レコードに何が起こるかを通知します。

	Name	Туре 🗢	Dependencies Current State		
✓	💬 /public/my.credentials	credential	Referenced by:	Will be overwritten	
✓	lo /public/Cleanup	job	Depends on: © /public/my.credentials ① /system/maintenance/cleanup-files	Will be created	
	() /system/maintenance/cleanup-files	function	Referenced by:	Exists	

4. 「インポート」をクトックします。

FlowForce Server のターゲナトインスタンス内で見つけることのできない外部インスタンスをアーカイブが特つ場合「現在の状態」列は「不在」の状態を表示します。対処方法に関しては不足する依存関係の処理方法を参照してくたさい。

6.19.4 不足する依存関係の処理方法

FlowForce Server からデータをエクスポートする際、エクスポートから特定のオブジェクトを常に除外することができます。しかしなから、オブジェクトの一部は地のオブジェクト上に依存関係を持つ場合かあります。依存関係を依存するオブジェクトと共にエクスポートしていと、データを FlowForce Server に再度エクスポートする際にエラーを引き起こす可能性かあります。下のサンプルはこの影響を理解し、問題の対処す ることを意図としています。 2つのレコードが以下に表示されており、FlowForce Server から資格情報レコードをエクスポートせずにジョブのみエクスポートすると仮定します。

エクスポート					
	名前 🗣	型	依存間係	リモートサーバー	
V	Jpublic/Jobs/AddNumbers	ಶತರ	依存先: ☞ /public/Jobs/my.credentials ① /system/compute-string	リモートサーバーに存在しないかもしれません	
	☞ /public/Jobs/my.credentials	資格情報	参照元:	リモートサーバーに存在しないかもしれません	

この場合、以下が発生します

- 1. このジョブはエクスポートされます(か不足する依存関係が存在します)
- 2. 資格情報レコードはエクスポートされません。

オブジェクト /public/Jobs/my.credentials か存在する FlowForce Server 環境にアーカインをインポートしようと試みると、不足する依存関係を解決することができるのでインポートは可能にないます。

1	インポート								
	名前 🗣	型	依存問係	現状					
2	Jpublic/Jobs/AddNumbers	ジョブ	依存先:	作成されます					
	> /public/Jobs/my.credentials	資格情報	参照元: 🙆 /public/Jobs/AddNumbers	存在する					

しかしなから、参照される資格情報がターゲト環境に存在しない場合、インポートダイアログボックスは以下のようにないます。

1	インポート								
	名前 🗣	型	依存問係	現状					
M	/public/Jobs/AddNumbers	ジョブ	依存先: ☞ /public/Jobs/my.credentials ① /system/compute-string	作成されます					
	/public/Jobs/my.credentials	資格情報	参照元: ② /public/Jobs/AddNumbers	存在しない					

上のイメージ内では、資格情報レコードは資格情報としては識別されません。具体的には型は「資格情報」ではよく「構成」となります。またアイエンが異なります。型「構成」は型が味知のジェネトックな構成オジェント(他のジョブませば資格情報)を指します。このサンプルでは 「my.credentials」構成オブジェントエクスポートされておらず、エクスポートされていやケージロは参照ノや以外の正確な型の情報は含まれ ていません*。このため、FlowForce Server へのデータのインポートは以下のようなエラーを引き起こす可能性があります。「操作に失敗し ました」、やいよ存在しません。」

このエラーをフィックスするコよ エラーメッセージことり示される ひで不足するレコード (この場合 "my.credentials」レコード)を作成しインポートを再度試みます。

* 資格情報の参照はスタンドアロン資格情報オブジェクトへの参照である可能性かあり、一部の場合ローカル資格情報を含むジョ ブへの参照である場合かあります<u>ジョブから資格情報を参照する</u>。

7 ジョブの実行のモニタリング

ジョブがトリガーの条件を満たすと、おけよ、Webサービスの呼び出しによりオンデマードでトリガーされると、そのジョブのインスタンスが作動を開始します。ジョブインスタンスの出力、1つの状態から他の状態への切り替え、および他の実行の詳細はFlowForce Webにより口づされます。ジョブの実行とその結果を2つのレベリから取得することができます:

- 1. ホームページから。ホームページはショブの実行に関する以下のデータのカイック概要を提供しています:統計最近および実行中のジョブ、アクティブトリガーとサービス。
- 2. <u>ログビュー</u>ページから。ログビューページは<u>メインログのビュー</u>で説明されている通り実行済みのショブ 切する完全な詳細を表示しています。

このトピックの残りまれームページ内で使用可能なショブ管理データここいで説明しています。

統計

ホームページの統計セクションは以下のチャートを表示しています

- 過去14日間に実行されたジョブ
- 過去 24 時間に実行されたジョブ
- 過去 60 分に実行されたジョブ

各チャートコンタブ実行結果に従い色付けされたい、そきんでいます。成功、失敗、その他。

チャートの特定の、・ーニマウスをポイントすると、対応する期間をカレーする詳細情報と共にとしたか表示されます。例えば、下のチャートでは、ビートは10:12 に3個のジョブインスタンスの実行が成功したことが表示されています。



同様に、下のことトレち件の実行が放功しており、2件の実行がその日の10時に行われたことを示しています:



チャート上の特定の情報を更に詳しく知るコはチャートからログビューページに簡単にナビゲートすることができます。これを行うコは、チャートでフォーカスされる、チをクトックします。これは、与えられた日、時間、分のためコ既にフィルターされているログと共にエグページを表示します。

メモチャートとログによりトラックされるログの詳細との間に若干、おとい影響の小さい差分が存在します。

更に統計を表示するリンクは統計専用の統計の詳細ページを開きます。

最近および実行中のジョブ

最近および実行中のジョブセクション内では、最近および実行中のジョブのステータスを(20件まで)確認することができます。すべての最近と実行中のジョブを個別のページで確認する」コは全てのアクティブなトリガーとサービスを表示するをクリックしてください。

Recent and Running Jobs								
φ								
Instance ID	Job	Activation Time	Last Action	Status				
<u>17</u>	/public/Examples/MyTask	2021-02-02 17:18:00	2021-02-02 17:18:29	Finished successfully after step 1				
<u>18</u>	lo /public/Examples/MyTask	2021-02-02 17:19:00	2021-02-02 17:19:29	Finished successfully after step 1				
<u>19</u>	/public/Examples/MyTask	2021-02-02 17:20:00	2021-02-02 17:20:00	Running step 1	Stop job			
φ								
Show	Show all recent and running jobs							

作動中のジョブインスタンスは加え、テーブルは、作動中のジョブドリナでなく何らかの理由で失敗したジョブを含む最近完了したジョブを表示します。このようなジョブは実行が完了した後短い期間(約1-2分)のみ表示されています。「ログ」ページからの各ジョブインスタンスの履歴を常にチェックすることができます。ジョブログの確認を参照してくたさい。

最近おより実行中のショブのグリンドには以下の列か含まれています

- インスタンスID。開始すると、各ジョブインスタンスコゴインスタンスID」列内で表示される一意のIDが割り当てられます。インスタンスID(1後に「ログ」ページからの各ジョブインスタンス0度行ステータスをトラックする手助けをします。ジョブの実行中テーブル内のインスタンスIDをクリックすると、「ログ」ページュリダイレクトされ、選択されたジョブインスタンス1関する情報のみが表示されます。(例えば、一意のファイル名を作成するなど)ジョブの一意のインスタンスIDをジョブ内で使用する」コは、instance-id 式関数を使用すると可能しています。。
- ジョブ。列はこのジョブインスタンスの構成を検索することが可能ないなを示しています。
- アクティベーション日時。ジョブインスタンスの実行が開始した日時。
- 最後のアクション。最後の実行ステータスの日時。
- ステータズショブステータスはページカ最後に更新された時のものです。すべてのジョブインスタンスのステータスを参照するこは、ジョ ブのステータスを参照してください。

ジョブの停止 ボタイは実行中のジョブのためこのみ表示されます。実行中のジョブインスタンスを停止することができます。これはデータの破損を招く場合があり、特別な場合のみ行ってください。

ジョブの停止を参照してくたさい。

複数のFlowForce Server インスタンスがプラスターとして実行される場合、グリッドには各ジョブを実行するクラスタメンバーは関する追加 詳細が含まれます、
<u>分散済みのグリッドのモニタリング</u>を参照してください。

[最近と実行中のジョブグリナドは自動的には更新されません。全てのジョブの最終のステータスを取得するには、「グリッドの再ロード」

🧐 ポシをクリックします。

アクティブなタイマー

ページのアクティブなタイマーセクションは<u>トリガー</u>を介しておける<u>Webサービス</u>として実行スケジュールされた最近のジョブを20件まで表示します。このようなジョブのフルリストを確認するはま全てのアクティブなトリガーとサービスを全て表示するをクリックします。

Active Timers								
φ								
Туре	Job	Next run 🗢	Info					
timer	/public/Examples/MyTask	2021-02-02 17:27:00	Fire from 00:00 to 24:00 every 1 minutes (as in Europe/Vienna)					
φ	φ							
Show all a	ctive triggers and services							

[アクティブよー」ガーとサービス]のブリンドには以下の列が存在します

- 型ーはいガー(ウォッチ、タイマー、サービスの型を指しています。ラベルビウォッチ」はファイルシステムトリガーませまHTTPトリガーを参照します。追加の詳細のセッカン「情報列を参照してくたさい。
- ジョブートリガーまけまサービスが定義されるジョブの、ひを指定します。ジョブの構成ページを開くためコレクをクリックします。
- 次の実行-ウォッチーリガーに適用されます。この列はトリガーが次を実行する際に示されます。
- *情報* Web サービスとして公開されるショブの場合 Web サービスのURL などの追加情報を提供します。ウォッチ、まけまなイ マート・リガーのために、この別は現在のト・リガーの構成をまとめています。
- サービス URL Web サービスが対応するジョブによい公開されている箇所で URL を指定します。これは<u>Web サービスを公開</u> <u>する</u>ジョブでのみ適応可能です。

アクティブホーリガーとサービスグリンドは自動的には更新されません。グリッドの再ロードしの ボタンをクリックしてページを更新します。

7.1 分散済みのジョブの実行のモニタリング

複数のFlowForce Server インスタンスがフラスターとして作動するように構成されている場合、マスターFlowForce Server インスタン スがジョブを実行し、詳細をログする役割を果たします。スタンドアロンモードに戻され、限り、ワーカーマシンノよ、ローカルジョブを実行せず、ログ ビューイは存在しません。「ワーカー」モードを停止する方法を参照してくたさい。

マスター FlowForce Server インスタンスによりクラスター済みのセオアップ内のすべてのログは行われます。更に、「ホーム」ページ内の 「実行中のジョブ」テーブルコよ、ジョブインスタンスが作動するクラスターメンバーを示す、「ワーカー」列か含まれています。これは、ジョブ構成により、マスター、おコよ、クラスターの一部であるワーカーマンンであることができます。配布された実行のセオアップを参照してください。

実行中のジョブ							
φ	φ						
インスタン	インスタン ジョブ アクティベーション日 最後のアクション ワーカー ステータス						
<u>39</u>	/public/MyDistributedTask	2019-02-13 14:27:00	2019-02-13 14:28:04	マスター <u>2B2A5B</u>	ステップ2の後、完了に成功しま।		
<u>40</u>	Jpublic/MyDistributedTask	2019-02-13 14:28:00	2019-02-13 14:28:00	ワーカー <u>BD4B47</u>	実行中のステップ 2	ジョブの停止	
φ	ϕ						

現在作動中、およよ、特定のケラスターメンバ・(ワーカー、およよ、マスター)のための最近完了したごョブインスタンスを確認するこよ、「ワー カー」列内のレクをクトックします。これにより、そのケラスターメンバー上で作動中のジョブインスタンスをモニターすることのできるケラスターメンバ ーページが開かれます。



7.2 ジョブのステータス

ライフタイム全体で、ジョブインスタンスは「実行中のジョブ」テーブル内の「ステータス」列により示されているとおり、多種のステータスを取得します。

Running Jobs								
φ								
Instance IE	Job	Activation Time	Last Action	Status				
<u>29</u>	/public/MyTask	2019-02-05 12:26:00	2019-02-05 12:26:29	Finished successfully after step 2				
<u>30</u>	lo /public/MyTask	2019-02-05 12:27:00	2019-02-05 12:27:00	Running step 2	Stop job			
φ								

ジョブインスタンスかどのようにつの状態からもうこの状態に変更するかについて次のダイアグラムは示しています。FlowForce Server サービスの喪失、おけよ、ネトワークの中断が発生していないことが前提です。ステータスの一部は、短いマルシである場合が防火、ユーザーインターフェイスで常に表示されているわけではないことに意してくたさい。



以下のテーブルは、ジョブのすべての可能なステータスをリストしています。

ステータス	説明
作成済み	これは、アウンョンかどられる前の、ジョブの最初の状態です。 次の可能な状態は「開始」、「スロッ ト待ち」、「失敗」、「失敗」、「中断済み」です。
待機中	ジョブインスタンスがファイルシステム おけま HTTP トリガーによトリガーされた結果によしの状態におき起こされる可能性があります。ジョブインスタンスがトリガーされていますが、オプションがまだ

ステータス	説明		
	「解決するまでN 秒待つ」 オプランコンと、指定さえている処理期間がまた経過していないことを示しています。		
スロナ待ち	ジョブインスタンスがトリガーされていますが、実行キューは現在フルであることを示しています。実行 キューの実行スロットには制限があり、ジョブインスタンスの指定された数のみか同じキューとで変更 して実行されることができます。 <u>キューの設定の定義</u> も参照してください。このキューに到達するイン スタンスは、スロットが空くまで待機し、「スロット待ち」ステータスが与えられます。		
	割り込みが起きな、限り、通常「スロット待ち」の後のステータスは「開始中」です。		
開始中	このステータスは通常とても短いです。これは、ジョブインスタンスが空いているキューイン割り当てられ、 最初のステップがすくに開始されることを意味します。		
実行中	ジョブインスタンスが現在実行中であり、実行が完了するまでこの状態であること、まけよ、外部イベントが発生して、完了前に実行が停止されていることを指します。このステータスコよ「実行中のステップ」{step}なと関連したステップ番号が存在します。		
失敗	このジョブインスタンスの実行は、完了し、失敗としてみなされます。これは最後のステータスで、ジョ ブインスタンスをこれ以上実行する試みは発生しません。		
	「ステップ{step} の後に失敗しました」ステータスは追加で失敗がステップ番号に関連することを示しています。		
正常二完了	これはショブが正常に完了したことを示す最後の状態です。		
中断中	FlowForce ユーザーが「ジョブの停止」ボタンをクリックしたさめ、ジョブインスタンスが実行を停 止しようと試みます。ジョブの停止は、非同期なアグションのため、時間がかかる可能性があります。 「中断済み」ステートに切り替えられる前に、ジョブインスタンスは正常に完了する可能性がありま す。これが発生すると、ジョブは正常に完了したものとして報告されます。		
	前のステータスにステップ番号か存在する場合、「ステップ後に中断」{step}ステータスが「中 止中」の代わりに表示されます。		
中止済み	ジョブインスタンスは実行を完了しました。通常、このステータスは、予期されないシャットダウン語に 直接発生することを示しますが、ユーザーがジョブを停止したことを示します。これは最後の状態で、 ジョブの一部が完了していないことを示しています。		
	前のステータスにステップ番号か存在する場合、「ステップ後に中断」{step} ステータスが「中 止済み」の代わりに表示されます。		
中断済み	ジョブインスタンスの実行が中断されました(例えば、ネトワークケーブルが外れている場合。ワーカ ーマンムがラッシュしている、おけま、シャトダウンされている場合、および、強制的に中断されてい るイベトの場合など)。中断済み状態の強制的ないシエーションです。ジョブインスタンスは再起 動することができず、失敗として扱われます。ジョブの一貫性は保証されませんので、結果を手動で 確認することが奨励されます。		
接続の切断	複数のFlowForce インスタンスかクラスターで実行されている場合適用することができます。		
	このステータスはマスターマンンがフーカーマンンへの接続を失ってことを示しています。このジョブインス タンスのステータスを合理的に決定する方法は存在しません。ワークが戻ると状態に関する情報が 提供される場合があります。		
同期中	複数のFlowForce インスタンスかりラスターで実行されている場合適用することができます。		

ステータス	説明
	クラスターされたセナアップ内で、マスターマシンは定期的にワーカーマンカらジョブインスタンスの現在のプログレスを取得します。このステータスはフーカーへの失われた接続が復元されて事を示し、 FlowForce はフーカーからの最新のステータスを取得することを試みています。
追跡対象外	FlowForce Server プロセスか現在ジョブをトラックできない ことを示しています。 このステータス は、ジョブがフーカーマシンで作動中にマスターが止まオニ場合に発生する場合があります。
	「中断済み」と「追跡対象外」はは若干の違いがあることに注意してくたさい、前者の場合、 FlowForce Server はプロセスとして中断済みであることか既知ですが、後者の場合、何か発 生したのか把握できていません。
	「中断済み」 同様に、ジョブの一貫性に保証はなく、結果を手動で確認することが奨励されています。
置き換え済み	このステータスはショブが実際に何かを行う機会を得る前にショブが中断されること示しています。これは、重要な状態ではありません。FlowForce Server かどったる変更を検知し、前のショブインスタンスをキャンセルして、新しいものと作成したことを意味します。
	例えば、このステータスはファイルシステム、おけよ、HTTPトリガーの結果としてトリガーされたショブ インスタンスのために発生する場合があります。「解決するまでN 秒待つ」オプションにお指定 されている期間の前にジョブか変更したファイルのトリガーか解決された場合、ジョブインスタンスはこ のステータスを取得する場合かあります。
	このステータスは、以前に開始されたインスタンスがスロットを待っている間タイムトリガーと共に発生することもできます。

7.3 ジョブの停止

使用中のユーザーアカウト(おさま使用中のユーザーアカウトカジンバーであるロール)か特権を有することが前提で、現在作動中のジョブを停止することができます。ジョブの停止特権を参照してくたさい。

警告: ジョブの停止はデータの破損を招く場合かあり、特別な場合のみ行ってください。

ジョブを停止すると、FlowForce Server は、最初にショブの正常な停止を試みます。ショブの停止が招く結果を想定できない場合、正常な停止プロセスは、(定義されている場合)ジョブに関連したエラーの処理を試みます。正常な停止が可能では内場合、FlowForce は、一定の待ち時間の後、ジョブを強制的に停止します。FlowForce Server に強制的にジョブを停止するように命令するには、「ジョブ の強制停止」、ポタンをクリックします。

実行されているジョブを停止する

Г

1. 「ホーム」をクリックします。現在実行中のショブは実行中のショブページのセクションに表示されています。

Running Jobs								
Instance ID	Job	Activation Time	Last Action	Status				
61	/public/db2osm.job	2014-06-18 14:58:00	2014-06-18 14:58:00	Running step 1	Stop job			
¢								

2. 「ジョブの停止」をクリックします。FlowForce Server は アクションを確認するようパプロンプトします:

Stop running insta	nce of job /publ	ic/db2osm.job?
	ОК	Cancel

3. 「OK」をクリックします。システムはショブを停止しようと試みます。ジョブインスタンスが停止されると、関連する通知メッセージが表示され、対応するジョブの記録が、イライトされます。

	Stopping instance 61 of jo	ob /public/db2osm.job		
Welcome!				
Running Jobs				
Instance IE Job	Activation Time	Last Action	Status	
61 🔕 /public/db2osm.job	2014-06-18 14:58:00	2014-06-18 14:58:00	Stopped while running step 1	
φ				

4. ジョブが通常に停止されなかった場合、「ジョブの強制停止」をクリックします。

7.4 メインログの確認

ログビューカらFlowForce Server によりログされた特定のジョブに関する詳細を確認することができます。「ログ」メニューアイテム、おけよ 「ログビュー」 ポタムを表示される他のコンテキストカらログビュービアクセスすることができます。(例えば、各ジョブの構成ページは「ログビュ ー」ボタンを表示します)。

メモ デンオルトで、読み取りアクセスのあるショブ全てのログをビューすることができます。FlowForce Server 内のすべてのショブとイベントのグロー・シルンログをビューするコよ、使用中のアカウトかシィルターされていたいログのビュー特権を持っていることが必須です。

Log View										
Date from: 🛍 🔹 🔍		Dat	e to: 🛍 💌	0	Object path:	•	Minimum severity: Inf	• •	Show	Reset Filter
Date	Severity	User		Instance ID	Message					
 Show 25 more records 	 Show 100 mo 	ore records	 Show 500 more records 							
2020-09-17 16:06:08	INFO	root		<u>16</u>	Finished job execution: job	/public/Examples/Co	<u>pyFiles</u> in queue <u>/public/E</u>	amples/Co	<u>pyFiles</u>	
2020-09-17 16:06:08	INFO	root		<u>16</u>	Starting instance 16.					
2020-09-17 16:06:08	INFO				Service /CopyFilesService o	of <u>/public/Examples/C</u>	opyFiles invoked			
2020-09-17 16:04:59	INFO	root		<u>15</u>	Finished job execution: job /public/Examples/HelloWorld in queue /public/Examples/HelloWorld					
2020-09-17 16:04:59	INFO	root		<u>15</u>	Starting instance 15.					
2020-09-17 16:04:59	INFO				Service /HelloWorldService of /public/Examples/HelloWorld invoked					
2020-09-17 16:04:57	INFO	root		14	Finished job execution: job /public/Examples/CopyFiles in queue /public/Examples/CopyFiles					
2020-09-17 16:04:57	INFO	root		14	Starting instance 14.					
2020-09-17 16:04:57	INFO				Service /CopyFilesService o	of <u>/public/Examples/C</u>	opyFiles invoked			
2020-09-17 16:04:55	INFO	root		<u>13</u>	Finished job execution: job	/public/Examples/Co	<u>pyFiles</u> in queue <u>/public/E</u>	amples/Co	<u>pyFiles</u>	
2020-09-17 16:04:55	INFO	root		<u>13</u>	Starting instance 13.					
2020-09-17 16:04:55	INFO				Service /CopyFilesService o	of <u>/public/Examples/C</u>	opyFiles invoked			
2020-09-17 16:04:53	INFO	root		<u>12</u>	Finished job execution: job	/public/Examples/Ch	neckPath in queue <u>/public/l</u>	xamples/C	heckPath	
2020-09-17 16:04:52	INFO	root		<u>12</u>	Starting instance 12.					
2020-09-17 16:04:52	INFO				Service /CheckPathService	of <u>/public/Examples/</u>	CheckPath invoked			
2020-09-17 16:04:37	INFO	root		<u>11</u>	Finished job execution: job	/public/Examples/He	elloWorld in queue <u>/public/</u>	Examples/	HelloWorld	
2020-09-17 16:04:37	INFO	root		<u>11</u>	Starting instance 11.					
2020-09-17 16:04:37	INFO				Service /HelloWorldService	of <u>/public/Examples</u>	/HelloWorld invoked			
2020-09-17 16:04:35	INFO	root		<u>10</u>	Finished job execution: job	/public/Examples/Co	pyFiles in queue <u>/public/E</u>	amples/Co	<u>pyFiles</u>	
2020-09-17 16:04:35	INFO	root		<u>10</u>	Starting instance 10.					
2020-09-17 16:04:35	INFO				Service /CopyFilesService o	f <u>/public/Examples/C</u>	opyFiles invoked			
2020-09-17 16:01:54	INFO	root		<u>9</u>	Finished job execution: job	/public/Examples/Ch	neckPath in queue <u>/public/l</u>	xamples/C	heckPath	
2020-09-17 16:01:54	INFO	root		<u>9</u>	Starting instance 9.					
2020-09-17 16:01:54	INFO				Service /CheckPathService	of <u>/public/Examples/</u>	CheckPath invoked			
2020-09-17 16:01:48	INFO	root		<u>8</u>	Finished job execution: job	/public/Examples/Ch	neckPath in queue <u>/public/l</u>	xamples/C	heckPath	
✓ Show 25 more records	✓ Show 100 me	ore records	✓ Show 500 more records							
Export View Exported	I Log for Instance	Copy Perm	nalink to Clipboard							

ロクビュー

[情報で設定された最小の重要度と共にデフォルトで「ログビュー]ページは最近にエグされたイベトを表示します。[25 件以上のレコードを 表示する] [100 件以上のレコードを表示する] おさは [500 件以上のレコードを表示する] ボタンをクルクして古いレコードをロード することができます。

日付 列はログされたイベトカ発生した日時を示しています。重要度 列はログされたイベトの重要度を目に報、「警告」、「エラー」によい示しています。ユーザー列はイベトのオリジンを指定します(これは Flow Force サービス、Python セキューリティサービス特定の Flow Force ユーザーであることができます)。

ジョブの各実行は一意のジョブインスタンスを生成します。一意のジョブインスタンスは インスタンス ID 列によりトラックされています。インスタン ス ID によるイベトをフィルターするオーダに インスタンス ID 列に表示されてい マレンケをクトックします。これは現在選択されてい るインスタンス に関する全ての清報を個別のページで開きます。 ログインスタンスの確認を参照してくたさい。これは適用不可能なオーダ、ログされたイベント の一部にはジョブインスタンス ID が存在しません。この場合、例えば、ジョブ構成のオーダにイベントを変更します。

メッセージ 列にはイベトをトリガーしたショブの構成ページへのレクか含まれている場合からます。

グリッド上の列のサイズ調整を行う場合、列の見出しを区切る垂直の、一の、ずれかをクリックし、左側のマウスを押したまま、左右にドラッグします。

永久リンクをクリップボードにコピーするボタノは選択されたッテメーターを含むログビューの現在のURLをクリップボードにコピーします。 例、?id=2773968&limit=25。これは同じ情報を後にページに素早くロードすることを希望する場合役に立ちます。例えば、永久 URLを他のブラウザーアドレスバーコ張り付け、まけは同じログビューを確認するために誰かに送信することができます。

ログのエクスポート

ログをディスク上のファイルコンクスポートする場合エクスポートをクリックします。ページ内で現在表示されているすべてのレコードはジップされた JSON (JavaScript Object Notation) ファイルとしてエクスポートされます。「エクスポート」をクリックすると、ZIP アーカイブをローカル ディノケーリニ保存する、おさは、ブラウザーのデフォルトのダウンロードディノケーリニ保存するようにプロンプトされます。

ログをフィルターする方法

以下の条件でログをフィルターすることができます

- 開始日 この日付以降のイベトが含まれます。
- 開始日-この日付までのイベントか含まれます。
- オブジェクトノロー選択されたいマでの上行性から発生するイベトのみか含まれます。ジョブませま重要なレコードなど FlowForce オブジェクトへの、マを選択することが可能です。
- *最小の重要度* 重要度が最小に設定されたイベトのみが含まれます。例えば、全ての重要度のイベトを含む場合、最小の 重要度として[情報を選択します。両方の情報メッセージと警告をスキップするために、最小重要度として[エラー]を選択します。

開始日と終了日 フィルターの両方を設定した場合、その範囲内のレコードが1000 件のレコードまで表示されます。追加のレコードを確認する場合、 [更にN件のレコードを表示] ボタンをクリックします。 最近の日付を持つレコードが常に最初に表示されます。

フィルターの、ずれかを変更後 表示 ポタンをクトックしてフィルターを適用します。フィルターを保ちつつ表示 ボタンを繰り返しクトックして最近 のログをロードします。更にN 件のレコードを表示 ポタンの、ずれかをクトックすると現在選択されているフィルターも保存されます。

フィルターのリセット ボタノはすべてのフィルターとログの更新をクリアします。フィルター無しの表示 ボタンをクリックしても同じ効果(ログの更新)が得られます。
7.5 インスタンスログの確認

[インスタンスログ]ページはショブ(ませましログインスタンス)).の特定のログされているインスタンスに関する詳細を提供します。このページを以下の方法で開くことができます:

- <u>ログビュー</u>ページ内のレコードのインスタンスID リンクをクトックします。
- ホームページ内の最近および実行中のセグョン内のレコードのインスタレスID リンクをクトックします。
- ジョブ構成ページからログをビューをクトックし ログビュー ページン移動し、レコードのクレスタレスID リンケをクトックします。

Log for instance 4			
Created instance:	2020-10-01 17:13:19		
Job Name:	/public/Examples/CopyFiles		
Queue Name:	/public/Examples/CopyFiles		
Current or Last Step:	2		
Postponed Steps Failu	res: 0		
Current State	Finished successfully		
current state.	Thistica successiony		
Data	Marrage		
2020-10-01 17:13:19	Starting instance 4.		
2020-10-01 17:13:19	Starting execution of job /public/Examples/CopyFiles in queue /public/Examples/CopyFiles		
2020-10-01 17:13:19	Running instance 4 locally.		
2020-10-01 17:13:19	Execute main job /public/Examples/CopyFiles		
2020-10-01 17:13:19	Execute function /system/compute		
2020-10-01 17:13:19	For each item in <i>list</i>		
2020-10-01 17:13:19	Iteration 1		
2020-10-01 17:13:19	Iteration 2		
2020-10-01 17:13:19	Execute function /system/filesystem/copy		
2020-10-01 17:13:19	Executing FlowForce.copy with parameters: Source: "C:\FlowForceExamples\CopyFiles \source\orders.txt", Destination: "C:\FlowForceExamples\CopyFiles\target", Overwrite: true(), AbortOnFail: true(), Working-directory: ""		
2020-10-01 17:13:19 Step FlowForce.copy completed successfully			
2020-10-01 17:13:19 Finished job execution: job /public/Examples/CopyFiles in queue /public/Examples/CopyFiles			
Export View Exported Lo	g for Instance		

インスタンスログ

適用可能な場合[インスタンスログ]ページは以下のデータのカテゴ」をレポートする場合かあります

- ジョブインスタンスの実行中に生成されるメッセージはステップごとごグループ分けされています。以下が含まれます:
 - ビルトイン関数おけはマンピングの実行からのメッセージと、ウメーター。
 - o compute とcompute-string 関数を実行するステップの結果。

- [成功/失敗を持つ実行]ハンドラーステップおとおうブの失敗内のエントリに導くエラーメッセージ。
- ステップの実行間に経過した時間に関する情報。
- [for-each] ステップの反復
- ジョブが再試行された回数に関する情報 エラー時に再試行を参照してくたさい。
- マピングの実行、おは commandline 関数により生成されたストリーム

エクスポート ボタノは現在のログインスタンスに関連したすべてのデータ.zip アーカイブを作成します。ログされたインスタンスをディスク上の .zip アーカイブニエクスポートした場合、インスタンスのためにエクスポートされたログをビューボタンをクトックして確認することができます。

インスタンスのためにエクスポートされたログをビューボタンにより以前に保存された.zipアーカイブをページコードすることができるようにないます。

.zip アーカイブからロードされたログインスタンスの確認が完了するとエクスポートされたログビューを閉じるボタンをクリックします。

7.6 統計詳細ページ

次の6個のチャートはとう系詳細ページを提供しています。

- 過去30日間に実行されたジョブーは過去30日間に実行されたショブの個数と、実行の結果を表示します。
- 過去24時間に実行されたジョブーは過去24時間に実行されたショブの個数と、実行の結果を表示します。
- 過去60分に実行されたジョブーは過去60分に実行されたジョブの個数と、実行の結果を表示します。
- 過去30日間にトリガーされたジョブーは過去30日間にトリガーされたショブの個数と、実行の結果を表示し、ジョブインスタンス がトリガーされた方法を表示します。
- 過去24時間にトリガーされたジョブーは過去24時間にトリガーされたショブの個数と、実行の結果を表示し、ジョブインスタンスがトリガーされた方法を表示します。
- 過去60分にトリガーされたジョブーは過去60分にトリガーされたジョブの個数と、実行の結果を表示し、ジョブインスタンスがトリ ガーされた方法を表示します。

上に示されている通り、チャートがセナあり、以下のようこう類することができます。

- 1. [実行の結果]チャート
- 2. [トⁱガー型チャート

各セナ内の3個のチャートは以下の期間をカバーしています:過去30日、過去24時間、および過去60分。

[実行の結果]チャート

ジョブインスタンスの実行の結果は以下の状態のつです

- 成功-ジョブインスタンスの実行が成功したことを示しています
- 失敗 ジョブインスタンスが実行中に失敗したことを示しています(例えば、エラーが存在しない、などのエラーに遭遇したなど。)
- 中断済み-ジョブインスタンスが中断されたことを示しています(例えば、ハードウェアおけよサーバーの失敗など)

サンプルとして、下に表示されるチャートは過去24時間内の実行の結果を示しています。チャート1は実行中に成功した1個のジョブと、失敗した4個のジョブである5個のジョブインスタンスをレポートしています。





チャート 1. 過去 24 時間に実行されたジョブ

ラベルをクトックして任意のデータセナをオフロジル替えすることができます。チャート2は成功データセナがウトックされており、レポートから除外されていることが表示されています。



チャート2.成功がオフニ切り替え済み

[トリガー型チャート

このセクタンはジョブインスタンスを開始するトリガーの3 つの種類にていて説明しています。

- タイマー-特定の時刻に実行されるようにプログラムされているためにジョブインスタンスが実行されます。 タイマートリガーを参照してくたさい。
- ファイル-HTTP おさなアイルシステムの変更が発生するためジョブインスタンスが実行されます(例えば、新規のファイルがデル/ ケリに追加される場合など)。ファイルシステムトリガーとHTTPトリガーを参照してください。
- サービス プログラムまけはユーザーかそのショブに関連したWebサービスを呼び出した場合、ジョブインスタンスが実行されます Webサービスとしてショブを公開するを参照してください。

下に表示されるチャートは過去24時間内はトリガー型により実行された統計を反映しています。具体的には、過去24時間に5個のジョブインスタンスが実行され、その内2個のインスタンスがタイマー時よりトリガーされ、3個がWebサービス呼び出しによりトリガーされていることを示しています。



(トリガー型により過去 24 時間内/よりガーされたジョブ

チャートデータの検索

統計ページ上のチャートはホームページ上のチャートと同じ性質を持っています、具体的には以下が挙げれます

- チャートの特定の、トーニマウスをポイントすると、対応する期間をカレデーする詳細情報と共によったも表示されます。
- チャート上の特定の情報を更に詳しく知るコはチャートからログビューページン、簡単にナビゲートすることができます。これを行うコよ チャートでフォーカスされるバーをクリックします。これは、与えられた日、時間、分のために既にフィルターされているログと共にエグページを表示します。
- メモ チャートとログによりトラックされるログの詳細との間に若干、および影響の小さ、差分が存在します。

8 FlowForce 式

FlowForce 式は、ジョブが作動する際にFlowForce Server にお計算され実行されるステートメントです。FlowForce 式は、ジョブ 内の複数のステップをつなける FlowForce にお理解される基本のスクリプト言語です。FlowForce 式はよ 以下のコンテキストが通常 必要とされます:

- ビルトイン関数の・ラメーター内(つまり、ジョブ構成ページ内の入力フィールドに式を書き込む、まけよ、埋め込むことができます)。
 以下は、例の一部です:
 - 実行ステップによ返された結果のデータ型を変更します。
 - 。 値の配列を返す結果から特定の値を選択します。
 - 文字列を作成するためご複数の値を結合します。
- 条件付きのステートメントを「when」ステップ内で作成します。これにより、ブール値がtrueを評価する式を与えるとステップを実行することが可能になります。
- 「for-each」ステップ内。「for-each」ステップを使用することにより、シーケンスが式により定義されている箇所のアイテムのシーケンスをループすることができます。

このセグランでは、上にリストされたシナリオのためにFlowForce式を構築する際に手助けたる概念にていて説明されています。

8.1 式の計算

①/system/compute 関数を呼び出す実行ステップを作成することにより、ジョブは埋め込む前にFlowForce 式を簡単にテストすること かできます。順序を追すゴ列は関しては、以下を参照してくたさい、<u>"Hello, World!" ジョブの作成</u>。

□/system/compute 関数は「式」、ウメーターの値を評価し、計算された結果を返します。 重要な点は、この関数は定義された 戻り値の型を持ちません。 実際の型は、計算される式により異なります。 例えば、この関数を式 1+1 につなすると、 関数は、 数値 2 を返し ます。 しかしなから、 例えば、この関数を式、 1+1、」につなすると、 関数は、 文字列の値 1+1 を返します。

この概念をより良く理解するために、 1/system/compute 関数を呼び出すステップを作成し、式フィールド内に "1+1"を 入力します。ジョブの戻り値の型を下に示されるように「文字列」として宣言してください。

実	【行	ステップ
	•	関数の実行 /system/compute
	1	バラメーター: 条件式: 1+1
	=	ステップの結果を割り当て名前 としてTO
	新規	見の実行ステップ 新規の選択ステップ 新規の For-each ステップ 新規エラー/成功処理ステップ
実	ミ行	の結果
	戻り	値の型を以下と宣言する: 文字列 ▼

ジョブを保存しようと試みると、FlowForce は、「型文字列と数値が一致しません」エラーを表示します。計算された 式が数値で、ジョブの戻り値の型が文字列の値として宣言されているため、このエラーが生じました。 入力に関する問題を解決するために、ジョブの戻り値の型を「数値」に変更するか、または数値を文字列に変換し ます。下の例は、数値を文字列の値に変換する FlowForce 式関数 string を呼び出します。

実行ステップ				
•				
4	関数の実行	/system/co	mpute	
	バラメーター:	条件式:	string(1+1)	

式を計算して、値を文字列として返す場合、代わりに①/system/compute-string 関数を使用することができます。この場合、文字列 から中かってを使用して区別される必要かあることに注意してください、以下を参照: 文字列フィールド内に式を埋め込む)。

8.2 式言語のルール

FlowForce 式内でエラーを回避するためコよ、以下のルールを守ってくたさい

- 許可されたまけは宣言された値のみを使用してくたさい。
- 文字列をそのまま使用するコよ 一重引用符で囲んでくたさい。
- 文字列フィールド内に式を埋め込むコよ 中かって、つまり、{と}文字で囲んでください。
- 式は入力された箇所のフィールド内で意味を持つ必要があります。

式言語のルールの詳細を確認してみましょう。

ルール#1:許可された、ませよ、宣言された値のみを使用します

FlowForce 式内では以下の構造文を使用することがで許可されています:

- FlowForce 式関数(完全なレファレンスに関しては、以下を参照してくたさい、式関数)
- FlowForce 演算子(以下を参照:<u>演算子</u>)
- 数値
- 文字列の値
- 前に宣言された値

FlowForce 式か許可されているフィールドのテキスト内部に入力すると、リアルタイムで構文のチェックが行われます。構文が正確ではない場合、FlowForce は、文字を赤でいイライトします。以下は構文検証エラーのサンプルです:

関数の実行	/system/filesystem/co	ру	▼ 2	
パラメーター:	ソース:	{source}	<u></u>	文字列 (必須) セットする値 🕨
	ターゲット:	{target}	0	文字列 (必須) セットする値 🕨

ソース おけまターゲット かジョブ内で宣言されていない サムウ、 FlowForce か気を解釈できない サムウ、エラーか発生します。問題によれらの 値を「例えば、ジョブ入力・デメーターとして〕宣言することによ、解決されます:

ジョブ入力パラメーター					
名前:	source	型: 文字	<u>ارة ا</u>	▼ デフォルト: 🔸 説明:	
● 名前:	target	型: 文字	- <u>7</u>]	▼ デフォルト: 🔸 説明:	
•					
実行	実行ステップ				
•	関数の実行	/system/filesystem/o	ору	▼ @	
	バラメーター:	ソース:	{source}		
		ターゲット:	{target}		

ルール#2: 文字列を一重引用符で囲みます

文字列をそのまま使用するコよ、一重引用符で囲んでくたさい。それ以外の場合、式は、期待しない結果を出す場合があり、また、検証 に失敗する場合があります。次の例を考慮してくたさい

式	評価	説明
1+1	2	値のデータ型は、数値です。
'1+1'	1+1	値のデータ型は、文字列です。
1+1==2	true	値のデータ型は、ブール値です。

データの型を他の型に変換する必要がある場合、FlowForce 式関数を使用してください(以下も参照してください ルール#4)。

ルール#3: 文字列フィールド内で中かっこを使用します

文字列フィールド内に式を埋め込むける、中かって一囲んでくたさい。下のサンプルでは、中かっては(FlowForce 式関数である)式 instance-id()を文字列の残りから区切ります。

	関数の実行 /system/mail/send		
	バラメーター:	差出人:	someone@example.org
一 一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一		· 劳开.	
件名:	90元:	someone@example.org	
	件名:	Job{instance-id()}has completed.	
			· · · · ·

フィールド全体が型「式」の場合、中かってを使用しないでくたさい。例えば、system/compute ビルトイン関数の「式」、デメーターでは、関数はこの型を持ちます。(中かってが使用されていない)下の例はこのフィールドのための正しい値の使用の例です:

関数の実行 /system/cor	mpute		▼ 🛃	
バラメーター: 条件式:	concat('a','b','c')			条件式 T0 (必須)
ステップの結果を割り当て	名前	としてT0		

式フィールド内に中かってを入力することは、構文エラーをトリガーします:

関数の実行 /system/cor	npute	▼ 2	
バラメーター: 条件式:	(concat('a','b','c')		🚺 条件式T0 (必須)
 ステップの結果を割り当て	名前	としてT0	

以下も参照してくたさい、文字列フィールド内に式を埋め込む。

ルール#4:正確なデータ型を使用してくたさい

最後に、FlowForce は、ジョブを保存する際に、データの型のチェックを行います。フィールドによ期待されるデータ型に一致しないフィール ドに式が入力されると、エラーが発生します。各フィールドできたしされるデータ型は、各フィールドの右端に表示されています。例:

関数の実行	/system/filesystem/move 🗸 🖉	
バラメーター:	ע-ג:	文字列 (必須) セットする値 >
	保存先:	文字列 (必須) セットする値 🕨

ですから、1+1 などの式は、明示的に数値とて評価されるため、文字列のフィールドでは有効ではありません。その一方、式 '1+1' は、有効な文字列フィールドです。次の例を考慮してくたさい

式	評価	説明
1/4	0.25 (数値とて)	値のデータ型は、数値です。
		この武をフィーリド内、おけよ数値を期待するコンテキストで使用してく たさい。それ以外の場合、ジョブの検証は失敗します。
1+1==2	true(ブール値とて)	値のデータ型は、ブール値です。
		この式をフィーリド内、おけま、ブーリ値の値を期待するコンテキストで使用してください。それ以外の場合、ジョブの検証は失敗します。
'apple'	apple(文字列とて)	値のデータ型は、文字列です。
		この武をフィーリド内、おけよ、文字列の値を期待することテキストで使 用してください。それ以外の場合、ジョブの検証には失敗します。
concat ('1','2','3')	123 (文字列として)	値のデータ型は、文字列です。
		この武をフィーリド内、おけよ、文字列の値を期待することテキストで使 用してください。それ以外の場合、ジョブの検証には失敗します。
1+'apple'	-	この式は有効ではかません、FlowForce は、ジョブを保存しようとす ると、エラーを返します。2つの異なるデータ型(文字列と数字)が比較され ているため、検証を行うことはできません。
{content	[](文字列とて)	この式は、2つのネストされた式関数を使用します:
(stdout (<i>result</i>))}		 関数 stdout は シェルコマイの標準出力をストリームとして取得します。 関数 content は ストリームの値を文字列に変換します。
		式は、正確ですが、ジョブよ、以下の条件が満たされている場合のみ検 証を成功することができます:
		 値「結果」は既に宣言されています。 値「結果」シェルロマ・ドの標準出力を含んでいます。 式は文字列フィールドに生め込まれています。
		以下も参照してけさい、式関数の呼び出し。

8.3 文字列フィールド内に式を埋め込む

文字列フィールド内でFlowForce 式を使用するコよ、式を中かって囲んでください。つまり、 "{ " and "} " 文字で囲んでください。文字列フィールドの式の部分は、通常、文字列の他の部分特別しやすい 薄し紫色の背景色で表されてします。 例

関数の実行 /system/mail/send		
バラメーター: 差出人:		someone@example.org
宛先: 件名:	宛先:	someone@example.org
	件名:	Job{instance-id()}has completed.

文字列フィールド内では、式のみが中かつに囲まれ、FlowForceにおばして扱われます。FlowForceに"{" and "}" 文字をそのまま解釈するように指示する場合は、一重引用符の代わりに中かってを使用してくたさい。以下のケースを考慮してくたさい

次の値を持つ文字列フィールド	評価	説明
echo Hello, World!	echo Hello, World!	(文字列は、埋め込まれた式を含んでいません)文字 列は中かってを使用しませんですから、そのまま評価 されます。
echo {<i>Hello</i>, World	-	文字列を評価することができません。埋め込まれた式 が構文的に正確ではおいため、FlowForce は、 構文エラーを表示します。
echo {'Hello, World!'}	echo Hello, World!	文字列は、構文的に正し、埋め込まれた式を含み ます。しかしながら、式は、文字列フィールドの内部 にあり、評価結果は、式が使用されていない場合と 同じにてみます(最初の例を参照してくたさい)。
echo {{'Hello, World!'}}	echo {'Hello, World!'}	エスケープ文字 {{ and }} が使用されたさめ文字列 は、式を含みません。

8.4 式関数の呼び出し

FlowForce 式言語は、(主に、実行すてぷこと版される値を扱う基本的なオペレーションを実行するための一連の関数を含んでいます。 FlowForce 式が有効なコンテキスト)からこれらの関数を呼び出すことができます、(例えば、関数の、ラメーターを表すテキストボックス内に入 力することなど)。

FlowForce 式関数とFlowForce ビルトイン関数を区別する必要がみます。ビルトイン関数は、FlowForce 実行ステップから呼び出されますが、(つまし、ステップとして実行されますが)、式関数はFlowForce 式から呼び出されます。

式関数を呼び出す典型的なシナリオは2つの実行ステップから構成されている、下て説明されているショブが挙げれるてしょう。

最初のステップはシェルコマンドを実行します(具体的には、"Hello, World!"を出力します)。このステップによし返されるデータ型は、「結果として」であり、戻り値は、var1として宣言されています。

2番目の実行ステップは 11/system/compute-string ビルトイン 関数を呼び出します。この関数は、var1 を文字列に変換するため に呼び出されます。 式自身は、文(中かってで示されている)字列フィールドに埋め込まれており、2つのネストされた式関数を呼び出します。

- 関数 stdout は、シェルコマドの標準出力をストリームとして返します。
- 関数 content は ストリームの値を文字列に変換します。

実	行	ステップ				
	•	関数の実行 /sv	ystem/shell/comma	ndline		
		パラメーター:	コマンド:	echo Hello, Wo	orld	
			エラー時には中断: 作業用ディレクトリ:	+ +		
	=	ステップの結果を	を割り当て var1		以下としてresult	
	•	関数の実行 /sy	ystem/compute-stri	ng		
		パラメーター:	条件式: {content	(stdout(var1))}		
	=	' ステップの 結果を	を割り当て var2		以下としてstring	

データ型の変換が完了し、文字列の値 var2 をジョブの処理ロジックム従い、更に(例えば、電子メールへの送信などに)使用することができます。

使用することのできる式関数に関しては、以下を参照してくたさい、式関数。

8.5 FlowForce データ型

次のデータ型を使用してFlowForce を操作します。

文字列

文字列の値を表します。例: 'Hello, World!'。

数値

数値を表します。例: -1, 0, 56, 0.45565.。

ブール値

true おっはfalse 値を表します。

結果

実行ステップにより生成された結果を表す抽象的な型です。

実行ステップは、MapForce マンピング、StyleVision 変換ファイル、シェル関数、など、多種の実行可能ファイルを処理します。 result データ型は、ですから、そのようなファイルの出力を表します。

実行ステップが、MapForce マッピングを実行する場合、出力は以下の可能性があります:XML、XBRL、テキスト、JSON、および MapForce にお生成される他のファイル型。

実行ステップが、StyleVision 変換を実行する場合、出力は以下の可能性があります: PDF、W ord、HTML ファイル および StyleVision により生成される他の出力の型。

結果の値にアクセスするコよ名前を与え、(例えば、「output」)、{result} 式関数にパレます。これにより、ストリーム式関数を使用して値をストリームに変換し、更にこの値を処理することができます(以下も参照してくたさい、式関数の呼び出し)。

実行ステップがシェルコマイを実行すると、出力に応じ特定のステップ結果式関数を呼び出します。例えば、ストリームとしての標準出力を 返すためゴよ式 {stdout(output)} を使用します。標準エラーをストリームで返す場合、式 {stderr(output)}を使用します。 詳細に関しては、以下を参照してくたさい、ステップ結果関数。

結果

MapForce マピング おけま Style Vision 変換か複数のオブジェクトを返す場合かあります。このようなステップにより生成される結果は、データ型として results を持ちます。

このような出力を扱うコよ、ストリームの配列を返す {results (output) } 式関数を使用します。nth 関数を使用して、配列から特定のストリームを選んでください。

例えば、下で説明されるジョブはFlowForce にデプロイされたStyleVision 変換ファイルから作成されました。このジョブは、入力パラメーターとして、XML ファイルを取り、多種のフォーマナの複数の出力を返します。

関数(0)実1」/P	ublic/AutoCalc.transformat	tion		▼ 🖻					
パラメーター:	InputXml:	•	altova://packagedfile/Data.xml			xs:string	(必須)	セットする値 🕨	
	OutHtml:		AutoCalc.html			xs:string	(オプション)	セットする値・	
	GenerateHtmlOutputAsM	1ime:				ブール値	(オプション)	セットする値 ▶	
	OutRtf:	₽ RTF	AutoCalc.rtf			xs:string	(オプション)	セットする値・	
	OutFo:	FO	AutoCalc.fo			xs:string	(オプション)	セットする値・	
	OutPdf:	PDF	AutoCalc.pdf			xs:string	(オプション)	セットする値・	
	OutDocx:		AutoCalc.docx			xs:string	(オプション)	セットする値・	
	Working-directory:		c:\temp			xs:string	(必須)	セットする値 ▶	
関数の実行 /s)	vstem/compute			 ∎ 					
11=1-4-	冬件式,		0))		冬件式TO(以)(2)				
パラメーター: ステップの結果	条件式: as-file(nth(resu を割り当て step2	ılts(step 1)	.0)) 以下としてT0		条件式T0(必須) 👌	セットする値			
パラメーター: ステップの結果 関数の実行 /sy	条件式: as-file(nth(resu を割り当て step2 ystem/filesystem/copy	ults(step 1)	0)) 以下としてTO	•@	条件式T0(必須)	セットする値	•• 2		
パラメーター: ステップの結果 関数の実行 /sy パラメーター:	条件式: as-file(nth(resu を割り当て step2 ystem/filesystem/copy ソース: (s	Ilts(step 1)	0)) 以下としてTO		条件式T0(必須) 文字列 (セットする値 必須)	 レントする 	ă • 2	
パラメーター: ステップの結果 関数の実行 /s パラメーター:	条件式: as-file(nth(resu を割り当て step2 ystem/filesystem/copy ソース: (s ターゲット: cc	Ilts(step 1) step2}	0)) 以下としてTO AutoCalc.html		条件式T0(必須) 1 文字列(文字列(セットする値 必須) 必須)	 レットする/ セットする/ 	ă ▶ 2 ă ▶ 2	
パラメーター: ステップの結果 関数の実行 /s パラメーター:	条件式: as-file(nth(resu を割り当て step2 ystem/filesystem/copy ソース: (s ターゲット: cc 上書き: 5	Ilts(step 1) step2} :\archive\	0)) 以下としてTO AutoCalc.html		条件式TO (必須) 1 文字列 (文字列 (ブール値(セットする値 必須) 水須) オプション		ă ▶ 2 ă ▶ 2 ă ▶ 2 (ă	
パラメーター: ステップの結集 関数の実行 /s パラメーター:	条件式: as-file(nth(resu を割り当て step2 ystem/filesystem/copy ソース: (s ターゲット: cc 上書き: E エラー時には中断: 4	ults(step 1) step2} :\archive\	0)) 以下としてTO AutoCalc.html	. C	条件式TO (必須) 1 文字列 (文字列 (ブール値 (セットする値 必須) 水須) オプション	 セットする(セットする(セットする(ă → 2 ă → 2 ă → 2 ē	

このショブコおつの実行ステップが存在します。

- 1. 最初の実行ステップは実際のデータ変換をおこないます。
- 2番名の実行ステップはこの変換の結果を取得するけっと、FlowForceのcompute 関数を呼び出します。具体的には式 as-file(nth(results(otput), 0)) はストリームの列挙内の最初のアイテムをアイルとて取得します。列挙にはゼンベースのインデックスが存在するける、式は、「1」ではよく、「0」を使用します。
- 3. 3番目のステップは c:\archive ディレクトリにHTML ファイルをコピーします。

アイテム

リストをアセンブルまけは分解する式を作成する必要がある場合があります(以下を参照 リスト関数)。リストはジェネトックな型item の オブジェクトのリストから構成されています。アイテムはは、抽象的なデータ型を持ちます。リストを作成するオブジェクトの型に従い、(文字列、 数値、おけはストリームであることができる)itemのデータ型を決定することができます。リストは同じデータ型のアイテムのみを含むことしかで きないてとご注意してください。

下のイメージはリストか数値によく構成されているため、「item」か数値の型であるループを表しています。



リストを活用するステップバイステップのサンプルは関しては、ファイルをエピーするを参照してくたさい。

8.6 演算子

FlowForce 式を構築するコよ、下にリストされる演算子を使用することができます。ビルトイン関数 つsystem/compute を使用して全ての式をテストできることを忘れないでくたさい。

演算子	説明	例
==	a がっと等しいかチェックします(数値的に番号が同じです。文字列のためのコードポイント)。	2 + 3 == 5 は true を算出します。
		2 + 3 == 4 は false を算出しま す。
!=	a がb と等しくない かチェックします。 次の3 つの式 は同じで a != b not (a == b) a <> b	2 + 2 != 5 は true を算出します。 3 + 2 != 5 は false を算出しま す。
<	a がb より小さ めチェックします (数値的に番号が小さいです。文字列に 関しては 以下を参照してくたさい)。	4 < 5 は true を算出します。
<=	a がっより小か等しし かチェックします。	5 <= 5 は true を算出しま す。
>	a がb お大き かチェックします	5 > 1 は、true を算出します。
>=	a がっより大か等しし かチェックします。	5 > = 5 は true を算出しま す。
+	加算	1+1は2を算出します。
-	減算	2 - 1 は、1 を算出します。
*	• · · · · · · · · · · · · · · · · · · ·	3*2 は、6を算出します。
/		6/3は2を算出します。

文字列の比較は以下のようこ行われます

- 2つの文字列の共通のプレフィックスは無視されます(コードポイントで評価されます)
- 残しつの文字列が空では内場合、最初のロードポイントが数値的に比較されます。
- 空の文字列は空では内文字列よりに小さい考えられます。

かってを使用して、FlowForce にかっての中の式を最初に計算するように命令します。例

2+3*4は、14を算出します。

(2 + 3) * 4 は、20 を算出します。

9 Altova 製品との統合

動作の説明内では、Altova 製品間での共同作業の概要が説明されて、ますが、基本的には、Altova MapForceにお作成されたマレングアイルとAltova Style Visionにお作成された変換ファイルは、次のサーイ・製品を使用することにお自動的に作成することができます。MapForce Server(おけはMapForce Server Advanced Edition)とStyle Vision Server。MapForce Server とStyle Vision Server は、サーイ・環境(Windows、macOS、Linux)内でマレングと変換をマイドラインで、おけはAPI からの呼び出し、実行することを可能にしまた。これらの製品は、FlowForce とは同時に実行されません。自動化けよAPI を呼び出す、おけよ、MapForce Server おけま Style Vision のコマイラインを呼び出すプログラムを開発、おけよスクリプトを作成することをのます。更に、FlowForce Server 管理下で作動している場合、RaptorXML Server 内で使用可能な関数はFlowForce Server ジョブから呼び出すことができます。

MapForce Server とStyleVision Server はコマドライン、おけよ API 呼出から複数のプラルフォーム(Windows、macOS、 Linux) でマピングと変換を作動することができます。FlowForce と共にこれらの製品が作動していない場合、自動化はAPI を呼び出す スクリプトの作成、おけよ MapForce Server おけはStyleVision Server のコマドラインの呼び出しを含みます。

MapForce Server とStyleVision Server がFlowForce Server 管理下で実行される場合、自動化を次のレベルで実行すること ができます。すなオち、マピングと変換を直接にFlowForce Server にデプロイすることができ、ジョブを実行できることを意味します(次を 参照してくたさい、しくみ)。このように、マピングまけは変換は以下を含むFlowForce Server ジョブの全ての利点を活用することができ ます:スケジュールされた、まけは、オンデマードのジョブの実行、WebサービスAS2 統合、としてのジョブの実行、FlowForce 式を使 用した構成、エラーの処理、条件付けられた処理、電子メールによる通知など。

FlowForce Server にデプロイされると、デプロイレたマメピング、ませま、変換がエレテナー内に表示されます。以下に示されるとおり、マメピングコは.mapping 拡張子、変換コは.transformation 拡張子が存在します。

C / O C public					
□ 名前	型 🗢				
🛛 🗊 AutoCalc.transform	nation StyleVision 変換				
CompletePO.mapp	ing MapForce マッピング				

FlowForce の観点からよ このようよオブジェクトは関数であり、このよか、新規のショブに変更することができます。これらよ 既存のショブか ら呼び出すことができ、パラメーターとして、通常はファイルである)多種の入力 を受け入れることができます。FlowForce Server はこのような マピング、おけよ 変換を自身で実行しません。 MapForce Server おけよ Style Vision Server (おけよ 場合によに両方) は、実際の実行を行うよりコ呼び出されます。

RaptorXML 関数を RaptorXML コンテナー内で使用することができます。RaptorXML Server との統合も参照してくたさい。

次のセクションでは、サーバー実行のためにどのようにマンピングと変換を準備するか、これらをジョブはし、FlowForce Server 内で結果を 処理する方法について説明されています。

9.1 サーバー実行のためにファイルを準備する

MapForce を使用してデザインされ、プレビューされるマンピングは、現在使用中のマシンと(データベースなどの) オペレーティングシステムの 外部にあるりノースを参照することができます。 更に、 MapForce 内では、全てのマンピングの マかデフォルト でWindows スタイルの規則 に従ってします。 おこ、 MapForce Server が作動するサーバーでは、マンピングがデザインされたマシンと同じデータベースをサポートしない 場合はあります。 このため、サーバー環境でマンピングを実行するこは、 おこ、ターゲットマシンがノースマンンと異なる場合には特に、若干の準備 が必要とされます。

メモ ソースマンともう用語は MapForce がインストールされているマンを指し、ターゲトマンノは、MapForce Server、おこ は、FlowForce Server がインストールされているマンを指します。最もシンプルなシナルオは、ソースマンンとターゲトマシンが 同じコンピューターの場合を指します。最も複雑なシナルガは、MapForce がWindows マシンで作動し、MapForce Server、お」は、FlowForce Server がLinux、お」は、macOS マンンで作動する場合を指します。

FlowForce Server にデプロイする前にMapForce 内でマンピングの検証を常に行うことが奨励されます。おけよ MapForce Server 実行ファイルにコン イルすることが奨励されます。

MapForce Server がスタイアロンで(FlowForce Server 無しで作動する場合、必要とれるライセンスは、以下のとおりです:

- ソースマシン上で、MapForce Enterprise、おは、Professional Edition がマピングをデザイン、サーバー実行ファイル(.mfx) にコンパルするよりに必要とれます。。
- ターゲナマシン上で、MapForce Server、おは、MapForce Server Advanced Edition がマピンが実行するため に必要とれます。

MapForce Server がFlowForce Server 管理下で作動する場合、次の条件が適用されます。

- ソースマシン上で、MapForce Enterprise、おは、Professional Edition がマピングをデザイルて、ターゲトマシンに デプロイするために必要です。。
- MapForce Server とFlowForce Server かターゲトマンノニインストールされている必要があます。MapForce Server の役割は、マピングの実行であり、FlowForceの役割はマピングを、スケジュールませま、オンデマドの実行、としての実行、 エラーの処理、条件付きの処理、電子メールの通知などの機能を活用することのできるジョブとして使用できるようにすることです。
- FlowForce Server は構成されているやホワークボレスとポートで作動している必要があります。すなわち、「FlowForce Web Server」サービスが開始され、構成されており、HTTP(おけよ、構成されている場合はHTTPS)クライアナカシの接続をファイアウォールにブロックされることなく受け入れる必要があります。「FlowForce Server」サービスも開始され、指定されたアドレスとポートで作動されている必要があります。
- コンテナーの1つへのペーミッションか与えられている Flow Force Server ユーザーアカウトを所有している必要がおます(デフォ ルトでは、/public コンテナーイ認証済みのユーザーであればアクセスすることができます)。

全般的な注意点

- スタイドアロン MapForce Server で作動するターゲナマンとのマピングを作動するコは、マピングには参照される全ての 入力ファイルをターゲナマンノこもピーする必要があります。MapForce Server がFlowForce Server 管理化で作動する 場合、ファイルを手動でピーする必要にありません。この場合、インスタンスとスキーマファイルはターゲナマンノニデプロイされたパッ ケージ内に含まれます。。
- マピンガニ特定のデータベースドライバーを必要とするデータベースエンポーネトか含まれる場合、このようよドライバーはターゲト マシンノミインストールされている必要があります。例えば、マピングがMicrosoft Access データベースからデータを読み取る場合、Microsoft Access、ませは、Microsoft Access Runtime かターゲットマシンノミインストールされている必要があります(<u>https://www.microsoft.com/en-us/download/details.aspx?id=50040</u>)。
- マッピングを非Windows プラナフォームニデプロイする場合、ADO、ADO.NET とODBC データベース接続は自動的に JDBC に変更されます。ネイティブ SQLite とネイティブ PostgreSQL 接続は保存され、追加の構成を必要としません。「データベース接続」も参照してください。
- (例えば、.dll、ませま、.class ファイルなどのカスタム関数呼び出しをマピングが含む場合、このような依存は、ランタイム前に 既知ではないため、マピングと共にデプロイされません。この場合、これを手動でターゲトマシノニピーすることができます。サー

バー上の.dll おは.class ファイルの みはMapForce 内のライブラの管理」ウイドウと同じてある必要がみます。例

Manage Libraries	×	
🛱 🛃 New Design1	*	
Own Library Imports Add		
🖃 😢 Global Library Imports 🛛 Add		
🕞 🔯 C:\Libraries\Converters.dll 🔀		
- 🖵 Functions		
f _x Converters, Version=1.0.0.0, S	how All Open <u>D</u>	ocuments
f _x Converters, Version=1.0.0.0, 🗸 S	ihow <u>F</u> ile Paths	
Own Library Imports		
	_	
4		

- マンピングの一部は、ワイルドカード、ウを使用して複数の入力ファイルを読み取りますこの場合、入力ファイル名はランタイム前に 既知ではおよりまな、デプロイされません。マンピングの実行に成功するには、入力ファイルがターゲットマンン上に存在する必要があります。
- マッピングの出力パマニディレケリが含まれている場合、これらのディレケリはターゲトマシン上に存在している必要があります。それ以外の場合、マッピングを実行するとエラーが生成されます。この振る舞いは、「一時的なファイルに出力を生成する」オプションが有効化されている場合既存ではないディレケリが自動的に生成されるMapForce とは異なります。
- マピングがクライアト証明書を使用するHTTPS 認証を必要とするWeb サービスを呼び出す場合、証明書をターゲトマン ンに手動で移動する必要があます。次も参照してください、デジタル署名の管理。
- Microsoft Access とSQLite などのファイルをベースコンたデータベースコマンピングが接続される場合、データベースファイルは 手動でターゲナトマンノン移動、おさよ参照することのできる共有されるディレオーリン保存される必要があります。次を参照「ファ イルをベースコントデータベース」。

パマをポータブルにする方法

サーバー上でマピングを実行する場合、マピングが適用することのできる変換パスを使用していること、および、サポートされるデータベース接続を使用していることを確認してください。

ノンを非Windows オペレーティングシステムへ移動可能にするためコよ、MapForce内でマンピングをデザインする際に絶対ノンの代わり に相対ノンを使用してくたさい。

- 1. 希望するマピングデザインファイル(.mfd)をWindows 上のMapForce で開きます。
- 2. 「ファイル」メニューでマッピングの設定を選択し、チェックされている場合「生成されたコード内でいたを絶対パレにする」 チェックボックスをクリアします。
- 3. 各マルングコンポーネトのオンビンプロ・ティダイアログボックスを開き、(例えば、コンポーネトのタイトル・・をダブルクトックして 全てのファイル マを絶対・マから相対・マムご変更します。「MFD ファイルニオして全ての・マを相対的に保存する」チェック ボックスを選択します。便宜上、マッピングと同様全ての入力ファイルとスキーマを同じファイルニエピーし、ファイル名で参照することが できます。

マピングのデザイン時に相対および絶対パンを処理するコはMapForceドキュメートを参照してくたさい。

更に MapForce Server とFlowForce Server は、すてくの相対ノジカ解決されるしたかる「作業ディレクトリ」をサポートします。。 作業ディレクトリは以下のとおノマンングのランタイム・指定することができます:

- FlowForce Server では、ジョブの「作業ディレクトリ」、デメーターを編集します。
- MapForce Server API では COM と.NET API のWorkingDirectory プロ ディーには、おは Java API の setWorkingDirectory メンドには行うことができます。

• MapForce Server コマバラインでは作業デルケーリはコマバシェルの現在のデルケーリです。

データベース接続

ADO、ADO.NET、およびODBC 接続はLinux とmacOS マンンではサポートされていないては注意して下さい。このため、ターゲットマンンがLinux おけまmacOS の場合、このような接続は、マメピングがFlowForce にデプロイされると、おけま、MapForce Server 実行ファイルにマメピングをコンパルオると、JDBC に変換されます。この場合、マメピングをデプロイ、おけま、サーバー実行ファイル にマメピングをコンパルオる前に以下のオプィンを使用することができます。

- MapForce 内では データベースへの JDBC 接続を作成します
- MapForce 内では、データベースコンポーネントの「JDBC 固有の設定」セクション内のJDBC データベース接続の詳細に入 カしてください。(データベースコンポーネント設定を参照してください)。

マピングがPostgreSQL おけまSQLite データベースへのオイティブな接続を使用する場合、ネイティブな接続が保存され、JDBC 変換は発生しません。。マピングがMicrosoft Access とSQLite などのファイルベースのデータベースは接続される場合、追加構成が必要とされます。下記の「ファイルベースのデータベース」を参照してくたさい。

JDBC 接続を使用してマピングを作動することは、Java Runtime Environment おとしま Java Development Kit がサーバーマシン ンニインストール されていることを必要とします。 Oracle JDK おとしの cole OpenJDK などのオープンノースビルドであることができます。

- JAVA HOME 環境変数はJDK インストーリディレクト」を指している必要があります。
- Windows 上では、Windows レジストリ内で検出された Java Virtual Machine パスは JAVA_HOME 変数 よ 化 優先順位を有します。
- JDK プラナオーム(64ビナ、32ビナ)はMapForce Server を同じてある必要かあます。それ以外の場合は、 理由かあるエラーか挙にたれる可能性かあます。「JVM へのアクセスか不可能です。」

Linux または macOS 上での JDBC 接続のセット アップ:

- 1. データベースペンダーには提供される 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から2 ビナトで作動する場合は32 ビナト版バージョンを、オペレーティングシステムが64 ビナトで作動する場合は64 ビナ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こは、JDBC ドライバーニ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 ライブラが/Library/Java/Extensions ディレクトリニ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ン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 ライブラルが JDBC ドライバーをインストーリしたい 父を検索するようにする必要があります。

macOS 上でのOracle Instant Client 接続

macOS 上で、Oracle Database Instant Client を介して Oracle データベース 接続する場合これらの命令を適用することができます。必要条件:

- Java 8.0 おけよ 以降がインストールされて る必要が決ます。Mac マシンが Java 8 以前の Java バージョンで作動作動している場合、「JDBC Thin for All Platforms」 ライブラ を介して接続し、以下の命令を無視することも可能です。
- Oracle Instant Client がインストールされている必要があます。Oracle Instant ClientをOracleの公式ダウムードページからダウムロードすることができます。Oracle ダウム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、シケージが選択されていることを確認してくたさい、例えばInstant Client Basic)。また、オペレーティングシステムが82-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たさい。

ダウンロードした後、Oracle Instant Client をアンパンし、インストーラと共にパン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のなまポイントします。例えば

変数	サンプル値
CLASSPATH	/opt/oracle/instantclient_11_2/ojdbc6.jar:/opt/oracle/instantcli ent_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
え	<pre>\$PATH:/opt/oracle/instantclient_11_2</pre>

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たさい。

ファイルをベースユたデータベース

Microsoft Access とSQLite などのファイルをベースコンナデータベースは FlowForce Server にデプロイされる シケージ内に含まれません。オゴム コン イルされた MapForce Server 実行ファイルにきまれません。このため、 ソースとターゲルマンンが同じでは内場合、次のステップに従ってください

- 1. MapForce 内では マッピングを右クリックして、「生成されたコード内でい ひを絶対いひにする」のチェックボックスのチェックを解除します。
- 2. マピング上のデータベースコンポーネントを右クリックし、相対パマを使用してデータベースファイルへの接続を追加します。。マピン グデザイン(.mfd ファイル)をデータベースファイルと同じディレクトリニ保存し、ファイル名別(相対パマを使用して)にマピングから 参照すること」といいなご見連した問題を回避することができます。
- 3. データベースファイルをターゲオマンス内のディレオリニピーします("作業ディレオリ」と呼ぶことします)。以下に表示されるよう にマッピングを実行する際必要とされるため、このディレオを忘れないてください。

このようなマピングをサーバー上で実行するコお以下を行います

- FlowForce Server エトロール下で作動するMapForce Server によりマピングが実行される場合、FlowForce Server ジョブが前に作成された作業ディレケリをポイントするように構成します。データベースファイルは作業ディレケリ内に存在 する必要があります。
- コマボラインでスタボアロン MapForce Server によりマピンが実行される場合、MapForce Server 上でrun コマボ を呼び出す前に、現在のディレカリを作業ディレカリニ変更します(例えば、cd path\to\working\directory)。
- MapForce Server API によりマピングが実行される場合、作業ディレケーをプログラング的にマピングを実行する前に設定します。これを効果的に行うために、COM と.NET API 内のMapForce Server オブジェクトのかっかにプロ・ディ WorkingDirectoryを使用することができます。Java API 内では、メノンド setWorkingDirectoryを使用することができます。 できます。

ローカルネ・トワーク上のWindows マンン上にノースとターゲットマンンが存在する場合、共有されるディレクトリからデータベースファイルを読み取るためにマンピングを以下のように構成することができます。

- 1. ソースとターゲオマンパンとリアクセスすることのできる共有されるディレオリニデータベースファイルを保管します。
- 2. マピング上のデータベースコンポーネトを右クリックし、絶対パスをしようしてデータベースファイルは接続を追加します。

グロー・ジレノノース

マンピングが直接の、マませまデータベース接続の代わりにグローイ、シリノノースへの参照を含む場合、サーイ、一側でもグローイ、シリノノースを使用

することができます。MapForce Server 実行可能ファイル(.mfx) にマンピングをコンパルする場合、グロー・シリソースへの参照は変更 されないので、マンピングランタイムこサー・「側に提供することができます。FlowForce Server にマンピングをデプロイする場合、リソースを サー・トー上で使用するかを任意で選択することができます。

マピング(お出まFlowForce Server の場合マピング関数)の実行に成功するために、グロー・ドレソースとして提供する実際のファイル、フォルダー、お出まデータベース接続の詳細はサーバー環境で互換性を持つ必要かあります。例えば、マピングがLinux サーバーで作動する場合、パマのためにLinux 変換を使用する必要かあります。同様に、データベース接続として定義されているグロー・ドレノースはサーバーマンンで利用が可能である必要かあります。

詳細は見してはリノースを参照してくたさい。

XBRL タケノノミパッケージ

マピングがFlowForce Server へのXBRL タクノノシ、シケージを参照する場合MapForce はマピングから全ての外部レフォンンを集め、現在の構成と現在インストールされてしるタクノノシ、シケージを使用して解決します。タクノノシ、シケージを指す解決された。外部レフォンンスが存在する場合、タクノノシ、シケージはマピングと共にデプロイされます。FlowForce Server は、その、シケージを、デプロイ時と同様、マピングを実行するために使用します。FlowForce Server によし使用されるタクノノシ、シケージを更新するには、MapForce内で変更し、マピングをもう一度デプロイします。

MapForce Server のルートカタログはターゲルマンと内でタケノノミの解決方法に影響を与えます。ルートカタログは MapForce Server インストールディレクトリニ対して相対的に以下の なで見つけることができます: etc/RootCatalog.xml.

MapForce Server のルートカタログかごのようなカタログを含んでいたい場合、おけよ、同 URL じプレフィックスのために定義されているい や ケージを含まれ、場合、マメピングと共にデプロイされるタケノノミッシケージが使用されます。MapForce Server のルートカタログはデプロイ されたタケノノミよ・凡優先順位か高いです。

MapForce Server がスタイアロンで(FlowForce Server 無しで作動する場合、次に用にマピングにより使用されるルートカタログを指定することができます:

- コマボラインではオプション catalog をrun コマボに追加することが可能です。
- MapForce Server API で、メンド SetOption を呼び出し、文字列「catal og」を最初の引数として、ルートカタログを 2番目の引数として与えます。

マンピングかテーブルリンクベースを使用するXBRLコンポーネントである場合、タケノノミッシケージ、ませよ、タケノノミッシケージ構成ファイル かランタイムでマンピングに提供される必要があります。

- MapForce Server コマバライノこて、オプション -- taxonomy-package、おけた -- taxonomy-packagesconfig-fileをrun コマバに追加します。
- MapForce Server API 内で、メノボ SetOption を呼び出します。最初の引数は「taxonomy-package」、おこは、「taxonomy-packages-config-file」である必要があます。2番目の引数は、タケノノミ・シケージ(、おさは、タケノノミ・シケージ構成)ファイルへの実際のいてである必要がありますファイル。

9.2 FlowForce Server にマッピングをデプロイする

FlowForce Server へのマピングのデプロイはMapForce か特定のマピングにお使用されるハースを整理し HTTP(おけば構成されている場合 HTTPS)を介して FlowForce Server が作動するマンソン 次するこを意味します。メモ「ソースマシン」と う用語は、 MapForce がインストールされているコンピューターを指し、「ターゲナトマシン」と う用語は、 FlowForce Server がインストールされてい るコンピューターを指します。マピングがデプロイされると、完全機能を搭載した FlowForce Server ジョブを作成することができ、(例えば、 ジョブのナックコンカスタムト・リガーを定義する、ジョブをサービスとして公開するなど)ジョブ固有の機能全てを活用することができます。

メモ 「ソースマシン」とう用語は、MapForce がインストールされているエピューターを指しており、「ターゲト・マシン」はFlowForce Server がインストールされているマシンを指しています。最もシンプルなシナルは、ソースマシンとターゲト・マシンが同じエピュータ ーの場合を指します。最も複雑なシナルオは、MapForceがWindows マシンで作動し、FlowForce Server がLinux、 およ、macOS マシンで作動する場合を指します。

FlowForce にデプロイされる シケージ お以下の通りです:

• マピング自身ユポーネトは入力引数となり、ターゲトコポーネトはこの関数の出力引数となります。全てのノースコンポーネ ントは入力引数になり、全てのターゲトコポーネトはこの関数の出力関数になります。

Execute fund	tion /public/Comple	etePO.man	nin	a	
Parameters:	Customers:	(input)	e	altova://packagedfile/C:/Users/	/MapForceExamples/Customers.xm
	Articles:	(input)	뎚	altova://packagedfile/C:/Users/	/MapForceExamples/Articles.xml
	ShortPO:	(input)	뎥	altova://packagedfile/C:/Users/	/MapForceExamples/ShortPO.xml
	CompletePO:	(output)	욚	CompletePO.xml	
	Working-directory	:		+	

• マピングによ使用される入力インスタンスファイル(XML、CSV、テキスト)。

必要条件

サーバー実行のためこマピングを準備するを参照してくたさい。

Flow Force Server にマンピングをデプロイする

- 1. MapForce を作動し、Built-in デジン・ノレトーボタンをクトック、おけま出力 | Built-in 実行エンジン メニューコマンドを選択して、変換言語がBUILT-IN に設定されていることを確認してくたさい。
- 2. 「ファイル」メニューから「FlowForce Server にデプロイ」を選択します。マッピングのデプロイダイアログボックスか開かれます。

♥ Deploy Mapping ×					
Enter the host name mapping.	e and port of a FlowForce Administration Interface to deploy the curre	ent			
<u>S</u> erver:	localhost V Port: 8082				
<u>U</u> ser:	root Use SSL				
Pass <u>w</u> ord:	••••				
Login:	Directly				
Global Resources	on the Server				
Use <u>R</u> esource	S				
Resource Path:	/public/				
	The path must start with a slash character.				
Deploy As					
<u>P</u> ath:	/public/CompletePO.mapping Browse				
	The path must start with a slash character.				
S <u>a</u> ve mapping	j before deploying				
Open web bro	owser to create new job				
	OK Cancel				

3. (下で説明される通りデプロイ設定を入力し [OK]を押します。「新しいジョブ作成のためにウェブブラウザを開く」 チェックボック スが選択されている場合、FlowForce Server 管理インターフェイスが開かれ、すくに FlowForce Server ジョブの作成を開 始することができます。

以下のテーブルは、マパングのデプロイダイアログボックスで使用することのできるマパングデプロイ設定をリストしています。

設定	説明
サーバー、ポート、SSL の使用	FlowForce Server のサード・ホスト名(おけまI.P. アドレス)とポートを入力します。これ はFlowForce Server かデフォルトのポートと同じマンンで実行されている場合、 localhost と8082 である可能性があります。疑問かある場合は、FlowForce Server Web 管理インターフェイス ロングインして、Web ブラウザーのアドレスバーに表示される I.P. アドレスとポートを確認してくたさい。
	接続エラーか生じた場合、FlowForce Server か実行されているマシンが入力接続を許可するように構成されていることを確認してくたさい。
	SSL 暗号化済み接続を介してマピングをデプロイする場合 SSL の使用 チェックボックスを 選択しますこれよ FlowForce Server か既にSSL 接続を受け入れるようご構成されて いるものの想定しています。詳細に関しては、FlowForce Server ドキュメント (<u>https://www.altova.com/ja/documentation</u>) を参照してくたさい。

設定	説明
ᠴ᠆ᡃᡃᠮ᠆ᡃ᠔ᡃᠺᡃ᠋᠋ᡔ᠆ᡟ	ユーザー名と マワードは、ログインドロップダウンリストの値(次のオプション参照)により入力されます。ログインドロップダウンリストが >Default> おけま Directly に設定されている場合、FlowForce Server ユーザー名と マワードを入力します。それ以外の場合ドメインユ ーザー名と マワードを入力し、ドメイン名をログインドロップダウンリストから選択します。
ログイン	FlowForce Server 内でDirectory Service か有効化されているとこのドロップダウン リストカシドメイン名を選択し、ユーザーと スワードフィールド にドメイン資格情報を入力します (前のオプノョンを参照してください)。
リノースの使用、リノースパス	マメンク関数がサーバーにデプロイ後に <u>リソース</u> を使用する場合リソースの使用 チェックボックスを選択しますチェックボックスを選択すると、「リソース・ペス」テキストボックスとの対応するリ ソースの、やを入力する必要があります。リソースを選択し、「省略記号」 ボタンをクトックしま す。
	サーバー上に選択するハースが存在しない場合、「グローバレリソースのデプロイ」をクトックして、必要とされるグローバレリノースをサーバーにデプロイします。
	「リソースの使用」チェックボックスを選択しない場合、グロー・シレトノースは現在選択されて しる構成をベースに解決されます。サー・チーとでは、マッピング関数はグロー・シレトノースを必 要とせず、代わりに解決された値を使用します。
/☆	「参照」をクックして、FlowForce Server エンテナーの階層内でマメピング関数が保存 される 文を選択します。 デフォルトで 文はFlowForce Server の/public エンテナーに 設定されています。
	必要な FlowForce Server パーミッションと特権が与えられている場合、既存のコンテナー とマッピングを削除できることもできます。
デプロイ前ニマンピングを保存する	保存されていないマンピングをデプロイする時にこのオプションを使用することができます。デプロイ 前にマッピングを保存する場合このチェックボックスを選択します。
新しいジョブを作成するな」 ブランザを開く	このチェックボックスを選択した場合、FlowForce Server Web 管理インターフェイスかデプロイ後開かれ、FlowForce Server ジョブの作成の開始をすることができます。

トラブルシューティング

次のテーブルはマンピングをデプロイする際に発生する一般的な問題をリストしています。

問題	ソリューション
マノピングのデプロイは以下のエラーを返します: ファイル上の 1/0 オペレーションに失敗しました。 1/0 エラー 28: 〈ser ver〉ポー 8082 への接続に失 敗しました。タイムアウト システムエラー 10060: 接続されたパーティが一定の時間内に 正確にユスポイトしなかった。または、接続されたホストがレスポイド に失敗し確立された接続が失敗したこの接続の試みは失敗しまし た。	ターゲトマシン上で FlowForce Web Server サービスが作動し ており、指定されているポート(デフォルトで 8082)で接続のオンジョン スンするよう 3 構成されていることを確認してくたさい。 ヨーファイアウォー ルがこのポートを介して受信される接続をブロックしないよう こてくたさ い。 デプロイを可能にするオーングに FlowForce Server サービスも作動し ている必要があります。
マノングのデプロイは以下のエラーを返します	デプロイされるマピングの入力ファイルがFlowForce Server により 許可されているHTTP リクエストの最大量(大体 100 MB)を超える

問題	ソリューション
ファイル上のI/O オペレーションに失敗しました。 I/O エラー 413: ペイロードが大きすぎます	とエラーが発生する場合があります。「ミナを max_request_body_size オプションを(in bytes) flowforceweb.ini とflowforce.ini ファイル内で設定して増量 することができます。詳細に関しては構成ファイルレファレンスを参照し てくたさい。

サーバーバージョンの選択方法 (Windows のみ)

マルングをデプロイするサー・・・・でFlowForce Server 管理下で、複数の・・・ジョンのMapForce Server を作動している場合、 (Windows サー・・・のみに適用することができます) このマルングを実行する MapForce Server の・・・ジョンを指定するよう こプロンプトされます。

Select MapForce	Server			×			
Multiple versions of MapForce Server were found which can all execute this mapping.							
Select the most	t appropriate version	on <u>a</u> utomatically					
O Choose versio	n <u>m</u> anually:						
Version:	2017r3	\sim					
			OK Cancel				

メモ FlowForce Server インホールディレケルに、tool ファイルは含まれている場合、FlowForce Server の管理下で実行され る各 MapForce Server の・ジュンのために、ダイアログボックスが表示されます。デフォルトでは、MapForce Server を FlowForce Server の一部とてインネールすると、MapForce Server .tool ファイルは、このディレケトルに自動的に追加 されます。FlowForce 内でファイルが保管される、ペロシストのとおしです: C:\Program Files\Altova\FlowForceServer2021\tools。FlowForce Server 管理下での他のMapForce Server バージョン を作動する場合、、tool ファイルを手動で上記のディレケールこと、MapForceServer2021、 は以下に保管されて、ます: C:\Program Files\Altova\MapForceServer2021。

9.3 マッピングと変換をジョブとして実行する方法

MapForce マピング、おけよ、StyleVision 変換からFlowForce Server ジョブを以下のように作成することができます:

- 1. 最初に、マルング、おけよ、変換をFlowForce Server にデプロイレます。このステップはMapForce (とStyleVision でそれぞれ)で行うことができます:
 - 「ファイル」メニューから、「Flow Force (Server) へデプロイする」をクトックします。

デプロイの設定の説明は、MapForceとStyleVisionドキュメトを参照してくたさい (https://www.altova.com/documentation)。

2. FlowForce Server 内で、マピング、おけよ 変換がデプロイされている個所にFlowForce コンテナーをナビゲートします(例 えば、コンテナー "/public")。

C / O C public						
	名前	型 🔷				
	AutoCalc.transformation	StyleVision 変換				
	CompletePO.mapping	MapForce マッピング				

3. 必要なマピング、およま、変換をクリックし、「ジョブの作成」をクリックします。およま、関数の実行 ボックスコ ひを入力して既存 のジョブからマピング、およま、変換を参照することができます:

実	実行ステップ							
	+							
	4	関数の実行						
	=	ステップの約	/ O public					
	新規	の実行ステップ	名前◆					
₽	2/∓/	の結果	(AutoCalc.transformation (StyleVision 変換)					
~	2110		(① CompletePO.mapping (MapForce マッピング)					
	戻り	値の型を以下と						

必要に応じてションを構成することができます。例えば、<u>Webサービス</u>として、おけよトリガーを使用して実行することができます。 StyleVision 変換のデプロイ、および、変換からションを作成する方法について説明するステップ、イステップのサンプルは関してよ StyleVision 変換からのショブの作成を参照してくたさい。MapForceのための類似したサンプルは関しては、<u>MapForceマンとし</u>がから のショブの作成を参照してくたさい。MapForce ServerとStyleVision Serverの両方を呼び出すサンプルジョブは関しては、サンプル 複数のPDFを複数のXMLから生成するを参照してくたさい。

変換、おけよ、ジョブのマンピングの実行の最も重要なパートは、ジョブの入力ファイルの処理です。2つのアプローチをとることができます:入力 ファイルを静的にジョブは提供する、おけよ、ジョブのランタイムに(例えば、パマから、動的に提供する。必要に応じて使用する正確なアプロ ーチを決定します。ジョブが同じ入力データを使用して毎度実行される場合、最初のアプローチが適切です。それ以外の場合、パマから動的 は提供されるファイルからデータを取得するけかに、FlowForce ジョブが必要になるけっか、2番目のアプローチが適切です。

MapForce マピング

MapForce からマピングがデプロイされる場合、(XML、CSV、JSON、Excel などの) インスタンスファイルはマピングと共にデプロイされ、明示的に静的に シケージされます。 これは、ジョブが作動すると、 FlowForce はデフォルト で シケージ済みのファイルから静的にデー タを読み取ります。 ここで2つのシナリオが考えられます:

 MapForce 内でマピングを右クトックし、「生成されたコード内で」なを絶対にする」チェックボックスをマピングをデプロイする 前に選択します。FlowForce Server 内ですべての入力ファイルはプレフィックスaltova://packagefile/と共に明示 的に表示されます。

実行	ステップ					
•	関数の実行 /p	ublic/CompletePO.	mapping			
	パラメーター:	Customers:	(input)	덆	altova://packagedfile/C:/User:	/Documents/Altova/MapForce2019/MapForceExamples/Cus
		Articles:	(input)	륩	altova://packagedfile/C:/Users/	ı/Documents/Altova/MapForce2019/MapForceExamples/Art
		ShortPO:	(input)		altova://packagedfile/C:/Users/	/Documents/Altova/MapForce2019/MapForceExamples/Sho
		CompletePO:	(output)	턭	CompletePO.xml	
		Working-directory	y:		+	
=	ステップの結果	を割り当て名前			以下としてCompletePO	

FlowForce Server に、シケージファイルからデータを読み取らないように命令するコスプレフィックス altova://packagedfile を、なから削除します。絶対まける相対ファイルを使用してファイルへ参照することができます。相対、文を使用する場合、作業ディレクトリ・プメーター(つけして、やいは相対的しています。例えば、C: \FlowForce\CompletePOからファイルを入力として提供する場合、C:\FlowForce\CompletePOを作業ディレクトリーに設定し、以下に示される通り入力ファイルの名前を入力します。

実	実行ステップ									
	+									
	4	関数の実行 /р	ublic/CompletePO.n	napping						
		パラメーター:	Customers:	(入力) 🚦	Customers.xml					
			Articles:	(入力) 🚦	Articles.xml					
			ShortPO:	(入力) 🚦	ShortPO.xml					
			CompletePO:	(出力) 🚭	CompletePO.xml					
			Working-directory	/:	C:\FlowForce\CompletePO					

2. 「生成されたコード内でいてを絶対にする」チェックボックスがFlowForce ヘマッピングをデプロイする前に選択されていない場合、FlowForce 内で入力ファイルは相対パマで表示されます。FlowForce はこの場合作業ディレクトリ内の同じ名前を持つ ファイルであっても、パッケージ済みのファイルからデータを読み取ります。FlowForce にパッケージ済みのファイルからデータを読み取ります。FlowForce にパッケージ済みのファイルからデータを読み取ります。FlowForce にパッケージ済みのファイルならデータを読み取ります。FlowForce にパッケージ済みのファイルならデータを読み取ります。

Š	»Э.	ブ入カパラ	メーター						
	+								
	名前:	customers	× 型:	ファイルと	しての文字列	`	デフォルト:	C:\Customers.xml	
	+								
	名前:	articles	× 型:	ファイルと	しての文字列	`	/ デフォルト:	C:\Articles.xml	
	+								
	名前:	ShortPO	× 型:	ファイルと	しての文字列	`	デフォルト:	C:\ShortPO.xml	
	•								
e	·/	7f							
夫	E172	ヘナツノ							
	+								
		関数の実行 /p	ublic/CompletePO.	mapping				▼ 🖻	
		パラメーター:	Customers:	(入力) 🚭	Customers.xml				
			Articles:	(入力) 闧	Articles.xml				
			ShortPO:	(入力) 🚭	ShortPO.xml				
			CompletePO:	(出力) 🚭	CompletePO.xml				
			Working-director	y:	+				

代わりに、入力ファイル名がマッピングの入力・ラメーターコンるようにMapForce内でマッピングデザインを変更することができます。例えば下で説明されるマッピングは入力と出力ファイル名を・ラメーターとしてとます。

C= "Altova_Hierarchical"	C= "Altova_Hierarchical"
*8 InputFileName	₩ OutputFileName
◆ default InputFileName へ & Altova_Hierarchical	default OutputFileName
> → Name > ⇒ ⊖ ♦ Office ►	C= "Organization Chart" →

FlowForce Server にデプロイすると、ジョブ構成ページン デメーターかそのように表示されます(ファイル自身は、シケージされてしません)。

実	実行ステップ									
	+									
	4	関数の実行 /p	ublic/FileNamesAsPar	ame	neters.mapping	• 🛃				
		パラメーター:	OutputFileName:	₩	output.xml					
			InputFileName:	₩	Altova_hierarchical.xml					
			Working-directory:		C:\Users \MapForceExample					
		1								

上記のマピングは FileNamesAsParameters.mfd と呼ばれ、MapForce に含まれるサンプルファイルの一つです。このマピングのデザイン方法に関しては MapForce ドキュメントを確認してけさい。

StyleVision 変換

StyleVision 変換の場合、入力ファイルを以下のように処理することができます:

- 1. PXF (Portable XML Form) ファイルを Style Vision 内で開きます。SPS (Style Vision Power Stylesheet) か存在 する場合、FlowForce Serverにデルイレようとすると、Style Vision は PXF 書式に変換するよう これしかします。
- 2. デザイン概要ウインドウ内で、「埋め込まれたファイルの構成」クトックします。ダイアログボックスか表示されます。

グローバル構成	
SPS ファイルの埋込み	\checkmark
イメージの埋込み	✓
cssファイルの埋込み	✓
xstrファイルの埋込み	✓
SPS モジュールの埋込み	✓
Authentic カスタム ボタンを埋め込み	✓
スキーマソース \$XML (メイン)	
スキーマファイルの埋込み	✓
作業用 XML ファイルの埋込み	✓
一時 XML ファイルの埋込み	

3. オプロン「作業 XML ファイルの埋め込み」 に注意してくたさい。このチェックボックスを選択すると、作業 XML ファイルカデプロ イされるパッケージの一部 なり、デフォルトで FlowForce Server はジョブの実行の都度データを読み込みます。 FlowForce 内でパッケージファイルカ表示されます:

Į	実行ステップ								
	+								
	4	関数の実行 /pu	ublic/AutoCalc.transformation		▼ 2				
		パラメーター:	InputXml:	•[8	altova://packagedfile/Data.xml				
			OutHtml:	<mark>₽</mark> HTHL	+				
			GenerateHtmlOutputAsMime:		•				
			OutRtf:	₽ BTF	•				
			OutFo:	FO	•				
			OutPdf:	POF	•				
			OutDocx:	0 008	•				
			Working-directory:						

ジョブニファイルを動的に提供するコよプレフィックスaltova://packagedfile/
を削除、おけよ パマを絶対パマ変更 します。相対的なパマを使用する場合、相対的なパマを使用する場合、作業ディレクトリパラメーターに対してパリオ相対的で す。代わりに、FlowForce Server に変換をにデプロイする前に、「作業 XML ファイルに埋め込む」チェックボックスをクリアし ます。

CSS ファイルませまイメージなどのリソースのために「埋め込み」チェックボックスをクリアすると、FlowForce Server はジョブの作業ディレカ・リ内でこれを検索します。

9.3.1 マッピング関数内の資格情報

資格情報の紹介で説明されているとおり、FlowForce Server内だけではスマピングデザイン時に内でMapForceでも資格情報を 作成することができます。

MapForce からFlowForce Server に資格情報を含むマンピングをデプロイする場合、サーバーに資格情報もデプロイされます。デプロイ

された情報には資格情報レコードを作成した際に記入されたフィールドのみか含まれます。例えば資格情報名のみを保管することを選択し、空の資格情報である場合、ませま、ユーザー名とマワードの両方を含む資格情報オブジェクトの場合。

MapForce からFlowForce Server へ資格情報オブジェクトをスタイアロンオブシェクトとしてデプロイすることができます。。 MapForce から直接デプロイ先のターゲトコンテナーを選択することができます。詳細に関しては、MapForce ドキュメントを参照してくだ さい(<u>https://www.altova.com/ja/documentation</u>)。

次のフィールドは重要なデータとて考えられます

- (型「パワード」の資格情報のための)パマワード
- (型「OAuth 2.0」の資格情報のための。 クライアントの秘密、アクセストークン、および更新トークン

MapForce 内のデザイン時に「実行可能ファイルとマッピングのデプロイに含まれる」チェックボックスを選択した場合のみ重要なデータ かデプロイされます。これはマッピングをデプロイする際、およびスタイドアロン資格情報をデプロイする際に適用されます。

FlowForce Server 内では、対応するマピンク関数のページを開くことにより、マピンク関数が資格情報を必要とするか確認します。例

Function Input Parameters										
Name: airportCode	📲 Type: string	optional								
Name: AirportStatus	(output) 5000 Type: string	Default: AirportSta	itus.json							
Name: my.credentials	🖙 Type: 🛛 creder	tial								
Name: Working-directory	Type: string	as directory Default:								

「実行可能ファイルとマッピングのデプロイに含まれる」チェックボックスを資格情報を作成する際に選択した場合、マッピングと共に資格 情報をジョブはデプロイします。この場合、ジョブ構成ページから指定する必要はありません。例えば、次の実行ステップは存在する場合保管 された資格情報と共にマッピング関数を実行します(「my.credentials 」、「ラメーターか展開されていないてとご注目してくたさい):

Exect	ution Step	S			
<u> </u>	 Execute function /public/AirportStatus.mapping 				
	Parameters:	airportCode: 🕺 🕂			
		AirportStatus: (output) 5500 +			
		my.credentials: 🛛 👄 🕂			
		Working-directory: +			

FlowForce Server 内で直接定義された、おさよ ローカルの資格情報により定義された他の資格情報オブジェクトと保管された資格情報を上書きすることができます。これを行うさよ「+」ボタンを押して、FlowForce Server 内に既存の資格情報オブジェクトを選択、おこ は、直接ユーザー名と、スワードを入力します。例

E	xecu (+)	ution Step	DS					
	Execute function /public/AirportStatus.mapping							- 2
		Parameters:	airportCode: AirportStatus:	اي⊭ (output) ا	+ +			
			my.credentials.		Select existing credentia			
					Define local credential:	User name:	admin	
						Password:	•••••	

実行ステップコーラーとて提供された資格方法は、マピンク関数内に保管されている資格情報より上優先順位が高いです。

「実行可能ファイルとマッピングのデプロイに含まれる」チェックボックスをMapForce内で資格情報を作成する際に選択しなかすた場合、実行ステップにプラメーターとして資格情報を提供することは必須です。それ以外の場合、実行は失敗します。

OAuth 2.0 認証が必要なマピンク関数の場合、アクセストークンの期限が切れる、おけよ W eb サービスプロ・イダーによい呼び出される ことがあります。この状況が発生すると、ジョブインスタンスの作動中 FlowForce Server は新規のものを自動的に取得しようと試みます。 複数の作動中のジョブが同じ資格情報を使用し、ランタイム要素が許可する場合、FlowForce Server はアクセストーケンを中央化され た方法で更新し、必要に応じて影響を受けるジョブインスタンスを全て同期化します。

9.3.2 例: OAuth 2.0 認証

このサンプルはOAuth 2.0 認証を必要とするREST-スタイルWebサービスを呼び出す方法について説明しています。 クライアントアプリケーションはGoogle カレンダーAPI (<u>https://developers.google.com/calendar/</u>)を使用してのイベントを抽出するFlowForce Server ジョブです。 簡素化するためにジョブ はカレンダーの情報を行のまま抽出し、他の処理無しに JSON 結果を生成します。

必要条件:

- MapForce Enterprise Edition
- MapForce Server Advanced Edition
- MobileTogether Server Advanced Edition
- このサンプルをステップ・イステップでフォローするコはGoogle アカウトが必要です。他のWeb サービスを呼び出す必要がある 場合 OAuth 2.0 資格情報 Web サービスプロ・イダーから取得し、以下のステップで使用してくたさい。

OAuth 2.0 認証情報を取得する

Webサービスにアクセスするために必要なOAuth 2.0認証情報が既存の場合このステップをスキップすることができます。それ以外の場合、取得方法はマッピングが呼び出すWebサービスのプロバイダーにより異なります。このサンプルのようなGoogleカレンダーAPIを呼び出す場合以下のステップに従ってくたさい

- 1. Google API コンノールコヴイルてけさい(<u>https://console.developers.google.com/)</u>。
- 2. 新規のプロジェクトを作成します。

≡ Google APIs							
New Pr	New Project						
You have 12 projects remaining in your quota. Request an increase or delete projects. <u>Learn more</u> <u>MANAGE QUOTAS</u>							
Project n MapFore	Project name * MapForce Demo						
Project ID: mapforce-demo. It cannot be changed later. EDIT							
Location * BROWSE							
Parent organisation or folder							
CREATE	CANCEL						

- 3. OAuth 同意画面をクルクしてくたさい。
- 4. 所属機関内のユーザーのみにAPI アクセスを許可することを可能にするG Suite アカウトカ既存の場合を除き外部をユーザー型として選択します。



5. [mapforce-demo]をアプリケーション名とて入力し、設定を保存します。

OAuth consent screen	
Before your users authenticate, this consent screen will allow the whether they want to grant access to their private data, as well to your terms of service and privacy policy. This page configure screen for all applications in this project.	hem to choose as give them a link es the consent
Verification status	
Not published	
Application name	
mapforce-demo	
Application logo ② An image on the consent screen that will help users recognise your a	pp
Local file for upload	Browse

- 6. 認証情報の作成 をクルクし OAuth Client ID を選択します。
- 7. デスクトップアプリと[MapForce Client] をクライアント名とて入力します。

Create OAuth client ID					
A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See <u>Setting up OAuth</u> <u>2.0</u> for more information. Application type *					
Desktop app 🔹					
Learn more about OAuth client types					
MapForce Client					
The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.					
CREATE CANCEL					

8. 「作成」をクトックします。ID が作成され認証情報 ページで使用できるようプロリます。

	Google APIs S MapForce Demo ▼					0	<u>ب</u>	:	
API /	APIs & Services	Credentia	ls	+ CREATE C	REDENTIALS	👕 DE	LETE		
<\$> [Dashboard	4							Þ
Ш І	Library	OAuth 2.0 Client IDs							
0+ (Credentials	als Type Client ID Desktop -s140					Ť	+	
:2	OAuth consent screen						-	_	
	Domain verification								, ,
≡¢	Page usage agreements	Service	Accour	nts	Mana	age serv	vice acc	ounts	1
		Em Em	nail	Name 个	Usage with all s (last 30 days) 🌘	ervices			
		No service	accounts to	display					
<1									

9. をクリックして OAuth 2.0 認証情報を JSON ファイルとしてダウレロードします。

OAuth 2.0 認証情報がGoogle コンソールAPI から取得されました

- 1. 認証エドポイト
- 2. トークノエイポイント
- 3. *ケ*ライアント ID
- 4. ケライアトの秘密

Google カレンダー API の有効化

クライアナトからの呼び出しを受け入れるコよこのサンプルで使用されたGoogle カレンダーAPI か有効化されます。Google API コンソール内でライブラリをクリックし Google カレンダーAPI を検索し有効化します:



このサンプルではイベント エンティティのlist メンドが呼び出されます。

<u>https://developers.google.com/calendar/v3/reference/events/list</u> でこのAPI メソバへの詳細な参照を見つったことができます。以下の重要点に注意してください

- このドキュメンテーションで指摘されているとおり calendarId が Google カレンダーの識別子である個所で GET リカエストを https://www.googleapis.com/calendar/v3/calendars/calendarId/events に送信することによりメ ンド は呼び出される必要があります。calendarId リカエスト デメーターは次のステップ内で MapForce から構成されます。
- 2. API メンドの呼び出しは少なくとも以下のスコープの一つを必要とします:
 - https://www.googleapis.com/auth/calendar.readonly
 - https://www.googleapis.com/auth/calendar
 - https://www.googleapis.com/auth/calendar.events.readonly
 - https://www.googleapis.com/auth/calendar.events

OAuth 2 認証プロセス中で、マッピングは上記のスコープの一つを提供し、これは次のステップでも構成することができます。このサンプルの目的のためには読み取り専用ノスコープで十分です。
認証トーケーのリクエスト

MapForce内でマピングをプレビューするコよ下で説明される通りOAuth 2.0認証詳細をマピングに追加し、認証トーケンをリクエストする必要がみます。

- 1. MapForce 内でマピング上の空白のエリアを右クトックし 資格情報マネジャーを開くをコンテキストメニューから選択します。
- 2. + 認証情報の追加をクルクします。
- 3. 名前 (このサンプルでは "my.oauth") を入力して OAuth 2 を型として選択します。
- 4. 認証エンドポイント、トークンエンドポイント、クライアント Id、クライアントの秘密、クライアントの秘密 テキストボックスを以前 にダウムードされている JSON ファイルからの対応する値で入力します。
- 5. https://www.googleapis.com/auth/calendar.readonly をスコープテキストボックスに入力します。
- 6. 他の設定をそのままします。

😢 Edit Credential	×
Name: my.oauth	
Type: OPassword (<u>O</u> Auth 2
Authorization Endpoint:	https://accounts.google.com/o/oauth2/auth
Token Endpoint:	https://oauth2.googleapis.com/token
Client ID:	わちろんわた ししちん さいやく やしろく ひかん かくり しろん な
<u>O</u> lient Secret:	••••••
<u>S</u> cope:	https://www.googleapis.com/auth/calendar.readonly

- 7. アクセストーケンのリクエスト をクリックして認証サーバーからトーケンを取得します (このサンプルでは Google です)。 Google ア カウト (接続するよう) 問うブラウザーウィバ ウか開かれます。
- 8. Google アカウトノエヴィノます。アプルの検証リクエストがGoogle ノ提出されていないため、以下のページの表示されます。



Advanced

BACK TO SAFETY

9. 高度なをクリックして mapforce-demo (安全ではない)に移動するをクリックします。

G Sign in with Google					
mapforce-demo wants to access your Google Account					
This will allow mapforce-demo to:					
31 View your calendars (i)					
Make sure you trust mapforce-demo					
You may be sharing sensitive info with this site or app. Learn about how mapforce-demo will handle your data by reviewing its terms of service and privacy policies. You can always see or remove access in your Google Account .					
Learn about the risks					
Cancel Allow					

10. 許可をクリックします。ブラウザー内に確認か表示されます。



MapForce はOAuth 2.0 認証コードの抽出の成功も通知します。



11.「OK」をクルクします。アクセストークンと更新トークンフィールドにデータが入力されていることに注意して下さい。

Access Token:	•••••••
Refresh Token:	••••••
Sensitive parameters	2
Include encrypted in MapF	Force Server Execution File and Mapping Deployment
	OK Cancel

12. マピングをGetCalendarEvents.mfd asとて保存します。

このチュートリアルでは MFD ファイルに暗号化を保存する チェックボックスか資格情報の編集ダイアログボックスで選択されています。このナークム 機敏なフィールド クライアント の秘密、認証トークン、更新トークン はマメピングの保存時にマメピングデザインファイル(.mfd) 内で暗号化された書式で保存されます。

認証トーケイは一定期間が過ぎると期限が切れることに注意してくたさい。これが発生すると、マンピングを実行することができなくなります(この段階では、マンピングはデザインされず、これは次のステップで発生します)。認証コードを手動で取得する必要がある場合 アクセストークンのノクエストをクリックして上記のステップに従ってくたさい。

Web サービスの呼び出しのデザイン

作成されたマメピング GetCalendar Events.mfd はまだ河も行いません。含まれている唯一のアイテムはGoogle カレンダーAPI へのアクセスを有効化する OAuth 2.0 資格情報のみです。

Web サービスの呼び出しをMapForce 内でデザインしましう

- 1. GetCalendarEvents.mfd マピングを開きます。
- 2. 「挿入」メニューからWeb サービス関数」をクリックします。Web サービス呼び出し設定ダイアログボックスか表示されます。
- 3. 手動をクルクします。
- 4. GET をパケエストメンドとて選択し URL を前のステップで入力されたWeb サービスに入力します: https://www.googleapis.com/calendar/v3/calendars/calendarId/events。
- 5. calendarid は、デメーターとて提供される必要のあるプレース市ルダのため、以下に示される通り中かって囲みます。

😢 Web Service Call Se	ttings	×
Service definition <u>W</u> SDL (Manual	Import from WADL file Import from URL	
Request Method:	GET ~	
Connection Settings		
URL:	https://www.googleapis.com/calendar/v3/calendars/{calendarId}/events	
<u>T</u> imeout:	40 seconds Infinite Dynamic URL (supplied by mapping)	

6. 2 パラメーターの追加 ポタンをクリックし以下のようこ デメーターの詳細を定義します:

Туре	Mappable	Fixed Value	Required	Repeating	Description
ite 💌 string 💌			✓		
	Type ate ▼ string ▼	Type Mappable ate string V	Type Mappable Fixed Value	Type Mappable Fixed Value Required ate string Image: Comparison of the string Image: Comparison of the string Image: Comparison of the string	Type Mappable Fixed Value Required Repeating ate string Image: Constraint of the string Image: Constraint of the string Image: Constraint of the string

上記の構成ではテンプレート]スタイルにより中かってで囲まれたURL部分をランタイムの、ラメーター値と置き換えることができます。 す。[マンピング可能]はマンピングから値を提供できることを意味します(例えば、定数から、おけは入力/ラメーターから値を提供すること)。最後に、パラメーターガム必須とマークされます。これは呼び出しはこの、ラメーター無しでは発生したいからです。

- 7. HTTP セキュリティ設定の横の「編集」ボタンをクリックします。
- 8. [HTTP セキューティ設定] ダイアログボックスで、資格情報を使用するを選択し、以前に構成されている[my.oauth] 資格情報レコードを選択します。

😢 HTTP Security Settin	ngs		×
HTTPS (TLS)			
Check server certit	ficate		
Allow <u>h</u> ost nam	e mismatch between certificate and request		
Client certificate:			Select
HTTP Authentication			
○ <u>N</u> one			
• Use creden <u>t</u> ial:	my.oauth	~	E <u>d</u> it
Use a credential fo	r OAuth or for sharing login data among compone	nts.	
O <u>U</u> sername:	Password:		
O Dynamic authentic	ation (username & password are supplied by mapp	ping)	
Preemptive auther	itication		
		ОК	Cancel

ここまでに構成済みのWebサービスコお以下のマピングの外観を持っています:

Ð	Crede	entials:				
+		Туре	Name			
1	8	OAuth 2	my.oaut	th		
2						
×						
(i)						
21						
				GET https://www	w.googleapis.com/cale	ndar/v3/
			ſ			D
			I	> w calendarid	* Status co	de D
				Body 🗐	⊞⊞Headers	
					Body =	I Å

以下のステップに従いデザインを完了することができます

1. 挿入 メニューで入力の挿入をクルクして、以下のようニュレポーネトを構成します:

🙁 Create	Input X
<u>N</u> ame:	calendarId
Datatype:	string ~
	✓ Input is required
Design-ti	me Execution
✓ Speci	ify value
<u>V</u> alue:	primary
	OK Cancel

上で表示されている通り、入力コンポーネントはデザインタイム値プライマリを持っています。APIのドキュメントによると、値プライマリはAPIサーバーは現在ログインしているユーザーのプライマリGoogleカレンダーにアクセスするように命令します。この値はデザイン時の値でないでマンピングをプレビューする際にのみ適用することか可能です。マンピングがサーバー環境で作動する場合、ランタイムで希望される値を提供する必要があります。

- 2. decode-mime-entity 関数をライブラリウイドウからマピングエリアイボラッグします。この関数はサーバーから取得した MIME ボデを文字列に変換します。を参照してくたさい。
- 3. 挿入 メニューで出力の入力 をクトックレ、プレーンな文字列として結果を出力するロールを持つ単純型出力コンポーネントを追加します。
- 4. コポーネト間の接続が以下に表示されているようしなっていることを確認してくたさい。

📲 calendarid	GET https://www.googleag	pis.com/calendar/v3/calendars,		f_8 decode-mime-en	tity	I# result
default calendarid 🕨 🔨 🛛	> 🖂 🖂 Request	🕂 🖂 Response 🛛 🔅		mime-entity		result
	▶ ₩ alendarid	Status code	> / :	> default-encoding	esuit	
	> Body 🔚	• Headers	\$ /			
		Body 🗧				

こでMapForce 内でのデザインパートは終わりです。

マピングの実行のテスト

MapForce内でマピングの実行をテストする場合、click the 出力 タブをクリックし、メッセージウィンドウ内で表示される結果を確認してく ださい。

[承認されていません(401)]などの認証エラーが表示されると、以下のトラブルシューティングのヒントを考慮してくたさい

- 1. Google カレンダーAPI か有効化されていることを確認してくたさい。Google カレンダーAPI の有効化を参照してくたさい。
- 2. 以前に取得されたアクセストークンの期限が切れている場合、新規の認証トークンのパクエストしてくたさい。
- 3. 全てのOAuth 2.0 詳細がMapForce内に正確に入力されているか再度確認してくたさい。

実行とMapForce からのOAuth 2.0 認証に成功すると、マパング出力は以下に表示されるようなマパングになります:

1	{	
2	"kind": "calendar#events",	
3	"etag": "\"p32gbjdmvo63ek0g\"",	
4	"summary":	
5	"updated": "2020-06-16T14:10:43.876Z",	
6	"timeZone": "Europe/Vienna",	
7	"accessRole": "owner",	
8	"defaultReminders": [
9	{	
10	"method": "email",	
11	"minutes": 10	
12	},	
13		
14	"method": "popup",	
15	"minutes": 30	
16	}	
17		
18	"nextSyncToken": "CKC5tt_BhuoCEKC5tt_BhuoCGAU=",	
19	"items": []	
20	}	-
Mappi	ng DB Query Output	
🛃 GetC	alendarEvents.mfd	×
Overview	✓ ₽ × Messages	ąχ
	GetCalendarEvents.mfd; Mapping validation successful 0 error(s). 0 warning(s)	
	GetCalendarEvents.mfd: Execution successful - 0 error(s) () warning(s)	
		-

このサンプル内に存在するようなイベトが存在しないGoogle アカウトを使用すると「アイテム」配列ルスポンス内では空白です。 しかし ながら、 Google カレンダーにイベトを追加してマンピングを再度実行すると、出力はこれを反映します。 サイドメモとては、デフォルトのカレ ンダー以外の公開されているカレンダーからイベトを抽出することもできます。 例えば、 [Holidays in United States] などの公開されてい るカレンダーカらデータを抽出することができます。 これを行うコは calendarld パラメーターを primary の代わりに en.usa#holiday@group.v.calendar.google.com に変更してください。

API 呼び出しに追加可能な他の デメータの詳細に関しては

https://developers.google.com/calendar/v3/reference/events/list でAPI メノドのドキュメンテーションを参照していたい。

Flow Force Server にマンピングをデプロイする

このセクションはOAuth 2.0 FlowForce Server 管理下でインストールされた MapForce Server を使用してデモ OAuth 2.0 マピングを実行する方法について説明しています。以下の必要条件が満たされている必要がおります。

- 1. FlowForce Server Advanced Edition がインストールされており、ライセンス供与されている必要がおります。
- 2. MapForce Server Advanced Edition がんストールされており、ライセンス供与されている必要がおります。
- 3. FlowForce Web Server サービスも開始され、構成されたアドレスとポートをリッスンするように構成されている必要があります。 デフォルトの設定で FlowForce Server が使用中のコンピューターにインストールされている場合、アドレスは http://localhost:8082 です。
- 4. FlowForce Server ユーザーアカウトおよびコンテナーへの書き込み権限を所有している必要があります。簡素化するために、 このサンプルはデフォルトのFlowForce Server root アカウトを使用し、マメピングをデフォルトのpublic コンテナーにデプロ んます。これらの設定は構成可能です。

マピングをサーバー環境内でショブとして実行する場合、指定されている FlowForce Server インスタンスにデプロイする必要があります。 マピングにデプロイする前に OAuth 2.0 資格情報を以下の方法の一つで対応することができます:

• FlowForce Server にデプロイされたッシケージ内で(暗号化された書式で) OAuth 2.0トーケンを含む。このアプローチでは、

埋め込まれたーケンが使用されるためジョブが実行される都度にOAuth 2.0 資格情報を提供する必要があります。 FlowForce ジョブを認証トーケンの期限が切れるまで、おけば認証サーバーが呼び出すまで実行することができます。OAuth 2.0 認証の詳細を新規のたのとオーバーライドすることができる点に注意してくたさい(以下の点を参照してくたさい)。

OAuth 2.0トークをFlowForce にデプロイされオッシケージに含まえいでください。この場合、構成する際にごうブへのOAuth 2.0 資格情報レコードについてを提供する必要があります。これを達成するためゴよ、新規のOAuth 2.0 資格情報レコードを in FlowForce Server 内に作成、おさよ、既存のOAuth 2.0 資格情報レコードを MapForce から FlowForce Server にデプロイすることができます。

このチュートリアルでは、説明のためにOAuth 2.0 資格情報はデプロイされている。シケージに含まれていません。 代わりに、資格情報は個別にデプロイされ、FlowForce ジョブを構成して参照されます。この目的のために以下のステップに従います:

- 1. MapForce 内でマピング上の空白のエリアを右クトックし 資格情報マネジャーを開くを選択します。
- 2. 資格情報レコード(このサンプルの場合 [my.oauth]) ダブルクトックし MapForce Server 実行可能ファイルとマッピング のデプロイ内に含む チェックボックスをクレアします。
- 3. マゾングデザインファイル(.mfd)を保存します。

マンピングをFlowForce Server にマンピングをデプロイする

1. 「ファイル」メニューから、「Flow Force Server にデプロイする」をクルクします。

😢 Deploy Mappi	ng		×				
Enter the host name and port of a FlowForce Administration Interface to deploy the current mapping.							
Server:	localhost \checkmark	Port:	8082				
<u>U</u> ser:	root		Use SSL				
Password:	•••••						
Login:	<default></default>	~					
Global Resources	on the Server						
Resource Path:							
	The path must start with a slash character.						
Deploy As							
<u>P</u> ath:	/public/GetCalendarEvents.mapping		Browse				
The path must start with a slash character.							
S <u>a</u> ve mapping) before deploying						
Open web bro	owser to create new job						
		OK	Cancel				

2. 適用可能な FlowForce Server 詳細を記入して OK をクリックします。デプロイロ成功すると、メッセージウイドウは関連した メッセージカ表示されます:

Messages	×
✓ ▼ ▲ □ □ □ ● \$ \$ \$ ×	
GetCalendarEvents.mfd: Mapping validation successful 0 error(s), 0 warning(s)	
GetCalendarEvents.mfd: Mapping generation for deployment completed successfully 0 error(s), 0 war	ning(s)
🐨 🖂 🥑 GetCalendarEvents.mfd: Mapping deployment completed successfully 0 error(s), 0 warning(s)	
Function /public/GetCalendarEvents.mapping was successfully deployed onto server 'localhost:8082'	
Click here to create a new job with FlowForce.	-

個別に既存のOAuth 2.0 資格情報もデプロイレましう

- 1. MapForce 内でマピング上の空白のエアを右クリックし 資格情報マネジャーを開くを選択します。
- 2. 資格情報マネージャー内で[my.oauth] レコードを右クトックし Flow Force Server にデプロイ をコンテキスト メニューから 選択します。

😢 Deploy Creder	ntial	×				
Enter the host name and port of a FlowForce Administration Interface to deploy a credential.						
<u>S</u> erver:	localhost v Port: 8082					
<u>U</u> ser:	root Use SSL					
Pass <u>w</u> ord:	•••••					
<u>L</u> ogin:	<default></default>					
Deploy As						
Path:	/public/my.oauth.credential Browse					
	The path must start with a slash character.					
	OK Cancel					

3. 適用可能な FlowForce Server 詳細を記入して OK をクリックします。デプロイロ成功すると、メッセージウイボウは関連した メッセージが表示されます:



デプロイされた資格情報を確認する場合、FlowForce Server コログイルて上記の ひから資格情報ページを開きます。

Credential						
Credential type: O Pass	Credential type: O Password O Auth 2.0					
Redirect URI:	p://localhost:8082/oauth/authorized					
Authorization endpoint:	https://accounts.google.com/o/oauth2/auth					
Token endpoint:	https://oauth2.googleapis.com/token					
Client ID:						
Client secret:	Retrieve client secret Change client secret					
Scope:	https://www.googleapis.com/auth/calendar.readonly					
Access token:	Retrieve access token Change access token					
Refresh token:	Retrieve refresh token Change refresh token					
Interoperability O	ptions					
Token endpoint authent	ication: in POST request header (standard method) 🗸					
Allow usage for HTTP:						
Allow usage for FTP:						
Allow usage for job exec	cution:					
Authorize and Save	Save Save As Delete					

Flow Force Server ジョブの構成方法

前のステップでは GetCalendar Events.mfd マピングローカルで作動している Flow Force Server インスタンスニデプロイされまた。このステップではデプロイされたマピングを Flow Force ジョブニ変換します。このサンプルではジョブはオンデマトで素早くトリガーできるようこWeb サービスと呼ばれます。

 FlowForce Server コングイン GetCalendarEvents.mapping を[Public] コンテナーから開きます。FlowForce Server 内ではデプレイされたマッピングは関数になるため、下のインターフェイス内でこの用語が使用されます。関数は認証情報 を入力パラメーターとして期待することに注意してくたさい。認証情報の名前はMapForce内で与えられているものと同じで [my.oauth]です。

Function GetCalendarEvents.mapping in /public				
Function Input Parameters				
Name: calendarld Mil Type: string required				
Name: my.oauth 🐲 Type: credential				
Name: Working-directory Type: string as directory Default:				
The function will be executed with 'MapForce' version '2020r2'.				
Create Job				

- 2. [ジョブの作成]をケリックします。ジョブ構成ページか開かれます。
- 3. [ジョブ入カ/ ラメーター] で en.usa#holiday@group.v.calendar.google.com のデフォルの値と共に作成します。(おけお以前の実行で使用されている同じ値である primary をデフォルの値として入力することができます)。

Job Inp	out Parameters				
Name:	calendarld	Type: string	~	Default:	en.usa#holiday@group.v.ca

- 4. [実行ステップ]で calendarid / デメーターを検索し [設定]をクトックして calendarid を選択します。
- 5. my.oauth パラメーターのためこclick the 한 ポタンをクリックして 既存の資格情報を選択するを選択し以前にデプロイさ れている OAuth 2.0 資格情報を参照します。デプロイ時でのデフォルトの設定が変更されていない場合 public コンテナーで見 つけることができます:



6. [サービス] でチェックボックス HTTP を介して使用できるジョブにする...をクリックしサービス名を入力します(この場合は [GetCalendarEvents])。

Service	
Make this job available via	GetCalendarEvents
HTTP at URL http://< <i>FlowForce</i> server>/service/	HTTP(S) service V

7. [資格情報]でローカルの資格情報の定義を選択し、使用中のオペレーティングシステムの資格情報を入力します。これらよ FlowForce Server アカウト資格情報とは異なりションを実行するためことの思えていましてくたさい。

С	redential				
	Run job using credential:	O Select existing credential			
		Define local credential:	User name:	tournesol	
			Password:		SHOW

8. 設定をそのままして、ジョブを保存します。

ジョブを以下のように実行することができます

- [サービスで Pist URL を新規のウィンドウで開始するボタンをクリックします。
 資格情報を提供するようこプロンプトされると、FlowForce Server アカウト 資格情報を入力します。

実行の成功とOAuth 2.0 認証ではGoogle カレンダーAPI から抽出されたブラウザーはJSON レスポンズを表示します。例:

<u>F</u> ile <u>E</u> dit <u>V</u> iew Hi <u>s</u> tory <u>B</u> ookmarks <u>T</u> ools <u>H</u> elp	- 0	×		
localhost:4646/service/GetCalendari × +				
\leftrightarrow \rightarrow C $\textcircled{1}$ $\textcircled{1}$ $\textcircled{1}$ $\textcircled{2}$ $\textcircled{1}$ $\textcircled{2}$ $\rule{0}{0}$	\ ⊡ 9			
<pre>{ "kind": "calendar#events", "etag": "\"p330c7kdqpa5uk0g\"", "summary": "Holidays in United States", "updated": "2020-06-17T07:50:31.0002", "timeZone": "Europe/Vienna", "accessRole": "reader", "defaultReminders": [], "nextSyncToken": "CMC_kNCuiOoCENC9xZLAi-oCGAE=", "items": [{ } } </pre>				
<pre>"kind": "calendar#event", "etag": "\"310151357600000\"", "id": "20190101_60o30dr46oo30c1g60o30dr56c", "status": "confirmed", "htmlLink": "https://www.google.com/calendar /event?eid=MjAxOTAxMDFfNjBvMzBkcjQ2b28zMGMxZzYwbzMwZHI1NmMgZW4udXNhI2hvb GlkYXlAdg", "created": "2019-02-21T13:46:28.0002", "updated": "2019-02-21T13:46:28.0002", "summary": "New Year's Day", "creator": {</pre>				
<pre>"email": "en.usa#holiday@group.v.calendar.google.com",</pre>		~		

上記のWebサービスの呼び出しては calendarld のデフォルトの値が使用されています。オプションでURL へ入力パラメーターを追加することができます。例: http://localhost:4646/service/GetCalendarEvents?calendarId=primary。Webサービスの呼び出しはGoogle カレンダーAPIからデーターとて提供されているカレンダー識別子からデータを取得します。

このサンプルではブラウザーから直接 W eb サービスが呼び出されているすよめ calendarld パラメーターは HTTP GET メンバを介して提供されています。 W eb サービスをプログラミング的に呼び出す場合 HTTP POST メンバを使用することができます。詳細に関しては W eb サービスの公開を参照してくたさい。

9.3.3 動的な認証

MapForce 内では ユーザー名と なワードを入力/ ラメーターと て受け入れるために基本的な HTTP 認証のためのWeb サービスを呼 び出すマメピングを構成することができます。動的な認証の構成に関する詳細は MapForce ドキュメント (<u>https://www.altova.com/ja/documentation</u>)を参照してくたさい。

FlowForce Server に動的な認証を含むマルングをデプロイする場合、ユーザー名と マワードはマルング関数の入力/ ラメーター なります。このようなマルング関数を呼び出す FlowForce Server ジョブは実行に成功するセカにユーザー名と マワードが必要です。例:

実	衍	ステップ			
	•				
	4	関数の実行 /p	ublic/DynamicAut	hentication.m	apping
		パラメーター:	username:	₩.	admin
			password:	₩.	dj9JaVax
			Headers:	(出力) 国	+
			xml_schema:	(出力) 🗐	+
			Working-directo	ry:	C:\Work

上記のサンプルでは、ユーザー名と、マワードはそれそれのテキストボックス内に入力されています。しかしなから、ジョブに入力、ラメーターとして 提供することもできます。入力、ラメーターの管理を参照してくたさい。

9.3.4 リソース

Altova グロー・シリノースはファイル、フォルダー、ませはデータベースへのポータブルなレファレンスです。グロー・シリノノースとして保管されると、 ハマとデータベース接続の詳細は再利用できるようこなり Altova アプリケーション全体で使用可能にないます。例えば、複数の Altova デス クトップアプリケーション内で同じファイルを頻繁に開く必要かある場合、グロー・シリノースとして定義すると便利な場合かあります。このようこ すると、「ファイルを開く」ダイアログボックスから対応するグロー・シリノノースを開くことかできるさか、ファイルレンを覚えている必要かありません。こ の方法はファイルレンカ変更されると、ハンの一部を変更するのみという利点かあります。

グロー・シリソースの一般的な使用法は、データベース接続を一度定義し、グロー・シリソースをサポートする全てのAltova アプリケーション で再利用することです。例えば、MapForce マメピングがデザインされるマシンでデータベース接続を作成し、MapForce Server がマメピ ングを実行するマシン上で同じ接続を再利用することができます(これは、一部の場合、両方のマシントに同じデータベースクライアントソストウェ アがインストールされている場合があります)。

任意で、(「構成」として既知の同じグロー・シリソースの複数の・シエーションを作成することができます。これによりファイル、おけよフォルダ ー・マ、(おけまデータベース)を必要に応じて変更することができます。例えば、「開発」と「生産」の構成を持つ「データベース」リノースを作成 することができます。FlowForce Serverを例に挙げると、マノピング関数に希望する構成を与えることで、開発、おけは生産データベースか らデータを簡単に抽出することができます。

次のAltova デスケップアプリケーションからグロー・シリトノースを作成することができます:Altova Authentic、DatabaseSpy、 MobileTogether Designer、MapForce、StyleVision、およびXMLSpy。サードー製品では、次のAltova サードーアプリケー ションでグロー・シリトノースを作成することができます:FlowForce Server、MapForce Server、RaptorXML Server、 RaptorXML+XBRL Server。

グロー・ジリノノースの作成に関する詳細はMapForceドキュメントの「Altova グロー・ジリノノース」のチャプターを参照してくたさい。

Flow Force Server 内のリソース

FlowForce Server の観点からよ、デスケップアプリケーション同様コンのXML ファイルコ保管されないため、グロー、シリソースはある 程度グロー、シレビオよくなります。FlowForce 内では、各リソースファイルはオゴンオルダー、マ、オゴムデータベース接続詳細を含む可能性 のあるは再利用可能なオブジェクトです。リソースをエピー、エクスポート、オゴムイルポートすることができ、他のFlowForce Server オブジェ クト同様にコロビューザーアクセスの対象となることができます。つまり、FlowForce ユーザーはマッピング関数内のすべてのリソースを使用する ことができます。 MapForce 内でグロー、 シリノースを使用するマンピングを作成すると、FlowForce Server にデプロイすることができます。 デプロイ時に デプロイメトダイアログボックス上の「リノースの使用」 チェックボックスを選択することことり、マンピングがジロー、 シリノースを使用するかを選 択することができます。 チェックボックスを選択しない場合、 マンピングにより使用されるグロー、 シリノノースは現在選択されている構成をベースこ 解決されます。 チェックボックスを選択すると、 FlowForce Server 内でもマンピング関数はノノースを必要とします。 次のマンピング関数のサ ンプルはノノースの実行を必要とする FlowForce にデプロイされます。 (最初の) デメーターがリノースファイルからデフォルトのファイル ひを取 得することに注意してくたさい):

以下に関数 ReadJSON.mapping を作成 /public							
関数入力パラメーター							
	名前:	People	(入力) 🚥 型:	文字列	デフォルト:	altova://file_resource/SourceFile	
	名前:	Text file	(出力) 圓型:	文字列	デフォルト:	Text file.csv	
	名前:	Working-directory	型:	ディレクトリとしての文字列	デフォルト:		
ļ	-עו	ース					
	-עע	-スを使用した実行関	数: /public/Gl	obalResourcesForDocu_Default.	resources		- 2

上て説明されているとおりFlowForce Server 内では、ジョブではなくマピンク関数がジロー・シリソースを使用します。例えば、マピン グが、マから読み取り、おけよ、ノマへ書き込むように構成されている場合が挙げたれます。上のサンプルでは、マピンク関数はリノースから最初の入力ファイルを 読み取ります。ジョブ構成ページからの、マをオー・トライドしない限り、これはこの関数を使用するすべてのジョブが同じ、 ノマを使用することを意味します。

FlowForce Server にグロー・シリノリースをスタイドアロンオブジェクトととしてデプロイすることができます。つまり、グロー・シリノノースをデプロイ するオークリニマノピングを最初にデプロイする必要におりません。詳細に関しては FlowForce Server にグロー・シリノノースをデプロイする詳細 「関しては、MapForce ドキュメトを参照してくたさい(<u>https://www.altova.com/ja/documentation</u>)。

リソースの構造

すべてのAltova デスケップアプケーション内では、グロー・ドルソースはXML ファイルとして管理されます。デフォルトのファイルは GlobalResources.xml で、MapForce がインストールされているエピューター上のC: \Users\<username>\Documents\Altova ディノケトリで見つけることができます。グロー・ドルトノースファイルはエイリアス」として既知の複数のトノースを含んでいます。エイリアスはファイル マ、ディノケール マデータベース接続の詳細のグループです。前述の通り、構成により、マまけはデータベースを切り替えることができます。これは以下のサンプルグロー・ドルトノースファイルの構造を見ることに理解することができます(データの一部は簡素化のため」消除されていることに注意してくたさい。

```
<Resources>
<Resource Alias="MyFile">
<Configurations>
<Configuration Location="C:\test.json" ContentKind="File"
Configuration="Default"/>
<Configuration Location="C:\production.json" ContentKind="File"
Configuration="Production"/>
```

```
</Configurations>
      </Resource>
      <Resource Alias="MyDirectory">
         <Configurations>
            <Configuration Location="C:\Test" ContentKind="Folder"
Configuration="Default"/>
            <Configuration Location="C:\Production" ContentKind="Folder"
Configuration="Production"/>
         </Configurations>
      </Resource>
      <Resource Alias="MyDatabase">
         <Configurations>
            <Configuration ContentKind="DataSource" Configuration="Default">
               <DatabaseContextInfo vendor="sqlite" connection="C:</pre>
\Resources\Test.sqlite"/>
            </Configuration>
            <Configuration ContentKind="DataSource" Configuration="Production">
               <DatabaseContextInfo vendor="sqlite" connection="C:</pre>
\Resources\Production.sqlite"/>
            </Configuration>
         </Configurations>
      </Resource>
</Resources>
```

上記のファイルおつのハーズエイリアスを定義します:「MyFile」と呼ばれるファイル ス「MyDirectory」と呼ばれるディレクトリ ス「MyDatabase」と呼ばれる SQLite データベース。各エイリアスコ おつの構成が存在します:テストのために使用されるデフォルトの構成と 生産の構成です。

FlowForce Server 内では、複数のユーザーサード・環境の特性のな、リノースは若干異なって作動します。具体的はよ、 FlowForce 内でリノースオブジェナトになるような XML リノースファイルの場合、デスケトップアプリケーション何のように複数のエイリアスが存 在することができます。各エイリアスははつの構成のみ存在することができ、MapForce から FlowForce Server ヘデプロイする際に選 択されている構成を指します。

MapForce からFlowForce Server にグロー ジリノースをデプロイする場合、一度に1つの構成だだデプロイすることができます。

例えば、上記のグロー・Ňリハースファイルをデプロイレた場合、サーベービ「デフォルト」ませま「生産」構成のとちらかがデプロイされます(一度に両方の構成はデプロイされません)。「デフォルト」の構成を選択する場合、リノースオブジェクトはFlowForce Server 内で以下のよう しています:



FlowForce 内では、マピング関数はグローィ ジリリノースの特定の構成を1つ使用することができます。このナムな、このサンプルではサーィー上の「生産」構成が必要しなり、MapForce 内のデプロイダイアログボックスから構成「生産」を選択し同じリノースをもう一度デプロイします。 代わりに、サーィー上で以下で説明される通り直接リノースを作成し、マピング関数を「デフォルト」の代わりにポイントするように変更することができます。代替のアプローチはファイルとディレクトリリノースに対してのみ可能で、データベースに対してしば可能ではおりません。

マピング関数のリソースの変更

FlowForce内では、
3 アイコイによりリソースオブジェクトは識別されます。このため、上のサンプルからデフォルト」と
生産」構成がデプロ
イされている場合、FlowForce Server内の対応するリソースが以下のように表示されます。

Name		Type 🗢
ExampleGlobalResources_Defau	resources	
ExampleGlobalResources_Produ	resources	
Create Import Objects	Move or Rename	Selected Objects

マッピング関数により使用されたリソースを変更する方法:

- 1. マピンク関数がデプロイされているコンテナーイン移動し、クトックして関数を開きます。
- 2. 「リノース」の下で、新規リノースのを選択します。リノースの選択は関数や資格情報などの他のFlowForce オブジェクトと同様 に作動します。

Resources		
Run function using resources: /public/ExampleGlobalResources_Default.resources	- 2	
The function will be executed with		
Name 🗢		
Save Save As ExampleGlobalResources_Default.resources		
ExampleGlobalResources_Production.resources		

マピング関数が「リノース」セクションを持たない場合、(おけよ、デプロイ時に「リソースの使用」チェックボックスが選択されていない場合)マッピングはMapForce内のグロー・ジリノースのために構成されません。

以下の条件が満たされている場合マンピンク関数はすべてのハノースを使用することができます。

- リノース型か関数と互換性を持つ場合。例えば、「フォルダー」リノースは、ファイルリリノースを必要とするマンピング関数には適してしません。
- リノースエイリアス名はマピング関数により必要とされているものです。MapForce内では通常エイリアス名はマピングのデザイン時に選択されますが、FlowForce内では以下で説明されている通りオーバーライドすることができます。

リソースとショブ構成

上記の通り、リノースはショブ構成レベルではなくマンピンク関数レベルで消費されます。ショブが実行されると、ショブによい呼び出される関数内 で定義されているグロー・ ジルリノースを消費します。このために、ショブ構成ページからジョブを編集する場合、リノースエイリアスのオー・・・ライド などリノースに対して最小の構成オプションのみを使用することができます(下で詳しく説明されています)。

ー部の場合は、(フォルダーませまファイルのようコリノースを直接ジョブ構成ページから参照することが可能な場合があります。これは、全ての コンテキストで可能な場合がますが、このようなレフォレンスを使用する良い理由がよい限り一般的には回避さえるべきです。

メモ 実行ステップの作業ディレケリル・ラメーターからリソースを参照してはなりません。これはリソースの処理がMapForce Server 処理が既に開始されていることを必要としますが、作業ディレケトリはMapForce Server が開始する前に設定されているからです。

リソースエイリアスの上書き

ファイル、おオコフォルダーリノースは複数のエイリアズを持つことができますが、ジョブランタイムでは、つのエイリアスのみ使用することができます。 マノピングのデザイン時にMapForce内でランタイム時に使用されるエイリアスが選択されます。例えば、次のMapForceエンポーネント は "MyDirectory" と呼ばれるディレストリエイリアスに output.csvを生成するようは「構成されます。FlowForce Server にこのマノピ ングをデプロイする場合、サーバー上のマノピング関数は "MyDirectory" エイリアズを含むリノーズを指している必要があります。

😢 Component Settir	ngs		×
Component name:	Text file		<u>Q</u> K <u>C</u> ancel
		Input file	
altova://folder_reso	purce/MyDirectory/output.csv	Output file	
✓ Sa <u>v</u> e all file paths	relative to MFD file		

エイリアスを変更する都度 MapForce 内でマピングを編集する代替としてはジョブ構成ページから FlowForce Server 内でエイリアスを オーバーライドすることができます。ジョブ内のファイルませまフォルダーエイリアスをオーバーライドする場合、MyFile ませま MyDirectory を必要とされるエイリアス名に置き換えます。

必要とされる型	サンプル
ファイル	altova://file_resource/ MyFile
ディレクトリ	altova://folder_resource/ MyDirectory

例えば、下のジョブの構成ではディレクトリエイリアスは TestDir」に変更されています。

E	xecu	ition Step	os		
	+				
	4	Execute funct	tion /public/FolderR	lesource.map	pping
		Parameters:	production:	(input) ⁵⁵⁰⁰	+
			Text file:	(output) 🗐	altova://folder_resource/ <mark>TestDir</mark> /output.csv
			Working-directory:		+

メモ 上記の通りエイリアスの上書きはデータベースリソースのためにサポートされていません。複数のデータベースエイリアスが存在する場合、マンピングをFlowForce Server にデプロイする前にMapForce 内で必要とされるデータベースに切り替えます。

リソースの作成

FlowForce Server 内でファイルおけはフォルダーリンースを作成することができます。FlowForce Server 内でグロー・シルトノースを作成することができます。

Resources name:	Resource1
Resources descrip	ption: Defines miscellaneous input or output paths used in mapping jobs.
Resources	
Resources	
File Resour	ces
4	
- Resour	ce alias: SourceFile
File:	C:\Resources\data.txt
new File resou	irce
Folder Reso	burces
new Folder re	source
Save	

メモ データベースリソースの作成はサーバー環境ではサポートされていません。データベースリソースを作成するこは、MapForce おうよ 他のAltova デスケトップアプリケーションのグローイ ジリリソースエディターを使用し、MapForce からFlowForce Server ヘリン ースをデプロイレてください。

リノースエイリアスはこのリノースを使用する箇所のマンピング関数により比津町とされているものに一致する必要があります。それ以外の場合、「リノースエイリアスの上書き」で説明されている通り、正確なエイリアスを指すようにごうごを手動で行う必要があります。

同じリソースオブシェクト内で、必要な場合、「新規のファイルリソース」 おさよ「新規のフォルダーリソース」 ボタンをクリックして複数のエイ リアスを作成することができます。これはオプションです。 複数のエイリアスを作成する場合、 どのエイリアスか使用されるか指すために ジョブを変 更する必要かあることを忘れないでくたさい。

リソースの編集

MapForce 内で行い 再デプロイする代わりに FlowForce Server 内でファイルまけはフォルダーリソースを編集することができます。リノー スを編集するコよ 対応するレコードをクトックし、ノス(おけよ、データベース接続の詳細を更新し、「保存」をクトックします。

メモ データベースリソースの場合、接続文字列、おさま、デフォルトのデータベースなどの特定のフィールドのみをFlowForce内で編集 することができます。データベースペンダーと接続メンバを変更することはできません。

リノースを更新すると、この更新は以下に影響を及ぼします

- ・ リノースを参照する全てのマパンク関数
- 対応するマピング関数を呼び出す全てのジョブ

9.4 マッピング/変換結果へのアクセス

MapForce マルング おけま Style Vision 変換がFlowForce Server にデプロイされた後、実行ステップから呼び出すことのできる FlowForce 関数になります。 例えば、 下のジョブの最初のステップは、 SimpleTotal.mapping と呼ばれ、 実行されます。

実行	マテップ	
	関数の実行 /	oublic/SimpleTotal.mapping
	パラメーター:	ipo: (入力) 鲳 🕂 Working-directory: 🔸
=	ステップの 結界	を割り当て output
4	関数の実行	ystem/compute
	バラメーター:	条件式: stdout (output)
=	ステップの 結界	を割り当て 名前 以下としてTO
新	規の実行ステップ	新規の選択ステップ 新規の For-each ステップ 新規エラー/成功処理ステップ

ジョブは 2つのステップにと構成されています

- ステップ1 は、MapForce Server は、実際に実行する SimpleTotal.mapping 関数を呼び出します。重要な点は 「このステップの結果を割り当てる」フィールドがマンピングの結果に名前を与える点に注意してくたさい、この場合、output。 しかしながら、名前を変更することは可能です)。
- 2. ステップ2は、マピングの出力をストリームに変換する/system/compute 関数を呼び出します。

デフォルトでは、マメング おうは変換 関数の出力は、ジェネルクな型「result」です。出力を使用するうは、「result」を必要なデー ダ例えば、文字列、ストリーム、ファイルに変換する必要があります。この目的のために、/system/compute ビルトイン関数と FlowForce 式関数を使用することができます。上のサンプルでは、ビルトイン関数 /system/compute が必要とするデータ型を変換す るためつ呼び出されています。すなわち、式 stdout (output) は、前のステップの結果をストリームに変換します。

下のテーブルは、マピングおけば変換関数の結果を処理するために使用される可能性のある FlowForce 式の例をリストしています。これ ら全てのサンプルは、内に入力されたoutputは、「このステップの結果を割り当てる」」内に入力されている名前であることに注意してください。

FlowForce 式	目的
stdout (output)	outputをオリーム」変換します。
as-file(stdout(output))	output をファイルニ変換します。
<pre>as-file(nth(results(output), 0))</pre>	output が複数のファイルこと構成されている場合このような式が 必要しています。(MapForce お日はStyleVision 内で)単一の 出力ではなく、複数の出力を生成するようこデザインされると発生しま

FlowForce 式	目的
	す。 式は、 output をストリームのシーケンスに変換し、シーケンス から最初のストリームをピックアップし、 ファイルに変換します。 サンプル に関しては、 次を参照してくたさい、 <u>Style Vision 変換からジョブを</u> 作成する。
as-file(nth(results(output , "CompletePO"), 0))	ゼロベースのインデックスではなく、名前別にストリームのシーケンスからファイルは抽出されます(この場合、、CompletePO、)。

ステップ おうまジョブの結果を処理するためのFlowForce 式関数に関するレファレンスについては 次を参照してくたさい、ステップ結果関数。FlowForce 式の概要については <u>FlowForce 式</u>を参照してくたさい。

9.5 RaptorXML Server との統合

RaptorXML がFlowForce に統合される場合、RaptorXML Server により公開されているすべての関数はようブウで呼び出すことが できるようこFlowForce で使用できるようゴンリます。具体的には、RaptorXML 関数はFlowForce の/RaptorXML コンテナー内 に存在しています。RaptorXML+XBRL Server の場合、コンテナー名は/RaptorXMLXBRL です。

			検索語の	入力	▲ 検索 🗹 再帰的
2 8 m	型 🗢	日付は変更されました	以下により変更されました	次の実行	
C C RaptorXML	コンテナー				パーミッション
C C RaptorXMLXBRL	コンテナー				パーミッション
🗆 💼 public	コンテナー				パーミッション
🗆 🗅 system	コンテナー				パーミッション

FlowForce ビルトイン関数の呼び出しに類似したショブから RaptorXML 関数を呼び出すことができます:

- /RaptorXML(まけは/RaptorXMLXBRL) コンテナー内で、興味のある関数を開き、「ジョブの作成」をクルクします。/RaptorXML/valjson などのジェネルクな関数を参照、おけは、/RaptorXML/2021/valjson などのを参照することができます。これら2つの相違点は以下で説明されています。
- ジョブ内で新規の実行ステップを作成し、実行ステップから希望する RaptorXML 関数を呼び出します。例えば、下のステップは valjson 関数を呼び出します。

実行ステップ		
・ ▲ 関数の実行 /Ra	aptorXMLXBRL/valjson	▼ 2
パラメーター:	作業用ディレクトリ:	C:Vobs
	エラー フォーマット: ?	テキストフォーマット 〜
	JSON ファイル: ?	orders.json
	エラー制限: ?	○ raptorxmlcmdl.core_error_limit_value_unlimited ● 他の値: 100
	詳細: 🕐	
	警告制限:?	•
	JSON5 サポート: 🕐	•
	JSON ラインサポート: 🥐	•
	書式のチェックを無効化する: 🕐	•
	JSON スキーマパス: ?	orders.json.schema
	JSON スキーマバージョン: ?	•
	XML ユーザーカタログ: 🕐	(+)

RaptorXML Server を呼び出すジョブのサンプルに関しては、以下を参照してくたさい

- <u>RaptorXML を使用してドキュメトを検証する</u>
- <u>エラーログのある XML を検証する</u>
- RaptorXML を使用してKey/Value ノウメーターペアをパスする

すべての Raptor XML 関数への参照に関しては Raptor XML Server ドキュメトを参照してくたさい (<u>https://www.altova.com/ja/documentation</u>)。

手動の統合

FlowForce Server とRaptorXML Server 間の統合は多くの場合自動的に行われます(例えば FlowForce Server インストールをWindows で実行し、RaptorXML Server もんこくールする場合など)。しかしなから、2つの間での統合か手動で行われる必要が

ある場合もあます。手動の統合は通常、個別に異なる、デジョンのFlowForce Server とRaptorXML Server がインストールされて しる場合に必要とされます。例えば、特定のRaptorXML Server バージョンの関数の定義がFlowForce Server インターフェイスで 不足している場合、RaptorXML Server のその、デジョンがインストールされている場合でも、手動の統合が必要とされます。

手動の統合を行うコよ 以下の 次で見つけることのできるスクリプトを実行します: RaptorXML installation directory} \etc\functions\integrate.bat。

メモ Unix システムでは、スクリプト名はintegrate.cs です。スー・レユーザー権限(sudo)はこのスクリプトを実行するためこ必要とされています。

スクリプトはつの引数を取ります: FlowForce Server インストールディレクトリークの やと、と FlowForce Server データディレクトリークの いな(FlowForce Server アプリケーションデータを参照してください)。 このスクリプトを実行すると、次の状態が発生します:

- 統合されたRaptorXML Server バージョンのすべてのリリース固有の関数はジョブとして呼び出すためにFlowForce Server で使用できるようてかます。
- ジェネトックな(リリースを指定しない)RaptorXML 関数は統合された RaptorXML バージョンのリース固有の関数のポイト までアップデートされます。

スクレフトカエラーを返すと、統合されたRaptorXML バージョンの関数の定義は、FlowForce Server と互換性を持っていません。もしもこの状態が発生した場合はサポートに連絡してくたさい。

ジェネリックとリース固有のRaptorXML 関数

RaptorXML、およ、RaptorXMLXBRL コンテナーで使用することができる関数は以下のようこ整理されています。

- /RaptorXML コンテナーからの関数は、(RaptorXML 関数をサポートする最初の、ージョンである) FlowForce Server の 2014 バージョンまで下位互換性があります。これらのジェネトックな関数は、/RaptorXML/{Release} コンテナーからのリース 固有の関数のラッペーとしての役割を果たします。これらはリノース間では互換性がありますが、最近インストールされている RaptorXML Server のすべての機能を提供しません。
- /RaptorXML/{Release} コンテナーからの関数は対応すr RaptorXML リレースのすべての機能を提供します。これらの関数 は、同じリレースのFlowForce Server と互換性があります。しかしなから、RaptorXML Server のすべての・・ジョンは FlowForce Server のすべての、・・ジョンと互換性を持つとは限りません。(「手動の統合」で説明されているとおり、統合スクレプトを実行して互換性を確認してくたさい。

ジョブがRaptorXML 関数を呼び出す場合、関数はRaptorXML Server のリース固有の関数と同等のラットとしての役割を果たします。選択されたRaptorXML リースは手動で統合されたものを含む最近 FlowForce に統合されたものです。上記の通り、(新規の引数、おけよ、関数などのこのような呼び出しは最新のRaptorXML 機能からの機能の利益を得ることはありません。FlowForce ジョブから最新のRaptorXML 機能を活用するはよ、リース固有の関数を触接呼び出します。



リリース固有の関数はRaptorXML.tool ファイルがRaptorXML 実行可能ファイルを検索するようと使用されるかを決定します。個別 の.tool ファイルは各 RaptorXML Server リリースのためご存在します。.tool ファイルはFlowForce Server にRaptorXML Server 実行可能ファイルのロケーションを指定し、環境変数を設定するようは主使用することができます。環境変数の設定を参照してくだ さ、

FlowForce ジョブがデンシン固有のRaptorXML 関数を参照する場合、そして FlowForce Server と RaptorXML Server の 新しいデジョンニアップグレードする場合、最新のリース固有の RaptorXML 関数の時点にまですべてのショブを変更、おけよ Raptor.tool ファイルを RaptorXML Server 実行可能ファイルの最新のデジョンコントの通りマップすることができます:

- 1. Raptor_<release>.tool を最近にインストールされたリリースのRaptorXML Server の{installation}\etc デルクト リカら同じリリースのFlowForce Server の{configuration data}\tools デルクトリコニピーします。
- 2. (ジョブがポイントする Raptor リリースを指すようこ古リリースのゾージョンに一致するようにファイルの名前を変更します。例えば、 古しリリースがRaptor XML 2017r3 の場合、ファイルを Raptor_2017r3.tool に名前を変更します。

マピングアプローチをとる場合、全ての既存のジョブはRaptorXML 2017r3 関数を呼び出すように継続され、.tool ファイルは最新の RaptorXML Server 実行ファイルまでマップします。

9.6 ツールファイル

他のAltova サーバーがFlowForce Server と共にインストールされている場合 .tool 拡張子がFlowForce Server 管理下で作動 する各実行可能ファイルのためにインストールされます。FlowForce Server は.tool ファイルのためにDATADIR\tools と INSTALLDIR\tools ディレストリをスキャンします。前のディレストリが後のディレストリよけ、優先順位を与えられます。

FlowForce Server アプリケーションデータディレクトリ(DATADIR)

Linux	/var/opt/Altova/FlowForceServer2021/data
macOS	/var/Altova/FlowForceServer2021/data
Windows	C:\ProgramData\Altova\FlowForceServer2021\data

FlowForce Server インストールディレクトリ(INSTALLDIR)

Linux	/opt/Altova/FlowForceServer2021/
macOS	/usr/local/Altova/FlowForceServer2021/
Windows	C:\Program Files\Altova\FlowForceServer2021\ C:\Program Files (x86)\Altova\FlowForceServer2021\

DATADIR\tools デルノオリは通常は空で、カスタマイズかされたソーリレファイルを配置する場所です。INSTALLDIR\tools デルノオリと含まれているソーリレファイルは変更される必要はありません。

アプリケーションプール

(名前によ) 識別される) 検出されたそれそれの.tool ファイルのために、FlowForce Server はアプリケーションプールを作成します。アプリケーションソールはですから実行中のソール処理インスタンスをグループ化し、FlowForce Server はこれらのインスタンスをソールファイル内で構成される通り管理します。

FlowForce がソール処理インスタンスのライフタイムに関数る情報を強制する場合、これらのイベトはログ内に情報メッセージを生成します。例

Starting instance {id} of {tool} for {session}.
Starting {commandline}.
Instance {id} of {tool} for {session} is now idle.
Shutting down instance {id} of {tool} for {session}; sitting idle for too long.
Shutting down instance {id} of {tool} for {session}; maximum reuse count reached.
Instance {id} of {tool} for {session} unexpectedly ceased communication.
Instance {id} of {tool} for {session} attached to job instance {instanceid}.

上記にリストされている情報メッセージはライセンスませるお問題を表示しません。代わりに特定の時刻に実行された処理に関する情報を提供することで可能な問題をトラックすることができます。ステップませるようが失敗すると、個別のログメッセージが生成されます。

ツールファイル構成

常に必要ではありませんが、ツールファイルを編集することができます。具体的に、以下の構成オプノョンを提供します

1. [ツール] セグションの下の実行可能、な古いレージョンの.tool ファイルか新しいレンションを実行する、おけての逆の場合など 特定のケースではこの、な変更する必要があります。 2. [Environment] セグションソールによい必要とされる環境変数を定義するためにこのセグションを追加ませば編集することができます。環境変数の設定を参照してくたさい。

重要点:

- Altova サポートにアドバイスされた場合を除き上記以外のソールファイル設定を変更しないでくたさい。
- カスタムソールを定義することは可能です。

9.6.1 環境変数の設定

FlowForce Server 管理下で泊づ出て、MapForce Server マピング、おは、StyleVision Server スタイルシートが作動する 場合、では環境変数が設定される必要がある場合があます。(例えば、データベースは接続する際にJDBC ドライバーの場所を指定す なっかのCLASSPATH)。

MapForce Server マピング、おけよ StyleVision Server 変換に必要な環境変数を設定するけよ 対応する Altova サードー製品の.tool ファイルを編集してくたさい。..tool ファイルを編集するけよ DATADIR がアプリケーションのデータディレクトリである DATADIR\tools を最初にチェックしてくたさい

FlowForce Server アプリケーションデータディレクトリ(DATADIR)

Linux	/var/opt/Altova/FlowForceServer2021/data
macOS	/var/Altova/FlowForceServer2021/data
Windows	C:\ProgramData\Altova\FlowForceServer2021\data

.tool ファイルがDATADIR\tools に存在しない場合、INSTALLDIR がアプリケーションのインストールディレクトリである FlowForce Server のINSTALLDIR ディレクトリからピーしてください。

FlowForce Server インストールディレクトリ(INSTALLDIR)

Linux	/opt/Altova/FlowForceServer2021/
macOS	/usr/local/Altova/FlowForceServer2021/
Windows	C:\Program Files\Altova\FlowForceServer2021\ C:\Program Files (x86)\Altova\FlowForceServer2021\

MapForce Server おはStyleVision Server がFlowForce Server の後にインストールされている場合、 INSTALLDIR\tools ディレクトリ内に.tool ファイルを見つけることができます。.tool ファイルがFlowForce アプリケーションデータディレク トリ、おけま FlowForce インストールディレクトリニ存在しない場合、FlowForce Server がMapForce Server おけま StyleVision Server の後にインストールされた可能性が高いです。MapForce Server まけま StyleVision Server しま対的な etc ディレクトリ内で.tool ファイルを見つけることができます。

(インストールデルクトリではなく)FlowForce Server アプリケーションデータデルクトリに.tool ファイルをコピーしてくたさい。 FlowForce Server アプリケーションデータディノカリ内の.tool ファイルは他のディレクトリエ既存の.tool ファイルより上優先順位が 高いです。

警告: migratedb コマイを FlowForce の最新のメジャー・・・ジョンへのアップグレード中に実行すると、前の、・・ジョンのアプリケー ションデータディノオリカらの.tool ファイルが新しし 、・・ジョンのアプリケーションディレクトリニコピーされます。これは期待されない結果を 引き起こす可能性がありますので、アプリケーションデータディレクトリ内で必要な.tool ファイルのみを保管するようしてくたさい。 .tool ファイル内の[Environment] セクションの下に必要な環境変数を追加することができます。例えば、CLASSPATH 変数を設定する.tool ファイルは、以下のようになります:

```
[Environment]
CLASSPATH=.:/usr/local/jdbc/oracle/ojdbc6.jar
```

サンプル.tool ファイル(Linux)

.tool ファイル内出設定された環境変数は、他の方法で定義された環境変数をオーバーライドします。

シェルコマドの実行、おは、FlowForce Server ジョブとてスクリアとて実行するには、/system/shell/commandline関数を参照してけない。

10 AS2 との統合

AS2 (Applicability Statement 2) はインターネット上でファイルを安全に交換することを可能にする仕様です。AS2 はHTTP HTTPS を使用してEDIINT (インターネット上のEDI)とXML ファイルを伝送することができます。

このドキュメノトロは以下に関連するレファレンスか含まれています:

 RFC 4130、「MIME-Based Secure Peer-to-Peer Business Data Interchange Using HTTP, Applicability Statement 2 (AS2)」。次を参照してたさい、<u>https://www.ietf.org/rfc/rfc4130.txt</u>。

主な機能

- FlowForce Server Advanced Edition を使用すると、機関のAS2取引パートナーにFlowForce ジョンを使用して、 AS2フォーマトのシッセージを送信することができます。取引パートナーからAS2メッセージを受信することもでき、必要に応じて更に処理し、FlowForce Server をAS2 Server にすることもできます。
- 証明書を使用してパトナーは送信されるAS2 メッセージを任意で暗号化し署名することができます。暗号化と署名をサポト するはよ (AS2 データの送信、おけよ 受信パトナーとして)全ての取引パトナーから受信される公開証明書を中央でイン ポトし管理することのできる証明書ストアを有し、公開と秘密証明書の組合せば所属機関により作成されることが必要です。こ の結果、他の取引パトナーからサイン済み、かつ、暗号化されたAS2 メッセージを受信する際、FlowForce Server は、こ のようなメッセージの署名を暗号化し検証することができます。同様に、暗号化済み、署名済みのデータを送信する際、 FlowForce Server はそれぞれの証明書を使用して、ストアにインポートされたデータを準備します。
- From FlowForce から、FlowForce Server から送信されるAS2 メッセージの応答として、任意でパトナーイコ同期メッセージ処理通知 (MDN)を送信するようごを送信することをリクエストすることができます。パトナーイこ MDN の署名をリクエストすることもできます。When FlowForce Server がAS2 メッセージの受信者としての役割を果たす場合、MDN を受信済みのAS2 リクエストの応答として自動的に送信します。
- FlowForce Server は次のアルゴリズムを使用してデータを暗号化、おけよ、暗号化の解除を行うことができます: DES、 3DES、AES-128、AES-192、AES-256、RC2-40、RC2-64、RC2-128、RC4-40、RC4-128。次のアルゴ リズムを使用して署名済みのデーダを署名、おけよ、検証することができます: MD5、SHA-1、SHA-224、SHA-256、 SHA-384、および SHA-512。
- 任意で、送信されたシッセーンの圧縮を行うことができます(署名の前後に圧縮を行うかを任意で指定することができます)。圧縮 済みのAS2 デーダを他の取引パトトナーから受信すると、必要な場合、FlowForce Server は(デー外署名の前後に圧 縮されたかに関わらず)自動的にデータの圧縮を解除することができます。
- AS2 データを受信、おけよ、送信するショブをビジネスデータフローイン統合し、他のFlowForce ジョブと同様にカスタマイズすること ができます。 例えば、ジョブをオンデマンド、おけよ、計画されたとおりに複数の実行ステップ、条件付き処理、ユーザーアクセス権 などを使用してトリガーすることができます。 加えて、FlowForce ビリトイン関数 とFlowForce 式関数による全ての機能を 活用することができます。

制限

- 現在、FlowForce 同期 MDN (メッセージ処理通知)のみをサポートします。同期 MDN は現在サポートされていません。
- メッセージのサイズは、使用することのできるシステムメモルにより制限されています。
- 基本的な HTTP 認証はサポートされて ます (プリエンディブな資格情報は最初のリクエストに含まれています)。 クライアート証明書を使用するダイジェスト認証、 おさよ、HTTPS 認証はサポートされていません。
- (証明書を持たな)秘密キーを含むPEM ファイルのインポートはサポートされていません。

10.1 概要: AS2 データの送信

下の図はFlowForce Server Advanced Editionを使用してAS2 メッセージを送信する イレベルのプロセンを示しています。



FlowForce Server を使用したAS2 メッセージの送信

上記のプロセスは以下のように作動します

Step #	説明
1. AS2 パートナーと証明書の構成	AS2 パトナーとの通信をセオアップするコよ (URI とAS2 名など) AS2 接続の 詳細と交換証明書を取得する必要があります。証明書 (とパトナーの詳細)は

Step #	説明
	FlowForce Server にんポートされる必要が対ます。次を参照してくたさい <u>AS2</u> 証明書の構成とAS2パートナーの構成。
2. ジョブの作成	FlowForce ジョブショブはAS2 メッセージを送信するかとれ成される必要があま す。FlowForce Server ジョブは必要に応じて多種の方法で実行されるよう、構成 することができます。例えば Web サービスの呼び出しとして実行することができます。ま たま、ファイルシステム上のファイルに変更か加えられると、特定の日時に実行されるよう にスケジュールすることができます。次を参照してくたさい、トリガーの管理。このプロセス は完全に自動化されています。
3. ジョブの実行とAS2 メッセージの送信	AS2 メッセージを送信するコよ your ジョブ(おコよ ジョブ内の実行ステップ) は FlowForce Server ビルトイン 関数 /system/as2/send を呼び出す必要がありま す。この関数は ステップ1 で定義されてい るパトナー・オブジェイト、および送信するメッセ ージェンテンンを含む、AS2 メッセージを送信するオッカコ必要なパラメーターをとします。 ジョブは多種のFlowForce Server AS2 式関数 をマメピングの出力を必要とされる 書式に変換するすっかコ呼び出す必要があります、例えば、ファイルからストリームニ変換 する場合など)。
4. 同期 MDN を使用してパトナーに返答 する	AS2 パトナーオブジェクトをステップ3 で作成すると、FlowForce Server により送 信されるAS2 メッセージの返答とて メッセージ処理通知 (MDN) を返すようコット ナーゴ注意でリクエスト することができます。パトナーは、を FlowForce Server から 送信される HTTP 呼び出しと同じセッション内に MDN を送信する必要する必要かあ ります(すなわち、「同期」として構成されている必要があります)。

上のダイアグラムよ簡単な構成を表示して、ます。このダイアグラムはAS2 メッセージロシ要なコンテンンが準備されており、FlowForce Server ジョブに入力として提供されるのみであることが想定されています。多種のノースからマピングデータにより自動的にAS2 メッセージ コンテンンを生成する必要かある場合、AS2 プロセスは、Altova MapForce とMapForce Server を使用して更にプロセスを自動化 することができます。次を参照してくたさい、MapForce とMapForce Server を使用したAS2 との統合。

詳細な手順に関しては、次を参照してくたさい、AS2 メッセージの送信.

10.2 概要: AS2 データの受信

下の図はFlowForce Server Advanced Edition を使用してメッセージを受信する イレインルのプロセンを示しています。



FlowForce Server を使用したAS2 メッセージの受信

上記のプロセスは以下のように作動します

Step #	説明
1. AS2 パートナーと証明書の構成	AS2パトナーとの通信をセオアップするコよ(URIとAS2名など)AS2接続の 詳細と交換証明書を取得する必要があります。証明書(とパトナーの詳細)は

Step #	説明
	FlowForce Server にんポートされる必要があります。次を参照してくたさい AS2 証明書の構成 と AS2 パートナーの構成。
2. ジョブの作成	A FlowForce ジョブよ AS2 リクエストのナカニFlowForce がリッスンするAS2 サービスを公開されるナカコ作成される必要があります。
3. パトナーにおAS2 データの送信	パートナーとサービスのURLを共有すると、AS2 リクエストの送信を開始することができます。
4. 同期 MDN を使用してパートナーニ返答 する	FlowForce Server がAS2 メッセージを受信すると、成功、おけよ 多種の検証エ ラーを通知する同期 MDN レポートを返します。
	それ以外の場合、受信されるデータはAS2 サービスを公開するショブにより処理されま す。必要に応じてショブがデータを処理するよう、構成することができます(例えば、メッセ ージをストリームから文字列に変換、メッセージから特定のヘッダーの読み取り、カスタム 名を持つファイルにデータを保存、送信元のパートナーの名前を取得するなどの処理が 挙にちれます)。
	AS2 仕様によると、MDN ではメッセージのコンテンツではなくメッセージのデリパリ ーのみか铐慮するべきです。この理由により、AS2 受取ジョブは、できるだけ最小 である必要があります(通常、メッセージはファイル、または、データベースに保存さ れます)。
	メッセージのコンテンソが理由で AS2 受取ジョブは失敗することはありません。この ため、余分なステップは(メッセージの受信と保存以外で) 個別のジョブとして定 義される必要があります。これ以外の場合、受取ジョブがメッセージにデリ、シーに 関連しないステップを含まず、ステップが失敗すると、AS2 仕様にでは期待され ない(ネガティブな) MDN の失敗を引き起こします。
5. 受信済みのAS2 データの処理	AS2 サービスを公開する同じジョブこと受信されるデータは処理されます。必要に応じてデータを処理するようごショブを構成することができます(例えば、ストリームから文字 列にシッセージを変換する、メッセージから特定のヘッダーを読み取る、カスタム名と共にデ ータをファイルに保存する、取引パートナーの名前を取得するなど)。

FlowForce をAS2 サーバーとて構成する詳細に関しては、次を参照してくたさい、AS2 メルセージの受信。

10.3 MapForce と MapForce Server を使用した AS2 との統合

FlowForce Server Advanced Edition は、取引パトナーにAS2 メッセージを送信、おけよ、取引パトナーからAS2 を受信する ためこ必要な機能を提供します。更に、FlowForce Server はビルドインの関数を使用してAS2 データを処理し、ローカルで保管する ことができます。更に高度な必要性に関しては、既存のノーズ例えば、データベースからAS2 データを準備する場合、おけよ、他の書式 に変換する場合、おけよ、Web サービスに送信する場合、MapForce とMapForce Server をAS2 プロセスに含むことができます。

MapForce とMapForce Server か必要なシナリカシつ存在します:

- 1. AS2 パートナーイン送信する前に、MapForce によりサポートされる書式(XML、XBRL、Excel、データベース、Web サービス)でデータをマップおけませ成する場合。
- 2. AS2 パートナーから受信下デーケを多種の方法で変換する場合 (例えば Excel に変換する。異なる XML スキーマに変換する場合。 データベース に保管する Web サービス に保管するなど)。

AS2 データの生成と作成

パートナートご送信する前に、 MapForce を使用して AS2 デーダを準備、おけよ 生成するシナリオでは、高度なプロセスは以下のようよな ります:



ハイレベルのAS2 プロセスのダイアグラム

上のダイアグラムでは MapForce Server とFlowForce Server は同じマンイニインストールされている必要があります(Windows、 Linux、おけよ macOS オペレーティングシステムを使用することができます。システムの必要条件を参照してくたさい(Windows マシン であることが前提で MapForce はMapForce Server とFlowForce Server と同じマシン、おけよ 異なるマシン上でHTTP おこ はHTTPS を介して FlowForce Server に接続することができます。AS2 パートナーは、HTTP(S)を使用して FlowForce Server が通信する遠隔のサーバーです。

上記のAS2 プロセスお以下のようご作動します。

ステップ #	説明
1. EDI/XML データマッピングのデザインとテ スト	MapForce を使用すると、(テキスト、CSV、JSON、XML、多種のEDI フレーバ ー、データベース Web サービス)異なるフォーマトのデータを入力として取るデータマッ ピングの変換をデザイン人、(例えば、UN/EDIFACT)目的のフォーマト内にコ つまた は複数のファイルとして出力することができます。データマノピング EDI 目的のかっかにデー タマノピングをデザインすることは、他のデータマノピング EDI 目的のかっかにデー タマノピングをデザインすることは、他のデータマノピング EDI 目的のかっかにデー タマノピングをデザインすることは、他のデータマノピング EDI 目的のかっかにデー タマノピングをデザインすることは、他のデータマノピング EDI 目的のかっかに クローマング EDI 目的のかっかい テー マノピングのデザイン たます。シスを参照してくたさい、EDI チャプタ ー。マノピングのデザイン中、出力タブをクトックして、MapForce 上で直接マノピングを 検証し プレビュー することができます。マノビングがサーィント環境内での実行に、適するよう に、BUILT-IN 変換言語をデザインしてストする必要があります。を参照してくたさい。
2.2. マナングをFlowForce Server にデ プロイする	FlowForce Server はW eb インターフェイス内で定義されているオンデマンド、また は、計画されたショブにより多種のショブを自動化します。MapForce Server が FlowForce Server の管理下で作動することを前提に、FlowForce Server は MapForce でデザインされたマングの実行も自動化することができます。 MapForce マンングが必要な出力を生成すると、FlowForce Server にデプロイ することにより実行を自動化することができます。
3.AS2 パートナーと証明書の構成	AS2 パトナーとの通信をセナアップするコよ (URI とAS2 名など) AS2 接続の 詳細と交換証明書を取得する必要があります。証明書 (とパトナーの詳細)は FlowForce Server につパートされる必要があります。 <u>AS2 証明書の構成</u> と <u>AS2 パトナーの構成を参照してくたさい。</u>
4. ジョブの作成	FlowForce ジョブはマピングを実行し、(a) 必要な出力を生成(b) AS2 メッセージを送信するかかご作成されます(ステップ7 を参照してくたさい)。これらの2つのアクションは、同じジョブの実行ステップ、まわよ 2つの異なるジョブの実行ステップです。 MapForce マピングを実行する FlowForce Server ジョブの例に関しては、次を参照してくたさい、MapForce マピングからジョブを作成する。
5.ジョブの実行	前のステップで作成されたFlowForce Server ジョブは必要に応じて多種の方法で 実行されるように構成することができます。例えば、Webサービスの呼び出しとして実行 することができます。おさよ、ファイルシステム上のファイルに変更が加えられると、特定の 日時に実行されるようにスケジュールすることができます。次を参照してくたさい、トリガー の管理。このステップは完全に自動化されています。
6.データマメングの実行	このステップはMapForce Server により自動的に実行することできます。(オンデマンド、おけよ、計画されているデータマンビングを実行するようごうゴが構成されている場合、MapForce Server への内部呼び出しか発生します。この結果、MapForce Server はマンビングを実行し、出力をFlowForce Server に返します。
7.出力をピックし、AS2 メッセージを送信す る	AS2 メッセージを送信するコよ、ジョブ(おコよ、ジョブ内の実行ステップ)は FlowForce Server ビルトイン関数 <u>/system/as2/send</u> を呼び出す必要がありま す。この関数は、ステップ3で定義されているパトナーオブンエクト、パトトナーURI、お よび送信するメッセージコンテンンを含む、AS2 メッセージを送信するオンタコンの要なパラ
ステップ #	説明
--	---
	メーターをとます。ジョブは多種のFlowForce Server AS2 式関数をマピングの 出力を必要とされる書式に変換するためコ呼び出す必要があります、例えば、ファイルか らストリームに変換する場合など)。
8. 同期 MDN を使用しての ^ペ トナーの応 答	AS2 パトナーオブジェクトをステップ3 で作成すると、オプションでパトナーに、 FlowForce Serverにお送信されるAS2 メッセージの応答してMessage Disposition Notification (MDN) をリクエストすることができます次を参照してくたさい い <u>AS2 概念</u> 。パトトナーは、をFlowForce Server から送信されるHTTP 呼び 出しと同じセッション内にMDN を送信する必要する必要があります、すなわち、「同 期」として構成されている必要があります)。

AS2 データの受信と処理

取引パトナーからのAS2 デーダを受信する場合、ワークフローを受信するデーダを追加して構成することができます。このンナオの場合、 AS2 デーダを受信し保管する付けてはなく、他のフォーマナに変換し、データベースに保存し、他のWeb サービスに送信することができま す。例えば EDI おけはXML 書式のファイルをAS2 取引パトナーから受信し、FlowForce Server ジョブに再帰して実行されるマッ ピングに入力として提供することができます。このセグションでは、サンプルのAS2 プロセスが示されています:



= 手動の構成または処理

AS2 データの受信と処理

上記のAS2 処理は以下のように作動します:

<i>ス</i> テップ #	説明
1, 2, 3	これらは前のテーブル内のステップと同じです。唯一の違いは、マッピングが入力として AS2 取引パトトナーから受信を期待するファイル例えば EDI おせはXML ファイ ルを期待することです。
4.(AS2 サービスとしてのジョブの作成	これは度きりのステップです。このステップではAS2サビスを公開するFlowForce Server ジョブを作成します。AS2サビスは構成されたHTTP(S)アドレスとポト で、AS2パトナーからのリクエストをトッスします。

ステップ #	説明
5.AS2 データの送信	このステップでは、取引パトナーはAS2 サービスをAS2 メッセージに送信します。コ ミュニケーションコ成功するコよ パトナーのAS2 名と証明書がFlowForce Server 内て既に定義されている必要があります。
6. 同期 MDN を使用しての回答	FlowForce Server は操作の出力(成功、お出よ エラー)を示す同期 MDN を使 用して AS2 パートナーコ応答します。
7. データの処理と保存	メッセージが受信されると、FlowForce Server ジョブは、受信されたデータを文字列、 ますは、ファイルに変換し、デルノケリに保存、ますは、引数として他のショブに、なしま す。正確な処理ロジックは、FlowForce Server ビルトインと式関数を使用して構成することができます。
8.データマンピングの実行	AS2 データを受信する Flow Force Server ジョブは最初のステップで作成されたデー タマメングジョブを任意で呼び出すことができます。マメングジョブは入力としてパトナ 一から受信した AS2 データを取り Map Force によりサポートされる方法で処理します : 例えば、他の書式への変換、他のWebサービスに送信するなど。

10.4 AS2 概念

AS2 メッセージを取引相手に送信する場合、取引相手からデータの暗号化図書名のためこ必要なデジタル証明書を含むAS2 接続の詳細を取得する必要があります。次の項目か構築される必要があります。

- パートナーはHTTP、おけは、HTTPSを介した接続を必要としますか?
- パートナーはAS2 メッセージを暗号化することを必要としますか?
- パートナーはAS2 メッセージを署名することを必要としますか?
- AS2 メッセージか受信されたことの「メッセージディスポジョン通知」からのMDN) 確認を必要としますか?

HTTP(S) 接続

HTTP 接続暗号化は実際のAS2 メッセージの暗号化と異ないます。AS2 メッセージは通常個別に異なるレイヤーで暗号化されている ため取引相手はプレーンHTTP を受け入れ、HTTPS 接続を必要とたい場合があります(次の段落を参照してくたさい)。取引相手が AS2 メッセージがプレーンHTTP の代わりにHTTPS を使用して送信されることを必要とする場合、取引相手のサーバーはプライアントから のSSL 暗号化されている接続を受け入れるように構成されている可能性があり、追加の構成は必要とされません。

AS2 暗号化

AS2 メンセージの「暗号化」は、転送前にデータを変更(暗号化)し、目的とするパーティ(すなオち取引相手) か暗号化を解除して読み 取ることを意味します。 AS2 メンセージ 暗号化の証明書は、取引相手への安全な接続を確立するために使用される証明書とお異なりま す(前の段落を参照してくたさい)。 AS2 メンセージ 暗号化を可能にするために、取引相手の公開された証明書か必要とされ、この証明 書を FlowForce Server 証明書ストアに追加する必要かあります、次を参照してくたさい、<u>AS2 証明書の構成</u>。

AS2 署名

「署名」はメッセージの受け取り相手(すなオち、取引相手)が送信元(すなオち所属機関)を確認し、メッセージが変更されていないことを 確認するすなのメッセージの処理方法を意味します。署名を可能にするコよ、署名の目的のすなの作成された所属機関の公開された証明 書(おけよ、公開キー)が必要になります。他方、所属機関の秘密の証明書(おけよ、秘密キー)FlowForce Server 証明書スト ア、次を参照してくたさい、<u>AS2 証明書の構成</u>。取引相手にAS2 メッセージと共に送信されるデジタル構成(ハッシュ)を作成するすっかに 秘密キーをFlowForce は使用します。取引相手にメッセージかある場合、デジタル構成と所属機関の公開された証明書はメッセージが実際に送信され、データが改ざんされていないかを検証することができます。

MDN

メッセージ ディスポジュン通知 (MDN) はAS2 転送の一部として送信されるメッセージです。これにより、所属機関(封ま)よ、所属機関の 取引相手) は実際のAS2 メッセージの受信の成功を知ることが出来ます。 おこ、送信済みのメッセージのトランスミッションは関連しているため、送信済みのメッセージをトラックすることができます。 MDN にはおい下の2つの型が存在します: (AS2 メッセージとして同じ HTTP セッション内に存在する) 同期 および、個別の HTTP セッションを介して送信された? 同期。 Flow Force Server はAS2 メッセージの受信の 承認を取引相手から、同期 MDN をリクエストします。 これはパトナーの詳細の構成ページでおこなうことができます、 次を参照してくたさい、 AS2 パトナーの構成。 同期 MDN のリクエストは、現在サポトされていません。 次を参照してくたさい、制限。

取引相手をデータの送信と転送の方法に合意すると、次のステップでは、関連する証明書とペートナーの詳細がFlowForce Server に 追加されます、次を参照してくたさい、AS2 証明書の構成とAS2 パートナーの構成)。

10.5 AS2 証明書の構成

AS2 メッセージ交換プロセス内に多種のレベルでのセキュー」ティを電子証明書は提供します。AS2 通信のエレテキストで、証明書は次の目的のために使用することができます:

- AS2 メッセージ 暗号化
- AS2 メッセージ署名
- AS2 署名の検証

AS2 シッセージの署名と暗号化と署名の検証に使用される証明書についてのみこのトピックで説明されます。FlowForce Server には、 FlowForce Server が作動するオペレーティングシステムの証明書ストアから独立した証明書ストアが存在します。FlowForce Server 内では、証明書はエンテナー内に保管されます(このためにFlowForce 内での他の一連のオブジェクトとして同じユーザーアクセスを活用す ることができます。次を参照してください、パーミッションのしくみ)。AS2 プロセスのために必要なすべての秘密、まけよ、公開された証明書は にFlowForce Server インポートされる必要があります(ターゲットコンテナーとアクセスすることのできるユーザーを決定することができます)。

AS2 メッセージ暗号化のための構成ステップは 以下のとおしです:

- 1. 暗号化おける署名の検証のために使用される公開証明書を取引相手から取得します。多くの場合同じ証明書がしようされます。
- 2. FlowForce Server 証明書ストアにインポートする方法は、以下に示されるとおりです。FlowForce 内でパートナーの詳細を 作成する際にこの証明書を参照する必要があります。(次を参照してください) AS2 パートナーの構成)。

AS2 メッセージ署名のための構成ステップは以下のとおりです:

- (FlowForce Server に対して外部のプログラムで)所属機関の公開されご証明書と秘密キーを作成します。署名のよめの所属機関の証明書が既にオペレーティングシステムの証明書ストア内に存在する場合、ファイルにごクスポートしまず(公開証明書と秘密キーの両方がファイルに含まれている必要があります)。Windows 上の作成方法は次を参照してください。 https://technet.microsoft.com/en-us/library/cc754329(v=ws.11).aspx。Linux の場合、証明書ファイルが証明書 ストアとしての役割を果たすディレイトリからごとつされる必要があります。例えば、Ubuntu 上で/etc/ssl/private、まこには、/etc/ssl/certs。macOSの場合、次を参照してくたさい、https://support.apple.com/kb/PH20122? locale=en US。
- 2. パートナーへの秘密キー無しで公開証明書。秘密キーは所属機関の外部と共有されてはおません。
- 3. (秘密キーを使用して)FlowForce Server 証明書ストアに証明書を以下のとおりインポートします。

署名済みのMDN パトナーが送信すると、(MDN 署名を検証するためコ パトナーの公開された証明書がFlowForce にんパト される必要が別はす。パトナーオブジェクトを作成する際にこの証明書を参照する必要が別はす。次を参照してくたさい AS2 パトナ 一の構成。

FlowForce Server に証明書をインポートする方法:

- 1. FlowForce Server Web 管理インターフェイス コングオンます。
- 2. 「構成」をクリックして、証明書を作成するコンテナートご移動します。
- メモ デフォルトでは、「ハブトック」コンテナーは全ての認証済みのFlowForce Server ユーザーによりアクセスすることができるような 機密な情報を保管するために最も適した場所ではありません。「ハブリック」コンテナーへのアクセスを制限する、おけよ、パーミッショ ンを有するユーザーのみがアクセスすることのできる別のコンテナーに機密なオブジェクトを保管するように定義することが奨励されま す。次を参照してください、ハーミッシュンティー。
 - 3. 「作成」をクリックして、「証明書を作成」をクリックします。

証明書の作成 /	
証明書の名前: 証明書の詳細:	ExampleCertificate
証明書イン	ィポート
ファイルからイ パスワード:	ンポートする: 参照
保存	

- 4. 名前を入力し、オプションで、証明書の名前を入力します。証明書を後に簡単に識別するために分かりやすい名前を与えてくたさい。詳細を後に変更することは可能です。
- 5. 「参照」をクリックして、証明書ファイルを選択します。

インポートされるファイルはPEM、DER、おけよ PKCS#12 書式である必要があります、ファイル拡張子と区別してくたさい。ファイル拡張子以下のノストの1つであることができます: .pem、.der、.cer、.crt、pfx、p12。FlowForce はファイルを次のように処理します:

- 拡張子が.pem、.cer、.crtの場合ファイルはPEM書式とて処理されます。ファイルは"----BEGIN "、おこは、"---- BEGIN "により開始するラインを含みます。
- 拡張子が.der、.cer、.crtの場合ファイルはDER書式とて処理されます。ファイルは上記のラインを含みません。
- 拡張子が.p12、おけよ.pfx の場合、PKCS#12 とて処理されます。

(証明書ではなく秘密キーのみを含むファイルをインポートすることはできません。

6. 証明書ファイルが、アワードを必要とする秘密キーを含む場合、対応するフィールドレ、アワードを入力してくたさい。証明書ファイ

いか保護されていない秘密キーを含む場合、「削除」 💷 をクリックしてこのフィールドを削除してくたさい。

7. 「保存」をクリックします。

証明書のたポートに成功すると、ページ内に詳細が表示されます。例

詛	b明書	
	発行先	
	共通名 (CN)	John Doe
	組織 (O)	Example Organization
	組織単位 (OU)	Example Department
	地域 <mark>(L)</mark>	Vienna
	国名 <mark>(</mark> C)	オーストリア (AT)
	emailAddress	example@example.org
	シリアル番号	b1:ab:13:da:ce:83:98:03
	発行元	
	共通名 (CN)	John Doe
	組織 (O)	Example Organization
	組織単位 (OU)	Example Department
	地域 <mark>(</mark> L)	Vienna
	国名 <mark>(</mark> C)	オーストリア (AT)
	emailAddress	example@example.org
	有効な日数	
	開始期限	2017-09-13 12:55:06
	有効期限	2018-09-13 12:55:06
	セキュリティ	
	指紋	f4:2c:7e:3a:cf:77:e5:b3:b0:ac:77:29:5a:32:9d:ae:ac:42:77:4c
	指紋アルゴリズム	SHA1
	署名アルゴリズム	sha1WithRSAEncryption
	Public key algorithm	rsaEncryption
	Public key size	4096 bits
	秘密キーを含んでいる	\checkmark
	自己署名	\checkmark
	亦再 31184	
	文 文]]] 萨	

(C) 2015-2021 Altova GmbH

証明書は一定期間内で有効期限が切れるため、FlowForce Server W eb 管理インターフェイスから、定期的に置き換えられる必要が あります。これは、所属機関により発行された証明書と取引相手より受け取った証明書の両方に適用されます。(公開証明書の期限が切 れると、取引相手に通知されることが想定されています。同様に、公開証明書の期限が切れると、取引相手に通知され、新し、証明書が 送信される必要があります)。証明書の有効期限と他の関連情報は(FlowForce Server に証明書がインポートされた後)W eb 管理 インターフェイスで確認することができます。

FlowForce Server 内での証明書の置換えまこの証明書を使用するパートナー全てし影響を与えます。AS2 操作の整合性のために、事前に取引相手と所属機関の証明書の変更について調整してくたさい。

証明書の置換えの方法:

- 1. FlowForce Server ログイン後、「構成」をクリックして、証明書を保管するコンテナーに移動します。
- 2. 証明書のエトリをクルクします。証明書詳細ページがロードされます。
- 3. 「証明書のインポート」をクリックします。
- 4. 「参照」をクトックして、新規の証明書を選択します。
- 5. 「保存」をクリックします。これには店い証明書が新しい証明書と置き換えられます。

FlowForce Server に以前にインポートされた証明書は、他のFlowForce Server オブンエクトと同様に削除することができます(特定のレコードの横のチェックボックスを選択し、「削除」をクリックします)。証明書の複製、おけよ、エクスポートは不可能です。

署名と暗号化のための証明書を交換する2つの取引パートナーを含むAS2転送のサンプルに関しては、サンプルフルAS2メッセージ交換(シンプル)を参照してくたさい。

10.6 AS2 パートナーの構成

「パートナー」とう用語は AS2 通信内の一部であるパーテルを指します。所属機関がAS2 取引相手と通信するためにFlowForce Server 内で詳細が最初に定義されている必要があります。AS2 パートナーの詳細を定義すると、ジョブ内で再利用することができます。具体的には、AS2 メッセージを送信するショブを作成する際に(FlowForce ジョブのためについトナーの詳細を入力する代わりにつけて定義されている取引相手のリストからパートナーを選択することができます。

メモ 暗号化と署名の有効化が必要な場合、必要とされる所属機関のパートナーの証明書をFlowForce Server にんポートしてくたさい次を参照してくたさい。AS2 証明書の構成。

AS2 パートナーの構成方法:

- 1. FlowForce Server <u>Web 管理イクターフェイス</u> コングオノます。
- 2. 「構成」をクリックして、パートナーオブジェクトを作成するコンテナーに移動します。
- メモ デフォルトでは、「ハブトック」コンテナーは全ての認証済みのFlowForce Server ユーザーによりアクセスすることができるナムな 機密な情報を保管するナーヴに最も適した場所ではありません。「ハブリック」コンテナーへのアクセスを制限する、おけよ、パーミッショ ンを有するユーザーのみがアクセスすることのできる別のコンテナーに機密なオブジェクトを保管するように定義することが奨励されま す。次を参照してくたさい、ハーミッシュンティー。
 - 3. 「作成」をクリックして、「AS2 パートナーの作成」をクリックします。

パトナー構成ページ内の設定は、FlowForce Web 管理インターフェイスの他の部分と同じ振る舞いを持つグループごとご整理されています。例えば、グループが任意の場合は ・ をクトックして必要とされるオプションを設定します。グループを再度任意に設定するコス ・ オタンをクトックして、この設定のグループを非表示して、関連性をなくします。

パトナー構成ページは次の設定のグループにお構成されています

+	説明
パ トナ- 名	必須のフィールド。FlowForce Server への取引パトトナーを識別する名前です。この名前はFlowForce グラフィカルなユーザーインターフェイスで表示され、この取引パートナーを識別するための手助いをします。
パートナーの説明	任意のフィールド。パートナー機関の詳細に関するテキスト(例えば、郵便番号、連絡先など)。

パートナーの設定

+	説明
AS2 名	必須のフィールド。FlowForce Server がAS2 データを送信すると、この値はデー タ交換(「AS2-To」 ヘッダーの値)。の受信者を識別します。FlowForce Server が AS2 データを受信する際、この値はデータ交換(「AS2-From」 ヘッダーの値)の送 信者を識別します。

298 AS2 との統合

+	説明
	通常この名前はAS2取引パートナー間で合意されている必要があり、システム上で 一意の名前である必要があります。RFC 4130, 96.2を参照してくたさい。

ローカル側の設定

フィールド	説明
AS2 名	必須のフィールド。FlowForce Server がAS2 データを送信する際、この値はデー タ交換(「AS2-From」 ヘッダーの値の送信者を説明します。FlowForce Server がAS2 データを送信すると、この値はデータ交換(「AS2-To」 ヘッダーの 値)の受信者を説別します。
	通常この名前はAS2取引パートナー間で合意されている必要があり、システム上で 一意の名前である必要があます。RFC 4130, 95.2を参照してくたさい。

AS2 サービス設定

フィールド	説明
フィールド	任意のフィールドこのチェックボックスを選択するとFlowForce Server はAS2 パートナーからのメッセーンを受信することができます。
	AS2 データを受信しないAS2 データを送信するのみのAS2 パートナーを作成する 場合、このチェックボックスを解除します。
	この機能により同じ「ローカルAS2名」と「AS2名」組み合わせを持つパトナーを一人以上作成するエラーを回避することができます。これが発生する場合、AS2メッセージをこのボックスが選択されているパトナーからのみ受信することができます。

HTTP エボポイトの設定

+	説明
URL መሥታエスト	必須のフィード。このフィードはAS2 メッセージが送信されるパトナーURLを指 定する必要が決ます。例 http://example.org:8080/as2/HttpReceiver。
	値は「http://」ませま「https://」から開始する必要があります。
י/ýብ/ንት ቲ ተ	任意のフィールド。セキューティ上の理由から、HTTP リクエストの/ダイレクトを禁止、 ますま、同じホスト上での/ダイレクトのみを許可する必要がある場合があります。有効 な値 ・ リダイレクトは許可されていません[デフォルト] ・ 同じホストでリダイレクト ・ 任意の/ダイレクト(リダイレクトを許可する場合、異なるホスト間でもこの 値を設定してくたさい)。
チャンクされた転送エンコードを使用する	任意のフィールド。有効な値

+	説明
	 はい、FlowForceは送信のかりニチャンクされた転送エンコードを強制ではない、使用することができるようこかはます。このオプションが有効化されると、受信先のシステムもチャンクされた転送エンコードをサポートすることが期待されます。 しいえ[デフォルト]: FlowForceは、エンテンツの終わとを示すためにContent-Lengthと接続の終了のみを使用することができます。
HTTP 認証資格情報	任意のフィールド。HTTP 認証をパトナーのURI が必要とする場合のみ適用することができます。パトナーのサーバー認証するために必要とするHTTP 資格情報をこ コニ入力してくたさい。資格情報のレコードから専用のページからHTTP 資格情報を 定義し、このページから参照することもできます。資格情報を参照してくたさい。 メモ FlowForce Server は事前に資格情報を送信します。
<u> </u>	任意のフィールド。応答か受信されない場合、サーバーかタイムアナトする値を秒数で 指定します。デフォルトパンステム固有です。

圧縮の設定

+	説明
圧縮の使用	任意のフィールド。FlowForce Server がAS2 データをパートナーに送信する前に 圧縮する場合、このチェックボックスを選択します。

セキュリティ設定|暗号化

所属機関がこのパトナーには送信されるAS2メッセージを暗号化する必要がある場合この設定のグループを設定する必要があります。

+	説明	
アルゴリズム	任意のフィールド。暗号化に使用される対称アルゴリズムを指定します。有効な値: DES 3DES [デフォルト] AES-128 AES-192 AES-256 RC2-40 RC2-64 RC2-128 RC4-128	
パトナーの証明書	必須のフィールド。AS2 メッセージ の暗号化のナックコ使用される証明書を指定しま す。AS2 メッセージ の暗号化のナックコ使用される証明書を指定します。これは、取 引パトナーから受信し、FlowForce Server にインポートされる証明書である必 要があります。次を参照してくたさい、AS2 証明書の構成。	

セキュリティの設定|暗号化の解除

所属機関がこのパートナーにより受信されるAS2メッセージの暗号化を解除する必要がある場合、この設定のグループを設定する必要があります。

+	説明		
אגויבות	任意のフィールド。所属機関に送信されるメッセージの暗号化解除のためコペトナーの使用が許可されているアルゴリズムを指定します。		
	他のアルゴリズムまけは選択されていないアルゴリズムをパートナーが使用する場合、 FlowForce Server はエラー MDN を送信しジョブは開始されません。エラー MDN はこの場合以下のようなテキストを含んでいます: "automatic- action/MDN-sent-automatically; 失敗 / エラー: insufficient- message-security"		
	このフィールドの有効な値		
	 DES 3DES AES-128 AES-192 AES-256 RC2-40 RC2-64 RC2-128 RC4-40 RC4-128 		
ローカル側の証明書	必須のフィールド。AS2 メッセージの暗号化の解除のためご使用される証明書を指定します。以前にFlowForce Server にインポート済みの秘密キーを持つ証明書 への参照である必要があります。AS2 証明書の構成を参照してくたさい。所属機関 がパートナーにより送信される MDN の署名を検証する必要かある場合、この設定の グループを設定する必要かあります。		
	Name Type 🗢		
	ApolloPublic certificate		
	HermesPrivate certificate + private key		

セキュリティ設定|署名の作成

所属機関がこのパートナーコ送信されるAS2 メッセージを署名する必要がある場合、この設定のグループを設定する必要があります。

+-	説明
דוובווד	必須のフィールド。署名 MIC (message integrity check)の計算のために使用 される ッシュアルゴリズムを指定します。有効な値

+	説明		
	 MD5 SHA-1 [デフォルト] SHA-224 SHA-256 SHA-384 SHA-512 		
ローカル側の証明書	必須のフィールド。このドトナーに送信されるAS2 メッセージとMDN を署名するために所属機関により発行される証明書を指定します。以前に Flow Force Server にインポート済みの秘密キーを持つ証明書への参照である必要かあります。AS2 証 明書の構成を参照してくたさい。所属機関がパトナーにより送信される MDN の署 名を検証する必要かある場合、この設定のグループを設定する必要かあります。		
	□ Name Type 🗢		
	ApolloPublic	certificate	
	🔲 📴 HermesPrivate	certificate + private key	

セキューティ設定|署名の検証

所属機関がドトナーによど、信されるMDNの署名を検証する必要がある場合、この設定のグループを設定する必要があります。

フィールド	説明	
דעויבווד	必須のフィールド。署名内の署名済みシッセージャッシュの計算のオータは受け入れられるハッシュアルロリズムを指定します。トレードパトナーが以下のアルロリズムを使用しない場合、FlowForce Server は以下のようなエラーテキストを含むMDN を返します: "automatic action/MDN-sent-automatically ; 失敗 / エラー: insufficient-message-security"この場合メッセージは受け入れられず、処理されません。 有効な値:	
	 MD5 [デンオルト] SHA-1 [デンオルト] SHA-224 [デンオルト] SHA-256 [デンオルト] SHA-384 [デンオルト] SHA-512 [デンオルト] 	
パートナーの証明書	条件付きフィールド。条件付きフィールド。パトトナーから送信されるメッセージとMDN の署名を検証するためはしようされる証明書を指定します。これは、取引先パトトナー から受信される公開証明書である必要かあり、FlowForce Server 次を参照してく たさいこインポートされる必要かあります。次を参照してくたさい、AS2 証明書の構 成。	
	「サイン済みのMDN リクエスト」チェックボックスが有効化されている場合、このフィールドも設定してくたさい。	

メッセージ処理通知

302

AS2 との統合

フィールド	説明	
MDN אדדלעי	オプション「同期」は、FlowForce がパトナーにAS2 メッセージの応答として、 同期 MDN の送信をリクエストすることを意味します。MDN をリクエストしなは場合、 「削除」 のトックして、オプションのこの部分を削除します。	
	メモ 非同期 MDN は現在サポートされていません。次を参照してくたさい、制限。	
サイン済みのMDNリクエスト	任意のフィールド。このチェックボックスを選択し、取引パトナーから署名済みのMDN をパクエストします。次を参照してくたさい、メッセージ処理通知。	

相互運用性の設定

+	説明
データの王縮	条件付きフィールド。「圧縮の使用」オプションが有効化されている場合、このオプ ションはAS2 パトナーへのトランスミッションのためにデータの署名が圧縮の前後に 実行されるかを指定します。
	送信するメッセージの場合、選択されているオプションはAS2パートナーがサポートするものである必要があります。
	受信するメッセージの場合、(すなわち、FlowForce Server が他のドトナーから メッセージを受信する場合)、このオブションは関連性がありません。FlowForce Server は、署名の前後にメッセージが汪縮されたかに関わらず、メッセージの汪縮を 解除します。
MIC 検証アルゴリズム	条件付きフィール・このフィールドは「MDN リクエスト」オプノョンか設定されている 場合適用することができます(上記参照)。AS2 データの署名のナムウニ使用されるメッ セージ性合成チェックを検証、おけよ、計算する際にFlowForce Server が使用す るアルゴリズムを指定します(次も参照してくたさい)
	AS2 パートナーがMicrosoft BizTalk で作動する場合、相互運用性の理由から「MDN 署名のアルゴリズムを使用する」を選択します。AS2 パートナーが mendelson AS2 おけまFlowForce Server 上で作動する場合、「元のメッ セージ署名のアルゴリズムを使用する」を選択します。
	通信する両方のAS2 サーバーがrun FlowForce Server を作動している場合、 このオプションは双方で同じである必要があります。
	AS2 メッセージませまMDN (例えばSHA-256)内の署名 MIC のためこSHA- 1 以外のアルゴリズムが使用されるとこのフィールドの値は異なる結果をもたっします。
<u> </u>	このチェックボックスが選択されているとFlowForce Server will はMIME メッセージを MIME ヘッダーとメッセージボデを含む可能性のある正規メッセージフォームのた めの MIME ルールゴ従い 再フォーマットします。
	このオプションの下のテキストボックスを使用してメッセージボディが正規フォームニ再フォ ーマナトされる必要のあるコンマで区切られた追加コンテンン型のリストを指定します。 HTTP Accept ヘッダーに類似し <u>is-mime-content-type</u> 式関数の

accept / ケメーターに一致するフィハドカードを受け入れられて型のJストはサポート します。 メッセージボディは以下の条件で正規フォームに再フォーマナされます: 1.MIME ヘッダー Content-Transfer-Encoding が「大文字と小文字を区別) "base64"の値若寺つ場合。 2.MIME ヘッダー Content-Transfer-Encoding が「7bit"、"8bit"、 " guoted-printable"(大文字と小文字を区別)であり Content-Type が (text/plain を含み、text/かご開始する text/* の場合。 3.MIME ヘッダー Content-Transfer-Encoding が「7bit"、"8bit"、 " guoted-printable"(大文字と小文字を区別)であり Content-Type が (text/plain を含み、text/かご開始する text/* の場合。 3.MIME ヘッダー Content-Transfer-Encoding が「7bit"、"8bit"、 " guoted-printable"(大文字と小文字を区別)であり Content-Type が以前 に説明済されているテオストボックが内で定義されているたのの場合。 4.マリティトトーのシルージの場合、プローグとユビローグが再フォーマトされ、ヘッダー に違い同じ過程がすべての、レイン適用されます。 Content-Transfer-Encoding が「パイナリ」の箇所のシッセージのケッセージ ポディル正規フォームご再フォーマトされません。 メッセージへッダーの正規フォームは取っしたうす。すなすた、ヘッダーが存在しない 場合、「パイナリ」は予測されたず、10年で、コオー、ヘッダーが存在しない 場合、「パイナリ」は予測されたではに見フォームで再フォーマトされません。 メッセージへッダーの正規フォームは取了のようてがはます。 1.OR LF 行来文字によいッダーは終了しています。 2.ヘッダーは折げ力されています(値を持つへ)ダー全体は一行で表示されています 3.ヘッダーナでの前はコロンナの後の空白文字によいな切られています。	+	説明
メッセージボディは以下の条件で正規フォームに再フォーマトされます: 1.MIME ヘッダーContent-Transfer-Encoding が、大文字と小文字を区別) であse64 の値を持つ場合。 2.MIME ヘッダーContent-Transfer-Encoding が、7bit、、8bit、、 「quoted-printable"(大文字と小文字を区別) であり Content-Type が (text/plain を含み、text/から開始する text/* の場合。 3.MIME ヘッダーContent-Transfer-Encoding が、7bit、、8bit、 「quoted-printable"(大文字と小文字を区別) であり Content-Type が してまた/から開始する text/* の場合。 4.マルチャトーのシャレージの場合、プロークとエビローグが再フォーマトされ、ヘッダー して、「マトンは見フォーム」、「シーン・クリークが用フォーマトされ、ヘッダー して、「シーン・クリーク」の 第一次ゲーク には、同じ過程がすべての いっぱいです。すなけた、ヘッダーが存在しない 場合、「ハイナリ」は予測されたでは、 クッケージへ・ダーの に規フォームは以下のようてなります: 1.CR LF 行来文字によりへ、ダーは終了して、ます。 2.ヘッダーは新げがった。 2.ヘッダーク にはたけっしょうにより、アッケーション・パートーー 3.ヘッダーク での に見フォームは以下のようてなります。 3.ヘッダーク が「クローレーション・クローン・クローン・クローン・クローン・クローン・クローン・ 3.ヘッダーク によりに、 なっか		accept パラメーターに一致するワイルドカードを受け入れられた型のノストはサポート します。
 1. MIME ペッダーContent-Transfer-Encoding が大文字と小文字を区別) であまe64″の値を持つ場合。 2. MIME ペッダーContent-Transfer-Encoding が "7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が (text/plain を含み、text/から開始する text/* の場合。 3. MIME ペッダーContent-Transfer-Encoding が "7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が (text/plain を含み、text/から開始する text/* の場合。 3. MIME ペッダーContent-Transfer-Encoding が "7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が (text/plain を含み、text/から開始する text/* の場合。 4. マリチットーンの場合、プローグとロビローグが取フォーマトされ、 ペダー に従い同じ過程がすべての キンパ道用されます。 Content-Transfer-Encoding が 「パイナリ」の箇所のかっセージのかっセージ ポディオル正規フォームンは適用されます。 Content-Transfer-Encoding が 「パイナリ」の箇所のかっセージのかっセージ ポディオル正規フォームンはおしゃ A S2 のよめのデオルトの Content-Transfer-Encoding ゴンイサリ」です。すなオを、 ペッダーが存在しない場合、「パイナリ」は予測さればディオエルフォーマトされません。 メッセージ ペダーの正規フォームは以下のようプがはます: 1. CR LF 行素文字により ペダーは終了してます。 アッダーは折げ オーズコンパンダーは終了してます。 アッダーナイの値にコレンゲの後の空白文字により なりのたれています。 		メッセージボディは以下の条件で正規フォームニ再フォーマナされます:
Content-Transfer-Encoding が「バイナリ」の箇所のかッセージのかッセージ ボディは正規フォームニ再フォーマトされません。AS2 のけるのデフォルトの Content-Transfer-Encoding はバイナリ」です。すなオち、ヘッダーが存在しない 場合、「バイナリ」は予測されボディは正規フォームニ再フォーマトされません。 メッセージヘッダーの正規フォームは以下のようになります: 1.CR LF 行末文字によりヘッダーは終了しています。 2.ヘッダーは折りたまれています(値を持つヘッダー全体は一行で表示されていま す)。 3.ヘッダーとその街にコロンとその後の空白文字により区切られています・		1.MIME ヘッダー Content-Transfer-Encoding が大文字と小文字を区別) "base64" の値を持つ場合。 2.MIME ヘッダー Content-Transfer-Encoding が"7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が (text/plainを含み、text/から開始する) text/* の場合。 3.MIME ヘッダー Content-Transfer-Encoding が"7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が いずつ Content-Transfer-Encoding が"7bit"、 "8bit"、 "quoted-printable" (大文字と小文字を区別) であり Content-Type が い前 に説明済されているテキストボックス内で定義されているものの場合。 4.マルチット・のかっセージの場合、プロローグとエーローグが再フォーマナされ、 ヘッダー に従い同じ過程がすべての いうに適用されます。
メッセージャッダーの正規フォームは以下のようこなります: 1.CR LF 行末文字により、ッダーは終了しています。 2. ヘッダーは折りナーまれています(値を持つ、ッダー全体は一行で表示されています)。 3. ヘッダーとその値はコロンとその後の空白文字により区切られています・		Content-Transfer-Encoding が「バイナリの箇所のシッセージのシッセージ ボデイは正規フォームニ再フォーマトされません。AS2のオタのデフォルトの Content-Transfer-Encoding はバイナリです。すなオム、ヘッダーが存在しない 場合、「バイナリ」は予測されボディは正規フォームニ再フォーマトされません。
1.CR LF 行末文字により、ダーは終了しています。 2. ヘッダーは折けままれています(値を持つへッダー全体は一行で表示されています)。 3. ヘッダーとその値はコロンとその後の空白文字により区切られています。		メッセージへッダーの正規フォームは以下のようてなります
		1.CR LF 行末文字により、ダーは終了しています。 2. ヘダーは折けままれています(値を持つ、ダー全体は一行で表示されています)。 3. ヘッダーとその値はコロンとその後の空白文字により区切られています:

10.7 AS2 メッセージの送信

AS2 メッセージを遠隔の、トトナーに送信するジョブも他のFlowForce Server ジョブ。と類似しています。具体的には、多種の実行ステップを含む、ウメータを取ることができ、計画された、おけよ、オンデマンドのショブとして、リガーすることができます。このサンプルは、AS2 サーバーにEDIFACT ファイルを送信するシンプルな AS2 ジョブ サンプルの作成方法について説明しています。

必要条件

- AS2 サーバーを使用することができ HTTP クライアントから AS2 メッセージを受け入れるように構成されている必要があります (この場合、FlowForce Server)。
- 遠隔のペトナーの詳細がFlowForce Server に追加される必要が決ます。次を参照してくださ、AS2 パトナーの構成. 最小限の基本的な接続テストのために、証明書無しのペトナーを定義する必要が決ます(暗号化されて れ 署名されていない 接続を受け入れることが前提とされます)。この場合、このペトナーと通信するはよ パトナーのURL、パトナーAS2名、と所属機関のAS2名のみか必要となます。

ジョブの作成

新規のFlowForce Server ジョブを通常の方法で作成します(コンテナー内の「作成 | ジョブの作成」 をクックします。 次を参照してく たさい、 ジョブの作成)。 次に、 ^①/system/as2/send 関数を呼び出す実行ステップを追加します。 関数を素早く検索する コよ 「関数の実行」 ボックス内をクックし、 関数名を入力します、 例えば

実行ステップ	
•	
▲ 関数の実行 as	2
= ステップの結:	
新規の実行ステップ	
	名前 🗢
実行の結果	by system/as2
	() /system/as2/send
戻り値の型を以下と	

ジョブニ関数を追加すると、ページとフィールドに構造がロードされ、全ての必要とされるパラメーターが使用できるようゴネルます。ページとフィー ルドに構造がロードされ、必要とされるパラメーターが使用できるようゴネルます。AS2トランスミッションが正確に設定されていることを確認する ためコンラメーターを次のようこ設定します:

- パートナーーこのフィールドは以前に構成されたパートナーオブジェクトを参照する必要が設ます。次を参照してくたさい、AS2 パ <u>ートナーの構成</u>。フィールドをクリックしてパートナーオブジェクトを参照してくたさい。
- メッセージ このフィールドはシッセージ内に含むストリームを開く FlowForce 式 を含んている必要があります。例えば C: \as2\orders.edi で検索されたEDIFACT ファイルを送信するゴは Content-Type へダーを持つ application/EDIFACT、次の式を入力してくたさい。

```
stream-open("c:\as2\orders.edi", "application/EDIFACT")
```

FlowForce 内の式詳細に関しては、<u>FlowForce 式</u>を参照してくたさい。ソースファイル(EDI、おけま XML) も(例えば 以前にFlowForce Server にデプロイされたマナピングを実行する前の実行ステップなど) MapForce を使用して生成される ファイルコなることができます。次を参照してくたさい、<u>MapForce とMapForce Server を使用したAS2 との統合</u>。

エラー時には中断する - ジョブはAS2 メッセーンを送信するステップパナではよく、多種の実行ステップから構成されることができます。例えば、パートナーにより返されたMDNを処理するために、現在のステップの後に他の実行ステップを定義するとます。この、ウメーターをTRUE(有効化済み)に設定し、現在の実行ステップに失敗するとジョブの実行の続行を破棄することができます。
 エラー時には中断する、ウメーターがTRUE(有効化済み)で、現在の実行ステップに失敗した場合、次に続く実行ステップは実行されず、ジョブ全体が中断されます。次を参照してくたさい、順番にステップを処理する。

下のイメージは、「サンプルパトナー」を参照し、FlowForce Server 式を使用してメッセージボディにEDIFACT ファイルを提供する実行ステップのサンプルを現しています。

実行	ステップ					
	関数の実行	/system/as2/send	▼ 2			
	パラメーター:	パートナー: メッセージ: エラー時には中断:	/public/Example Partner stream-open('C:\as2\ORDER.EDI', 'application\EDIFACT') +	•	 AS2 パートナー (必須) セットする値 条件式ストリーム (必須) セットする値 	
= 新規	ステップの結界 の実行ステップ	長を割り当て 名前 新規の選択ステップ	以下としてAS2 MDN 新規の For-each ステップ 新規エラー/成功処理ステップ			

上記のようこ、FlowForce ジョブはオンデマイで作動するようご構成することできます。おけよ 計画されたショブとして実行されるように 構成することもできます。構成することのできるジョブのトリガーは関する詳細は、次を参照してくたさい、トリガーの管理。このサンプルでは Webサービスとしてプラウザーからジョブをオンデマイで実行するようこAS2ジョブを以下に示されるように構成します。Webサービスの名 前を確認してくたさい、このサンプル内では、「sendAS2」ですが、必要に応じて異なる名前を与えることもでき案巣。詳細に関しては、次 を参照してくたさい、Webサービスとしてジョブを公開する。

サービス 🗹 このジョブを HTTP を介して URL 上で利用可能にする http://*<FlowForce server>/s*ervice/_sendAS2

最後に、ジョブを保存する前に、FlowForce Server が作動しなければならな、オペレーティングシステムアカウトに証明書を入力します (これらはFlowForce Server コングオンするためコ使用される証明書と同じ証明書ではないてとご注意してくたさい)。このサンプルでは、 証明書にはショブ内に直接挿入されますが、証明書レコードとして個別に保管資、便利にジョブ内で選択、おけよ参照することもできます。 次を参照してくたさい、資格情報。

シー	發格情報		
	資格情報によりジョブを実行:	◎ 既存の資格情報を選択:	
		◎ ローカルの資格情報を定義: ユーザー名	altova
		パスワード	パスワードの変更

ジョブをページの下の「保存」ボタンをクリックして保存します。

FlowForce Server はデータの整合性をチェックし、構成されていないプロレティであるショブを保存することを回避することができます。 多くの場合、エラーはレテメーターの値として正確ではない式が適用されることにより引き起こされます。次を参照してくたさい、 ステップで処理する。FlowForce Server の初心者の場合、FlowForce 式とジョブ構成のサンプルセクションを参照してくたさい。

メモ 類似している複数のジョブを作成する場合、FlowForce ジョブを簡単に複製することができ時間を削減することができます。次を参照してくけざい、ジョブの複製。

ジョブの実行

W eb サービスとしてジョブが公開されるため、ブラウザーのアドレスバー内にW eb サービスURL を入力して実行することができます。 FlowForce Server サービスが実行されるURL でW eb サービスURL は構成されます(例えば、

http://localhost:4646/)、および、Webサービスの名前がsendAS2であるservice/sendAS2パトー。最終URLは 以下になけます。http://localhost:4646/service/sendAS2。FlowForce Serverサービスが異なるホストとポートで実行 されるようご構成されている場合、このURLが調整されていることを確認してくたさい。<u>ネットワーク設定の定義</u>を参照してくたさい。下のイメ ージはブラウザー内に表示される実行の成功例の結果を表示しています。

AS2 Reponse × +			-		×						
← ⓒ localhost:4646/service/sendAS2 ☑	☆	•	Â	◙	≡						
AS2 Response											
Message-ID: <5f7e5ebc94434c32a774b43dab44d398@DOC-W10x64>											
HTTP-Status: 200											
Disposition: automatic-action/MDN-sent-automatically; processed											
Signed: no											
Message:											
====================================											
The AS2 message has been processed. Thank you for exchanging	AS2 me	ssages	s with	n Pyas	2.						
====================================											
Reporting-UA: Bots Opensource EDI Translator Original-Recipient: rfc822; pyas2plain Final-Recipient: rfc822; pyas2plain Original-Message-ID: <5f7e5ebc94434c32a774b43dab44d398@DOC-W3 Disposition: automatic-action/MDN-sent-automatically; process	10x64> sed										

ジョブの実行の結果はFlowForce Server ログを介して確認することができます。ジョブのログを確認するを参照してくたさい。

AS2 ジョブの結果の処理

①/system/as2/send 関数を呼び出す1つの実行ステップには構成される簡単なジョブの作成方法が説明されていますが、実社会の シナリオでは、FlowForce Server ジョブは複数のステップには構成されます。

□ /system/as2/send 関数の戻り型は、AS2 MDN オブシェクトです。このオブシェクトから有益な情報を抽出するけっめこ、 FlowForce 式関数を使用して更に処理される必要があります。 例えば、元のAS2 メッセージのメッセージ ID を取得するコよ、実行ステップを以下のようご追加することができます。

実行	ステップ			
	関数の実行	/system/as2/send		▼ 2
	バラメーター:	バートナー:	/public/Example Partner	
		メッセージ:	stream-open('C:\as2\ORDER.EDI', 'application\EDIFACT')	
		エラー時には中断:	•	
=	ステップの結界	果を割り当て output1	以下としてAS2 MDN	
•	関数の実行	as2-message-id(outp	ut1)	•
=	ステップの 結界	果を割り当て 名前		
新規	の実行ステップ	新規の選択ステップ	新規の For-each ステップ 新規エラー/成功処理ステップ	

上のショブでは、(output1 とて宣言されている) 最初のステップの結果をパラメーターとして取ることによ、2番目のステップは元の AS2 メッセージID を文字列として取得します。これを達成するには、FlowForce で式を計算するための通常の方法である /system/compute 関数を呼び出します。 式は、<u>as2-message-id</u> 式関数を最初の実行ステップ(output1)の結果に適用し ます。

FlowForce Server はよ 多種の環境で役に立つ他の式関数が含まれています。例えば、AS2 呼び出しが成功であるかを決定するに は、<u>as2-success</u> 関数を呼び出すことができます。AS2 呼び出しのHTTP ステータスを取得するはよ、<u>as2-http-status</u>式関 数を呼び出すことができます。使用することのできる式関数は式関数 チャプター内にリストされています。AS2 とMIME に適用することので きる式関数は、それぞれ <u>AS2 式関数</u> と<u>MIME 式関数</u>にリストされています。

FlowForce 式と作業する際の重要なルールは、各関数の戻りデータ型に注意することです。データ型は呼び出し関数とステップと互換性を 有する必要があります。それ以外の場合、ジョブは検証エラーのナック保存することができません。使用するはは、FlowForce 式についての 基本的理解が必要になります。<u>FlowForce 式</u>を参照してくたさい。

10.8 AS2 メッセージの受信

FlowForce Server を使用して、期間のペトナーからAS2 サービスを受け取り、このデータを処理し、ローカルな場所に保存することが できます。一般的には、このようなジョブはFlowForce ジョブと同じような性質を共有し、更に次の追加機能を提供します:

- ジョブ構成ページから直接、リクエストをリハスンするAS2 サービスを作成することができます。
- 下記のとおり、2つの定義済みの、ラメーターを受信するAS2 データは、パートナーとメッセージを取ります。これらの、ラメーター ーは送信元のパートナーと受信されるメッセージに関する情報を提供します。

ジョ港AS2 サービスとして公開することは、ジョブをWeb サービスとして公開する方法とお話してす。次も参照してくたさい、ジョブを Web サービスとして公開。具体的には、AS2 サービスURL は以下のフォーマナのようておます http(s)://<flowforceserver>:<port>/service/<as2-service-name>:

- <http(s)> はHTTP おけまHTTPS を選択することのできるプロトコールを参照し、プロトコールにわらはFlowForce Server セルアップページで構成することができます、次を参照してくたさい、<u>ネルワーク設定の定義</u>)
- <flowforce-server> はFlowForce Server が作動するマンのホスト名、およ、IP アドレスです。
- <port> はポート名です(デフォルトでは、4646)。HTTPとHTTPS はセナア・プページ「FlowForce Server」セグタンで構成されているとおり異なるポート番号を持津ことに注意してください。次を参照してください、<u>ネットワーイ設定の定義</u>
- service- このURLパトは常に同じて変更することはできません。
- <as2-service-name> はAS2 サービスに与えるカスタム名です。このURLの部分をショブを作成する際に定義することができます。

必要に応じて、FlowForce Server を構成して、認証されていないクライアトカらのクエストを受け入れることができます、ますよ、クライア ントから基本的な HTTP 認証をリクエストすることもできます(従って、サービスを公開することができます)。AS2 サービスを認証無しにアクセ ス可能にするます、ユーザー 🌡 anonymous がのの トミッションを持つ FlowForce Server エンテナー内で AS2 サービスジョブを作 成します。 次の パーミッション、「サービス 使用」。 コンテナーと パーミッションの詳細に関しては、次を参照してくたさい、ノ<u>パーミッションとエンテナ</u> ー。このような構成のサンプルは関しては、次を参照してくたさい、サンプル、フル AS2 メッセージの交換(シンプル)。

必要条件

AS2 データをパートナーから受信する前に、次の必要条件を満たす必要かあります:

- データを受け取る各パートナーの詳細は、FlowForce Server に追加される必要が決ます。次を参照してくたさい、AS2 パートナーの構成。
- 「FlowForce Server」サービスは上記のとおり、目的 URL 上のノモート クライアノト からの接続を受け入れる必要があります。

デフォルトでは、FlowForce Server はポート 4646 上のローカルホスト からの接続を受け入れます。ローカルホスト以外のマシン からAS2 サービスにアクセスを可能にするさは、セットアップページを開き、FlowForce Server の・インドアドレスを、すべてのイン ターフェイス (0.0.0.0) に、おさは、特定のインターフェイス ご変更してくたさい。次を参照してくたさい、<u>ネットワーク設定の定義</u>更に、 FlowForce Server との通信がオペレーティングシステムのファイア・ケォールを介して、可能なことを確認してくたさ、。

メモ 「FlowForce Server」サビスを「FlowForce Web Server」サビス特別してけさい。後者はWeb 管理者インター フェイスニアクセスするために使用され、ポト 8082 上での接続を受け入れ、個別の構成を有します。次も参照してくたさいしく み。

AS2 サービスの作成方法

このサンプルは、AS2 サービスを公開するショブの作成方法について説明しています。最初に FlowForce Web 管理者インターフェイス 「コレゲオーします(次を参照してください) FlowForce Server ヘログオンする方法)。デフォルトの public コンテナー内にAS2 サービスを 作成することができます。個別のコンテナーを作成することは良案です(このサービスが個別の トーミッションを必要とする可能性かあるため)。 構成をクリックし、「作成 | コンテナーの作成」をクリックします。

コンテナーの作成場所 /
コンテナー名: as2service
保存 保存して移動

コンテナー名(例えば、「as2service」)を入力し、「保存して移動」をクリックします。次に、「作成 | ジョブの作成」をクリックします。 ジョブ構成ページが開かれます:



このジョブをAS2 サービスにするコよ チェックボックス「このジョブを利用可能にする」を選択し、サービスの名前を入力します(例えば、 「as2-receiver」)。更に、AS2 サービスをドロップダウンリストから選択します。

サービス ☑ このジョブを HTTP を介して URL 上で利用可能にする http://<FlowForce</p> AS2 サービス server>/service/ Ŧ

2つの新規の デメーターか自動的にごうブニ追加されます

ジョブ入力パラメーター					
•					
名前: partner	型: AS2 パートナー	•			
◆ 名前·message	型・フトリーム	•			
+					

パラメーター	目的
partner	この、デメーターは、メッセージを送信するAS2 パトナーについての情報を提供します。 パテメーターデータ型は 「AS2 パトナー」です。このオブジェクトを次のステップにて処理することができ、FlowForce 式関数 <u>as2-</u> <u>partner-local-name</u> おけは <u>as2-partner-remote-name</u> を使用してパトナーのローカル、また は、リモートの名前を文字列として取得することができます。
message	この、ラメーターは、受信するメッセージへのアクセスを提供します。メッセージのデータ型は、「stream」です。下記のとおり、FlowForce 式関数を使用してファイルへのストリームを変換することができます。

メモ 定義済みの、ラメーターpartnerとmessageを削除してはおりません。次のステップ内で定義済みの、ラメーターを使用しな し、場合、これらを無視することができ、ジョブが無効しなることはおりません。受信されるメッセージは通常何らかの方法で処理されま す(例えば、ファイルご保存するなど)。下記のとおり、これをFlowForce 式関数、具体的には、<u>MIME 式関数</u>、を使用して おこなうことができます。一部の場合、追加・ラメーターをジョブに追加する必要がある場合かあります(例えば、複数のステップ全 体で再利用するために定数の値の一部を定義する場合など)これを行う場合、パラメーターがデンオットレの値を持つことを確認して くたさい。それ以外の場合、AS2 メッセージが受信されるとどョブが開始されず、エラーメッセージがログされます

AS2 データを受け入れるようごうが構成されましたが、データの処理はおざ行われません。ストリームからペルセージエンテンクを読み取り、 ファイルご保存するこは、新規の実行ステップをジョブに追加する必要があります。新規の実行ステップをクリックして、/system/filesystem/copy 関数。を参照し、ソースとターゲット・パラメーターを下記のとおり入力します

実行	<u></u>	ステップ		
+)			
		関数の実行 /sy	vstem/filesystem/copy	
		パラメーター:	ソース:	{as-file(message)}
			ターゲット:	{substring(current-message-id(), 1, -1)}.msg
			上書き:	+
			エラー時には中断:	•
			作業用ディレクトリ:	C:\temp
=		ステップの結果	を割り当て名前	以下としてブール値

上の実行ステップは<u>/system/filesystem/copy</u> 関数を呼び出し、ソースからターゲット ヘデータをエピーします。ソースは FlowForce 式です。このサンプルでは、式

{as-file(message)}

は、以前に説明されたシッセージ、デメーターを読み取り、as-file 式関数を使用してファイル名に変換します。

式

{substring(current-message-id(), 1, -1)}

以下を行います

- <u>current-message-id</u>を使用してmessage-ID へダーフィールドの値を文字列とて取得します。例えば通常 Message-ID は以下のようよります <20180309125433018954-56c8aeb2fb4b478eb02f6f57662607da@somehostname>。
- 2. <u>substring</u> 式関数を使用して結果文字列の最初と最後の文字を取り除きます。これによりMessage-ID は次のようにより ます 20180309125433018954-56c8aeb2fb4b478eb02f6f57662607da@somehostname(山かご ^ベ< ["] and ["]> " か取り除かれていること注意してください)。

最後に、文字列「.msg」が式に追加され、これによりFlowForce か受信されるAS2 シッセージを保存する場所についた作成されます。 作業デルクトリC:\temp にいいか作成されることに注意してください。基本的には、AS2 シッセージが $http://<flowforce-server>:<port>/service/as2-receiver に送信されると、このジョブはシッセージェンテンルを読み取り、次のようないなご保存します C:\temp\20180309125433018954-56c8aeb2fb4b478eb02f6f57662607da@somehostname.msg。$

コメント:

• 「上書き」チェックが選択されていない場合、同じメッセージID が2回受信されると、ジョブはエラーを返すことを意味します。

• 「エラーが発生すると中断」設定が有効化されている場合、関数のコピーが失敗するとショブが失敗することを意味します。 失敗したジョブにより、FlowForce はオガティブな MDN をパートナーに送信します。 この場合、 このオプションは内部で有効化されます。

AS2 リクエストをリッスンする基本的な AS2 サービスを作成し、受信される AS2 サービスをローカルに保管します。この AS2 サービスかう ライアントによ 使用される方法に関してのサンプル 1 理しては サンプル フル AS2 メッセージの交換 (シンプル)。

実際のンナオでは、更に高度な処理のために、ジョブに更なる実行ステップを追加する必要があり、FlowForce内で他の武関数を使用 する必要があります。実行ステップ内で呼び出すことのできるFlowForce 関数に関するレファレンスに関しては、次を参照してくたさい、ビル トイン関数。FlowForce 式の基本的な紹介に関しては、FlowForce式、チャプターと参照してくたさい。

全てのジョブステップが実行される前にFlowForce か結果を返すように構成することは可能です。サービスとして呼び出されたショブが時間 を要す場合場合この機能は特に役に立ちます。早い結果は処理のためにFlowForce Server により受け入れられた呼出元により確認と して扱われます。詳細に関しては<u>延期されたステップ</u>を参照してくたさい。

10.9 例: フル AS2 メッセージの交換 (シンプル)

このサンプルでは、FlowForce Server のパースペクティブかションのAS2 パートナー間のAS2 メッセージ交換の構成方法について説明 されています。このサンプルでは、送信元 AS2 パートナーと受信先のAS2 パートナーは双方とも FlowForce Server インスタンスです。

送信する側のサー・〜「Hermes」と受信する側のサー・〜「Apollo」と呼ぶことします。Hermes はCentOS 上で作動、Apollo は Windows 上で作動することします(この詳細は なと、ファイア・オールの構成時に重要になります)。 このサンプルの目的は、以下のとおり です:

- 送信する側のサーバー(Hermes) が受信先のAS2 サーバー(Apollo) へのAS2 メルージの送信に成功する
- 受信する側のサーバー(Apollo)か受信するメッセージの処理に成功し、ローカルに保存する。

このサンプルは、2つのAS2パトナー間の単純な通信のシナリを表しています(RFC 4130 のセクション 2.4.2 に従い 2の可能な順列の最初の順列) 基本的には以下の点にていて説明されています:

- 送信者は暗号化が解除されたAS2 データを送信します。
- 送信者は署名されていないAS2 データを送信します。
- 送信者はシッセージへの応答内でMDN が返されることを必要としません。

他の前提

- Apollo とHermes は両者、ローカルのプライベートネトワーク上で作動しています。
- 受信側 AS2 サーバー(Apollo) は、認証されていない ウライアント からの HTTP リクエスト を受け入れます。(すなわち、サービス に公的にアクセスすることができます)

必要条件

- FlowForce Server Advanced Edition がインストールされている必要があり、 Apollo とHermes マントでライセンスが 供与されている必要があります。
- Apollo とHermes サーバー上で、FlowForce W eb 管理者インターフェイスが構成されているホストとポート上で作動している必要があります。(例えば、http://apollo:8082 とhttp://hermes:8082、「apollo」と「hermes」がそれそれ対応するホスト名であることが前提です)。次も参照してください、<u>ネットワーク設定の定義</u>。

AS2 サーバー(「Hermes」)の送信の構成方法

 FlowForce Web 管理者 イクターフェイスコングオン、「APOLLO」と ら名前の新規のAS2 パトナーを作成します。(次 も参照してくたさい: AS2 パトナーの構成)。このパトナーは AS2 サービスを受信するサーバーを識別します。暗号化、署 名、MDN はこの単純なサンプルでは必要とされず、パトナーの設定外かのように設定されるだけです。

J	ペートナーの名前:	APPOLO				
)	ペートナーの証明:	このオブジェクトは AS2	パートナー	[apollo]	として識別されます。	
ノ	ペートナー設	婝				
	AS2 名: Apollo					
]ーカル側の	設定				
	AS2 名: Hermes					
A	S2 サービス	、設定				
	メッセージの受信					
Н	TTP エンド	ポイント設定				
	リクエスト URL:		http://apollo	o:4646/servic	e/as2-receiver	
	リダイレクトモー	۲:	任意のリダ	イレクト	•	
	チャンクされた転	送エンコードを使用する:				
	HTTP 認証資格情報	反:	+			
	タイムアウト:		+			

上記のとおり、AS2 通信のために使用されたAS2パトナーの名前は「Apollo」で、FlowForce Server 内に保管されて いるパトナーオブジェクト名は「APOLLO」です。「Request URL」値はパトナーの大スト名がapollo であるこを想定 しています。ホスト名が異なる場合、URLを必要に応じて調整してくたさい。次のステップ内でこのURLの背景で実際のAS2 サービズを構成します。

2. AS2 メッセージを送信する新規のジョブを作成します。

a)「public」コンテナーを開き、「作成 | ジョブの作成」をクリックします。



(例えば、「send-as2」)ジョブ名を入力し、任意で詳細を入力します。

b) 新規のファイルシステムトリガーをクトックして、以下に示されるようト・リガー設定を設定します。デルクトリ/home/altova/as2/outgoing がHermes のマシン上に存在しない場合、作成してください。

トリガー							
チェック 開始: 有効期限:	□ンテンツ + +	 ファイルまたはディレクトリ: /hc 	ome/altova/as2/outgoing/*.edi	ポーリング間隔: 60	秒。解決するまで	0 秒	待つ。
タイムゾーン:	Europe/Berlin されました	•					

トリガーが追加されると、triggerfileとう名前の、ラメーターがショブに追加されます。この、ラメーター は/home/altova/as2/outgoing にファイルをコピーする都度ジョブを自動的にトリガーするファイル名を示しています。詳細に 関しては、次を参照してください、ファイルシステムのトリガー。

c) AS2 パートナーにお定義されているEDI ファイルをローカル なから送信する実行ステップを追加します。このステップの詳細に関しては、次を参照してください AS2 メッセージの送信。

Ę	実行ステップ							
	•	関数の実行 /sg	ystem/as2/send					
		パラメーター:	パートナー:	/public/APOLLO				
			メッセージ: メッセージ ID:	<pre>stream-open(triggerfile, 'application/EDIFACT') {new-message-id()}</pre>				
			エラー時には中断:					

d) 最後に、ローカルのマシン上のユーザーのアカウトに資格情報を追加します(通常、このマンメコングオンするために使用する ユーザー名と マワード)。これらはFlowForce Web 管理者インターフェイスのユーザー名と マワードと同じ資格情報ではない こ とご注意してくたさい。詳細に関しては、次を参照してくたさい、<u>資格情報</u>。

沙里	資格情報		
	資格情報によりジョブを実行:	 ○ 既存の資格情報を選択: ◎ □ーカルの資格情報を定義: ユーザー名 	altova
		パスワード	パスワードの変更

e)「保存」をクリックします。FlowForce Server ホームページ内の「アクティブよトリガー」の下にジョブカ表示されます。

ホーム構成ログ管理				
ようこそ!				
実行中のジョブ				
インスタンス ID ジョブ		アクティベーション日時	最後のアクション	ステータス
φ				
アクティブなトリガ)—			
型	ジョブ		次の実行 💲	情報
φ	Joublic/send-as2			Checking directory '/home/aitova/as2/outgoing/*.edi' for modification da

受信先 AS2 サーバー(「Apollo」)の構成方法

 FlowForce Server を目的のURL 上でAS2 クライアトから接続を受け入れるように構成します。このサンプルでは、AS2 クライアト デフォルトのポト 4646 上でプレーシな HTTP を介して Apollo に接続します。構成ページは以下のよう てかます (次も参照してください、ペトワーク設定の定義):

FlowForce Web 暗号化されていない	Server \接続	
有効化済み: バインドアドレス:	 ▼ ● 全てのインターフェイス (0.0.0.0) ▼ ● その他: 	ポート: 4646

2. FlowForce Server がオペレーティングシステムのファイアウォールを介して通信することが許可されていることを確認してくたさ。このサンプルでは、「Apollo」 FlowForce Server はWindows 上で作動するため、Windows Defender ファイアウォールを介して通信が許可されている必要があります。

P Allowed apps	_			Х
\leftarrow \rightarrow \checkmark \uparrow \textcircled{B} \ll Win \rightarrow Allowed a \checkmark \circlearrowright	earch Cor	ntrol Pane	el ,	Q
				^
Allow apps to communicate through Windows Def	ender F	irewall		
To add, change, or remove allowed apps and ports, click Change s	ettings.			
What are the risks of allowing an app to communicate?	- 😯 Ch	a <u>n</u> ge setti	ngs	
Allowed apps and features:				
Name	Private	Public	^	
✓ Altova FlowForce Server 2018	\checkmark			
✓ Altova license acquisition port (TCP)	\checkmark	~		
✓ Altova license acquisition port (UDP)	\checkmark	~		
Altova License Metering Port (TCP)	\checkmark	\checkmark		~
	OK	C	ancel	

3. 新規のFlowForce Server コンテナーを作成します。「as2service」と呼ぶことします。(FlowForce 内では、パーミッショ ンコンテナーレベルで設定されており、AS2 サービスを受け取るショブのためコ個別のコンテナーが作成されることが奨励されま す。このようして、既存のFlowForce ジョブご適用されているパーミッションに影響を与えることは、AS2 固有のパーミッション を必要とされるコンテナーのサーブして設定することができます)。

🗅 / O 📄 public O	
■ 名前	
as2-service	
作成 マ オブジェクトのインポート	選択されたオブジェクトの移動または名前の変更をする

4. 前に定義されている「as2service」コンテナーを開き、送信元のパトナーであるHermesを下記のとお兆作成します。 「Request URL」値はパトナーの大スト名がhermesであることを想定しています。ホスト名が異なる場合、URLを必要 に応じて調整してくたさい。

パートナーの名前: HERMES パートナーの証明:		
パートナー設定		
AS2 名: Hermes		
ローカル側の設定		
AS2 名: Apollo		
AS2 サービス設定		
HTTP エンドポイント	設定	
リクエスト URL:		
リダイレクトモード:	任意のリダイレクト	•
チャンクされた転送エンコードな	を使用する: 📃	
HTTP 認証資格情報:	•	
タイムアウト:	+	

両方のサーバーで相互運用性設定が同じてあることを確認してくたさい。例

柞	相互運用性の設定				
	データの圧縮:	署名前 ▼			
	MIC 検証アルゴリズム:	MDN 署名のアルゴリズムの使用			

5. 前に定義されている「as2service」コンテナーを開き、新規のジョブを作成します。このジョブの目的は、AS2 リクエストをナルス ンするAS2 サービスを公開することです。新規のAS2 メッセージが受信されると、このジョブは一時フォルダーイニピーします。

以下にジョブを作成 /as2service						
ジョブ名:	receive-as2					
ジョブの説明:	AS2 メッセージの受信。					

a) このジョブをHTTP を介して利用可能にする.. チェックボックスを選択し、AS2 サービスに名前を与えます(このサンプルでは、「as2-receiver」)。

サービス	
図 このジョブを HTTP を介して URL 上で利用可能にする http:// <flowforce server>/service/</flowforce 	AS2 サービス ▼

b)上記のとおり、オプィンAS2 サービスをドロップダウノストから選択します。この結果、2つの入力パラメーターがショブ、パートナーとメッセージに追加されます。これらは送信元のパトトナーとメッセージに関する情報を処理し保管するために使用されます。このサンプルでは、以下に示されるようにメッセージのみを保管します。

ジョブ入カパラ •	メーター	
名前: partner	型: AS2 パートナー	•
名前: message	型: ストリーム	•

c) ローカル スロラ信済みのシッセージをエピーする実行ステップを追加します。以下で使用される Flow Force Server 式は基本的にコンシャレージをファイル 変換し、Message-ID ヘッダーフィールドをベース ファイル名を作成します。これらの式に関する詳細に関しては、次を参照してくたさい、AS2 サービスの受信。

Ę	衍	ステップ		
	•			
	4	関数の実行 /sy	stem/filesystem/copy	
		パラメーター:	ソース:	{as-file(message)}
			ターゲット:	{substring(current-message-id(), 1, -1)}.msg
			上書き:	+
			エラー時には <mark>中</mark> 断:	+
			作業用ディレクトリ:	C:\temp
	=	ステップの結果	を割り当て名前	以下としてブール値

デルオリC:\as2\incoming が存在することを確認してくたさい。これが受信されたAS2 ニミュニケーションが保存されるディレオリです。

d) 最後に、ローカルのマシン上のユーザーのアカウトに資格情報を追加します(通常、このマンメコングオンするオークローサークアカウトに資格情報を追加します(通常、このマンメコングオンするオークローガー名と) マワードと同じ資格情報ではない ことに注意してくたさい。詳細に関しては、次を参照してくたさい、<u>資格情報</u>。

NH/	資格情報
	資格情報によりジョブを実行: ○既存の資格情報を選択:
	● ローカルの資格情報を定義: ユーリー名: altova パスワード: パスワード: パスワードの変更

6. コンテナーpublic / as2service に移動し、パーミッションをクリックします。パーミッションの追加をクリックして、パーミッションサービスを「as2service」コンテナー上のユーザー & anonymous に割り当てます:

~ _=	ミッションの編集				
検索:					
検索:	FlowForce Server		▼ コンテナー:	継承	•
	名前 🗢	説明	構成:	継承	•
	& all	FlowForce Server	サードス・	使田	•
	authenticated	FlowForce Server			
V	anonymous	FlowForce Server	貸格情報:	継承	•
	💄 root	FlowForce Server	関数:	継承	•
			証明書:	継承	•
			AS2 パートナー	継承	•



User or Role name 🗢		Permissions		
anonymous	Service:	Use		Change
& authenticated	Container: Configuration: Credential: Service: Function: Certificate: AS2 Partner: Security:	Read, Write Read, Write Use Use Use Use Use Read	inherited from i /public inherited from i /public	Change
🙎 root	Container: Configuration: Credential: Service: Function: Certificate: AS2 Partner: Security:	Read, Write Read, Write Use Use Use Use Use Read, Write	inherited from _ /public and & authen inherited from _ /public and & authen	Change

これによ効果的にAS2サービスを公開し、アクセスを可能し、認証無して使用できるよう」ないます。

AS2 メッセージの送信

Hermes マシン上で、.edi ファイルを前に構成された次のディレクトリにエピーします: /home/altova/as2/outgoing。ディレクトリのポー リング間隔が開始されると(デフォリトでは 60 秒)トリガーが実行され、 Apollo マシン上でジョブはファイルをAS2 サービスに送信します。

ジョブの結果を確認するゴま FlowForce Server ログをチェックし、次を参照してください、ジョブのログの確認。ジョブが失敗すると、ログ内に失敗の理由が表示されます。以下を含む理由がショブが失敗する理由として挙じたれます。

- Hermes 上のEDI ファイルへの なか正確ではかません。
- ジョブ内で指定されている Hermes オペノーティングシステムの資格情報が正確ではよい場合

- Apollo のマシン上でファイアウォールがジロックするために、Apollo サービスhttp://apollo:4646/service/as2receiverを使用することができたい場合
- サービスhttp://apollo:4646/service/as2-receiver のかめのFlowForce Server エンテナーパーミッションは 匿名のアクセスを禁止しています(すなわち、クライアトはAS2 サービスにアクセスすることができません)
- Apollo パートナーの URL のパロスト」 パラメーター か正確ではは、場合 (Hermes マシン上で、Apollo のマシン上で、まこ に両方)
- Apollo のマシン上で「相互運用性の設定」、デメーターがHermes パートナーのために正確に構成されていたい場合

成功すると、Apollo マンン上の受信ジョブは、受信するメッセージを処理し、次のパン新規のファイルを作成します: C: \as2\incoming。

10.10 例: フル AS2 メッセージ交換(高度)

このサンプルは、FlowForce Server 上で作動する2つのAS2パトナー間の暗号化、署名など、更に高度なAS2メッセージの交換 について説明しています。このチュートリアルを開始する前に、基本をカバーしている前のチュートリアルを確認してくたさい。次を参照してくだ さい、サンプル・フルAS2メッセージの交換(シンプル)。

このサンプルは、2つのAS2パートナー間の複雑な通信のシナリオを表しています(RFC 4130 のセグラン2.4.2 に従い12の可能な順列の12番目の順列) 基本的にはより下の点について説明されています:

- 送信者は暗号化されたAS2 データを送信します。
- 送信者は署名済みのAS2 データを送信します。
- 送信者は受信者がシッセージへの応答内でMDNを返すことを必要とします。

前提

- 前のサンプルと同じ送信者と受信者が使用されます: Hermes (Linux 上のFlowForce Server)とApollo (Windows 上のFlowForce Server)
- Hermes はApollo に暗号化済み、署名済みのメッセージをに送信し、署名済みのMDN が返されることを期待しています。
- Apollo とHermes は両者、ローカルのプライベートネトワーク上で作動しています。
- 受信側 AS2 サーバー(Apollo) は、認証されていない ウライアント からの HTTP リクエスト を受け入れます。(すなオカ、サービス に公的にアクセスすることができます)。

必要条件

- FlowForce Server Advanced Edition がインストールされている必要があり、 Apollo とHermes マントでライセンスが 供与されている必要があります。
- Apollo とHermes サーバー上で、FlowForce W eb 管理者インターフェイスが構成されているホストとポート上で作動している必要があります。(例えば、http://apollo:8082とhttp://hermes:8082、「apollo」と「hermes」がそれそれ対応するホスト名であることが前提です)。次も参照してください、<u>ネットワーク設定の定義</u>。

Apollo の資格情報のセオアプ

この構成ステップでは、以下が行われます:

- 1. Apollo は公開証明書と秘密キーを生成し、両方をFlowForce Server にんポートします。
- 2. Apollo は公開証明書 (秘密キー無しに) をHermes に送信します。
- 3. Hermes はApollo の公開証明書をFlowForce Server にんポートします。

上記が必要な理由

- Apollo にメッセージを送信する前に、Hermes は Apollo の公開キーを暗号化する必要があります。Hermes からのメッセージを受信すると、Apollo は自身の秘密気ーを使用して暗号化を解除します。
- Hermes によりルフェストされている MDN を送信する前に、Apollo は自身の秘密キーを使用して署名します。署名済みの MDN を受け取ると、Hermes は、署名を検証するために Apollo の公開証明書が必要になります。

このサンプルのスコープのするは、Cygwin (<u>https://cygwin.com/</u>)内に含まれるOpenSSL library (<u>https://www.openssl.org/</u>) を使用して自身にお屠名された証明書を作成します。現実のシナリオでは、他のソールを使用してを生成、ませよ、既に所属機関で使用 することのできる証明書を使用します。

Apollo のために自身で署名された証明書を生成するコよ Cygwin ターミナルを開き、以下を入力します:

openssl req -x509 -newkey rsa:2048 -keyout apollo_private.pem -out apollo_public.pem - days 365

・ プローズを入力するよう プロンプトされると、秘密キーを暗号化する プワードを入力し、記憶しておいてくたさい。FlowForce Server 「証明書をインポートする際「この プワードが必要「こなります。すべてのウィザードステップ」ご従い、すべての必要とされるフィールド (「Country」、「State or Province Name」、「Locality Name」、「Organization Name」、「Department Name」、 「Common Name」、および「Email」など」に記入します。

F ~ \times ~ altova@DOC-W10x64 🗻 \$ openss1 reg -x509 -newkey rsa:2048 -keyout apollo_private.pem -out apollo_publ ic.pem -days 365 Generating a 2048 bit RSA private key *** writing new private key to 'apollo_private.pem' Enter PEM pass phrase: Verifying - Enter PEM pass phrase: You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank For some fields there will be a default value, If you enter '.', the field will be left blank. Country Name (2 letter code) [AU]:AT State or Province Name (full name) [Some-State]:. Locality Name (eg, city) []:Vienna Organization Name (eg, company) [Internet Widgits Pty Ltd]:Apollo Organizational Unit Name (eg, section) []:. Common Name (e.g. server FQDN or YOUR name) []:apollo Email Address []:apollo@example.org altova@DOC-W10x64 ~

ウイザドが完了するとコマンドは、以下の2つのファイルをCygwinのホームデルクトリ内に生成します apollo_private.pem と apollo_public.pem (例えば、Cygwin をC:\cygwin64 にインストールしている場合 C:\cygwin64\home\<user>\ が作 成されます)。このペアはご単一のファイルとして FlowForce Server にアップロードすることができ、公開証明書を秘密キーファイルにことー するために次の追加コマンドを実行します:

cat apollo_public.pem >> apollo_private.pem

Apollo マシン上で、FlowForce Server コングオンし、構成 メニューをクリックし、作成 > 証明書の作成 をクリックします。

証明書名と詳細を入力し、参照をクリックして、前に作成されたapollo_private.pem ファイルを選択します。このステップで以前に作成された。 家ワードを入力します。 「保存」をクリックします。
証明書の作成 /			
証明書の名前: 証明書の詳細:	ApolloPrivate Apolloの公開証明書と秘密キー		
証明書イン	/ポート		
ファイルからイ バスワード:	ンポートする: 参照 ••••		
保存			

公開+秘密証明書ペアはApolloのFlowForce Server にんポートされます。アイコン
りまと詳細テキストはこの証明書ファイルが以下を含んていることに注意してくたさい

セキュリティ	
指紋	f4:2c:7e:3a:cf:77:e5:b3:b0:ac:77:29:5a:32:9d:ae:ac:42:77:4c
指紋アルゴリズム	SHA1
署名アルゴリズム	sha1WithRSAEncryption
公開キーアルゴリズム	rsaEncryption
公開キーサイズ	2048 ヒビット
秘密キーが含まれています	
自己署名	\checkmark

Hermes に公開キーを送信するこよ apollo_public.pem ファイルを Hermes マンノニピーします。次に、 Hermes マンン上で FlowForce Server (コングオンし、上と同じステップに従い、インポートします(今回は秘密キー(オファイル内に存在したいケンタ・ノ タワード が必要しています)。



アイコン 🧊 と詳細テキストは この証明書ファイルコは秘密キーを持たない)公開証明書のみか含まれていることを示していることご注意し てたれ、

セキュリティ	
指紋	e8:ce:cd:49:7a:e7:2b:43:90:45:22:5b:83:ca:6c:ea:39:f7:9d:b6
指紋アルゴリズム	SHA1
署名アルゴリズム	sha1WithRSAEncryption
公開キーアルゴリズム	rsaEncryption
公開キーサイズ	20481ビット
秘密キーが含まれています	
自己署名	

Hermes の証明書のセオアプ この構成ステップでは、以下が行われます

- 1. Hermes は公開証明書と秘密キーを生成し、FlowForce Server にんポートします。
- 2. Hermes は秘密キー使用せずコ公開証明書 をApollo に送信します。
- 3. Apollo はHermes の公開証明書をFlowForce Server にんポートします。

上記が必要な理由

- Apollo にシッセージを送信する前に、Hermes は自身の秘密キーを使用して署名します。
- Hermes からのシャセージを受信すると、Apollo はHermes の公開証明書を使用してメッセージの署名を検証します。

最初に、Apollo のためのステップと同じステップでHermes の公開証明書と秘密キーを作成します。ファイル名を置き換えてくたさい

```
openssl req -x509 -newkey rsa:2048 -keyout hermes_private.pem -out hermes_public.pem - days 365
```

更に、「Organization name」、「Common Name」などはHermes の物である必要が引ます:



```
次に、コマドを使用して両方のファイルを1つのファイルに結合します:
```

cat hermes_public.pem >> hermes_private.pem

次に、Hermes マンン上でhermes_private.pem をFlowForce Server にんポートします:

証明書の作成 /public					
証明書の名前:	HermesPublic				
証明書の詳細:	Hermes の公開証明書および秘密キーを含んでいます				
証明書イン	ンポート				
ファイルから	インポートする: hermes_private.pem 参照				
パスワード:	••••••				
保存					

次に、hermes_public.pem をApollo マンノニピーし、FlowForce Server にんポートします:

証明書の作成 /public				
証明書の名前:	HermesPublic			
証明書の詳細:	Hermes の公開証明書を含んでいます			
証明書イン	ンポート			
ファイルから	インポートする: hermes_public.pem 参照			
パスワード:	•			
保存				

Hermes 上でAS2 暗号化、署名、および、MDN 署名検証を有効化する方法 Hermes マシントで、APOLLO パトナー設定を以下のとお編集します:

Security Settings	この証明書は Hermes から Apollo へのメッセージが暗号化 されていることを保証します
Encryption Algorithm: AES-256 V Partner Certificate: /public/ApolloPublic	
Decryption +	この証明書により Apollo から送信されるメッセージを Hermes が検証することができます
Signature Creation Algorithm: SHA-1 V Local Side Certificate: /public/HermesPrivate	
Signature Verification Algorithms: MD5 SHA-1 SHA-224 SHA-2	CostDistational Section Apollo により送信された署名済みの MDN をApollo が検証 することができます
Partner Certificate: /public/ApolloPublic	• 2
Message Disposition Notification	pollo からの署名済みの MDN Hermes はリクエストしており、 J期して送信することができます
Request MDN: MDN Type: Synchronous 🔽 💼 Request Signed MDN: 🗹	
Interoperability Settings	
Compress Data: Before signing 🗸	(used if compression is enabled for outgo
MIC Verification Algorithm: Use algorithm of original message signature	(used if MDN is requested for incoming r

Apollo 上でAS2 暗号化、署名、および、MDN 署名検証を有効化する方法 Apollo マシン上で、HERMES パトナー設定を以下のとお編集します:

Security Settings				
Encryption + この証明書により Apollo は Hermes から送信されたメッセージの暗号化				
Decryption Algorithms: DES 3DES AES-128 AES-192 AES-256 RC2-40 RC2-64 RC2-128				
Local Side Certificate: /public/ApolloPrivate アクジェントがあます				
Signature Creation Algorithm: SHA-1 V				
Local Side Certificate: /public/ApolloPrivate ・ 2 2の証明書により Apollo は.				
Signature Verification Algorithms: MD5 SHA-1 SHA-224 SHA-256 SHA-384 SHA-512 Image: Constraint of the state of the				
Partner Certificate: /public/HermesPublic				
Message Disposition Notification				
Request MDN: +				
Interoperability Settings				
Compress Data: Before signing 🗸 (used if compression is enabled for outgoing messages; for in				
MIC Verification Algorithm: Use algorithm of original message signature 🗸 (used if MDN is requested for incoming message or returned				

MDN の処理

上記の必要条件に従い、Hermes はApollo にAS2 伝送のMDN を認めるを送信するようごクエストします。 <u>as2-success</u> 式 関数。を使用して、受信されるMDN (成功、失敗) の状態を計算することができます。これを達成するこよ、Hermes マントで FlowForce にエヴオノレ、サンプル フルAS2 メッセージの交換 (シンプル) 内で作成された「send-as2」 ジョブを開きます。 次に、ジョブ を下記のとお 変更します:

実行	ステップ		
+			
	関数の実行 /s	ystem/as2/send	▼ 2
	パラメーター:	パートナー:	/public/APOLLO
		メッセージ:	<pre>stream-open(triggerfile, 'application/EDIFACT')</pre>
		メッセージ ID:	+
		エラー時には中断:	•
=	ステップの結果	また割り当て mdn	× 以下としてAS2 MDN
	関数の実行 /s	ystem/compute	▼ 🖻
	パラメーター:	条件式: as2-succe	ess(mdn)
=	ステップの結果	。 観を割り当て 名前	以下としてTO

以下の点に注意してくたさい

- 「AS2 MDN」最初の実行ステップの結果が宣言されました。最初の実行ステップの結果が宣言されました(次を参照してくたさい、「このステップの結果を割り当てる」)。
- 実行が次のステップを続行する必要があるため「エラー時に中断」チェックボックスはクリアされます。
- 2番目の実行ステップは、/system/compute 関数を呼び出します。この関数は <u>as2-success</u> 関数を使用してブール式 を計算します。後者は最初の実行ステップにより返されたMDN を引数として取ります。

AS2 メッセージの送信

Hermes から Apollo 「暗号化され、署名済みのAS2 メッセージを送信する準備ができまた。Hermes マシン上で.edi ファイルを以前に構成された ディレオリ /home/altova/as2/outgoing にピーします。 ディレオ・リポーレング間隔が開始され (デフォルト では、60 秒)、 Apollo のマシン上でトリガーが実行され、 ジョブはAS2 サービス こアイルを送信します。 Apollo のマシン上のディレオ・リ C: \as2\incoming には Hermes により送信されたメッセージが含まれてい るはずです。 例

incoming			_		<
$\leftarrow \ \rightarrow \ \checkmark \ \uparrow$	- ≪ as2 > incoming	~ Ū	Search incoming	م ر)
Name	^	Туре	Size		
20180321142	154344152-7b9444fab04149669e5fb86dbd9	MSG File		2 KB	
1 item					

ジョブの実行の失敗、おけま、成功を確認するは、システムのエヴをチェックします(Hermes のマシンだけではなく、Apollo マシンのマシンでこの操作を行ってくたさい)。詳細に関しては、次を参照してくたさい、ジョブのエグの雑認。

ログゴは、この伝承に関連して発生した可能性のあるエラーは関する情報が含まれています。例えば、Hermes が暗号化されていないデーケを送信し、Apollo が暗号化されたデーケを期待する場合、ジョブは失敗し、関連するメッセージがログされます。

11 コマンド ライン インターフェイス

上記のとおりFlowForce ソリューションはつのサービスにと構成されています: FlowForce Server と FlowForce W eb Server.こ れらのサービスのそれぞれのために、コマンドラインで実行可能な管理コマンドをサポートする実行可能ファイルを使用することができます。両方の実行可能ファイルを次の、次で見つけることができます:

Linux	/opt/Altova/FlowForceServer2021/bin/		
macOS /usr/local/Altova/FlowForceServer2021/bin/			
Windows <programfilesfolder>\Altova\FlowForceServer\bin\</programfilesfolder>			

実行可能ファイル名は以下の通りです

Linux	flowforceserver flowforcewebserver	
macOS	flowforceserver flowforcewebserver	
Windows	FlowForceServer.exe FlowForceWebServer.exe	

コマボラインインターフェイス(CLI)をライセンス供与、トラブルシューティング、内部データベースのバックアップなど管理の目的のために使用することができます。CLIによりサポートされるコマンドは下にリストされています:

メモ 省略 [FFS] と [FFW] はコマドが Flow Force Server おは Flow Force Web Server のかり 使用可能かもして指定ます。

بر≿د	FFS	FFW	説明
<u>help</u>	こ	ほい	引数として与えられたコマドのためのへいた表示します。
<u>assignlicense</u>	は		W indows プラオフォームのみここのコマイドを適用することができます。 FlowForce Server にライセンスファイルをアップロード、およびライセンス ファイルを割り当てるために使用されます。
<u>compactdb</u>	ま		削除済みのレコードか含まれる場合 FlowForce .db ファイルのサイズを 削減します。
<u>createdb</u>	ほ		新規のFlowForce データベースの作成
<u>debug</u>	ほ	ほく	デッシグモードでアプリケーションを開始します。
<u>exportresourcestrings</u>	こ	ほく	XML ファイルに全てのハノース文字列をエクスポートします。
foreground	こ	ほ	前景モードでアプリケーションを開始します。
<u>initdb</u>	は、		FlowForce データベースを作成おけま更新します。
<u>install</u>	はい	はい	アプケーションをWindows サービスとてインストールます。

<u>licenseserver</u>	は		ローカルネトワーク上のAltova LicenseServer にFlowForce Server を登録します。
migratedb	ほ		FlowForce Server データを前の、ージョンから最新の、ージョンに移行します。
<u>repair</u>	こ		修復モドでアプケーションを開始します。
<u>resetpassword</u>	は、		^と root ユーザーの ペワードをデフォルトの直 こ セートし、 ^と root ユー ザーニ全ての特権を与えます。
<u>setdeflang sdl</u>	こ	はい	デフォルトの言語を設定します。
<u>start</u>	こ	はい	アプリケーションをサービスとして開始します。
<u>uninstall</u>	こ	はい	アプリケーションをWindows サービスとしてアンインストールします。
<u>upgradedb</u>	は		FlowForce Server データベースを最新の・ジョンニアップグレードします。
<u>verifylicense</u>	はい		Windows プラオフォームのみこのコマドを適用することができます。 FlowForce Server にライセンスが与えられているか、与えら得たライセンスキーが既にFlowForce Server に与えられているかを検証するすよめ に使用されます。

規則

規則としては、このドキュメンテーションは特定のコマンドを説明する際に実行可能ファイルのフルレンを省略し、実行可能ファイルのなめを使用する代わりにflowforceserverを使用します。例:

flowforceserver help

flowforceserver は実行可能ファイルの やまけは名前です。絶対ル やを使用する場合、コマイ・プロンプトウイイトウ(ターミナル)かある現在のディレクトリニ関わらずコマイドを実行することができます。しかしなから、実行可能ファイルを名前を入力するだけで呼び出す場合、以下を最初に行ってくたさい

- ターミナルの現在のディレケートを Flow Force Server インストールディレケールで変更します。
- ディレクトリを実行可能ファイルがPATH環境変数である場所に追加します。

これらのシナリオの詳細は以下で説明されています。

ビトムリク

コマボラインの初心者の場合、以下のヒナよリックに注意してくたさい。

- コマボラインウイドウかある現在のディレクトリを探すコはpwd をLinux とmacOS 上で入力します。Windows 上では echo %CD% を入力します。
- Tab キーを活用して素早く多種のファイルおけまデルノケリッジを全て入力することなく素早く入力してくたさい。例えば cd c: 、prog をコマ・ドライノに入力し、Tab を押すとC: \Program Files が自動的に入力されます(おけま、[Prog]で開始 するC:¥の下の他のディレクトリが入力されます)。
- Windows 上でC: \Program Files などの空白が含まれる、びを入力する場合、引用符で囲んでたさい。
- [このコマイドは内部また」おり部コマイ、実行可能なプログラム、また」おうシッチファイルとして認識されていません」に類似したメッセージが表示されると、パスまた」コマイドを間違って入力した可能性があります。

- Linux 上ではファイルおけまディレケリ名のために正確な大文字と小文字が使用されているとを確認してください。例えば、ディレケリ名が/home/nikita/Downloadsの場合 /home/nikita/downloads はエラーを返します。
- Linux おけまmacOS 上でやを入力する場合、Windows でのドックスラッシュの使用とは異なりスラッシュを使用します。

コマドの実行方法

- 1. コマドプロンプトウィドウを開く
 - a. Windows 上でコマドプロンプトを開くには Windows キーを押し、 cmd の入力を開始します。 key and then start 表示される コマンドプロンプト 提案をクトックします。
 - b. Mac 上でターミナルを開き ファインダー アイコンをクトックし 移動 > ユーティリティ をメニューから選択します。ターミナルア イコンをユーティリティンインドウ内でダブルクトックします。
 - c. Linux をグラフィカルなユーザーインターフェイスから作動する場合、ターミナルコマイドを使用中のLinux の配布に適用できるようコケートし実行します。Linux をコマイドラインインターフェイスから実行する場合、このステップを無視してください。
- 2. 実行するコマボが後に続く実行可能ファイルへのフルクを入力します。例えば、下のコマボはコマボラインでヘルを提供します。

Linux	/opt/Altova/FlowForceServer2021/bin/flowforceserver help
macOS	/usr/local/Altova/FlowForceServer2021/bin/flowforceserver help
Windows	C:\Program Files (x86)\Altova\FlowForceServer2021\bin\FlowForceServer.exe help

上のサンプルでは、コマンド help がオプションおけお |数無しこ実行されています。他のコマンド は |数おけはオプションを持っている場合があり、これななお |数とオプションは必須おけはオプションです。各コマンドの詳細に 異しては ルファレシスセグションをチェックしてください。

FlowForce Server をインストールディレクトリから呼び出す方法

実行可能ファイルをフル マを入力することなく呼び出すコよ、現在のディレクトリを FlowForce Server 実行可能ファイルがインストールされている個所に変更します。例

Linux	cd /opt/Altova/FlowForceServer2021/bin
macOS	cd /usr/local/Altova/FlowForceServer2021/bin
Windows	cd "C:\Program Files (x86) \Altova\FlowForceServer2021\bin\FlowForceServer.exe"

実行可能ファイル名のみを入力してコマイを実行することができます。例

Linux	./flowforceserver help
macOS	./flowforceserver help
Windows	FlowForceServer.exe help

メモ Linux とmacOS システムとではプレフィックス./ は実行可能ファイルは現在のディレクトリ内にあることを示しています。

FlowForce Server をディレクトリから呼び出す方法

実行可能ファイルをディレクトリから呼び出すココ総対パタを使用して参照してくたさい。ませる、プログラムを実行可能ファイル名を入力する

だけで呼び出す場合、使用中のオペレーティングシステムのPATH環境変数を編集することことり、 FlowForce Server インストールディレクトリへのフルレ マか含まれるようごといます。PATH環境変数を変更する方法に関しては、使用中のオペレーティングシステムのドキュメンテ ーションを参照してください。

メモ PATH 環境変数の変更後、ターミナルウイドウを閉じ、変更の効果が反映されるように新規のウイドウを開く必要かある場合 かあります。

11.1 help

目的

引数とて与えられたコマドのためのへいた表示します。

構文

FlowForceServer help Command

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

引数

ヘルプコマドは単一の引数を取ります: ヘルプが必要なコマド名。コマドの正確な構文と正確なコマドの実行に関連する他の情報を表示します。

サンプル

FlowForceServer help createdb

-helpを他のコマドのためのオプションとして使用する

コマドロ関するへいプ情報は、そのコマドの--help オプションで使用することができます。例えば、createdb コマドを使用した--help オプションの使用方法は、以下の通りです:

FlowForceServer createdb --help

以下と同じ結果が得られます

FlowForceServer help createdb

11.2 assignlicense

目的

W indows プラナトフォームのみにこのコマナドを適用することができます。ライセンスファイルを Flow Force Server にアップロードし割り当てるために使用されます。

構文

FlowForceServer assignlicense [options] FILE

引数

FILE	ライセノスファイルはアップロードされる、マを指定します。
------	------------------------------

t,test-only=true false	true に設定されていると、ライセンスはアップロードされ検証されます。
	false に設定されていると、ライセンスはアップロード、検証、および 割り当てられます。
	このオプノョンが指定されてなと、デフォルトの値はtrueです。

11.3 compactdb

目的

削除されたシコードを含むFlowForce .db ファイルのサイズを縮小します。このコマンドは、archive-log おけはtruncate-log システム管理関数を実行した後とても役にたます。

構文

FlowForceServer compactdb [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

$\frac{1}{2} \frac{1}{2} \frac{1}$	datadir=VALUE	VALUE は、縮小される.db ファイルのデータディレクトリの、なです。この オプションが指定されていない場合、/data ディレクトリがデフォルトで 使用されます(次も参照してくたさい: <u>FlowForce Server アプリケ</u> <u>ーンコンデータ</u>)。
--	---------------	---

11.4 createdb

目的

新規データベースを作成します。データベースが既存の場合、コマンドは失敗します。デフォルトのデータベースはインストール時に作成され、通常このコマンドを使用する必要はありません。

構文

FlowForceServer createdb [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

datadir=VALUE	VALUE は データディークトリの 次です。
---------------	-------------------------

11.5 debug

目的

一般使用のナダのコマイドではありません。このコマイドはFlowForce Server デバッグモードで開始します(つまり、サービスではなく開始します)。このモードを停止するはよ「CTRL+C」を押します。

構文

FlowForceServer debug [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

datadir=VALUE	VALUE はデータディレクトリの やです。
---------------	------------------------

11.6 exportresourcestrings

目的

FlowForce Server のハース文字列を含むXML ファイルを出力します。以下の引数を取ります。(i) 出力 XML ファイル内のハース文字列の言語、および(ii) 出力 XML ファイルの やと名前。有効なエクスポート言語は以下の通りです(言語コードはかっこの中です): 英語(en)、ドイン語、(de)、スペイン語(es)、および日本語(ja)。

構文

FlowForceServer exportresourcestrings Language XMLOutput

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

引数

Language	エクスポートされたXML ファイル内のハース文字列の言語を指定します。 許可される言語: en, de, es, ja
XMLOutput	エクスポートされた XML ファイルの場所と名前を指定します。

サンプル

このコマイは、英語のFlowForce Server アプリケーションの全てのリノース文字列を含むこと、にStrings.xml とう名前のファイ ルを作成します。

FlowForceServer exportresourcestrings en c:\Strings.xml

11.7 foreground

目的

一般使用のためのコマドではありません。このコマドは、Altova FlowForce Serverを前景で開始します。Linux のためのスタートア・プスクリプトにより内部で使用されます。

11.8 initdb

目的

新規データベースを作成、ますま、既存のたのを最新の、デジョンにアップデートします。データベースが既存の場合、コマンドは失敗します。デ フォルトのデータベースはインストール時に作成され、通常このコマンドを使用する必要はありません。

構文

FlowForceServer initdb [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

datadir=VALUE	VALUE は データベースディレクトリの みです。
---------------	----------------------------

11.9 install

目的

このコマイはFlowForce Server インストーラーにお自動的に実行され、一般の目的には使用することができません。このコマイド はAltova FlowForce Server をサービスとしてWindows にインストールします。このコマイドをLinux とmacOS に適用すること はできません。

11.10 licenseserver

目的

FlowForceServer をLicenseServer に登録します。FlowForceServer をLicenseServer に登録するは、管理者の特権(ルート)が必要です。詳細に関しては、以下を参照してたさい、LicenseServer ドキュメント。

構文

FlowForceServer licenseserver [options] Server-or-IP-Address

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

サンプル

FlowForceServer licenseserver DOC.altova.com

上のコマドは、DOC.altova.com とう名前のマンがAltova LicenseServer を動作してます。LicenseServer がユーザーのマンで作動している場合、次のコマドも使用することができます:

FlowForceServer licenseserver localhost FlowForceServer licenseserver 127.0.0.1

オプション

オプションは下にリストされており、短い書式(最初の列)と長い書式(2番目の列)はそれそれ詳細を記載しています。コマバライン上では、1つまけは2つのダッシュか短い、および、長い書式のために使用されています。

j	json	JSON オブジェクトとして登録の結果を生成しようと試みます。
		אָדעjson=true false

11.11 migratedb

目的

前のアプリケーションデータディレクトリ から FlowForce Server データを現在のディレクトリニエピーし、 FlowForce データベースを必要な 場合最新バージョンに更新します。 FlowForce Server の前のバージョンがうていて、「したれている場合に FlowForce インストー ルスクリプトによりこのコマンドは呼び出され、通常実行する必要にはかません。 FlowForce Server を新ししマシンに移行する場合、 おまは アプリケーションデータディレクトリをバックアップから復元する場合、 このコマンドの実行は役に立つかもしれまれん。 <u>バックアップと復元</u>を参照し てください。

FlowForce データベースバージョンを最新のバージョンに更新するのみが必要な場合、upgradedb を実行するだけで十分です。

構文

FlowForceServer migratedb [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

オプション

datadir=VALUE	VALUE は データベースデルノトリの みです。
olddatadir=VALUE	VALUE は データベースディレクトリの みです。

サンプル

FlowForce Server 2017 のアプリケーションデータディレクトリからデータをFlowForce Server 2021 に移行する場合、以下を実行します:

"C:\Program Files(x86)\Altova\FlowForceServer2021\bin\FlowForceServer.exe" migratedb --datadir=C:\ProgramData\Altova\FlowForceServer2021\data --olddatadir=C: \ProgramData\Altova\FlowForceServer2017\data

11.12 repair

目的

FlowForce Server を全てのトリガーとゴブの実行プロセスを無効化した状態で、トラブルシューティングを有効化するために開始します。

構文

FlowForceServer repair [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

オプション

datadir=VALUE	VALUE はデータベースディレケーリの みです。
---------------	---------------------------

サンプル

FlowForceServer repair --datadir=C:\ProgramData\Altova\FlowForceServer2021\data

11.13 resetpassword

目的

^Lroot ユーザーの 次ワードをデフォルの値にルケレ、^Lroot ユーザーに全ての特権を与えます。このオペレーションを開始する前に、実行中のFlowForce Server インスタンスを停止することが奨励されます。I (以下を参照 Linux、macOS、および Windows 上でサービスを開始ませ」は停止するための命令)。

構文

FlowForceServer	resetnassword	[options]
TTOMPOLCEPETVEL	reserpassworu	[Options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

オプション

datadir=VALUE	VALUE は データベースディレクトリの なです。
---------------	----------------------------

サンプル

FlowForceServer resetpassword --datadir=C:\ProgramData\Altova\FlowForceServer2021\data

11.14 setdeflang (sdl)

目的

setdeflang コマボ (略して sdl) は FlowForce Server のデフォルの言語を設定します。デフォルの言語を変更するコよ FlowForceServer とFlowForceWebServer サービスのためこのコマボを実行してくたさい(以下を参照 構文)。

構文

FlowForceServer setdeflang | sdl LanguageCode
FlowForceWebServer setdeflang | sdl LanguageCode

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

LanguageCode の値には以下からます。

- en **英語**
- es **水心語**
- de **ドイン**語
- fr **万次語**
- ja 日本語

サンプル

FlowForceServer setdeflang de

11.15 start

目的

FlowForce Server をサービスとして開始します。このコマンドは、内部でスタートアップスクリプト、ませよ、Windows サービスインストールにと使用され、一般の使用のためではありません。

構文

FlowForceServer start [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

datadir=VALUE	VALUE は データベースディレクトリの なです。
---------------	----------------------------

11.16 uninstall

目的

このコマナドはFlowForce Server インストーラーにより自動的に実行され、一般の目的には使用することができません。このコマナドは Altova FlowForce Server をサービスとしてWindows にインストールします。このコマナドをLinux とmacOS に適用することはできません。

11.17 upgradedb

目的

データベースを最新のバージョンイニアップグレードします。デフォルトのデータベースは自動的にインストール時にアップブレードされるため、手動でこのコマイドを使用する必要におりません。

構文

FlowForceServer upgradedb [options]

メモ Linux システム上では、全て小文字のflowforceserverを使用して実行可能ファイルを呼び出します。

オプション

datadir=VALUE VALUE はデータベースデルクトリの	のみです。
-----------------------------------	-------

サンプル

FlowForceServer upgradedb --datadir=C:\ProgramData\Altova\FlowForceServer2021\data

11.18 verifylicense

目的

W indows プラオフォームのみにてのコマイドを適用することができます。FlowForce Server にライセノスか与えられているか、与えら得たライセノスキーが既にFlowForce Server に与えられているかを検証するために使用されます。

構文

FlowForceServer verifylicense [options]

1,license-key=VALUE	このオプションにより、特定のライセンスキーが既にFlowForce Server に与えられているかを検証することができます。
	検証するライセンスキーに値が設定されている必要があります。

12 ビルトイン関数

このセクションでは、FlowForce Server に内蔵されているシステム関数について説明しています。ビルトイン関数を使用することにより、ファ イルをコピー、移動し、ディレクトリを作成し、シェルコマンドを実行するなど多種のアクションを実行することができます。FlowForce Server ビルトイン関数は、/system コンテナー内で使用することができます。

以下のトピックはルトコンテナーは相対したのマンジレープ化されたビルトで関数について説明しています。

- /system/abort
- /system/compute
- /system/compute-string
- <u>/system/as2</u>
- <u>/system/filesystem</u>
- /system/ftp
- /system/mail
- <u>/system/maintenance</u>
- <u>/system/shell</u>

RaptorXML Server がFlowForce Server に統合されている場合、すべてのRaptorXML Server 関数を含む追加コンテナーを使用することができます。これはRaptorXML+XBRL Server にも同様に適用されます。詳細に関しては、<u>RaptorXML Server との統合</u>を参照してくたさい。

Windwos ネトワーク いの参照

ジョブを作成する際、通常 FlowForce Server が作動するマシン上で、おゴよ、やトワーク上のファイル ジを参照します。マップされたネットワークドライブなどの、Windows ネ・トワーク・ジを参照する場合、Universal Naming Convention (UNC)構文を使用します。これは、ドライブ文字は、システムに対してグロー・ジレではおよりか、おこ、各ログインセッションは自身のドライブ文字に書い当てられているため、これはドライブ文字に書い当てられているため、これはドライブ文字がシステムに対してグロー・ジレではおく、ログオンセッションが自身のドライブ文字に書い当てられているため必須です。

UNC 構文は以下のフォームをとます:

\\server\sharedfolder\filepath

該当する箇所

- server が(DNS によに定義される)ネトワーク内のサーバー名を参照する箇所.
- sharedfolder が管理者にお定義されるラベルを参照する箇所例えば、admin\$ が通常オペレーティングインストールのルートディレクトリである箇所)。
- filepath がっての下のサブディレクトリを参照する箇所。

12.1 /system

/system コンテナーゴは全てのFlowForce ビルトク関数が含まれています。abort、compute、および compute-string 関数 のみが直接このコンテナーで見つけることができます。(例えば、AS2 関数、ファイルシステム関数、など)他の関数は適用できるコレア別に サブコンテナーイン分別されています。

12.1.1 abort

フルな:/system/abort

ジョブの実行を中断します。ジョブの実行を中断します。この関数は、通常条件(つまり、選択ステップ)内で条件が満たされた場合、ジョブを意図的に中断するために使用されます。プログラム言語内のthrow おけまraise 関数と同じです。この関数は、値を返しません。

パラメーター

名前	型	説明
メッセージ	string	必須の文字列/ ラメーターです。 ジョブを停 止する際に出力されるメッセージを指定しま す。

サンプル

次のジョブでは、abort 関数は、チェックされた値が0個のアイテムを超えると、ジョブがエラーを返して完了するために使用されます。アイテムの数が、10個よりたりない、または、10個の場合、ジョブボリストのアイテムは0個よりた少ないです」というテキストをローカルシステム上のファイルにか書き込みます。

実行ス	ステッ	プ					
•							
•	■ 新知の美行「/system/abort						
•							
▲ For each item シーケンスで list(1,2,3,4,5,6,7,8,9,10,11)			3,9,10,11)				
	•						
	4	選択					
		When	item>10				
		•					
		4	関数の実行 /	system/abort			▼ 🖻
			バラメーター:	メッセージ: リストカ	[×] 10個のアイテム	を超えるため中断します	
	= ステップの結果を割り当て 名前						
	• When <i>item</i> <+10						
関数の実行 /system/shell/commandline			▼ 2				
			バラメーター:	コマンド:	echo"ファイルは	こは10個より少ないアイテムが存在します">Fi	le.txt
				エラー時には中断:	+		
				作業用ナイレクトリ:	c:\temp		
		=	ステップの 結果	を割り当て名前		以下としてresult	
		•					
	新規 When Otherwise						
	 ● = ステップの結果を割り当て 名前 						
	•						
=	ステップ	プの結果	を割り当て名前	ίj			

12.1.2 compute

フルな:/system/compute

式の結果を計算し、計算された値を返します。計算され値は、パラメータ内、おけま他の実行ステップの式内で使用することができます。サービスとして使用されるジョブの出力を定義するためにこの関数を使用することができます(例を参照)。

この関数は、任意の型を示す値 TO を返します。つまり、返されたデータ型は、式パラメーター内で使用された式から推定されます。つまり、返されたデータ型は、式 パラメーター内で使用された式から推定されます。

パラメーター

名前	型	説明
式	Expression of TO	計算される Flow Force Server 式。式 の詳細・現しては <u>The Flow Force 式</u> 言語を参照してくたさい。

サンプル

このサンプルは、2つの実行ステップを使用するジョブに関する説明をします。最初のステップは、c:\temp デルノトリ内のシェルコマンドを実行、結果はhello として宣言されています。

次にこの結果は2番目の実行ステップンマされます。2番目の実行ステップは式言語(特に stdout とcontent 関数)を使用して 以下を行います:

- 最初のステップの結果の標準出力を取得します。
- 出力を文字列に変換します。

compute 関数は Expression 式テキストボックスに入力された式を評価します。 the text box.

Ę	ミ行:	ステップ			
		関数の実行ス	、テップは削除されまし	ite	
		関数の実行	/system/shell/comm	andline	
		バラメーター:	コマンド: エラー時には中断: 作業用ディレクトリ:	echo"Hello wo + c:\temp	orld"
	=	ステップの 結界	Rを割り当て hello		以下としてresult
	4	関数の実行) バラメーター:	/system/compute 条件式: content	(stdout (<i>hello</i>))	
	=	ステップの 結界	長を割り当て 名前		以下として T0

<u>"Hello, W orld!" ジョブの作成</u>を参照してくたさい。

12.1.3 compute-string

フルな:/system/compute-string

式の結果を文字列とて出力します。このステップ関数は、フォーマナが式ではな、文字列である以外、compute 関数と基本的につじ処理を行います。

パラメーター

名前	型	説明
式	string	計算される文字列としての)FlowForce Server 式

サンプル

/system/compute/と/system/compute-string 関数の違いを理解するコよ 以下のサンプルコ留意してけさい
ジョブ	「入力パラメーター	_	
◆ 名前: ◆	inputname	型: 文字列	▼ デフォルト: 🛨 説明:
実行	ステップ		
•	関数の実行 /system/co	mpute	
	バラメーター: 条件式:	concat(filename(inputn	ame), 'txt')
=	ステップの結果を割り当て	outputname1	以下として TO
	関数の実行 /system/co	mpute-string	
	バラメーター: 条件式:	{filename(inputname)}.t	ist
=	ステップの 結果を割り当て	outputname2	以下としてstring
	関数の実行 /system/co	mpute-string	
	バラメーター: 条件式:		
=	ステップの 結果を割り当て	名前	以下としてstring
	関数の実行 /system/co	mpute-string	
	バラメーター: 条件式:	Both expressions are {if	(outputname1==outputname2,",'not')}identical
=	ステップの 結果を割り当て	名前	以下としてstring

上記のジョブでは、3つの実行ステップがあります。

最初のステップは、/system/compute/関数を呼び出します。中かったが使用されていないことご注意してくたさい。(背景色により示されるようにフィーリド全体が式を保管する場合、中かってを示します。式は、2つの値を連結し、ジョブ入カパラメーターにより文字列を作成します。例えば、入カパラメーターが、"c: ¥emp ¥nvoices.txt"の場合、ステップは、(outputname1 と宣言された)文字列の値 "invoices.txt"を返します。

2番目のステップは、/system/compute-string 関数を呼び出します。この関数は、埋め込まれたFlowForce 式を含む文字列を 処理します。ここでは、中かっこは、文字列の残りから式を区別するために使用されています。埋め込まれた式は、文字列の残りの部分と背 景色が異なることに注意してくたさい。異なる技術が使用されていますが、ステップの結果(outputname2)はoutputname1 と同じで す。 最後に、3番目のステップは、outputname1 とoutpuname2 を比較するために/system/compute-string 関数をもう1度呼び 出します。両方の値が同じの場合、結果は文字列の値「両方の式は同等です」となります。それ以外の場合、結果は「両方の式は同等 ではありません」となります。

12.2 /system/as2

/system/as2 コンテナーゴは AS2 パートナーにAS2 メッセージを送信するためコ使用されている send 関数が含まれます。

12.2.1 send

フルな:/system/as2/send

AS2 メッセージをリモートのAS2 サーバーに送信します。ジョブからこの関数を呼び出すコよ (適用することのできる証明書を含む) AS2 パートナーの詳細が Flow Force Server 内で構成されている必要があります。次も参照してくたさい、AS2 ジョブの作成。

この関数はサーバーとプロトコーリからの補助情報により返された実際のMDNをカプセリ化するAS2 MDN オブジェクトを返します。AS2 MDN オブジェクトから追加情報を取得するはま、実行ステップを追加し(例、HTTP ステータス、おはま、オバジナリのメッセージのMDN) 必要とする AS2 式関数を呼び出す実行ステップを追加します。

名前	型	説明
<u>パートナー</u>	AS2 Partner	「AS2 パートナー」オブジェクトを参照する にコよ、次を参照してくたさい、 <u>AS2 パートナ</u> <u>ーの構成</u> 。
メッセージ	stream	AS2 メッセージのエンテンツ。このフィールド はFlowForce 式を期待します。送信さ れるファイルを意味する「ストリームの式」が 期待されるデータ型がストリームご変換され る必要かあることご注意してくたさい(例え ば、XML またはEDI ファイル)。これを達 成するこは、このフィールドにストリーム関数 を呼び出す FlowForce 式を入力してくた さい。例:
		<pre>stream-open("C: \files\myfile.edi", "application/EDIFACT")</pre>
		上のstream-open 関数は、
		FlowForce 内の式の詳細は関しては <u>FlowForce 式言語</u> を参照してくたさい。
エラー時には中断する	Boolean	このブール デメーターはショブが失敗した場合の戻り値を決定します。「エラー時には中断」がFALSEの場合、関数は、ブール値のFALSEを返します。「エラー時には中断」がTRUEの場合、ショブの実行は中断されます。デフォルトの値は、

364 ビルトイン関数

名前	型	説明
		TRUE です。

サンプル

以下の例を参照してくたさい

- <u>サンプル フルAS2 メッセージ交換(シンプル)</u>
- サノプル・フルAS2 メッセージ交換(高度)

12.3 /system/filesystem

/system/filesystem コンテナーは、FlowForce Server が作動するオペレーティングシステム上のファイルとディレクトリを管理するため に使用される関数を含んでいます。

ジョブの実行内のファイル いよ、使用中のローカルズシンではよく、FlowForce Server が作動するオペレーティングシステム上のいてある必要があります。

12.3.1 copy

フル☆:/system/filesystem/copy

ソースからターゲナディレオリニファイルをコピーします。任意で、ファイルを新しい名前と共にターゲナディレオリニピーすることができます。

シンプルな実行ステップから呼び出された場合、この関数は、このファイルをその都度コピーします。FlowForceを使用して複数のファイルをコ ピーする場合、copy 関数を呼び出すステップをステップ内でファイルのコピー サンプル内で説明されているとおり、囲んでくたさい。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異ないます。

- 「エラー時には中断」 パラメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSEの場合、関数は、FALSEを返します。

名前	型	説明
у— д	string as file	ピーするノースファイルのファイル名とな。
ターゲット	string as file	保存先ディレクトリのウンファイル名。ファイ ルの名前を変更する場合、保存先フィール ドに異なる名前を入力することもできます。
上書き	boolean	true の場合は、保存先ファイルが上書き されます。デフォルトの値はfalse です。
エ ラー時 にコま中断する	boolean	このブール ラメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
作業ディレクトリ	string as directory	作業デルクトリを指定します(例えば c: \somedirectory)。相対的ないなか使 用されると、作業デルクトリニ対して解決さ れます。

サンプル ファイルのコピーを参照

12.3.2 delete

フル☆:/system/filesystem/delete

引数として与えられたいなからファイルを削除します。

シンプルな実行ステップから呼び出された場合、この関数はコンフファイルをその都度削除します。FlowForceを使用して複数のファイルを 削除する場合、delete 関数を呼び出すステップを「for-each」ステップ内でファイルのコピーするサンプル内で説明されているとおり、囲 んでくたさい。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異ないます。

- 「エラー時には中断」 パラメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSEの場合、関数は、FALSEを返します。

名前	型	説明
え	string as directory	削除するファイルのファイル名とな。
エラー時 には中断する	boolean	このブール・ラメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSEの場合、関数は、ブー 川値のFALSEを返します。「エラー時に は中断」がTRUEの場合、ショブの実行 は中断されます。デフォリトの値は、 TRUEです。
作業ディレクトリ	string as directory	作業ディレクトリを指定します、例えば、c: \somedirectory)。相対的ない ぷか使 用されると、作業ディレクトリーズオして解決さ れます。

パラメーター

12.3.3 mkdir

フルな:/system/filesystem/mkdir

指定されたり、マニデルノトリを作成します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ウメーターの値 により以下のように異なります:

• 「エラー時には中断」、デメーターが TRUE (デフォルの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護

されたブロックを使用して処理することができます(以下を参照: ステップのエラーの処理)。

• 「エラー時には中断」、デメーターがFALSE の場合、関数は、FALSE を返します。

パラメーター

名前	型	説明
え	string as directory	新規デルケークな。
親にする	boolean	チェックボックスを選択して、アクティブ化する ことにより、 c: \dir1\dir2\dir3 のような 階層・なが1つのステップで作成されます。
エ ラー時 には中断する	boolean	このブール ラメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ショブの実行 は中断されます。デフォットの値は、 TRUE です。
作業ディレクトリ	string as directory	作業デルクトリを指定します(例えば c: \somedirectory)。相対的ないなか使 用されると、作業デルクトリニズ扎て解決さ れます。

サンプル

作業ディレクトリがc:\temp、パスがtemp2\temp3の場合、関数は、新規ディレクトリc:\temp\temp2\temp3を作成します。

12.3.4 move

フルな:/system/filesystem/move

ファイルを移動ませるファイルの名前を変更します。

シンプルな実行ステップから呼び出された場合、この関数は、このアイルをその都度移動、およよ、名前を変更します。FlowForceを使用して複数のファイルを移動する場合、move 関数を呼び出すステップを「for-each」ステップ内でファイルのエピーサンプル内で説明されているとおり、囲んでくたさい。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ウメーターの値により以下のように異なります。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」パラメーターがFALSE の場合、関数は、FALSE を返します。

1	<u>بر</u>	-4
•	·//	

名前	型	説明
у — д	string as file	移動するノースファイルのマとファイル名。
保存先	string as file	保存先ディレケリの名前。このフィールドに ディレケリ名のみが与えられている場合、 元の名前が保持されます。
ターゲットの上書き	boolean	true の場合は、保存先ファイルが上書き されます。デフォルトの値はfalse です。
エ ラー時 には中断する	boolean	このブール ラメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
作業ディレクトリ	string as directory	作業ディレケリを指定します(例えば c: \somedirectory)。相対的ないなか使 用されると、作業ディレケリに対して解決さ れます。

12.3.5 rmdir

フルな:/System/filesystem/rmdir

デルケリの削除。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ウメーターの値により以下のように異ないます。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSEの場合、関数は、FALSEを返します。

名前	型	説明
バス	string as directory	削除するファイルのファイル名。
エ ラー時 にコま中断する	boolean	このブール ウメーターはショブが失敗した場合の戻り値を決定します。「エラー時には中断」がFALSE の場合、関数は、ブー ル値のFALSE を返します。「エラー時には中断」がTRUE の場合、ジョブの実行は中断されます。デフォルトの値は、

名前	型	説明
		TRUE です。
作業ディレクトリ	string as directory	作業ディレクトリを指定します、例えば c: \somedirectory)。相対的ないなか使 用されると、作業ディレクトリニズルて解決さ れます。

12.4 /system/ftp

/system/ftp コンテナーゴはFTP おゴはFTPS サーバー 接続し、ファイルをアップロード、抽出、削除、おゴはノモートディレクトリと他を 作成おゴゴ削除するゴッジコ使用される関数が含まれています。

FTP 関数内の ぷ

FTP 関数の一部コま作業ディレクトリップメーターが存在します。この、デメーターはFlowForce 関数の多くで共通です。作業ディレクトリを設定するコま FTP 関数に デメータとて提供されるすべてのローカル なか絶対的 でおよく相対的であることに注意してくたさい。 ファイルをアップロードするコま作業ディレクトリとして C:\Upload とソースファイルとして file.txt を入力することができます。最後のノースパスは C:\Upload \file.txt 「よいます。

更に、FTP 関数の一部はJモート FTP サーバー上で作業ディレクトリと同じ役割を果たすホスト上のディレクトリ・ラメーターを有します。このためコホスト上のディレクトリを設定するコは、FTP 関数にン ラメータとして提供されるすべてのローカル なが絶対的ではなく相対的であることご注意してください。例えば、ファイルをアップロードするコはホスト上のディレクトリとしてuploadsとターゲットファイルとして file.txtを入力することができます。最後のターゲットパスは/uploads/file.csv になります。

ファイルシステムトリガー おけま<u>HTTPトリガー</u>としてジョブを構成することも重要です。このようよジョブおショブをトリガーしたファイルの やを提供する triggerfile ノ デメーターを有します。FTP 関数内で triggerfile ノ デメーターを使用する場合、その やれま *絶対的であることを*忘 れないでください。

triggerfile から拡張子を持つファイル名を取得するコよ 次のFlowForce 式を使用してくたさい

{filename-with-extension(triggerfile)}

サンプルに関してはFTP store 関数を参照してくたさい。

FTP 関数内のフイルドカード

以下のFTP 関数はフイルドカードを デメーターとて受け入れます

- /system/ftp/delete-wildcard
- /system/ftp/retrieve-wildcard
- /system/ftp/store-wildcard

このような関数を使用する場合、専用のワイルドカード・パラメーター内にフイルドカードを入力します。使用可能なワイルドカードは以下の通りです

ワイルドカード	使用法	サンプル
*	ゼロおうはより多くの文字を一致します。	*.htm はhome.htmとindex.htm を一致します。
?	単一の文字を一致する。	*.xm? はindex.xmlとproject.xmi に一致します。

ワイルドカード + (1つまけは複数) はサポートされています。代わりに?*を使用して同じ効果を得ることができます。例えば、*.c?* は.cs、.cpと.csproj ファイルに一致しますが.c ファイルコは一致しません。

12.4.1 delete

フルな:/system/ftp/delete

FTP サーバーからファイルを削除します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ラメーターの値により以下のように異ないます。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」パラメーターがFALSEの場合、関数は、FALSEを返します。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいは絶対的であることを忘れないでくたさい。サンプルは関してはFTP store 関数を参照してくたさい。

名前	型	説明
FTP サー バー	string	URL おはIP バレスとての圧ト FTP サーバーのバレス
		必須の ラメーターです。
*-+	number	FTP サーバー:接続するけっかっ使用される ポート番号です。デフォルトの値は21 で す。
ホスト側のディレクトリ	string	ファイルを削除するホスト上のディレクトリの 名前です。
		任意の・ラメーターです。
ログインの資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが視覚情報を必要とない 場合、この デメーターをスキップします。
受動モードを使用する	boolean	接続エラーが発生すると、パッシブモードを 使用します、例えば、ルーターおけはファイア ウォールがアクティブは接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の ラメーターです。デフォルトの値は いえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をいいえに設定しま す。それ以外の場合、この値を以下に設定 します:

 暗号化されたコマンド・ ルと共に明示的 暗号化されたコマンドと チャンネルと共に明示的 	チャンネ トデータ り マチると マメーむ 証 されま
上の2つのオプションの、ずれかを設 証明書は下で説明されて、るサー	証されま
	訪され
す、FlowForce 内で されていません。	ᡃᢆᢧᡯ᠆ᢣ
サーバー証明書の検証 string 任意の、ウメーターです。FlowFor のようにFTP サーバーの証明書を るか指定します。有効な値:	rce かど 検証す
 検証無し - 全ての証明 け入れます。 証明書ストアに対して まず(デフォルの値) - W indows ではどう交 のユーザーアサウトの値) W indows ではどう交 のユーザーアサウトの面 トアとステムストアを注明 名を検証するとがは明 名を検証するとがは 書ストアを使用にます。。 通常 /usr/lib/ssl/certs SSL_CERT_FILE と SSL_CERT_FILE と SSL_CERT_FILE を SSL_CERT_FILE を SSL_CER	唐 検 実明明肌シンnt.対 観す、ていに証 研装改をしていたまでごいとなった。 しているいでは、「おいい」では、「おいい」では、「おいい」では、「おいい」では、「ないい」では、「ないい」では、「 いい しょう
サーバーの証明書 Certificate イ注意の、ラメーターです。FlowFor 証明書オブシェクトへの、なを指定 前の、ラメーターを選択されたサー 明書に対して検証に設定してい、 指定されているFlowForce 証明 FTP サーバー証明書に対して検証 す。それ以外の場合、この、ラメー	- ce 内の します。 マン マン はま しま しま しま しま しま

名前	型	説明
		無視されます。
ターゲットファイル	string	サーバーかり削除されるファイル名です。
		必須のゲメーターです。
エラー時には中断する	boolean	このブール デメータ ばショブが失敗した場合の戻り値を決定します。「エラー時には中断」がFALSE の場合、関数は、ブール値のFALSE を返します。「エラー時には中断」がTRUE の場合、ジョブの実行は中断されます。デフォルトの値は、TRUE です。
ፖክታታት	string	リモトサードー上でアクセスすることのでき る ユーザ か許可されている FTP アカウ ント名です。 任意の ラメーターです。

12.4.2 delete-wildcard

フル☆ /system/ftp/delete-wildcard

FTP サーバーからワイルドカードに一致するファイルを削除します。例、*.xml。成功すると、関数に削除されたファイルのリスト(パスの無いファイル名)を返すませま、一致が見つからえ、場合は空のリストを返します。実行に失敗すると、結果におい下で説明されているエラー時に中断ノウメームとと異なります。

FTP 関数内でtriggerfile、デメーターを使用する場合、その、なは絶対的であることを忘れないでください。サンプルゴ関してはFTP store 関数を参照してください。

名前	型	説明
FTP サーバー	string	リモート FTP サーメーのボレス。 URL ま たはIP ブドレスの、ずれかです。
		必須の ラメーターです。
*-+	number	FTP サーバーは接続するなめに使用される ポート番号デンォルトの値は21 です。
ホスト上のディレクトリ	string	ファイルを削除するヤスト上のディレクトリの 名前
		任意の「ラメーターです。
ログイン資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。

名前	型	説明
		FTP サーバーが資格情報を必要とない 場合、この・ラメーターをスキップしてくたさい い。
受動モードの使用	boolean	接続問題が発生すると受動モードを使用し てくたさい(例えば、ルーターおけはファイア ウォールがアクティブは接続を回避するようこ セットアップされている場合)。
SSL/TLS の使用	string	(任意の ラメーターです。デフォハトの値は いいえです。)情報を転送するはまFTP はコマ・ドチャンネルビデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をいいえ(設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネ ルと共に明示的 暗号化されたコマンドとデータ チャンネルと共に明示的
		上の2つのオブションの、ずれかを設定すると 証明書は下て説明されているサーバー証 明書の検証・パラメーターに従い検証されま す。
		ノモ 明示的な暗号化は奨励され ず、FlowForce内でサポート されていません。
サーバ - 証明書の検証	string	任意の、ラメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値: ・ 検証無し - 全ての証明書を受 け入れます。 ・ 証明書ストアに対して検証し ます(デフォルトの値)- Windows ではジョブを実行中 のユーザーアカウトの証明書ス トアとステムストアを証明書署 名を検証するよがは使用してくだ さい、Linux では、システム証明 書ストアを使用します。これらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certs まけは SSL_CERT_FILE と SSL_CERT_FILE と

名前	型	説明
		- FlowForce はサーバー証 明書 パテメーター内で指定され ている FTP サーバーの証明書と 比較します。
		この、ラメーターの使用はサー、・証明書と 安全な接続が必要です。安全な接続を構 築できない場合、FTP 関数は失敗しま す。
サーバーの証明書	certificate	任意の ウメーターです。FlowForce 内の 証明書オブジェク への やを指定します。 前の ウメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されている FlowForce 証明書は FTP サーバー証明書に対して検証されま す。それ以外の場合、この ウメーター値は 無視されます。
ワイルドカード	string	必須の ウメーターです。 ワイルドカードを指 定します。 例、 * . xm1。 ワイルドカードに一 致するファイルは削除されます。 <u>FTP 関数</u> <u>内のフイルドカード</u> を参照してくたさい。
エ ラー時 には中断する	boolean	実行に失敗するとこの、ラメーターは関数の 振る舞しを指定します。具体的には、実行 が失敗した場合、関数には以下のしつを返し ます:
		 エラー時に中断パラメーターが falseの場合、削除されこファイ ルのノストは返されません。 エラー時に中断パラメーターが trueの場合、失敗が発生する まで関数はファイルを削除し、実 行を中断します。
		このために、ファイルの一部は実行か失敗し ても削除される場合かあります。
ፖታታ	string	リモートサーバー上のファイルニアクセスを許可されているユーザーのFTP アカウト名
		任意の・ラメーターです。

12.4.3 mkdir

אר אד:/system/ftp/rmdir

FTP サーバー上にディレクト」を作成します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ウメーターの値により以下のように異ないます。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSE の場合、関数は、FALSE を返します。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいれ、絶対的であることを忘れないでくたさい。サンプルは関してはFTP store 関数を参照してくたさい。

名前	型	説明
FTP サー バー	string	URL おはIP バレスとての圧ト FTP サーバーのバレス
		必須の ラメーターです。
*-+	number	FTP サーバーは接続するために使用される ポート番号です。デフォルトの値は21 で す。
ホスト側のディレクトリ	string	ホスト上で新規のデルクトリを作成するディ レクトリ名です。
		任意のマテメーターです。
ログインの資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが視覚情報を必要としない 場合、このパラメーターをスキップします。
受動 モード を使用する	boolean	接続エラーが発生すると、パッシブモードを 使用します(例えば、ルーターおけはファイア ウォールがアクティブな接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の ラメーターです。デフォルトの値は いえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をしいえに設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネルと共に明示的 暗号化されたコマンドとデータチャンネルと共に明示的
		上のソンのオフションのようれかを設定すると

名前	型	説明
		証明書は下て説明されているサーバー証 明書の検証・デメーターに従い検証されま す。
		メモ 明示的な暗号化は奨励され ず、FlowForce内でサポート されていません。
サーバー証明書の検証	string	任意の ラメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値
		 検証無し - 全ての証明書を受 け入れます。 証明書ストアに対して検証し ます(デンオ)トの値) - Windows ではジョブを実行中 のユーザーアカウトの証明書ス トアとンステムストアを証明書署 名を検証するよがは使用してくだ さい、Linux では、システム証明 書ストアを使用します。こわらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certs おけは SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントする/文レコシルます。 サーバー証明書に対して検証 -FlowForce はサーバー証 明書, プンーター内で指定され ているFTP サーバーの証明書と 比較します。
		安全な接続が必要です。安全な接続を構築できな、場合、FTP 関数は失敗します。
サーバーの証明書	certificate	任意の、ラメーターです。FlowForce内の 証明書オブジェクトへの、文を指定します。 前の、ラメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されているFlowForce証明書は FTPサーバー証明書に対して検証されま す。それ以外の場合、この、ラメーター値は 無視されます。
ターゲットディレクトリ	string	作成するデルクトリ名です。
		必須の、ラメーターです。
エ ラー時 には中断する	boolean	このブール ウメーターはショブが狭敗した場 合の戻り値を決定します。 「エラー時には

名前	型	説明
		中断」がFALSE の場合、関数は、ブー ル値のFALSE を返します。「エ フ・時 に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
ፖታታ	string	リモトサッドー上でアクセスすることのでき る ユ ーザ か許可されているFTP アカウ ント名です。 任意の ウメーターです。

12.4.4 move

אנ:/system/ftp/retrieve

FTP サーバーからファイルを移動します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異なります。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSEの場合、関数は、FALSEを返します。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいは絶対的であることを忘れないでくたさい。サンプルは関してはFTP store 関数を参照してくたさい。

名前	型	説明
FTP サ━ッシ━	string	URL おはIP バレスとてのJモート FTP サーバーのバレス
		必須の┝メーターです。
*-+	number	FTP サーバー:接続するために使用される ポート番号です。デフォルトの値は21 で す。
ホスト側のディレクトリ	string	ファイルを移動するホスト上のディレクトリの 名前です。
		任意の・ラメーターです。
ログインの資格情報	credential	FlowForce 資格情報レコードとての FTP アカウトのユーザー名と スワードで す。 <u>資格情報</u> を参照してくたこい。

名前	型	説明
		FTP サーバーが視覚情報を必要とない 場合、このパラメーターをスキップします。
受動 モード を使用する	boolean	接続エラーが発生すると、パシンプモードを 使用します(例えば、ルーターおイオンアイア ウォールがアクティブは接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の ウメーターです。デフォルトの値は しいえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をしいえに設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネ ルと共に明示的 暗号化されたコマンドとデータ チャンネルと共に明示的
		上の2つのオプションの、ずれかを設定すると 証明書は下て説明されているサーバー証 明書の検証、パラメーターイン従い検証されま す。
		メモ 明示的な暗号化は奨励され ず、FlowForce 内で サポー されていません。
サーバー証明書の検証	string	任意の ラメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値
		 検証無し-全ての証明書を受け入れます。 証明書ストアに対して検証します(デンオ)小の値)- Windows ではジョブを実行中のユーザーアカウトの証明書ストアとンステムストアを証明書署 名を検証するよがは使用してくだささ、Linux では、システム証明書ストアを使用します。これらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certsまけは SSL_CERT_FILE と SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントする。気にあります。 サーバー証明書に対して検証 -FlowForce はサーバー証明書、デメーター内で指定され

名前	型	説明
		ているFTP サーイーの証明書と 比較します。
		この、ラメーターの使用はサーバー証明書と 安全な接続が必要です。安全な接続を構 築できない場合、FTP 関数は失敗しま す。
サーバーの証明書	certificate	任意の ちメーターです。FlowForce 内の 証明書オジェクへの なを指定します。 前の ちメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されている FlowForce 証明書は FTP サーバー証明書に対して検証されま す。それ以外の場合、この ちメーター値は 無視されます。
ソースファイル	string	移動するデルケー名です。
		必須の デメーターです。
ターゲットファイル	string	ターゲオの場所でのコピーされナファイルの 名前です。コピーされナファイルの名前を変 更する場合、異なる名前を使用してくたさい。
		必須の♪ラメーターです。
エ ラー時 にコま中断する	boolean	このブール デメーターはジョブが失敗した場合の戻い値を決定します。「エラー時には中断」がFALSEの場合、関数は、ブー ル値のFALSEを返します。「エラー時には中断」がTRUEの場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUEです。
アカウント	string	リモトサーバー上でアクセスすることのでき るユーザーが許可されているFTP アカウ ント名です。
		社息リノマメージーじゅ。

12.4.5 list

フルな /system/ftp/list

FTP サーバー上のディレクトリのエレテンンをリストします。実行に成功するとこの関数は文字列のシーケンスを返します。それ以外の場合、結果は以下に説明される通りエラー時に実行を中断ノデメーターにより異なります。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいれ、絶対的であることを忘れないでくたさい。サンプルニ関しては

FTP store 関数を参照してくたさい。

名前	型	説明
FTP サーーン゙ー-	string	リモート FTP サーバーのボレス。 URL ま たはIP ブドレスの、ずれかです。
		必須の♪ラメーターです。
ポ ー ト	number	FTP サーバーは接続するながに使用される ポート番号デンオルトの値は21 です。
ホスト上のディレクトリ	string	コンテンンをリストするホスト上のディレクトリ の名前
		任意の ラメーターです。 デフォルトの値は 現在のディレオトリ, "/" です。
ログイン資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが資格情報を必要としない 場合、この デメーターをスキップしてくたさ い。
受動モードの使用	boolean	接続問題が発生すると受動モードを使用してください(例えば、ルーターおけはファイア ウォールがアクティブは接続を回避するようこ セットアップされている場合)。
		任意の 「メーター です。 デフォルトの値は true です。
SSL/TLS の使用	string	(任意の、ラメーターです。デフォルトの値は しいえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をしいえに設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネルと共に明示的 暗号化されたコマンドとデータチャンネルと共に明示的
		上の2つのオブションの、ずれかを設定すると 証明書は下て説明されているサーバー証 明書の検証 パラメーターに従い検証されま す。

名前	型	説明
		メモ 明示的な暗号化は奨励され ず、FlowForce 内で サポト されていません。
サーバー証明書の検証	string	任意の ウメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値
		 検証無し - 全ての証明書を受 け入れます。 証明書ストアに対して検証し ます(デンオ)トの値) - Windows ではジョブを実行中 のユーザーアカウトの証明書ス トアとンステムストアを証明書署 名を検証するかが1支用してくだ さいしいいいでは、システム証明 書ストアを使用します。こわらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certs おけよ SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントする いつかけます。 サーバー証明書に対して検証 -FlowForce はサーバー証 明書,ウメーター内で指定され ているFTP サーバーの証明書と 比較します。 この、ウメーターの使用はサーバー証明書と 安全な接続が必要です。安全な接続を構 築できない場合、FTP 関数は失敗しま す
サーバーの証明書	certificate	任意の、ラメーターです。FlowForce内の 証明書オブジェクトへの、文を指定します。 前の、ラメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されているFlowForce証明書は FTPサーバー証明書に対して検証されま す。それ以外の場合、この、ラメーター値は 無視されます。
ワイルドカード	string	ホスト上のディレクトリ・ウメーターにより指定されるディレクトリからすべての js ファイルを抽出する必要がある場合使用するワイルドカード文字列は、例えば*.js です。 任意の、ウメーターです。デフォルトの値はフィルドカードフィルターが実行されたいことを

名前	型	説明
エラー時 にコま中断する	boolean	この、ラメーターは実行が失敗すると何か起 こるかを管理します。実行に失敗するとエ ラー時に中断がtrue になり、ジョブ実行 が中断され、保護されたプロックを使用して エラーを処理できます。実行に失敗すると エラー時に中断がfalse で、関数は空 のシーケンスを返します。 任意の、ラメーターです。デフォハトの値は、 true です。
ፖታታ	string	リモートサーバー上のファイルニアクセスを許 可されているユーザーのFTP アカウナト名 任意の・ラメーターです。

12.4.6 retrieve

フルな:/system/ftp/retrieve

FTP サーバーからファイルを抽出します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異なります。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: ステップのエラーの処理)。
- 「エラー時には中断」パラメーターがFALSEの場合、関数は、FALSEを返します。

FTP 関数内でtriggerfile、ラメーターを使用する場合、そのパスは絶対的であることを忘れないでください。サンプルに関してはFTP store 関数を参照してください。

名前	型	説明
FTP サー バー	string	URL おはIP バレスとてのJモート FTP サーバーのバレス
		必須の┝メーターです。
*-+	number	FTP サーバー 接続するために使用される ポート番号です。デフォルトの値は21 で す。
ホスト側のディレクトリ	string	ファイルを抽出するホスト上のディレクトリの 名前です。
		任意の ラメーターです。

名前	型	説明
ログインの資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが視覚情報を必要としない 場合、この デメーターをスキップします。
受動 モード を使用する	boolean	接続エラーが発生すると、パッシブモードを 使用します(例えば、ルーターおイオファイア ウォールがアクティブな接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の ラメーターです。デフォルトの値は しいえです。)情報を転送するコはFTP はコマ・ドチャンネルとデータチャンネルを使 用します。FTP デーダを暗号化無しこ転 送する場合、この値をいいえし設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネルと共に明示的 暗号化されたコマンドとデータチャンネルと共に明示的
		上の2つのオプションの、ずれかを設定すると 証明書は下で説明されているサーバー証 明書の検証 パラメーターに従い検証されま す。
		ノモ 明示的な暗号化は奨励され ず、FlowForce 内で サポー されていません。
サーバー証明書の検証	string	任意の デメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値:
		 検証無し-全ての証明書を受け入れます。 証明書ストアに対して検証します(デンオ)トの値)- Windows ではジョブを実行中のユーザーアカウナルの証明書ストアとシステムストアを証明書署名を検証するかが1支用してください。Linux では、システム証明書ストアを使用します。これらは通常/usr/lib/ssl/cert.pemと/usr/lib/ssl/certsまけるはSSL_CERT_FILEとSSL_CERT_FILEとSSL_CERT_DIR環境関数が

		で削
ፖታታ	string	児 る ン
		任意
(0) 2015 2021 Alton Crahl		

名前	型	説明
		ポイトする いっかます。 ・ サーバー証明書に対して検証 - FlowForce はサーバー証 明書 パラメーター内で指定され ているFTP サーバーの証明書と 比較します。
		この、ラメーターの使用はサーバー証明書と 安全な接続か必要です。安全な接続を構 築できない場合、FTP 関数は失敗しま す。
サーバーの証明書	certificate	任意の ちメーターです。FlowForce 内の 証明書オジェクへの なを指定します。 前の ちメーターを選択されたサーバー証 明書に対して検証に設定して る場合、 指定されている FlowForce 証明書は FTP サーバー証明書に対して検証されま す。それ以外の場合、この ちメーター値は 無視されます。
ソースファイル	string	抽出するデルケー格です。
		必須の、テメーターです。
ターゲットファイル	string	抽出後のファイル名です。
		必須の、テメーターです。
ターゲットの上書き	boolean	true の場合は、保存先ファイルが上書き されます。デフォルトの値はfalse です。
エ ラー時 には中断する	boolean	このブール デメーターはジョブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
作業ディレクトリ	string	ジョブの作業ディレオリを指定します、例え ば、c:\somedirectory)。相対的な パマが使用されると、作業ディレオリニズ扎 て解決されます。
ፖታታ	string	リモトサーバー上でアクセスすることのできる ユーザーが許可されている FTP アカウント名です。
		任意の デメーターです。

ビルトイン関数

12.4.7 retrieve-wildcard

フル次 /system/ftp/retrieve-wildcard

FTP サーバーからワイルドカードに一致するファイルを抽出します。例、*.xml。成功すると、関数は書き込まれたファイルに絶対ローカルパス のノストを返すませるよ 一致が見つからない場合は空のノストを返します。実行に失敗すると、結果にお以下で説明されている エラー時に 中断 パラメーターにお興ないます。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいれ、絶対的であることを忘れないでくたさい。サンプルは関してはFTP store 関数を参照してくたさい。

1	₩-	_5_
	11	

名前	型	説明
FTP サー バー	string	リモト FTP サーバーのボレス。URLま たはIP ブドレスの、ずれかです。
		必須の┝メーターです。
*-+	number	FTP サーバーは接続するながに使用される ポート番号デンォルトの値は21 です。
ホスト上のディレクトリ	string	ファイルを削除するホスト上のディレクトリの 名前
		任意の ラメーターです。
ログイン資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが資格情報を必要としない 場合、この デメーターをスキップしてくたさ い。
受動モードの使用	boolean	接続問題が発生すると受動モードを使用し てくたさい(例えば、ルーターおイコンアイア ウォーリのアクティブは接続を回避するようこ セットアップされている場合)。
SSL/TLS の使用	string	(任意の、ラメーターです。デフォルトの値は しいえです。)情報を転送する」コよFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無しに転 送する場合、この値をしいえに設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャシネ ルと共に明示的 暗号化されたコマンドとデータ チャンネルと共に明示的

名前	型	説明
		上の2つのオプションの、ずれかを設定すると 証明書は下で説明されているサーバー証 明書の検証・デメーターに従い検証されま す。 メモ 明示的な暗号化は奨励され ず、FlowForce内でサポー
		されていません。
サーバー証明書の検証	string	任意の、ラメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値 ・ 検証無し - 全ての証明書を受 け入れます。 ・ 証明書ストアに対して検証し ます(デフォルトの値)- Windows ではジョブを実行中 のユーザーアナウストの証明書ス トアとステムストアを証明書署 名を検証するよがは使用してくだ さい、Linux では、システム証明 書ストアを使用します。これらは 通常/usr/lib/ssl/cert.pem と/usr/lib/
サーバーの証明書	certificate	任意の、ラメーターです。FlowForce内の 証明書オブジェクトへの、なを指定します。 前の、ラメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されているFlowForce証明書は FTPサーバー証明書に対して検証されま す。それ以外の場合、この、ラメーター値は 無視されます。
ワイルドカード	string	必須の ウメーター です。 ワイルドカードを指 定します。 例、 <mark>* . xm1</mark> 。 ワイルドカードに一

名前	型	説明
		致するファイルが抽出されます。 <u>FTP 関</u> 数内のフイルドカードを参照してくたさい。
エ ラー時 には中断する	boolean	実行に失敗するとこの、ラメーターは関数の 振る舞しを指定します。具体的には、実行 が失敗した場合、関数は以下のしつを返し ます:
		 エラー時に中断・デメーターが falseの場合、抽出されたファイ ルのノストは返されません。 エラー時に中断・デメーターが trueの場合、失敗が発生する まで関数はファイルを抽出し、実 行を中断します。
		このために、ファイルの一部は実行が失敗しても抽出される場合があます。
作業ディレクトリ	string	FTP サーバーから抽出されるすべてのファイ ルのあるディレクトリ
ፖታታ	string	リモトサーバー上のファイルニアクセスを許 可されているユーザーのFTP アカウト名 任意の ウメーターです。

12.4.8 rmdir

フル次:/system/ftp/rmdir

FTP サーバーからディレクトリを削除します。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ラメーターの値により以下のように異なります。

- 「エラー時には中断」 パラメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSE の場合、関数は、FALSE を返します。

FTP 関数内でtriggerfile、デメーターを使用する場合、そのいは絶対的であることを忘れないでください。サンプルは関してはFTP store 関数を参照してください。

名前	型	説明
FTP IJ —у ў—	string	URL おはIP バレスとての圧ト FTP サードのバレス

名前	型	説明
		必須の┝メーターです。
*-+	number	FTP サーバー:接続するために使用される ポート番号です。デフォルトの値は21 で す。
ホスト側のディレクトリ	string	ファイルを削除するホスト上のディレクトリの 名前です。
		任意の、ラメーターです。
ログインの資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが視覚情報を必要としない 場合、このパラメーターをスキップします。
受動 モード を使用する	boolean	接続エラーか発生すると、パッシブモードを 使用します(例えば、ルーターお」はファイア ウォールがアクティブな接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の、ラメーターです。デフォルトの値は しいえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無して転 送する場合、この値をしいえて設定しま す。それ以外の場合、この値を以下に設定 します:
		 暗号化されたコマンドチャンネルと共に明示的 暗号化されたコマンドとデータチャンネルと共に明示的
		上の2つのオブションの、すれかを設定すると 証明書は下て説明されているサーバー証 明書の検証 パラメーターに従い検証されま す。
		ノモ 明示的な暗号化は奨励され ず、FlowForce 内でサポート されていません。
サーバー証明書の検証	string	任意の ラメーターです。FlowForce かど のようこFTP サーバーの証明書を検証す るか指定します。有効な値:
		 検証無し - 全ての証明書を受け入れます。 証明書ストアに対して検証します(デンオ)トの値) -

名前	型	説明
		Windows ではジョブを実行中 のユーザーアかつたの証明書ス トアとシステムストアを証明書署 名を検証するために使用してくださ さい。Linux では、システム証明 書ストアを使用します。これらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certs おけは SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントする。文ロンあります。 ・サーバー証明書に対して検証 -FlowForce はサーバー証 明書、ウメーター内で指定され ているFTP サーバー証明書と 比較します。 この、ウメーターの使用はサーバー証明書と 安全な接続が必要です。安全な接続を構 築できれ、場合、FTP 関数は失敗しま す。
サーバーの証明書	certificate	任意の ウメーターです。FlowForce 内の 証明書オブジェクト への 文を指定します。 前の ウメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されている FlowForce 証明書は FTP サーバー証明書に対して検証されま す。それ以外の場合、この ウメーター値は 無視されます。
ターゲットディレクトリ	string	削除するデルノトリ名です。 必須の らメーター です。
エ ラー時 には中断する	boolean	このブール ウメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
アカウント	string	リモトサーバー上でアクセスすることのでき る、ユーザーが許可されているFTP アカウ ント名です。 任意の・ラメーターです。

12.4.9 store

אל:/system/ftp/store

FTP 削除ーバーファイルをアップロードします。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、ラメーターの値により以下のように異ないます。

- 「エラー時には中断」、ウメーターが TRUE (デフォルトの値)の場合、ジョブの実行は中断されます。この場合、エラーを保護されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSE の場合、関数は、FALSE を返します。

名前	型	説明
FTP サ━ッシ━	string	URL おはIP バレスとての圧ト FTP サーバーのバレス
		必須の ラメーターです。
*-+	number	FTP サーバー 接続するかが 使用される ポート番号です。 デフォルトの値は 21 で す。
ホスト側のディレクトリ	string	ファイルを保管するホスト上のディレクトリの 名前です。
		任意の・テメーターです。
ログインの資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してくたさい。
		FTP サーバーが視覚情報を必要としない 場合、このパラメーターをスキップします。
受動モードを使用する	boolean	接続エラーが発生すると、パッシブモードを 使用します、例えば、ルーターおけよファイア ウォールがアクティブな接続を回避するようこ 設定されている場合)。
SSL/TLS 暗号化の使用	string	(任意の ラメーターです。デフォルトの値は いくえです。)情報を転送するコはFTP はコマボチャンネルとデータチャンネルを使 用します。FTP データを暗号化無して転 送する場合、この値をいいえて設定しま す。それ以外の場合、この値を以下に設定 します: ・ 暗号化されたコマバチャンネ
		ルと共に明示的

名前	型	説明
		・ 暗号化されたコマンドとデータ チャンネルと共に明示的 上の2つのオプションの、ずれかを設定すると 証明書は下で説明されているサーバー証 明書の検証、パラメーターに従い検証されま す。
		ず、FlowForce内でサポート されて、ません。
サーバー証明書の検証	string	任意の、ラメーターです。FlowForce かどのようこFTP サーバーの証明書を検証するか指定します。有効な値: ・ 検証無し - 全ての証明書を受け入れます。 ・ 証明書ストアに対して検証し
		ます(デオハトの値)- Windows ではジョブを実行中 のユーザーアカウトの証明書ス トアとンステムストアを証明書署 名を検証するかかご使用してくださ さい、Linux では、システム証明 書ストアを使用します。これらは 通常/usr/lib/ssl/cert.pem と/usr/lib/ssl/certs おけよ SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントするパンマントます。 サーバー証明書に対して検証 -FlowForce はサーバー証 明書、デメーター内で指定され ているFTP サーバーの証明書と 比較します。
		安全な接続が必要です。安全な接続を構築できな、場合、FTP 関数は失敗します。
サーバーの証明書	certificate	任意の、ラメーターです。FlowForce内の 証明書オブジェクトへの、文を指定します。 前の、ラメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されているFlowForce証明書は FTPサーバー証明書に対して検証されま す。それ以外の場合、この、ラメーター値は 無視されます。

名前	型	説明
ソースファイル	string	FTP サーバーニアップロードされるファイル 名です。
		必須の ラメーターです。
ターゲットファイル	string	FTP サーバーへのアップロード後のファイル 名です。これはソースファイルとは異なる場 合かあります。
		必須の ラメーターです。
エラー時 にコま中断する	boolean	このブール テメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。
作業ディレクトリ	string	ジョブの作業デルクトリを指定します(例え ば、c:\somedirectory)。相対的な パマカ使用されると、作業デルクトリーマオし て解決されます。
ፖታታ	string	リモトサッドー上でアクセスすることのでき る ユーザ か許可されている FTP アカウ ント名です。 任意の ウメーターです。

サンプル

このサンプルはショブ構成時にファイル名と拡張子を知ることなく、リモート FTP サーバーにファイルをアップロードする方法を説明しています。これは下記の通りファイルシステムトリガーとしてジョブを実行するように構成することにより可能にないます。

Tr	iggers				
	Check	Content	✓ of file or directory:	C:\FlowForce\Upload	polling interval: 60 seconds.
	Start:	+			
	Expires:	+			
	Time zone:	Europe/Berlin	-		
	🗹 enabled	I			

上記のトリガーはデルノケトリC:\FlowForce\Uploadの変更をモニターします。ファイルをこのデルノケーリニピーすると、ジョブが実行され、ジョブをトリガーしたファイルの絶対・ぷがtriggerfile入カ・ラメーター内で使用できるようゴなります。これにより下記のとおりジョブ内でこのファイルを名前および拡張子を知ることなく使用できるようゴムります。

Job Input Para	meters			
• Name: triggerfile •	Type: str	ing 🔽 Default: 💼		
Execution Steps				
Execute funct	ion /system/ftp/stor	e		
Parameters:	FTP Server: Port: Directory on host:	10.100.63.200 21 uploads		
	Login credentials:	 Select existing credential: /public/my.ftp.credentials Define local credential: 		
	Use passive mode:	: +		
	Source file:	{filename-with-extension(triggerfile)}		
	Target file:	{filename-with-extension(triggerfile)}		
	Abort on error:	•		
	Working directory:	C:\FlowForce\Upload		
	Account:	+		

上記のジョブ構成ではstore 関数は以下の デメータと共に呼び出されます

- FTP サーバー-FTP サーバーのアドレス(このサンプルの場合 I.P. アドレス)
- ポート デフォルトのポートは21 です。
- ホスト上のディレクトリーこのサンプルでは、アップロードされたすべてのファイルをFTP ルートディレクトリニ相対するサーバー上の 「アップロード」サブディレクトリニ置きます。
- ログインの資格情報 FTP ユーザー名と マワードが FTP サーバー 接続する コンシ要です。 再利用の目的のために 資格 情報 として以前に定義されており、 ここでは public コンテナーから参照されています。
- ソースファイル アップロードされるローカルファイルのパマです。作業パマカ設定されているため、このサンプルではこれは相対的なパマである必要があります。
- ターゲットファイル アップロード後のFTP サーバー上のファイルの みです。このサンプルではホスト上のディレクトリ か設定され ているため、これは相対的ない みである必要があります。
- 作業デルクトリーローカルコンピューター上のデルクトリです。ローカルの相対パスはこのデルクトリニ対して相対的であると仮定されます。トリガー内で定義されているポーレクデルクトリと同じであることに注意してくたさい。

作業ディレクトリカ設定されていると、ソースファイルは絶対的ではなく相対的である必要かあります。同様にホスト上のディレクトリカ設定されていると、ターゲットファイルは相対的ない、次である必要かあります。

このサンプルは作業ディレクトリとホスト上のディレクトリを使用するためtriggerfileの絶対的パスを相対的パスで変更する必要があります。

これを達成するためにソースファイルはFlowForce 式を使用します。この式はtriggerfile を引数(これは絶対)なてあることご注意してください)として取りファイル名と拡張子のみを返します。例えば、triggerfile がC:\data.txt の場合、式はdata.txt のみを返します。ターゲットファイル内の式を使用しても同様に作動します。FlowForce 内の式の詳細に関しては、The FlowForce 式言語を参照してください。

上記の構成では、(拡張子に関わらず)ファイルを作業ディレクトリニピーすると以下が発生します:

- data.txt と呼ばれるファイルカロピーされた上仮定すると、ジョブが実行され C:\FlowForce\Upload\data.txt を triggerfile として取得します。
- 式を使用することでノースファイルはdata.txt により、ターゲットも同様にまます。
- アップロードするファイルの実際のマは作業ディレクトリをソースファイルと結合することによ取得されます。
- サーバー上のファイルの目的パマはホスト上のディレクトリをターゲホファイルと結合することに以取得されます。
- FlowForce は提供済みのFTP 資格情報は接続しようと試みます。成功するとファイル data.txt を FTP サーバー上のアップロード ディレクトリニアップロードします。

12.4.10 store-wildcard

フルゲ /system/ftp/store-wildcard

ワイルドカードに一致する場合、FTP サーバーファイルをローカルディレオ・しからアップロードします。例、*.xml。成功すると、関数はアップ ロードされたファイル総対ローカルののの人を返すおまし、一致が見つからない場合は空のノストを返します。実行に失敗すると、結果は 以下で説明されているエラー時に中断ノウメーターにお興ないます。

FTP 関数内でtriggerfile、ラメーターを使用する場合、その、マは絶対的であることを忘れないでくたさい。サンプルゴ関してはFTP store 関数を参照してくたさい。

名前	型	説明
FTP サーーン゙ー	string	リモート FTP サーバーのボレス。 URL ま たはIP ブドレスの、ずれかです。
		必須の♪テメーターです。
*-	number	FTP サーバーは接続するなめに使用される ポート番号デンォルトの値は21 です。
ホスト上のディレクトリ	string	ファイルを削除するホスト上のディレクトリの 名前
		任意の、ラメーターです。
ログイン資格情報	credential	FlowForce 資格情報レコードとしての FTP アカウトのユーザー名と ペワードで す。 <u>資格情報</u> を参照してください。
		FTP サーバーか資格情報を必要としない 場合、このパラメーターをスキップしてくたさい い。

名前	型	説明
受動モドの使用	boolean	接続問題が発生すると受動モードを使用し てくたさい(例えば、ルーターおイオンアイア ウォールがアクティブは接続を回避するようこ セットアップされている場合)。
SSL/TLS の使用	string	(任意の、ラメーターです。デフォルトの値は いくえです。)情報を転送するはまFTP はコマベチャンネルとデータチャンネルを使 用します。FTP データを暗号化無して転 送する場合、この値をいいえに設定しま す。それ以外の場合、この値を以下に設定 します: ・ 暗号化されたコマンドチャンネ ルと共に明示的 ・ 暗号化されたコマンドとデータ チャンネルと共に明示的 上の2つのオプションの、ずれかを設定すると 証明書は下て説明されているサーバー証
		明書の検証、デメーターご従い検証されます。 メモ 明示的な暗号化は奨励され ず、FlowForce内でサポート されていません。
サーバー証明書の検証	string	任意の ラメーターです。FlowForce かど のようこ FTP サーバーの証明書を検証す るか指定します。有効な値 ・ 検証無し - 全ての証明書を受 け入れます。 ・ 証明書ストアに対して検証し ます (デフォルトの値) - Windows ではジョブを実行中 のユーザーアカウトの証明書ス トアとステムストアを証明書署 名を検証するために使用してくだ さい、Linux では、システム証明 書ストアを使用します。これらは 通常 /usr/lib/ssl/cert.pem と/usr/lib/ssl/certs まけは SSL_CERT_FILE と SSL_CERT_FILE と SSL_CERT_DIR 環境関数が ポイントする。マンニカルます。 ・ サーバー証明書に対して検証 -FlowForce はサーバー証 明書、ラメーター内で指定され ているFTP サーバーの証明書と 比較します。
名前	型	説明
-------------------------	-------------	---
		この、ラメーターの使用はサーバー証明書と 安全な接続が必要です。安全な接続を構 築できない場合、FTP 関数は失敗しま す。
サーバーの証明書	certificate	任意の ちメーターです。FlowForce 内の 証明書オブシェクト への 文を指定します。 前の ちメーターを選択されたサーバー証 明書に対して検証に設定している場合、 指定されている FlowForce 証明書は FTP サーバー証明書に対して検証されま す。それ以外の場合、この ちメーター値は 無視されます。
ワイルドカード	string	必須の ちメーターです。 ワイルドカードを指定します。 例、 *.xm1。 このフイルとカードに 一致すると作業ディレクトリ パラメーターに より指定されるディレクトリ パラメーターに アップロードされます。
		<u>FTP 関数内のフイルドカード</u> を参照してく たさい。
エ ラー時 には中断する	boolean	実行に失敗するとこの、ラメーターは関数の 振る舞しを指定します。具体的には、実行 が失敗した場合、関数は以下のつを返し ます: ・ エラー時に中断、ラメーターが
		false の場合、アッフロートされ たファイルのノストは返されませ ん。 ・ エラー時に中断 / ウメーターが true の場合、失敗が発生する まで関数はファイルをアップロード し、実行を中断します。
		このために、ファイルの一部は実行が失敗し てもアップロードされる場合が決ます。
作業ディレクトリ	string	ワイルドカードが一致する場合、FTP サー バーニアップロードされるファイルが存在する ディレオトリ。
ፖክታታ	string	リモートサーバー上のファイルニアクセスを許可されているユーザーのFTP アカウナ名
		任意の「ラメーターです。

12.5 /system/sftp

/system/sftp コンテナーはセキュアFTP(SFTP)を使用したFTP サーバーへの接続に使用された関数を含んでおり、ファイルのアップロードと取得、ディレクトリの作成と削除、ファイルの削除などの操作を行います。

パマの規則に関しては/system/sftp 関数は/system/ftp コンテナーからの関数と同様の性質を有しています。それ以外の場合、完全 に異なるプロトコルが使用され、サーバーへのその接続は専用のFlowForce ステップ内で確立されることが必要となります。SFTP 接続が 確立されると、FTP サーバー上で必要とされる操作を実行するオンの次のステップ内で使用できます。例:

Execu +	ition Step	os	
	Execute funct	tion /system/sftp/co	nnect 🔹
	Parameters:	Host:	test.rebex.net
		Port:	22
		Login credentials:	Select existing credential: /public/dsa
			O Define local credential:
		Server certificate:	(+)
		Abort on error:	(+)
=	Assign this st	ep's result to my_sft	p_connection as SFTP connection
4	Execute funct	tion /system/sftp/ret	rieve 💌 💌
	Parameters:	SFTP connection:	my_sftp_connection
		Source file:	-
		Target file:	-
		Overwrite target:	
		Abort on error:	+
		Working directory:	C:\Local

上で説明されているサンプルジョブ内では、2つのステップか定義されています:

- 最初のステップはSFTP 接続を構築しmy_sftp_connection とてオブジェクトを宣言します。
- 2番目のステップはFTP サーバー上の現在のディレオリからすべてのファイルをローカルの作業ディレオリC:\Local に取得します。最初の、ラメーターは最初の実行ステップ内で宣言されているmy_sftp_connection オブンエクトをポイントしています。

SFTP 関数内のフイルドカード

以下の関数はフイルドカードをパラメーターとして受け入れます

- /system/sftp/delete-wildcard
- /system/sftp/retrieve-wildcard
- /system/sftp/store-wildcard

このような関数を使用する場合、専用のワイルドカードノデメーター内にフイルドカードを入力します。使用可能なワイルドカードは以下の

通りです

ワイルドカード	使用法	サンプル
*	ゼロませまより多くの文字を一致します。	*.htm はhome.htmとindex.htm を一致します。
?	単一の文字を一致する。	★.xm? はindex.xml とproject.xmi に一致します。

ワイルドカード + (1つま)は複数) はりポートされています。代わりに?*を使用して同じ効果を得ることができます。例えば、*.c?* は.cs、.cpと.csproj ファイルに一致しますが.c ファイルコは一致しません。

12.5.1 connect

フルな /system/sftp/connect

SFTP サポートを使用したSSH サーバーへ接続し、次のステップ内の他のSFTP 関数のために使用可能な SFTP 接続オブジェクトを返します。

メモ エラー時に中断 パラメーターがfalse の場合、この接続は未接続のSFTP 接続を返すかもしれません。SFTP 接続はジョブ の実行中に接続を切断し接続されていたい状態になる場合があります。両方の場合、SFTP 接続などを使用する全ての後の ステップは成功しません。

名前	型	説明
ポス ト	string	URL おはIP バレスの、ずれかとての リモート SFTP サーイトのバレス。
		必須の♪テメーターです。
*-+	number	SFTP サーバー 接続するか 使用され るポート番号デンホルト の値は 22 です。
ログイン資格情報	credential	SFTP アカウトのユーザー名と マワード を使用 ます」 はユーザー名と SSH キーを選択します。 FlowForce 資格情報レコードを選択し ます。詳細に関しては <u>資格情報</u> を参照 してくたさい。 FTP サーバーか資格情報を必要としてい 場合、この、デメーターをスキップしてくたさ い。
エ ラー時 には中断する	boolean	ブール値・ウメーターは実行に失敗した場 合、関数の戻い値を決定します。この・ウメ ーターがfalseの場合、関数はブール値 false も返します。・・ ウメーターがtrueの 場合、ジョブの実行は中断されます。デフォ

名前	型	説明
		ルトの値は true です。

12.5.2 delete

フルダ /system/sftp/delete

SFTP サーバーからのファイルを削除します。

実行に成功するとこの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、ラメーターにより異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する Flow Force オブ ジェクト SFTP 接続オブジェクトを取得する ために 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にコ呼び出します。 必須の デメーターです。
ターゲットファイル	string	削除するファイル
エ ラー時 にコま中断する	boolean	ブール値・ウメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ウメ ーターがfalseの場合、関数はブール値 falseも返します。、・ウメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。

12.5.3 delete-wildcard

שע איז /system/sftp/delete-wildcard

SFTP サーバーからフイルドカードに一致するファイルを削除します。例、*.xm1。

実行に成功するとこの関数は削除されたファイル名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断ノウメーターにより異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する Flow Force オブンエクト SFTP 接続オブンエクトを取得するかかに個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初に呼び出しま す。
ワイルドカード	string	必須の ^{らメーター} です。 ワイルドカードを指定します。 例、 *.xml。 ワイルドカードに一致するファイルが削除されま す。 <u>SFTP 関数内のフイルドカード</u> を参照してくたさい。
エ ラー時 には中断する	boolean	ブール値・ウメーターは実行に失敗した場合、関数の戻り 値を決定します。この、ウメーターがfalseの場合、関数 に消除に成功したディレクトリ名のノストを返し、何らかの理 由で削除できなかオンアイル名を省略します。この、ウメー ターがtrueの場合、削除できなかオー最初のファイル内の ジョブの実行が中断されます。デフォルトの値は、trueで す。

12.5.4 list-directories

フルゲ /system/sftp/list-directories

SFTP サーバーからディレクトリのノストを文字列のシーケンスとして返します。

実行に成功するとこの関数はディレクトリ名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断ノラメーターにより異なります。

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供するFlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にJ呼び出します。 必須の ウメーターです。
ワイルドカード	string	ー致するデルクトリワイルドカードを指定し ます。
エ ラー時 には中断する	boolean	ブール値・デメーターは実行に失敗した場合、関数の戻い値を決定します。この・デメ ーターがfalseの場合、関数はブール値

名前	型	説明
		false も返します。パラメーターがtrue の 場合、ジョブの実行は中断されます。デフォ ルトの値は、true です。

12.5.5 list-files

フルな /system/sftp/list-files

SFTP サーバーからファイルのリストを文字列のシーケンスとして返します。

実行に成功するとこの関数はファイル名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、ラメーターにお異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供するFlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にJ呼び出します。 必須の ウメーターです。
ワイルドカード	string	ー致するディレクトリワイルドカードを指定し ます。
エ ラー時 には中断する	boolean	ブール値・ウメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ウメ ーターがfalseの場合、関数はブール値 falseも返します。、パラメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。

12.5.6 mkdir

フルダ /system/sftp/mkdir

SFTP サーバー上でデルノオーを作成します。

実行に成功するとこの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、ラメーターに と、異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する FlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ためこ 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にコ呼び出します。 必須の ウメーターです。
ターゲットディレクトリ	string	作成されるディレクトリの名前を指定します。
エ ラー時 には中断する	boolean	ブール値・ラメーターは実行に失敗した場合、関数の戻い値を決定します。この、ラメ ーターがfalseの場合、関数はブール値 falseも返します。、パラメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。

12.5.7 move

フルな /system/sftp/move

SFTP サーバー上のファイルを移動します。

実行に成功するとの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、プラメーターにより異なります。

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する FlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にコ呼び出します。 必須の ウメーターです。
ソースファイル	string	移動されるファイルのノース・マを指定します。
ターゲットファイル	string	移動先に又。
エ ラー時 にコま中断する	boolean	ブール値・ラメーターは実行に失敗した場合、関数の戻い値を決定します。この・ラメ ーターがfalseの場合、関数はブール値

名前	型	説明
		false も返します。パラメーターがtrue の 場合、ジョブの実行は中断されます。デフォ ルトの値は、true です。

12.5.8 retrieve

フルゲ /system/sftp/retrieve

SFTP サーバーからのファイルを抽出します。

実行に成功するとこの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、ウメーターに と、異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する Flow Force オブ ジェクト SFTP 接続オブジェクトを取得する ためこ 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にJFFび出します。 必須の テメーターです。
ソースファイル	string	必須の ラメーターです。抽出されるファイル のノース・マを指定します。
ターゲットファイル	string	必須の ラメーターです。ローカル(目的) ファイル 文を指定します。
ターゲットの上書き	boolean	任意の ウメーターです。同じ名前のファイ ルが上書きされる場合これをtrue に設定 します。デフォルトの値はfalse です。
エ ラー時 には中断する	boolean	ブール値・ラメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ラメ ーターがfalseの場合、関数はブール値 falseも返します。、・ラメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。
作業ディレクトリ	string	全てのコーカル相対パスが解決される作業 ディレクトリを指定します。

12.5.9 retrieve-wildcard

フル 次 /system/sftp/retrieve-wildcard

SFTP サーバーからワイルドカードに一致するファイルを抽出します。例、*.xml。

実行に成功するとこの関数はコーカルファイル名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断ノウメーターにと、異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供するFlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にJ呼び出します。 必須の ウメーターです。
ワイルドカード	string	必須の ウメーターです。 ワイルドカードを指 定します。 例、* . xml。 ワイルドカードに一 致するファイルは削除されます。 <u>SFTP 関</u> 数内のフイルドカードを参照してくたさい。
ターゲットの上書き	boolean	任意の ラメーターです。同じ名前が目的 の箇所で上書きされる場合これを true に 設定します。デフォハトの値は false です。
エ ラー時 には中断する	boolean	ブール値・ラメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ラメ ーターがfalseの場合、関数はブール値 falseも返します。、・ラメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。
作業ディレクトリ	string	ファイルか抽出される元のコーカルディレクト リを指定します。

12.5.10 rmdir

フルダ /system/sftp/rmdir

SFTP サーバー上のディレクト」を削除します。

実行に成功するとこの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断、パラメーターに とと異なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供するFlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ためこ 個別の実行ステップ内

名前	型	説明
		で <u>/system/sftp/connect</u> 関数を最初 ロ呼び出します。
		必須の┝メ─タ─です。
ターゲットディレクトリ	string	必須の ラメーターです。削除されるディレク トリの名前を指定します。
エ ラー時 にコま中断する	boolean	ブール値/ ラメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ラメ ーターがfalse の場合、関数はブール値 false も返します。/ ラメーターがtrue の 場合、ジョブの実行は中断されます。デフォ ルトの値は、true です。

12.5.11 rmdir-wildcard

フル次 /system/sftp/rmdir-wildcard

SFTP サーバーからフイルドカードに一致するディレクトリを削除します。例、*.xm1。

実行に成功するとこの関数は削除されたディンケー・名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断ノデメーターにと、異なります。

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する FlowForce オブンエクト SFTP 接続オブシェクトを取得するかかに個別の実行ステップ内 /system/sftp/connect 関数を最初に呼び出します。 必須の ウメーターです。
ワイルドカード	string	必須の ウメーターです。 ワイルドカードを指定します。 例、 TEST*。 ワイルドカードに一致するディレクトリが削除されま す。 ディレクトリはオペレーションを成功するかかこ空である必 要があります。 <u>SFTP 関数内のフイルドカード</u> を参照してく たさい。
エ ラー時 にコま中断する	boolean	ブール値・ウメーターは実行に失敗した場合、関数の戻り 値を決定します。この、ウメーターがfalseの場合、関数 に消削除に成功したデルノケリ名のノストを返し、何らかの理 由で削除できなかオーデルノケリ名を省略します。この、ウ メーターがtrueの場合、削除できなかオー最初のデルク トリ内のジョブの実行がや断されます。デフォルトの値は、 trueです。

12.5.12 store

フルな /system/sftp/store

SFTP サーバーファイルをアップロードします。

実行に成功するとの関数はブール値 true を返します。それ以外の場合、結果は以下に説明される通りエラー時に中断 パラメーターに お興なります。

パラメーター

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する FlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にコ呼び出します。 必須の デメーターです。
ソースファイル	string as file	必須の ラメーターです。 アップロードされる ローカルファイルへの やを指定します。
ターゲットファイル	string	必須の ラメーターです。 リモートサーバー上 にアップロードされるファイルへの 文を指定 します。
ターゲットの上書き	boolean	任意の デメーターです。 同じ名前か目的 の箇所で上書きされる場合このチェックボッ クスを選択します。 デフォルトの値は false です。
エ ラー時 には中断する	boolean	ブール値/ デメーターは実行に失敗した場 合、関数の戻い値を決定します。 この デメ ーターがfalse の場合、関数はブール値 false も返します。 / デメーターがtrue の 場合、ジョブの実行は中断されます。 デフォ ルトの値は、true です。
作業ディレクトリ	string	全てのコーカル相対/ なか解決される作業 ディレケトリを指定します。

12.5.13 store-wildcard

フルな /system/sftp/store-wildcard

ワイルドカードに一致する場合、SFTP サーバーファイルをローカルディレクトリからアップロードします。例、*.xml。

実行に成功するとこの関数はアップロードされたローカルファイル名を持つ文字列のノストを返します。それ以外の場合、結果は以下に説明される通りエラー時に中断 パラメーターにより異なります。

1	- برج	-5-	
•	·//		

名前	型	説明
SFTP 接続	SFTP Connection	SFTP 接続を提供する FlowForce オブ ジェクトSFTP 接続オブジェクトを取得する ために 個別の実行ステップ内 で <u>/system/sftp/connect</u> 関数を最初 にJ呼び出します。 必須の デメーターです。
ワイルドカード	string	必須の ラメーターです。 ワイルドカードを指 定します。 例、 * . xml 。 ワイルドカードに一 致するファイルは削除されます。
ターゲットディレクトリ	string	必須の ラメーターです。ファイルがアップロー ドされるJモートシステム上のディンケトリ名を 指定します。 <u>SFTP 関数内のフイルドカー</u> 上を参照してください。
ターゲットの上書き	boolean	任意の デメーターです。 同じ名前が目的 の箇所で上書きされる場合このチェックボッ クスを選択します。 デフォルトの値は false です。
エ ラー時 には中断する	boolean	ブール値・ラメーターは実行に失敗した場 合、関数の戻い値を決定します。この、ラメ ーターがfalseの場合、関数はブール値 falseも返します。、パラメーターがtrueの 場合、ジョブの実行は中断されます。デフォ ルトの値は、trueです。
作業ディレクトリ	string	ファイルがアプロードされる元のローカルディレクトリを指定します。

12.6 /system/mail

/system/mail コンテナーは電子メールを送信するために使用される関数 send と send-mime を含んでます。

12.6.1 send

フルな:/system/mail/send

指定された宛先、通常は管理者、に電子メールを送信します

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異ないます。

- 「エラー時には中断」 パラメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSE の場合、関数は、FALSE を返します。

この関数を使用する前に、メールサーバー設定が構成されていることを確認してくたさい(メール・ウメーターの設定を参照してくたさい)。

名前	型	説明
差出人	string	電子メールが送信されるアドレス。例 flowforce@ <hostname>。</hostname>
宛先	string	電子メールアドレスの宛先必須の、ウメータ ーです。このフィールドは複数の宛先電子メ ールアドレスの ユンマで区切られたリストを 含んている場合があります。
件名	string	メッセージの件名必須の ちメーターです。
本文	string	文字列とてメッセージの本文を提供する 任意の、ラメーター。メッセージの本文は ASCII とUnicode 文字をサポートしま す。 テキストが1行以上ある場合(すなわち、改 行を入力後) メッセージ本文のすめのテキス トボックスコは寸番機能かあります。更に、 波{} かっコニ囲まれた式を入力すると、 FlowForce 式の基本の構文の イライト 機能を使用することができます。 このテキストボックス内に入力する場合以下 の制限があることにごの本文を提供する

名前	型	説明
		 FlowForce 式は同じラインで 開始し終了する必要があります。 元に戻す/やり値しサポートを使用することはてきません。 数千のラインを含む長い本文はエディターの、プォーマンスは影響を与えます。 任意の、ラメーターです。
添付	string as file	電子メールと共に送信される添付ファイルの 名前。
エラー時には中断する	boolean	このブール デメーターはショブが失敗した場合の戻い植を決定します。「エラー時には中断」がFALSE の場合、関数は、ブール値のFALSE を返します。「エラー時には中断」がTRUE の場合、ジョブの実行は中断されます。デフォルトの値は、TRUE です。

サンプル エラー処理をショブ こ追加するを参照

12.6.2 send-mime

フルな:/system/mail/send-mime

指定済みの宛先に電子メールを送信します。通常宛先は管理者です。

この関数を使用する前に、メールサーバー設定が構成されていることを確認してくたさい(メール・デメーターの設定を参照してくたさい)。

実行に成功すると、この関数はブール値のTRUEを返します。ジョブの実行に失敗すると、結果は「エラー時には中断」、デメーターの値により以下のように異なります。

- 「エラー時には中断」 パラメーターが TRUE (デフォルトの値) の場合、ジョブの実行は中断されます。この場合、エラーを保護 されたブロックを使用して処理することができます(以下を参照: <u>ステップのエラーの処理</u>)。
- 「エラー時には中断」、デメーターがFALSEの場合、関数は、FALSEを返します。

send 関数とは異なり、この関数のメッセージ本文ノラメーターは、文字列ではなくストリームを生成する式を期待します。これにより、ストリームから(例えば、HTML として)メッセージを取得することができます。

HTML コンテンンをペッセージ本文のために取得するには、HTML 出力をMIME とて生成する、Style Vision Server 変換を呼び出すことが強く奨励されます。FlowForce Server 自身は、イメージ、スタイルシート、おけよ、HTML ファイルにより参照されている類似したシノースを内蔵 MIME ストリームに集めません。

内蔵 HTML メッセージ本文を Style Vision Server を使用して生成するゴよ 以下を行います:

1. Altova Style Vision 内で電子メールのHTML 本文をデザインします。デザインコココーカルイメージとスタイルシートか含まれる場合かあ ります。

2.StyleVision 変換をFlowForce Server にデプロイレます。FlowForce 内でよ 変換はStyleVision Server にお実行が可能な内蔵 FlowForce 関数になります。

3. 上記のStyleVision Server 変換を呼び出すジョブを作成します。GenerateHtmlOutputAsMime オプションカジョブ構成ページで選択されていることを確認してくたさい。

4.ジョブ構成ページ内でFlowForce Server 式関数を呼出、生成されたMIME ストリームをピックアップし、send-mime 関数の「メッセージ本文」、テメーターコンとます(下の「例1」を参照してくたさい。

HTML ファイルによ参照されている外部リソースはMIME ストリームコ埋め込むことはできず、電子メールの添付として追加されます。

HTML 出力をMIME ストリームとて生成するジョブのサンプルは、以下で説明されています。StyleVision 変換をFlowForce Server にデプロイする方法をステップバイステップで説明しているサンプルは関しては、<u>StyleVision 変換からのジョブの作成を参照してくだ</u> さ、StyleVision Server 統合の詳細に関しては、他のAltova サーバーとの統合。統合を参照してくたさい。

メッセージ本文のためにストリームを直接 FlowForce 内で作成するコよ stream-open おけまstream-from-string などの式 関数を呼び出すこともできます。同様に、<u>MIME 式関数</u>を使用して電子メール おけよ 添付メッセージへッダーをカスタマイズすることができます。

宛先側の迷惑メールフォルダーに電子メールが振り分けられないように、MIME ヘッダーを受信先のサーバー、ませよ プログラムで許可されている方法で構成してくたさい。

名前	型	説明
差出人	string	電子メーリは送信されるアドレス。例: flowforce@ <hostname>.</hostname>
宛先	string	電子メールアドレスの宛先必須の ラメータ ーです。このフィールドは複数の宛先電子メ ールアドレスの ユンマで区切られたリストを 含んでいる場合かみます。
件名	string	メッセージの件名必須の ラメーターです。
本文	stream	ストリーム型を返す Flow Force 式上てのメッセージの本文テキスト。
添付	sequence of stream	電子メールと共に送信される添付ファイルで す。各添付は、ストリームを生成する FlowForce 式である必要があります。文 字列ますはファイルからストリームを作成する ために、ストリーム式関数を呼び出しま す。MIME 式関数を呼び出して、MIME へッダーを追加、変更、ますま、削除するこ とができます。

名前	型	説明
エ ラー時 には中断する	boolean	このブール ラメーターはショブが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、 TRUE です。

サンプル

サンプル1: HTML 書式で電子メールを送信するコよ下で説明されるショブは send-mime 関数を呼び出します。

最初の実行ステップはHTML 出力をStyleVision Server 変換を呼び出して生成します。この変換はStyleVision を使用してデザインされ、StyleVision 変換からのジョブの作成で説明されている通り、FlowForce Server にデプロイされます。 GenerateHtmlOutputAsMime チェックボックスが選択されています。最初の実行ステップの結果は、「出力」と呼ばれています(この 文字列は以降のステップで必要とされ、式内にJoutput」を入力することのとり参照することができます)。

2番目の実行ステップはFlowForce 式を計算するために compute 関数を呼び出します。具体的には、式 nth (results (output), 0) はインデックス0 を持て この場合、 OutHtml) MIME ストリームを前のステップにより生成された結果 かピックアップします。

最後に、3番目の実行ステップは電子メールメッセージを送信します。「メッセージ本文」フィールドは、ステップで前に計算された結果である FlowForce 式を含んでいます。

実行	ステップ					
+	(+)					
-	関数の実行 /public/BiggestCitiesPerCity.transformation			tion 💽 💌		
	パラメーター:	InputXml:	₽{₿	altova://packagedfile/BiggestCities.xml		
		OutHtml:	₽ HTHL	•		
		GenerateHtmlOutput	tAsMime:	•		
		OutRtf:	<mark>.</mark> ₽TF	•		
		OutFo:	₽ F0	•		
		OutPdf:	₽ •	•		
		OutDocx:	00088 (1000	•		
		Working-directory:				
=	ステップの結果	を割り当て output		以下としてReturnTypeDocx, ReturnTypeFo, ReturnTypeHtml, ReturnT		
+						
	■ 関数の実行 /system/compute ■ ピ			_ ₫		
	パラメーター:	条件式: nth(result	ts(output), <mark>0</mark>)			
_	ステップの結果	を割り当て message		以下として10		
+	7(7 7 7 07)HJ					
	関数の実行 /sy	/stem/mail/send-mime	e	▼ 2 ³		
	パラメーター:	差出人:	flowforce@yo	purhostname		
		宛先:	to@example.	org		
		件名:	Example mail			
		本文:	message			
		添付:	+			
		エラー時には中断:	+			
=	ステップの結果	を割り当て名前		以下としてブール値		

サンプル2:下で表示されているジョブはHTML書式のイメージ添付を含む.png書式で電子メールを送信するsend-mime関数を呼び出します。

実行	ステップ				
-	関数の実行 /sy パラメーター: ステップの結果	ystem/compute-string 条件式: HilTh を割り当て message	▼ ご ismessage contains a link , 文字列 以下として文字列	(必須) セットする値	i 🕨
+	関数の実行 /sy	ystem/mail/send-mim	e		
	パラメーター:	差出人:	flowforce@yourhostname	文字列	(必須)
		宛先:	to@example.org	文字列	(必須)
		件名:	Example mail	文字列	(必須)
		本文:	stream-from-string(message, 'UTF-8', 'text/html')	条件式ストリー	ム (必須)
		添付: エラー時には中断:	 add-mime-header(stream-open('C:\sample.png', 'image/png'), 'Content-Disposition', 	'attachment; filenam	ie="samp
=	ステップの結果	。 見を割り当て 名前	以下としてブール値		

最初の実行ステップはメッセージの内容のナダのHTMLコードを準備します。簡素化のナダに、このサンプル内のHTMLコードは直接テキストボックスに入力されます。HTML出力を取得するコよ、奨励される方法は、前のサンプルで説明されている通りStyleVision Server 変換を呼び出すことです。この実行ステップは次のステップで使用される文字列出力を生成します。

2番目の実行ステップは電子メールのボデをストリームとて生成します。具体的はは最初の実行ステップ(body_html,型 string)の 結果を stream に変換します。Stream-from-string 関数はこの目的のためは呼び出されます。希望されるエンロードとMIME 型はこ の関数の引数としてのされています。

3番目の実行ステップは電子メールの添付をストリームとして生成します。イメージ添付はローカルレスからです: C:\sample.png。具体的 には、このステップは以下の武を計算します:

```
add-mime-header(stream-open('C:\sample.png', 'image/png'), 'Content-
Disposition', 'attachment; filename=sample.png')
```

上の式は以下を行います

- <u>stream-open</u> 関数はイメージをストリームとて開きます。
- <u>add-mime-header</u> 関数は、「コンテンソーディスポジション」 ヘッダーをストリームに追加します。このようし、て、イメージ添付は想定された名前(「sample.png」)が与えられます。

最後にこれおイナリファイルのため「添付」フィールドからの式が必要とされます。mime-content-encode 式関数を使用して転送中に保管されるためにこのようなファイルはbase-64 としてエンコードされる必要が決ます。

12.7 /system/maintenance

/system/maintenance コンテナーは、サーバー上でのオペレーションを管理するために使用される関数を含んています。

12.7.1 archive-log

フル 次:/system/maintenance/archive-log

古いログ記録をサーバー上のアーカイブファイルに移動します。作成されるアーカイブファイルの名前を文字列の植として返します。

		-
1	X-X-	
•	11	

名前	型	説明
日、よりも古い	number	ここに入力された日数より比古いアーカイブ ファイル。デフォルトの値は30です。
アーカイヴディレクトリ	string	アーカイブディレクトリ名 (例えば c: \temp)。必須。
アーカイブファイルプレフィックス	string	アーカイブファイルのプレフィックス。 デフォルト はflowforcelog です。
アーカイブされた記録の削除	boolean	FlowForce データベースからアーカイブされ た記録を削除します。
作業ディレクトリ	string	ジョブの作業ディレクトリを指定します、例え ば、c:\somedirectory)。相対的な パンか使用されると、作業ディレクトリニズ扎 て解決されます。

12.7.2 cleanup-files

フル 次:/system/maintenance/cleanup-files

使用されていない ファイル、おさは、デプロイされたオブジェクト(MapForce マノピングとStyleVision 変換など)により参照されるファイルを 削除します。数値として削除されたファイルの数量を返します。

デプロイされたオブジェクトを削除、まけよ、変更されたファイルを持つ既存のオブジェクトを再デプロイすると、以前デプロイされたオブジェクトに 関連するファイルは未使用になります。デフォルトでは、FlowForce Server は未使用のファイルを削除しません。ですから、デ特に複数 のユーザーがオブジェクトを FlowForce Server にデプロイする企業の環境では、ディスク容量をクレーンアップするためはは、この関数を定 期的に呼び出すジョブを作成することが奨励されます。

デプロイされたオブジェクトにより使用されている現在のディスク容量を確認するコは、FlowForce Server アプリケーションデータフォルダー 内にあるファイルフォルダーのサイズを確認してくたさい(以下を参照 FlowForce Server アプリケーションデータ)。

この関数にコお ラメーターか存在しません。

12.7.3 truncate-log

רקע א:/system/maintenance/truncate-log

与えられた日付よりにはしての記録を削除します。削除された記録の数を数量の値で返します。

名前	型	説明
日、より先古い	number	こコに入力された日数より比古いアーカイブ ファイルを削除します。デフォルトの値は30 です。

12.8 /system/shell

/system/shell コンテナーには、シェルコマンド、おけよ、スクリフトを実行する関数 commandline かきおれています。

12.8.1 commandline

フルな:/system/shell/commandline

シェルコマドおけおシチファイルを実行します。

FlowForce Server ジョブコ環境関数を読み込ませるコよ、スクレプト内で定義されている必要があります。おコよスクレプト は/system/shell/commandline 関数にお実行されなければなりません。FlowForce Server は、インタラクティブでは内 シェルにより作動されています。これは(Linux 上のexecuting .profile おコよ.bashrc などの) インタラクティブなシェル特有の振 る舞いコン適用することができないに注意してくたさい。

シェルコマイドからの終了コードが「0」 よりた大き、場合、出力は、以下の通りです:

ノウメーターエラー時には中断がtrue(デフォルト)の場合、この関数は、実行を中断します。この場合、エラーを保護されたブロックとして 処理することができます。(以下を参照: ステップのエラーの処理)。

ノラメーターエラー時には中断がfalseの場合、関数は、標準の出力、標準のエラーおよび終了コードを含むシェルコマイの結果を返します。

最後のコマイトから終了コードが「0」(成功)の場合、関数は最後のシェルコマイトの結果を、ジェネリックな型として返します。この関数により返された値を、他のステップませるエンラブで処理するコよ、以下を行ってくたさい

1.「このステップの結果を以下に割り当てる」テキストボックス内に値を入力して返された結果に名前をつけます。(例えば、myresult)。

2. 必要とする戻り値の型により、関数 compute おけま compute-string を実行する新規ステップを作成します。 3. FlowForce ステップ結果関数を使用して。上記の関数の引数として、ジェネリングは結果から希望する部分を取得する式を入力します。 例えば、式 stdout (myresult)を入力して、ストリームとての結果の標準出力を取得する、おけま、標準エラー出力ストリームを取 得するためこ stderr (myresult)を入力します。文字列と同じ値を取得するけま、 content (stdout (myresult))と content (stderr (myresult))をそれぞれ使用します。

stdout 関数 (どうブ) は シェルコマボカ標準の出力を返され 場合、失敗することご注意してください。同様に、stderr 関数は、 標準のエラーかれ 場合でも失敗します。

詳細に関しては、以下を参照してください、データ型をステップで処理するとステップ結果関数

パラメータ	_

名前	型	説明
マゲ	string	実行するシェルコマンドを入力します。
エ ラー時 には中断する	boolean	このブール デメーターばらづが失敗した場 合の戻り値を決定します。「エラー時には 中断」がFALSE の場合、関数は、ブー 川値のFALSE を返します。「エラー時に は中断」がTRUE の場合、ジョブの実行 は中断されます。デフォルトの値は、

名前	型	説明
		TRUE です。
作業ディレクトリ	string as directory	ジョブの作業デルクトリを指定します、例え ば、c:\somedirectory)。相対的な パンか使用されると、作業デルクトリニマ扎 て解決されます。

サンプル

次のショブは DoTransform.bat と呼ばれる Windows バッチファイルを実行します。 DoTransform.bat が XML ファイルを入力として必要とすると仮定し、入力 XML ファイルは作業ディレクトリニニピーされる必要があります。 このサンプル内では、作業ディレクトリお以下のとおりです: C:\codegen\xslt2。

実	実行ステップ				
	•	関数の実行 /sy	/stem/shell/commandlin	e	
		パラメーター:	コマンド: エラー時には中断: 作業用ディレクトリ:	DoTransform.bat + C:\codegen\xslt2	
	=	ステップの結果	を割り当て名前		以下として結果

次のショブはXSLT 変換をパラメーターを使用して実行するオンジニRaptorXML Server を呼び出します。PATH 環境変数は RaptorXML Server 実行可能ファイルへの 少を含んでいます。例えば、C:\Program Files (x86) \Altova\RaptorXMLServer2021\bin。RaptorXML Server, に関する詳細は<u>https://www.altova.com/raptorxml</u>を参照 してくたさい。

/system/shell/comm	andline 🔹	C ₂
コマンド:	raptorsml xsltinput=c:/Test.xmloutput=c:/Output.xml c:/Test.xslt	

コマボライノによし返された出力を処理するステップバイステップのサンプルに関しては、パマか存在するか確認を参照してくたさい。

13 式関数

このセクションでは、FlowForce 式関数に関するレフォンスについて説明されています。式の使用方法を理解する概念に関する情報に関しては、FlowForce式を参照してくたさい。

式関数は以下に示されるようこグループ化されています:

- <u>ステップ結果関数</u>
- ホリーム関数
- ファイルシステム関数
- ファイル マ関数
- <u>リスト関数</u>
- <u>文字列関数</u>
- ブール値関数
- ラタイム情報関数
- AS2 関数
- <u>MIME 式関数</u>

13.1 ステップ結果関数

ステップ結果関数は、ジョブによ返された結果(おけよジョブ内での実行ステップによ返された結果)を処理します。

13.1.1 error-message

この関数はステップが遭遇するエラーメッセージのテキストを返します。この関数は典型的に保護されたブロックの内部、具体的はばエラー時」の、レドラーで使用されます。エラーは遭遇しない場合、おけおショブの性質からエラーのテキストを技術的に抽出できない場合、関数は空の文字列を返します。

署名

error-message(result:result) -> string

ジェネリックな署名

パラメーター

名前	型	説明
result	result	エラーテキストが抽出されるエラーのあるス テップを提供します。エラーのあるステップに 移動するコはfailed-step() 関数を 呼び出します。

サンプル

エラー処理をショブに追加するを参照してくたさい。

13.1.2 exitcode

結果の数値終了コードを返します。

署名

exitcode(result:result) -> number

ジェネリックな署名

名前	型	説明
result	result	取得する終了コードを持つステップの結果

式関数

サンプル

次のショブセンのステップから構成されています。最初のステップは現在の作業ディレクトリ内(C:\FlowForce)にdata と呼ばれるデル クトリを作成しようと試みるWindows コマイドラインコマイドを呼び出します。このステップの結果はoutcome とて宣言されます。2番目 のステップはoutcomeを取得し、exitcode 関数の手助けを受け、数値の終了コードを返します。string 関数の助けを受け、数値 の終了コードは文字列に変換されます。式のデータ型が文字列のためこの変換が必要にないます。

更に重要なことは、エラー時に中断オプションが選択されていないことです。それ以外の場合、実行はエラー発生時に中断され、処理のための2番目のステップのオーダの終了コードは存在しません。

実行	実行ステップ				
	Ð				
		関数の実行 /sy	stem/shell/commandlin	e	▼ 🗗
		パラメーター:	コマンド:	mkdir data	
			エラー時には中断:	+	
			作業用ディレクトリ:	C:\FlowForce	
-	-	ステップの結果	を割り当て outcome	× 以下として結果	
		関数の実行 /sy	stem/compute-string		• 🖻
		パラメーター:	条件式: The exit cod	le is: {string(exitcode (<i>outcome</i>))}	

ジョブカ最初に実行される場合 data ディレクトリの作成が成功すると、終了コードは0 になります。次の実行では、ディレクトリは既存のため、終了コードは1 になります。

エラー処理をジョブに追加するを参照してくたさい。

13.1.3 failed-step

ステップ結果を返します。保護されたブロックを有するエラーを処理する際にこの関数は役に立ちます。failed-step 関数は、「エラー時」ハンンドラーの一部である必要があります。それ以外の場合、ステップを使用している箇所で、エラーのステップが存在したいため、失敗します。

このように、この関数は、エラーカ発生したステップを指しません。属性の結果を確認するコよこの関数をstdoutませるtstderrなどの関数に引数としていてします。例:

```
stderr(failed-step())
stdout(failed-step())
```

署名

```
failed-step() -> result
```

ジェネリックな署名

サンプル

以下の例を参照してくたさい

- ジョブエラー処理を追加する
- <u>エラー処理を使用してXMLドキュメトを検証する</u>

13.1.4 results

オプションで名前でフィルターされた指定された結果のストリームの配列を返します。関数 nth を使用して配列内の特定の値にアクセスします。

署名

results(result:result, name:string) -> array of stream

ジェネリックな署名

パラメーター

名前	型	説明
result	result	必須の ラメーターです。ストリームの配列を 返すステップの結果です。
name	string	任意の ラメーターです。 与えられると、結 果内で特定な値を名前に従い フィルターし ます。

サンプル

サンプルA。FlowForce Server に出力とて単一のXML ファイルを生成するマピングをデプロイレナと仮定します。このようなマピン グのサンプルは Complete PO.mfd で MapForce サンプルに含まれています。 MapForce 内のターゲナ XML コンポーネント 名は Complete PO です。このマピングの結果を処理し、FlowForce からのファイルに保存するコよ、ショブを次のように構成します。

実	実行ステップ			
	+			
		関数の実行 /p	ublic/CompletePO.mapp	ning 🗾 🛃
		パラメーター:	Customers: (in Articles: (in ShortPO: (in CompletePO: (ou Working-directory:	put) 월 + put) 월 + put) 월 + itput) 월 +
	-	ステップの結果	を割り当て mapping_r	result × 以下としてCompletePO
	4	関数の実行 /sy	/stem/filesystem/copy	
		パラメーター:	ソース:	{as-file(nth(results(mapping_result), 0)))}
			ターゲット:	output.xml
			上書き:	+
			エラー時には中断:	•
			作業用ディレクトリ:	•
	=	ステップの結果	を割り当て名前	以下としてブール値

上記のジョブ構成では、最初のステップはマピングを実行し、結果をmapping_result として返します。2番目のステップでは、式は、

{as-file(nth(results(mapping_result), 0))}

mapping_result を処理し、ファイルニ変換します。具体的にresults 関数は、MapForce コンポーネントからストリームの配列を 選びます。nth 関数は、この配列から最初のアイテムを選びます。as-file 関数はストリームからファイルを生成します。

<u>copy</u> 関数は生成されたファイルを作業ディレクトリニニピーします。Target テキストボックスは生成されたファイルの名前を定義します。同じ 名前の既存のファイルは上書きされます。

サンプルB。FlowForce Server ロンプロゲーゲナ XML エンボーネント「MarketingExpenses」と「DailyExpenses」が存在す るマピングをデプロイしたと仮定します。このようなマピングのサンプルはMarketingAndDailyExpenses.mfd でMapForce サン プルに含まれています。「DailyExpenses」エンポーネントからファイルを生成するには、上記に類似したごらブを作成し、式を以下に変更し ます:

{as-file(nth(results(mapping_result, 'DailyExpenses'), 0))}

ここでの唯一の違いはマピングにより生成されたストリームの配列か希望するコンポーネトの名前によりフィルターされていることです(この場合「DailyExpenses」)。

サンプルC。FlowForce Server に複数のXML ファイルを動的に生成するマメピングをデプロイレナと仮定します。出力ファイル名はマッピング自身により生成され、ランタイムの前は不明です。このようなマメピングのサンプルは

DividePersonsByDepartmentIntoGroups.mfd でMapForce サンプルに含まれています。マピングの3番目の出力ファイルを生成するコよ、上記に類似したショブを作成し、式を以下に変更します:

{as-file(nth(results(mapping_result), 2))}

ここで、3番目のファイルは必要になるため、nth 関数に2番目引数として与えらえれたインデックスは、インデックスがゼロベースのため、(3 で はなく) 2です。

以下の例を参照してくたさい

- StyleVision 変換からジョブを作成する
- 複数のXML から複数のPDF を作成する

13.1.5 retry-count

FlowForce か再試行を行ったエラー/成功の処理(しかゆる「保護されたブロック」)を持つ「つまけは複数のステップの回数を示す数値を返 します。関数は持い関数を囲む最も内側の保護されたブロックを評価することに注意してくたさい。再試行が行われなかけた場合(すなわち、 保護されたブルックの最初の実行に成功した場合)戻り値は0です。エラー時に再度試すも参照してくたさい。

署名

retry-count() -> number

ジェネリックな署名

13.1.6 stdout

(シェルコマイを実行する)などの実行ステップの一部は標準の出力を返します。例えば、シェルコマイ、dir (Windows)はディレクトリのノストを返します。

ステップが結果を返すと、FlowForce Server は、自動的に結果を一般的な型のresultに割り当てます。stdout 関数に関しては、 resultの標準の出力は、以下の通りです:

stdout(result)

resultは実行ステップによ返された値です。

この関数は、result か標準の結果を与えない場合失敗します。

署名

stdout(result:result) -> stream

ジェネリックな署名

式関数

パラメーター

名前	型	説明
result	result	標準エラーを得るステップの結果。

サンプル

以下の例を参照してくたさい

- ジョブエラー処理を追加する
- エラー処理を使用してXMLドキュメトを検証する
- パマの存在を確認する

13.1.7 stderr

結果の標準エラーを返します。結果が標準のエラーを返ざえい場合失敗します。

署名

stderr(result:result) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
result	result	標準エラーを得るステップの結果。

サンプル

例に関しては、ジョブヘエラー処理を追加するを参照します。

13.2 ストリーム関数

ストリーム関数は、データのストリームを処理するために使用されます。ストリームをWebサービスませはステップ結果からFlowForce Server に、次することができます。

13.2.1 as-file

ストリームノースがファイルの場合ファイルを作成します。ストリームノースがファイルでは内場合、一時ファイルを作成します。

署名

as-file(stream:stream) -> string

ジェネリックな署名

パラメーター

名前	型	説明
stream	stream	ホリームノーズを指定します。

サンプル

次のショブはテキスト1行を持つfile.txt と呼ばれるファイルを作成します。最初にstream-from-string 関数は、引数とて提供されたテキストからストリームを生成します。次にas-file 関数は、ストリームを引数としてとり、一時的なファイルを生成します。一時的なファイルを 永続的ないマルニピーするには、内蔵のcopy 関数が実行ステップから呼び出されます。ジョブ(C:\FlowForce)の作業ディレクトリニファ イルよーピーされ、ジョブが作動する都度上書きされます。

実行	実行ステップ			
+				
-	関数の実行 /sy	ystem/compute	▼ 🖉	
	パラメーター:	条件式: as-file(strea	am-from-string('This is my file content.'))	
=	ステップの結果	また割り当て result	× 以下としてTO	
	関数の実行 /sj	ystem/filesystem/copy	 ∎ 	
	パラメーター:	ソース:	{result}	
		ターゲット:	file.txt	
		上書き:		
		エラー時には中断:	•	
		作業用ディレクトリ:	C:\FlowForce	

エラーログのあるXMLドキュメトを検証するも参照してくたさい。

13.2.2 content

指定されたストリームのコンテンツを指定されたエノコード内のテキストとして読み取ります。

署名

content(stream:stream, encoding:string="UTF-8") -> string

ジェネリックな署名

パラメーター

名前	型	説明
stream	stream	ネリームノーズを指定します。
encoding	string	使用するエンコードを指定します。デフォルト のエンコードは 'UTF-8'です。

サンプル

以下の例を参照してくたさい

• エラー処理をショブに追加する

13.2.3 empty-stream

空のストリームを作成します。

署名

empty-stream() -> stream

ジェネリックな署名

13.2.4 stream-from-string

与えられたエンコードを使用する文字列からストリームを作成します。引数とて与えられたコンテンンの型はストリームに関連しています。このストリームの型は自動的にファイルとして保存されません。

署名

stream-from-string(string:string, encoding:string="UTF-8", contenttype:string=contenttype=text/plain) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
string	string	ストリームを作成する文字列
encoding	string	使用するエンコードを指定します。デフォルト のエンコードは'UTF-8'です。
contenttype	string	ストリームご関連させるかかに contenttype を指定します。デフォルト は contenttype=text/plainです。

13.2.5 stream-open

既存のファイルからストリームを作成します。

署名

```
stream-open(name:string, contenttype:string=contenttype=application/octet-stream) ->
stream
```

ジェネリックな署名

パラメーター

名前	型	説明
name	string	このストリームのためのノースファイルのな
contenttype	string	ストリーム工関連させるかっめこ contenttype を指定します。デフォルト は contenttype=application/oct et-streamです。

サンプル

次のショブは、txt 拡張子を持つ既存のファイルを開き、.csv 拡張子を持つ同じディレクトリに書き込みます:

Ę	衍	ステップ		
	•			
	4	関数の実行 /sy	/stem/filesystem/copy	▼ 🖉
		パラメーター:	ソース:	{as-file(stream-open('C:\FlowForce\file.txt'))}
			ターゲット:	C:\FlowForce\file.csv
			上書き:	+
			エラー時には中断:	+
			作業用ディレクトリ:	+

13.3 ファイルシテム関数

ファイルシステム関数は、ファイルシステムへのアクセスを許可します。これらの関数を実行するために、ジョブはオペレーティングシステムの対応するアクセス権を使用してユーザーアカウストの資格情報を使用する必要があります。

13.3.1 list-files

(ワイルドカードにより中断される可能性のある)、マ内のファイルをリストし、結果の文字列のリストを返します。

パマが、マ区切り文字、およ、ワイルドカードで終了しない場合、指定されたアイテムの検索が親ディレクトリ内で行われます。

署名

list-files(path:string) -> list of string

ジェネリックな署名

パラメーター

名前	型	説明
path	string	ディレオトリ、おけよ ファイルへの 文を指定 します。

サンプル

サノプルは関してはファイルのエピーを参照してくたさい。

13.3.2 list-directories

(ワイルドカードにより中断される可能性のあるノマ内のサブディレクトリをリストし、結果の文字列のリストを返します。

署名

list-directories(path:string) -> list of string

ジェネリックな署名

名前	型	説明		
path	string	デルケリーのやを指定します。		

13.3.3 read-lines

与えられたファイルからラインを読み取り、文字列のリストとして返します。返される文字列は(\n などの)行末を含みます。下のサンプルで説明されている通り、処理前にtrim() 関数を使用して各ラインをトリムする必要がある場合があります。

署名

read-lines(filename:string, encoding:string="UTF-8") -> list of string

ジェネリックな署名

パラメーター

名前	型	説明		
filename	string	ファイルへの、文を指定します。		
encoding	string	使用するエンコードを指定します。デフォルト のエンコードは'UTF-8'です。		

サンプル

FlowForce Server がインストールされているエンピューター上の複数のディンケリ内にある複数のファイルを処理する必要がある上仮定します。ディレケリレスは各ラインカディノケリレスに対応する箇所でテキストファイルとして保存されています。例

C:\FlowForce\Examples\ListDirectories\1 C:\FlowForce\Examples\ListDirectories\2 C:\FlowForce\Examples\ListDirectories\3

下で説明されているショブはつのステップから構成されています。最初のステップはread-files 関数を呼び出し、全てのディレオ・リージを 上のテキストファイルから集めノストレーます。2番目のステップはジロノストを反復しlist-files 関数を呼び出します。ノジロ処理される 前にトリムされます。結果する文字列が空白文字まけば新規ライン文字を含んでいないことに注意してくたさい。

Execution Steps

•								
4	Execute function /system/compute							
	Parameters: Expression: read-lines('C:\FlowForce\Examples\ListDirectories\paths.txt')							
=	Assign this step's result to lines as TO							
	For each item in sequence lines							
	•							
	Execute function /system/compute							
	Parameters: Expression: list-files(trim(<i>item</i>))							
	= Assign this step's result to name as T0							
	J							

このジョブをWebサービスとして公開し、デフォルトのアドレスとポートでブラウザーからアクセスするとブラウザーは各ディレクトリのコンテンンを例えばJSON配列として出力します

<u>F</u> ile <u>E</u> dit <u>V</u> iew Hi <u>s</u> tory <u>B</u> ookm	arks <u>T</u> ools <u>H</u> elp		_		×
127.0.0.1:4646/service/ListDirectorie	× +				
← → C ^I	7.0.0.1:4646/service/ListDirectories	•••	111	• •	≡
JSON Raw Data Headers					
Save Copy Collapse All Expand	All 🗑 Filter JSON				
- e:					
0: "C:\\FlowForce\\Examp	les\\ListDirectories\\1\\A.txt"				
1: "C:\\FlowForce\\Examp	les\\ListDirectories\\1\\B.txt"				
▼ 1:					
0: "C:\\FlowForce\\Examp	les\\ListDirectories\\2\\C.txt"				
1: "C:\\FlowForce\\Examp	les\\ListDirectories\\2\\D.txt"				
▼ 2:					
0: "C:\\FlowForce\\Examp	les\\ListDirectories\\3\\E.txt"				
1: "C:\\FlowForce\\Examp	les\\ListDirectories\\3\\F.txt"				
13.4 ファイルパス関数

ファイル や関数は、アとファイル名の特定の一部を抽出するために使用されます。例えば、ディレクトリをポール、おけま triggerfile、ラ メーターからジョブをトリガー下ファイル名を抽出する場合、この操作を行う必要かある場合かあります(ファイルシステムトリガーを参照してくだ さ、)。

13.4.1 extension

パマからファイル拡張子を抽出します。

署名

extension(path:string) -> string

ジェネリックな署名

パラメーター

名前	型	説明
path	string	ファイルへの、マを指定します。

サンプル

次の式ば、txt を返します:

```
extension("c:\temp\file.txt")
```

13.4.2 filename

パマカシは振子の無い、ファイル名と拡張子を抽出します。

署名

```
filename(path:string) -> string
```

ジェネリックな署名

名前	型	説明
path	string	ファイルへの、文を指定します。

サンプル

次の式ばfile を返します:

filename("c:\temp\file.txt")

13.4.3 filename-with-extension

パマからファイル名と拡張子を抽出します。

署名

filename-with-extension(path:string) -> string

ジェネリックな署名

パラメーター

名前	型	説明
path	string	ファイルへの、又を指定します。

サンプル

次の式ばfile.txt を返します:

```
filename-with-extension("c:\temp\file.txt")
```

13.4.4 join-paths

パマコ 割とて提供される ひを結合します。

署名

join-paths(string1:string, string2:string, stringN:string) -> string

名前	型	説明
string1	string	ジョインする単一の なステップを指定しま す。全ての続き引数はエンマにより区切られ ます。
string2	string	上記と同様です。

名前	型	説明
stringN	string	上記と同様です。

サンプル

W indows 上では次の式が"C: \mp \test txt" を返します:

```
join-paths('C:\tmp', 'test.txt')
join-paths('C:\tmp\', 'test.txt')
join-paths('C:\', 'tmp', 'test.txt')
join-paths('C:\Users', '\tmp', 'test.txt')
join-paths('D:\Data', 'C:\tmp', 'test.txt')
```

Linux とMacOS では次の式が"/home/user/test txt」を返します:

```
join-paths('/home/user', 'test.txt')
join-paths('/var', '/home/user', 'test.txt')
```

13.4.5 parent-directory

パマから親ディレクトリを抽出します。

署名

parent-directory(path:string) -> string

ジェネリックな署名

パラメーター

名前	型	説明
path	string	デルケリーのな指定します。

サンプル

XML ファイルからデータベースを更新する MapForce マピングが存在すると前提します。FlowForce Server はままにデプロイされており、ジョブは既に作成されています。また、ディレクトリが変更されると実行されるジョブ共構成ずみです、つまり、ジョブはファイルシステムトリガーを使用します。ファイルシステムトリガーを参照してください。

ジョブの最初のステップは、データベースを更新するマッピングを実行します

Execute function /public/Tutorials/Altova_Hierarchical_DB.mapping		
Parameters:	rameters: Altova_Hierarchical: (input) 톏 🕂	
	Working-directory:	C:\FFSERV
Assign this step's result to name		

マピングステップが実行を完了すると、目的は、ソースXML ファイルを "processed" とう名前のサブディレオトリニ移動することです。これ を行うことにより、どのファイルが処理済みかを管理することができます。このエールを達成するために /system/filesystem/move 関数を 呼び出す新規のステップを追加し、 ソースとターゲット として下の値を入力します:

Execute function /system/filesystem/move			
Parameters:	Source:	{triggerfile}	
	Destination:	{parent-directory(triggerfile)}processed	
	Overwrite target:	\$	
	Abort on error:	+	
	Working directory:	C:\FFSERV	

ソースフィールドの、デメーター値 {triggerile} は、FlowForce にマッピングをトリガーしたファイルを移動するように命令します。 パテメーター値:

{parent-directory(triggerfile)}processed

保存先フィールドが現在のディレケリ内の「processed"とう名前のディレケリに設定します。これは、式と文字列にお構成されて、ます。式の部分が中かってこと区切られていることに注意してくたさい(以下を参照:文字列フィールド内に式を埋め込む)。式は、

{parent-directory(triggerfile) }

親ディレクトリ 関数を呼び出し、引数として値 "triggerfile" を与えます。

ですから、ジョブが実行されると、次のアグションが実行されます

1.スクリプト、おけよ ユーザーがファイルを現在の作業ディレクトリに(例えば、C:\FFSERV)コピーします source.xml。

2.トリガーが実行され source.xml は "triggerfile" になります。

3.FlowForce Server は、マピングを作動するステップを実行します。

4. FlowForce Server は、source.xmlをprocessed サブデルクトリン移動するステップを実行します。パスC:

\FFSERV\processed存在しなくてはなられてとご注意してくたさい。

13.5 リスト関数

リスト関数はリストを作成、および、分解するために使用されます。リストは常に、単一の型のアイテムを含んでいます(例えば、文字列のみ、数値のみ、ネストされた同じアイテムの型のリストなど)。

13.5.1 char

引数として与えられた数字のUnicode 文字を含む文字列を返します。例えば、char(10)は改行を返します。特定のUnicode 文字の数値コードを見つけるコキュード 関数を使用してくたさい。

署名

char(number:number) -> string

ジェネリックな署名

パラメーター

名前	型	説明
number	number	文字の数値コードこのコードはHTML 内で Unicode 文字を表すためご使用される数 値コードと同様です(例えば8734 は無限 のシンボルを表します)。

サンプル

次の実行ステップは無限シンボルを返します

実行ステップ
•
A 関数の実行 /system/compute
パラメーター: 条件式: char(8734)
= ステップの結果を割り当て名前
新規の実行ステップ 新規の選択ステップ 新規の For-e
実行の結果
戻り値の型を以下と宣言する: 文字列 ~

13.5.2 code

引数とて与えられた文字列の最初の文字のUnicode 値を返します。

署名

code(string:string) -> number

ジェネリックな署名

パラメーター

名前	型	説明
string	string	入カストリームを指定します。

サンプル

次の実行ステップはスペース文字を表す数値 32 を返します:

実行ステップ	
▲ 関数の実行 /system/compute	
パラメーター: 条件式: code('')	
= ステップの結果を割り当て名前 以下としてTO	
新規の実行ステップ 新規の選択ステップ 新規の For-each ステップ 新規	:=-
実行の結果	
戻り値の型を以下と宣言する:数値 🗸	

13.5.3 from-to

"from"と"to"の間の整数のノストを返します。"from"がto" よけき、場合、このノストは空です。

署名

from-to(from:number, to:number) -> list of number

ジェネリックな署名

パラメーター

名前	型	説明
from	number	開始インデックス("from")を指定します。
to	number	終了インデックス("to")を指定します。

サンプル

次の式は[3, 4, 5, 6, 7]を生成します:

from-to(3, 7)

13.5.4 join

2番目のノストを各リストのペアの区切り出て使用し、最初の引数のノストを結合します。

署名

join(lists:list of lists, separator:list=empty list) -> list

ジェネリックな署名

名前	型	説明
lists	list of lists	ジョインするノストを指定します。引数12つ 以上のノストのノストである必要があります。 ネストされているすべてのノストロに型であ る必要があります。
separator	list	ジョインされたリストを区切るセレーターを 指定する任意の引数与えらえていない場合、セレーターは使用されません。

名前	型	説明
		セ レーターは型 リスト である必要がみます。 リスト 関数を使用してセ レーターを作成します。 例えば 式 list(',') は単 ーのエンマ文字をセ レーターとして指定します。

サンプル

次の実行ステップは2つのノストをジョインする方法を説明しています。ステップ1 は最初のノストを作成します。ステップ2 は2番目のノストを 作成します。ステップ3 は両方のノストを含む」リストのノスト "のオブジェイトを作成します。 最後に、ステップ4 はセミコロン文字をセ レーター として使用しリストをジョインします。

実行	ステップ	
(
4	関数の実行 /system/compute	
	パラメーター: 条件式: list('a', 'b')	
=	、 ステップの結果を割り当て list1	× 以下としてTO
4	関数の実行 /system/compute	
	パラメーター: 条件式: list('c', 'd')	
=	ステップの結果を割り当て list2	× 以下としてTO
-	関数の美行 /system/compute	
	パラメーター: 条件式: list (<i>list1, list2</i>)	
=	ステップの結果を割り当て list_of_lists	× 以下としてTO
	関数の実行 /system/compute	
	パラメーター: 条件式: join(list_of_lists, list(':'))	
=	ステップの結果を割り当て 名前	以下としてTO

13.5.5 length

リスト内のアイテムの数を返します。

署名

```
length(list:list) -> number
```

ジェネリックな署名

パラメーター

名前	型	説明
list	list	入カリストオブジェクトを指定します。

13.5.6 list

単一アイテムのリストを構築します。全てのアイテムは同じ型である必要があり、、結果のリストはその型のアイテムのリストです。

署名

list(item1:any type, item2:any type, itemN:any type) -> list

ジェネリックな署名

パラメーター

名前	型	説明
item1	any type	単一のアイテムを指定します。次のアイテム はエンマイことに区切られる必要があります。
item2	any type	上記と同様です。
itemN	any type	上記と同様です。

サンプル

次の式はリスト[1, 2, 3]を返します。すべてのリストアイテムは数値の型です:

list(1,2,3)

次の式はリスト['a', 'b', 'c']を返します。すべてのリストアイテムは文字列の型です:

list('a','b','c')

13.5.7 nth

リストから指定されたアイテムを返します。インデックスは、ゼロベースです。インデックスか範囲外の場合は失敗します。

署名

nth(list:list, index:number) -> item

ジェネリックな署名

パラメーター

名前	型	説明
list	list	入カリストを指定します。
index	number	返されるアイテムのゼロベースインデックスを 指定します。

サンプル

次の式ば'b"を返します:

```
nth(list('a', 'b', 'c'), 1)
```

13.5.8 slice

リストからリストの一部を返します。

署名

slice(list:list, start:number, end:number=length(list)) -> list

ジェネリックな署名

名前	型	説明
list	list	入カリストを指定します。
start	number	スライス内に含まれる最初のノストアイテムの ゼロベースのインデックスを指定します。
end	number	スライス内で無視される最初のアイテムのゼ ロベースのインデックスを指定します。

サンプル 次の式はリスト(2,3)を返します:

slice(list(1,2,3,4),1,3)

13.6 文字列関数

文字列関数は結合、文字列からのサブ文字列の抽出、トリミング、分割などの基本的な文字列のオペレーションを行います。

13.6.1 concat

引数とて与えられた文字列を単一の文字列に結合します。string-join 関数を使用して、型"文字列のノスト"のオブジェクトのすべてのアイテムを結合します。

署名

concat(string1:string, string2:string, stringN:string) -> string

ジェネリックな署名

パラメーター

名前	型	説明
string1	string	ジョインする単一の文字列を指定します。 次のアイテムはコンマにより区切られる必要 があります。
string2	string	上記と同様です。
stringN	string	上記と同様です。

サンプル

次の式ばfile ~を返します:

```
concat('a', 'b', 'c')
```

13.6.2 contains

最初の文字列がサブストリングの発生を1度でも含む場合、TRUE を返します。

署名

contains(string:string, substring:string) -> Boolean

ジェネリックな署名

式関数

パラメーター

名前	型	説明
string	string	入力文字列
substring	string	チェックする文字列の値

サンプル

次の式はtrueを返します:

contains('cat','a')

次の式はfalseを返します:

contains('cat','b')

13.6.3 ends-with

string 引数内で与えられた文字列がend 引数内で与えられた文字列で終わる場合、true を返します。

署名

ends-with(string:string, end:string) -> Boolean

ジェネリックな署名

パラメーター

名前	型	説明
string	string	入力文字列
end	string	チェックする文字列の値

サンプル

次の式はtrueを返します:

ends-with('cat', 't')

次の式はfalseを返します:

ends-with('cat', 'a')

13.6.4 find-all

pattern か正規表現の箇所で、文字列内の全てのpattern の発生を抽出します。

署名

find-all(string:string, pattern:string) -> list of string

ジェネリックな署名

パラメーター

名前	型	説明
string	string	入力文字列
pattern	string	正規表現とての、ターシ

サンプル

次の式は文字列"apollo"からのすべての"。"の発生を抽出します。

find-all('apollo', 'o')

結果は文字列の次のノストです: ["o", "o"]

13.6.5 number

文字列の数字表記を計算します。例えば、引数として与えられてしる文字列を数値に変換します。

署名

number(string:string) -> number

ジェネリックな署名

パラメーター

名前	型	説明
string	string	変換する入力文字列の値

サンプル

次の式は文字列の値"1 ″を数値 1に変換します:

number('1')

13.6.6 split

引数とて与えられているセノシーターの発生ごとて文字列を分割します。

署名

split(string:string, separator:string) -> list of string

ジェネリックな署名

パラメーター

名前	型	説明
string	string	入力文字列
separator	string	セレーター文字列

サンプル

次の式はリスト["1", "2", "3"]を返します:

split('1;2;3', ';')

13.6.7 starts-with

文字列引数内で与えられた文字列が開始引数内で与えられた文字列で終わる場合、trueを返します。

署名

starts-with(string:string, start:string) -> Boolean

ジェネリックな署名

名前	型	説明
string	string	入力文字列
start	string	チェックする文字列の値

サンプル

次の式はtrueを返します:

starts-with('cat', 'c')

次の式はfalseを返します:

starts-with('cat', 'b')

13.6.8 string

与えられた数値の文字列表示を計算します。例えば、引数として与えられている数値を文字列に変換します。

署名

string(number:number) -> string

ジェネリックな署名

パラメーター

名前	型	説明
number	number	文字列に変換する数値

サンプル

次の式は数値"1"を文字列1に変換します:

string(1)

13.6.9 string-join

引数とて与えられた文字列のノストを単一の文字列にジョインします。任意で、各文字列間に引数とて与えられているセルーターを挿入します。

署名

string-join(list:list of string, separator:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明
list	list of string	文字列の入力リスト
separator	string	任意の引数すべてのジョインされている文字 列を区切るセルーターを指定します。

サンプル

次の式は文字列a;b;cを返します。

```
string-join(list('a', 'b', 'c'), ';')
```

13.6.10 string-length

文字列内の文字の数量を返します。

署名

```
string-length(文字列:string) -> number
```

ジェネリックな署名

パラメーター

名前	型	説明
文字列	string	入力文字列

サンプル

次の式は3を返します:

string-length('cat')

13.6.11 substring

指定された、開始文字のポジョンから、終了文字のポジョンまでの文字列からサブ文字列を返します。開始と終了インデックスはゼンベースです。

設定されていない場合はend は与えられた文字列の長さです。

end 引数は、負の整数であることもできます。負の値 -n は 文字列から最後のn 文字をトリムすること」を意味します。

署名

substring(string:sting, start:number, end:number) -> string

ジェネリックな署名

パラメーター

名前	型	説明
string	sting	入力文字列
start	number	ゼロベースの開始インデックス
end	number	ゼロベースの終了インデックス

サンプル

次の式は"Force"を返します:

substr('FlowForce',4)

次の式は"t" を返します:

substr('Altova',2,3)

次の式は"ltov"を返します:

substr('Altova',1,-1)

13.6.12 trim

先頭および末尾スペースを文字列から削除します(スペース、タブ、開業、リターン、フォームフィード、および垂直タブ)。

署名

trim(string:string) -> string

ジェネリックな署名

名前	型	説明
string	string	入力文字列

13.6.13 trim-start

引数として与えられている文字列から先頭の空白を削除します(トリム関数も参照してくたさい)

署名

trim-start(string:string) -> string

ジェネリックな署名

パラメーター

名前	型	説明
string	string	入力文字列

13.6.14 trim-end

引数として与えられている文字列から末尾の空白を削除します(トリム関数も参照してくたさい)

署名

```
trim-end(string:string) -> string
```

ジェネリックな署名

名前	型	説明
string	string	入力文字列

13.7 ブール値関数

TRUE/FALSE 式を検証するためブール値関数を使用します。

13.7.1 all

全てのブール値がtruevの場合 true を返します。最初のfalse 値の後に評価を停止し、falsevを返します。

署名

all(booVal1:Boolean, boolVal2:Boolean, boolValN:Boolean) -> Boolean

ジェネリックな署名

パラメーター

名前	型	説明
booVal1	Boolean	評価するブール値を指定します。次のアイテ ムはエンマにより区切られる必要があります。
boolVal2	Boolean	上記と同様です。
boolValN	Boolean	上記と同様です。

13.7.2 any

全てのブール値がFueの場合Fueを返します。最初のFue値の後に評価を停止します。すべての値がfalseの場合、falseを返します。

署名

any (boolVal1:Boolean, boolVal2:Boolean, boolValN:Boolean) -> Boolean

ジェネリックな署名

名前	型	説明
boolVal1	Boolean	評価するブール値を指定します。次のアイテ ムまこンマーこと、区切られる必要があります。
boolVal2	Boolean	上記と同様です。
boolValN	Boolean	上記と同様です。

13.7.3 false

ブール値 falseを返します。

署名

false() -> Boolean

ジェネリックな署名

13.7.4 if

ブール値がtrue の場合、valueTrueを返し、false の場合、valueFalse を返します。選択されたサブ式のみが評価されます。両方のサブ式は戻り値でもある同じ型である必要があります。

署名

if (condition:Boolean, valueTrue:any type, valueFalse:any type) -> any type

ジェネリックな署名

パラメーター

名前	型	説明
condition	Boolean	評価する条件を指定します。
valueTrue	any type	条件 がtrueを評価した場合、返されるサ ブ式を指定します。
valueFalse	any type	条件がfalseを評価した場合、返される サブ式を指定します。

サンプル

次の式はXMLスキーマン準拠した値とてブール値をパレます:

if(b, "true", "false")

これを行う代替法は以下の通りです

if(b, "1", "0")

13.7.5 not

引数として与えられたブール値の否定型を返します

署名

not(value:Boolean) -> Boolean

ジェネリックな署名

パラメーター

名前	型	説明
value	Boolean	否定するブール値を指定します。

13.7.6 true

ブール値 trueを返します。

署名

true() -> Boolean

ジェネリックな署名

13.8 ランタイム情報関数

ラノタイム情報関数は、現在実行されているジョブの詳細を処理するために使用されます。

13.8.1 instance-id

各ジョブの実行のナーダニー意の文字列を返します。これは、文字列かディレケー」名を定義するナーダニ使用される、各ジョブの実行のナーダニー 意のディレケーを作成するナーダニュと使用することができます。

署名

instance-id() -> string

ジェネリックな署名

13.8.2 log

引数として受信された式を変換し、システムログに書き込みます。この関数はステップにより生成される式を明示的にログする場合などのシ チュエーションで役に立ちます。この方法での値をログすると、ログされた値が長すぎる場合、システムログ内で切り捨てか発生しません、ログ <u>へ設定</u>も参照してくたさい。

署名

log(expression:T0) -> string

ジェネリックな署名

パラメーター

名前	型	説明
expression	то	FlowForce 式はログされます、型 IO (全ての型)。

サンプル

以下のようないなからファイルのリストを取得するジョブが作成されたと仮定します。

E	xecu	tion Steps
		Execute function /system/compute
		Parameters: Expression: list-files('C:\FlowForceExamples\LogFunction\source')
	=	Assign this step's result to name as TO

ソースディレオリ内のファイル数がリストのオメクのFlowForce デフォルトのログレミナを超えた場合、ジョブログ内のエトリは省略されます。 下記の通り、このサンプルでは、最初の10ファイル名のみが表示されます。また、ノマロちの文字のデフォルトの制限を超えるオメ各ファイルパスの最後の文字は切り詰められます。

Date	Message		
2020-09-21 15:20:17	Starting instance 67.		
2020-09-21 15:20:17	Starting job execution: job /public/Examples/Functions/log-function in queue /public/Examples/Functions/log- function		
2020-09-21 15:20:17	Running instance 67 locally.		
2020-09-21 15:20:17	Job /public/Examples/Functions/log-function		
2020-09-21 15:20:17	System function /system/compute		
2020-09-21 15:20:17	Computed list("C:\FlowForceExamples\LogFunction\source\File 01.tx[]", "C:\FlowForceExamples \LogFunction\source\File 02.tx[]", "C:\FlowForceExamples\LogFunction\source\File 03.tx[]", "C:\FlowForceExamples\LogFunction\source\File 04.tx[]", "C:\FlowForceExamples\LogFunction \source\File 05.tx[]", "C:\FlowForceExamples\LogFunction\source\File 06.tx[]", "C:\FlowForceExamples\LogFunction\source\File 07.tx[]", "C:\FlowForceExamples\LogFunction \source\File 08.tx[]", "C:\FlowForceExamples\LogFunction\source\File 09.tx[]", "C:\FlowForceExamples\LogFunction\source\File 07.tx[]", "C:\FlowForceExamples\LogFunction \source\File 08.tx[]", "C:\FlowForceExamples\LogFunction\source\File 09.tx[]",		
2020-09-21 15:20:17	Finished job execution: job <u>/public/Examples/Functions/log-function</u> in queue <u>/public/Examples/Functions/log-</u> function		

省略を回避するためこしの 関数内の式を囲み、ジョブの構成を保存します。

Execu	ution Steps		
+			
	Execute function /system/compute		
	Parameters: Expression:	log(list-files('C:\FlowForceExamples\LogFunction\source'))	
=	Assign this step's result to	name as TO	

システムによりログ済みのエトリニ加え、新規の構成と共にショブを実行すると、ログココログ済みの式のための新規のエトリか含まれます。切り捨ていまう発生しません。

Date	Message		
2020-09-21 15:21:41	Starting instance 68.		
2020-09-21 15:21:41	Starting job execution: job <u>/public/Examples/Functions/log-function</u> in queue <u>/public/Examples/Functions/log-function</u>		
2020-09-21 15:21:41	Running instance 68 locally.		
2020-09-21 15:21:41	Job /public/Examples/Functions/log-function		
2020-09-21 15:21:41	System function /system/compute		
2020-09-21 15:21:41	step 1 parameter 'Expression': C:\FlowForceExamples\LogFunction\source\File 01.txt C:\FlowForceExamples\LogFunction\source\File 02.txt C:\FlowForceExamples\LogFunction \source\File 03.txt C:\FlowForceExamples\LogFunction\source\File 04.txt C:\FlowForceExamples \LogFunction\source\File 05.txt C:\FlowForceExamples\LogFunction\source\File 06.txt C:\FlowForceExamples\LogFunction\source\File 07.txt C:\FlowForceExamples\LogFunction \source\File 08.txt C:\FlowForceExamples\LogFunction\source\File 09.txt C:\FlowForceExamples \LogFunction\source\File 10.txt C:\FlowForceExamples\LogFunction\source\File 11.txt C:\FlowForceExamples\LogFunction\source\File 12.txt C:\FlowForceExamples\LogFunction \source\File 13.txt C:\FlowForceExamples\LogFunction\source\File 14.txt C:\FlowForceExamples \LogFunction\source\File 15.txt C:\FlowForceExamples\LogFunction\source\File 16.txt C:\FlowForceExamples\LogFunction\source\File 17.txt C:\FlowForceExamples\LogFunction \source\File 18.txt C:\FlowForceExamples\LogFunction\source\File 19.txt C:\FlowForceExamples \LogFunction\source\File 12.txt C:\FlowForceExamples\LogFunction \source\File 18.txt C:\FlowForceExamples\LogFunction\source\File 19.txt C:\FlowForceExamples \LogFunction\source\File 20.txt C:\FlowForceExamples\LogFunction\source\File 21.txt		

このサンプルでは、log 式関数の呼び出しの代替として、ログするステップ・デメーターの横の ベログの有効化 2 パタンをクトックすることが

できます。これを行うことは10g 関数の使用と同様で、次回ジョブ構成ページを開くと、FlowForce は10g 関数を隠します。 2 パシンと 10g 関数の違いは、前者はテキストボックス内の式全体をログし、10g 関数はかさ、サブ式などのために厳選して使用することができます。 例

実	行	ステップ				
	+					
	4	関数の実行 /sy	stem/comp	ute		
		パラメーター:	条件式:	contains(log("haystac	:k"), "nee	dle")
	=	ステップの結果	を割り当て	名前		以下としてTO

13.8.3 slot-number

現在ジョブを実行しているキューの実行スロット番号を返します。ファイル名とて数値を使用してはなりません。平行ジョブを実行するけっとサー・ドープクセスするけっぱる男は使用されます(簡単にロードをパランスするけの)。

スロナキ実行が開始されたキューインとリスロナ番号は異なります。現在のショブ型のショブにとい呼び出される場合、呼び出しジョブのスロナ番号が継承されます。

署名

slot-number() -> number

ジェネリックな署名

13.9 AS2 式関数

AS2 式関数は遠隔のサーバーにAS2 メッセージを送信するジョブに適用することができます。AS2 統合を参照してくたさい。

13.9.1 as2-disposition

<u>/as2/send</u> 関数によし返された<u>MDN</u> から配置へッダー値を抽出します。同期の失敗の通知が返されるトランスミッションに失敗しない限し、ヘッダーの値は受信されたとおし返されます。処理値のサンプル

automatic-action/MDN-sent-automatically; processed/error: decryption-failed

署名

as2-disposition(result:AS2 MDN) -> string

ジェネリックな署名

パラメーター

名前	型	説明
result	AS2 MDN	型AS2 MDN 0值

13.9.2 as2-http-status

<u>/as2/send</u> 関数によし返された MDN から HTTP ステータスを抽出します。成功した MDN のためのHTTP ステータスは 200 範囲内です。 失敗した MDN は、 HTTP レベルで失敗した場合は、異なるステータスを含んでしる場合があり、 おた、 HTTP レスポースか受信されないというを含みます。

署名

```
as2-http-status(result:AS2 MDN) -> number
```

ジェネリックな署名

名前	型	説明
result	AS2 MDN	型 AS2 MDN 0值

13.9.3 as2-mdn-serialize

(更に処理して他の場所で保管できるようコシリアル化できるようコMDN をストリームとして返します。

署名

as2-mdn-serialize(result:AS2 MDN) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
result	AS2 MDN	型 AS2 MDN の 直

13.9.4 as2-message-id

<u>/as2/send</u> 関数により返された MDN からメッセージ ID 値を抽出します。この ID は MDN の ID とお同じではありません。失敗した MDN のために、メッセージ ID は空の文字列場合かあります。この関数はログのために役に立つ場合かあります。

署名

as2-message-id(result:AS2 MDN) -> string

ジェネリックな署名

パラメーター

名前	型	説明
result	AS2 MDN	型 AS2 MDN の直

13.9.5 as2-partner-local-name

AS2 メッセージを受け取るショブ内では、受信する側のAS2 パトナーの名前を取得するためにの関数を呼び出すことができます。AS2 パトナー構成ページ内の ローカル側設定」で定義されるAS2 名前です。AS2 パトナー名を抽出するこは、/system/computestring まさは、system/compute ビルドイン関数を実行する実行ステップを追加します:

/system/compute-string

{as2-partner-local-name(partner) }

/system/compute

as2-partner-local-name(partner)

partner カ型 AS2 パートナーの入力パラメーターの名前である個所。

型 AS2 パートナーの入力パラメーターはジョブ構成ページに自動的に追加されます。チェックボックス このジョブを URL. で HTTP を介し て使用できるようにするを選択し AS2 サービスを選択します。このようなジョブの詳細に関しては AS2 <u>メッセージの受信</u>を参照してくたさい。

署名

as2-partner-local-name(partner:AS2 Partner) -> string

ジェネリックな署名

パラメーター

名前	型	説明
partner	AS2 Partner	ローカル名が抽出される型 AS2 パートナー のオブンエクトを指定します。

13.9.6 as2-partner-remote-name

AS2 メッセージを受け取るジョブ内では、送信する側のAS2 パトナーの名前を取得するためにの関数を呼び出すことができます。AS2 パトナー構成ページ内の「パトナー設定」で定義されるAS2 名前です。AS2 パトナー名を抽出するには、/system/computestring おけは、/system/compute ビルドイン関数を実行する実行ステップを追加します:

/system/compute-string

{as2-partner-local-name(partner)}

/system/compute

as2-partner-local-name(partner)

partner が型 AS2 パートナーの入力パラメーターの名前である個所。

型 AS2 パートナーの入力パラメーターはショブ構成ページに自動的に追加されます。チェックボックス このジョブを URL. で HTTP を介し て使用できるようにするを選択し AS2 サービスを選択します。このようなジョブの詳細に関しては AS2 メッセージの受信を参照してくたさい。

署名

as2-partner-remote-name (partner: AS2 Partner) -> string

ジェネリックな署名

パラメーター

名前	型	説明
partner	AS2 Partner	リモート名が抽出される型 AS2 パートナー のオブジェクトを指定します。

サンプル

下で説明されるジョブは送信するAS2パートナーの名前を取得します。このジョブは<u>AS2メッセージの受信</u>内で説明されているサンプルよ ・
比若干複雑です。

Job Input Parameters								
•								
Nam	Name: partner Type:		AS2 partner	T	Description:	This parameter receives the partner object		
+								
Nam	e: mess	age		Туре:	stream	T	Description:	This parameter receives the stream object
+								
Exect	ition	Ster	ns.					
	ation	Step						
Ĭ	Execut	te funct	tion /system/f	ilesysten	n/copy			▼ 2
	Param	eters:	Source:		{as-file(message)}			
			Target:		{substring(current	t-message	-id(), 1, -1)}.ms	g
			Overwrite:					
			Abort on err	or:	√			
	Working directory: C:\as2\incoming							
=	Assign	n this st	ep's result to	name		as boolea	n	
+	_							
▲ Postpone								
	Execute function /system/compute-string							
Parameters: Expression: {as2-partner-remote-name(partner)}								
	= Assign this step's result to sending_partner as string							
	Everyte function /system/mail/send							
	Parameters: From: flauferer@leartheat							
		Faran	To:		name@evamp	le org		
			Subj	ect:	New AS2 mes	sage was re	ceived from (s	endina partner}
			Mess	sage bod	ly: +	age nas re	cerred noning	enang_paratery
			Attac	thment:	(+)			
			Abor	t on erro	or: 🔶			

上記のジョブはつのがAS パートナーオブジェクトである2つの入力パラメーターを有しています。ジョブはつの実行ステップから構成されています:

- 最初のステップはAS2 メッセージをファイルに保存し返します。
- 2番目のジョブはas2-partner-remote-name 関数を使用して partner 入力パラメーターから AS2 パトトナーの名前 を抽出します。
- 3番目ののジョブは受信者 name@example.org に新規のAS2 メッセージが以前に計算されたパートナー名から受信されたことを通知します。

2番目と3番目のステップは延期されたブロックにより囲まれていることに注意してくたさい。これは送信先のパトナーへのかッセージ処理通知 (MDN)の送信が遅延ませばかかれること回避するために行われます。最初にごョブは延期されていないステップ(AS2 メッセージの保存)を 実行します次に結果 (MDN)を送信するパトナーイに返し、延期されたステップ2と3を実行するためにのみ続行されます。ステップ2と3 が何らかの理由で失敗する場合、問題はローカルで解決されますが、AS2パトナーへ送信されるレスポンスには影響を与えません。

延期されたステップの実行に関しては延期されたステップを参照してくたさい。

13.9.7 as2-success

MDN がランスミッションコ成功すると、true を返します。

HTTPトランスミッションコズ切すると、トランスミッションは成功とみなされます。(有効化されている場合)MDNは署名に対して検証され、 MDNは成功を示します。/as2/send のためゴエラー時には中断」がオンコジリ替えられていると、この関数を使用する必要はありません。

署名

as2-success(result:AS2 MDN) -> Boolean

ジェネリックな署名

パラメーター

名前	型	説明
result	AS2 MDN	型 AS2 MDN の直

13.9.8 as2-signed

MDN か署名済みで、署名の検証に成功している場合、true を返します。

HTTP レイヤーで失敗したーランスミッションは正確に署名されません。この関数は以下の場合必要です:

a. /as2/sendのためゴエラー時には中断」が有効化されており、

b.「署名済みのMDN リケエスト」オプションがAS2 パートナーのために有効化されている場合、<u>AS2 パートナーの構成を参照してくたさ</u>い。

署名

as2-signed(result:AS2 MDN) -> Boolean

ジェネリックな署名

名前	型	説明
result	AS2 MDN	型 AS2 MDN の 直

13.10 MIME 式関数

MIME 式関数はMIME (Multipurpose Internet Mail Extensions) エンティティのヘッダーを操作する必要なジョブに適用することが できます。

13.10.1 add-mime-header

追加されたへダーkey: value を持つ、ひりームを返します。この関数はキーを持つ既存のへダーを削除しません。

署名

add-mime-header(s:stream, key:string, value:string) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	ヘッダーが追加されるストリーム
key	string	キー値へアからのキー。
value	string	キー値へアからの値。

13.10.2 add-mime-headers

追加されたヘッダーからの全てのストリームを返します。

署名

add-mime-headers(s:stream, headers:list of (string, string)) -> stream

ジェネリックな署名

名前	型	説明
S	stream	入カストリームを指定します。
headers	list of (string, string)	追加される~ッダーのノスト・ノスト 関数を使 用してリストを作成します。

サンプル

サンプル次の式は 2つの、ダーを含むストリームを返します: Content-Disposition と Content-Transfer-Encoding.

13.10.3 current-message-id

AS2 シッセージの Message-ID ヘッダーフィールドを返します。AS2 リクエストを受信するオークは構成されナジョブ内でこの関数は使用され る必要があります。すなオタ、チェックボックス「このジョブを HTTP を介して URL 上で利用可能にする」がジョブ構成ページ内で選択さ れてしる必要があります。これ以外の場合、この関数は、新たに生成された Message-ID を返します。(新規のジョブインスタンスか実行さ れ、終了までそのジョブインスタンスのオークにつな場合、新規の値が生成されます)。

署名

current-message-id() -> string

ジェネリックな署名

サンプル

次の式はMessage-ID をベースコンナファイル名を作成します。substring 関数は、山かこ(最初と最後の文字)をMessage-ID から削除します。

C:\temp\{substring(current-message-id(), 1, -1)}.msg

次の式は、上記と同じように作動し、更に現在のMessage-IDを分割関数を使用して、文字 @ の箇所で分割します。nth 関数は、 ランダムな16進数の値を持つ32文字の長さの最初の部分のみを抽出し、ファイル名の一部として使用します。

C:\temp\{nth(split(substring(current-message-id(), 1, -1), '0'), 0)}.msg

13.10.4 get-mime-content-disposition-param

このようなヘッダーと、ラメーターが存在する場合、ストリームの "Content-Disposition" ヘッダーから、ノラメーター paramを返します。これ 以外の場合、 default 引数の値を返します。この関数はオプションのAS2 プロファイル File Name preservation (FN) に続くメッセー ジを受信し、 MIME ヘッダーからの元のファイル名を取得するために使用されます。

署名

get-mime-content-disposition-param(s:stream, param:string, default:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明
s	stream	入カストリームを指定します。
param	string	返すパラメーター名を指定します。
default	string	指定されているノウメーターとヘッダーが 存在しない場合返される値を指定します。 デンオルトではこれは空の文字列です。

サンプル

ストリームmsg が、ダーContent-Disposition: attachment; filename="GETMSG.edi"を含むと想定して、次の式は GETMSG.edi を返します:

get-mime-content-disposition-param(msg, "filename")

13.10.5 get-mime-content-id

存在する場合、Content-ID ヘッダーの値を引数とて与えられたストリームから返します。それ以外の場合、default の引数が返されます。

署名

get-mime-content-id(s:stream, default:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入カストリームを指定します。
default	string	Content-ID ヘッダーカ存在しない場合 返す値を指定します。デフォルトではこれは 空の文字列です。

サンプル

ストリームmsg が次の、ダーContent-ID: <root.attachment> を持つと仮定します。式は この場合

get-mime-content-id(msg, "")

″<rootattachment>″を返します。このようなヘッダーが存在する場合、上の式は空の文字列(2番目の引数の値)を返します。

13.10.6 get-mime-content-type-param

ヘッダーと ラメーターが存在する場合、ストリームの "Content-Type" ヘッダーから ラメーター param を返します。それ以外の場合、デ フォルトの引数の値を返します。任意のAS2 プロファイル Multiple Attachments (MA)に続くメッセージを受信するため この関数 は使用されます。具体的には、開始ドキュメト Content-ID と Content-Type を multipart/related コンテンツ型に対する ラメ ータ 'start'と'type'通りに抽出します。下のサンプルに示される通り、文字セトを抽出するために使用することができます。

署名

get-mime-content-type-param(s:stream, param:string, default:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入カストリームを指定します。
param	string	返すパラメーター名を指定します。
default	string	指定されているノウメーターが存在しない 場合返される値を指定します。デフォルトで はこれは空の文字列です。

サンプル

ストリームmsg はヘッダーContent-Type: text/html; charset=utf-8 を含むことを前提として、次の式はutf-8 を返します:

get-mime-content-type-param(msg, "charset", "ascii")
13.10.7 get-mime-header

ヘッダーが存在する場合、特定のMIME ヘッダーを現在のストリームから取得します。これ以外の場合は、デフォルトの引数を返します。

署名

get-mime-header(s:stream, key:string, default:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明	
s	stream	入カストリームを指定します。	
key	string	ヘッダーを構成するキー値へアからのキー。	
default	string	返すデオパトの値を指定します。デオパト ではこれは空の文字列です。	

サンプル

ペリームmsg が、ダー Content-Disposition: attachment; filename="GETMSG.edi" を含むこう前提して、次の式は attachment; filename=\"GETMSG.edi\" を返します:

get-mime-header(msg, "Content-Disposition", "")

このサンプルでは、ストリームがContent-Disposition ヘッダーを持たよ、場合、上の式は、(3番目の引数の値である)空の文字列を返します。

13.10.8 get-mime-headers

ストリームからすべてのMIME へが一を取得し、タプル(キー、値)のノストを返します。返されるリストはadd-mime-headers 式関数の headers パラメーターとして与えることができます。

get-mime-headers(s:stream) -> list of (string, string)

ジェネリックな署名

パラメーター

名前	型	説明
s	stream	入カストリームを指定します。

署名

13.10.9 get-stream-filename

引数と共に提供されたストリームがファイルから作成される場合、拡張子と共にストリームのファイル名を返します。それ以外の場合 デンォルトの引数の値を返します。

署名

get-stream-filename(**ストリーム:stream**, default:string="") -> string

ジェネリックな署名

パラメーター

名前	型	説明	
ネリーム	stream	入カストリームを指定します。	
default	string	返すデフォルトの値を指定します。デフォルト ではこれは空の文字列です。	

13.10.10 is-file

関数 as-file か既存のファイル名を返す場合、trueを返し、as-fileが一時ファイルを作成するとfalseを返します。

例えば、関数、おけま、stream-open 関数、おけま、マピングから返された関数を使用してストリームを作成するとtrue か返されます。ファイルからストリームがサーブされない場合、おけま、一時的なファイルの場合、この関数はfalseを返します。

署名

is-file(s:stream) -> Boolean

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入カストリームを指定します。

13.10.11 is-mime-content-type

ヘッダーストリームの"Content-Type" ヘッダーに一致し、カスタム定義済みのルールを受け入れます。"Content-Type" ヘッダーか存在 し、ルールが値に一致する場合、trueを返します。それ以外の場合は、falseを返します。"Content-Type」ヘッダーを持たないストリー ムは、"application/octet-stream" として扱われます。 受け入れルールは次の書式を取ります: 拡張された Backus-Naur form (EBNF) 反復:

```
Match ::= Single ("," Single)*
Single ::= Spaces? Type-Match ( Spaces? ";" Spaces? Parameter )* Spaces?
Type-Match ::=
   "*/*" |
   Type "/*" |
   Type "/*+" Suffix |
   Type "/" Subtype
Parameter ::= Name "=" Value
```

署名

is-mime-content-type(s:stream, accept:string) -> Boolean

ジェネリックな署名

ノデメーター

名前	型	説明	
S	stream	入カストリームを指定します。	
accept	string	カスタムで定義された受け入れルールを指 定します。	

サンプル

ストリームmsg がヘダーContent-Type: text/html; charset=utf-8 おけたContent-Type: text/plain; charset=utf-8を含む場合、次の式はtrueを返します。

is-mime-content-type(msg, "text/*; charset=\"utf-8\"")

ホリームmsg が、ダーContent-Type: application/rss+xml おさContent-Type: application/svg+xmlを含む場 合、次の式はrueを返します。

is-mime-content-type(msg, "application/*+xml")

コンマで区切り複数のリレールを一致させることができます。ストリームmsg かやッダーContent-Type: text/xml おけよ Content-**Type:** application/xmlを含む場合、次の式はrueを返します。

is-mime-content-type(msg, "text/xml, application/xml")

13.10.12 mime-content-encode

Content-Transfer-Encoding とてのエンコートをストリームsに適用します。

サポートされるエノコードは

式関数

空の文字列: "binary"に等しい。

- "base64": Base64 エノコード
- "guoted-printable": 引用された印刷可能なエンコード

他の文字列: エノコード無し

現在のContent-Transfer-Encodingを使用してストリームのエンコードし、指定されたコードを使用して再コードします。新規の Content-Transfer-Encoding は結果ストリームのヘッダー内に保管されます。

関数はノースエンコード内のエラーが報告されることを保証しません。

```
署名
```

mime-content-encode(s:stream, encoding:string="") -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
S	stream	入カストリームを指定します。	
encoding	string	適用するエンコードを指定します。デフォルト ではこれは空の文字列です。	

13.10.13 mime-flatten

MIME ペッダーを持つストリームを取り、コンテンソ内の元のペッダーを含むストリームニ変換します。結果するストリームは message/rfc822 のコンテンツ型を持ちます。

署名

mime-flatten(s:stream) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入カストリームを指定します。

13.10.14 mime-multipart

無限の数量のストリームを取り、単一のmultipart/subtype に結合します。

境界は自動的に作成されます。ストリームはアセンブ前にフラナ化されます。追加ノラメーターを持つマルチパートはサポートされません。

Flow Force Server Advanced Edition ユーザーのためのメモ AS2 型のマルチパートメッセージを定義しないため AS2 のため にサブ型は常に 関連付けられています。mime-multipart-related 関数を確認してくたさい。

署名

mime-multipart(subtype:string, s:stream) -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
subtype	string	使用する <i>multipart/</i> subtype を指定し ます。	
S	stream	入カストリームを指定します。	

サンプル

次の式はEDIファイルとPDFの2つのファイルを含むストリームを返します。

```
mime-multipart("related", stream-open("c:
\example\order.edi", "application/EDIFACT"), stream-open("c:
\example\measuredetails.pdf", "application/pdf"))
```

Execution Steps

+	
4	Execute function /system/compute
	Parameters: Expression: mime-multipart("related", stream-open("c:\example\order.edi", "application/EDIFACT"), stream-open("c:\example\measuredetails.pdf", "application/pdf"))
=	Assign this step's result to name as T0
ne	w Execution step new Choose step new For-each step new error/success handling step

13.10.15 mime-multipart-related

無制限の数のストリームを取り、multipart/relatedに結合します。境界は自動的に作成されます。ストリームはアセンブ」前にフラナ化されます。

FlowForce Server Advanced Edition ユーザーのためのメモ 任意のAS2 プロファイル Multiple Attachments (MA) の 後にくるメッセージをアセンブルするためこの関数を使用することができます。最初のストリームが主要なパトトはないます。存在したは 場合、 マルチパトト のアセンブル前に全てのパトトは作成された一意の値を持つ Content ID ヘッダーを取得します。new-message-id 関数 に返された通い作成された値は新規の Message-ID です。ソースストリームは影響を受けていません。

署名

mime-multipart-related(s:list of stream) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
S	list of stream	ホリームの入力リスを指定します。

サンプル

次の式は、2つのストリームを含むストリームを返します。

mime-multipart-related(list(part1, part2))

ジョ	ブ入力パラメー	ター			
◆ 名前:	part1	型: ストリーム		•	説明:
◆ 名前: ◆	part2	型: ストリーム		•	説明:
実行	ステップ				
•	関数の実行 /system/co	ompute			
	パラメーター: 条件式	add-mime-headers(emp	oty-stream(), list(('Co	ntent-[Disposit
=	ステップの結果を割り	当て名前	以下としてTO		

13.10.16 mime-multipart-from-list

ネリームのリストを取り、multipart/subtypeと組み合わせます。

署名

mime-multipart-from-list(subtype:string, s:list of stream) -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
subtype	string	使用する <i>multipart/</i> subtype を指定し ます。	
s	list of stream	ネリームの入力リスを指定します。	

13.10.17 mime-parse

ストリームs内に保管されているMIME メッセージを解析し、MIME ヘッダーとメッセージボディを分割します。必要な場合、"Content-Transfer-Encoding" ヘッダーゴばい暗号化が解除されているメッセージボディコンテンンを持つストリームを返します。MIME ヘッダーニ get-mime-header、is-mime-content-type などの式関数によりアクセスすることができます。mime-flatten 関数により実 行された項目を元に戻します。関数はノースストリーム内のエラーが報告されることを保証しません。

署名

mime-parse(s:stream) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
s	stream	入カストリームを指定します。

13.10.18 mime-split-multipart

ストリームsがMIME マルチパトトのケッセージの場合、この関数は分割し、ストリームのリストを返します。ストリームsがマルチパトトのケッセージではない場合、(すなオス、is-mime-content-type(s, "multipart/*")がfalseを返す場合)、関数はしつの要素-ストリームs(変更無し)のリストを返します。関数はノースストリーム内のエラーが報告されることを保証しません。

署名

mime-split-multipart(s:stream) -> list of stream

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入力ストリームを指定します。

13.10.19 new-message-id

Message-ID ヘッダーフィールドのオゴンの新規の値を生成し返します。MIME メッセージのヘッダーを作成するオゴンコンの値を使用することが できます。current-message-id とは異なり、この関数は、常に新規のMessage-IDを返します。Message-ID は次の書式を有 します:

```
'<' UTC timestamp '-' random hex value 32 characters long '@' host name related text '> '
```

例: <20180306154822808383-5933b654b26c4495bb0b619ab72b3bc6@myservername>.

署名

new-message-id() -> string

ジェネリックな署名

13.10.20 reset-mime-headers

新しい、ダーを持つストリームを返します。ヘッダーリスト無しに、全てのヘッダーをクリアします。

署名

reset-mime-headers(s:stream, headers:list of (string, string)=empty) -> stream

ジェネリックな署名

パラメーター

名前	型	説明
S	stream	入カストリームを指定します。

名前	型	説明	
headers	list of (string, string)	作成されるへッダーのノストを指定します。デ フォノトの値は空です。	

13.10.21 set-mime-content-disposition

s 内で検索されたMIME "Content-Disposition」 ヘッダーの デメーターを設定します。

FlowForce Server Advanced Edition ユーザーのためのメモ この関数は任意のAS2 プロファイル FileName preservation (FN)を使用して、AS2 メッセージの送信時に役に立ちます。ファイル名の読み取りのために、関数 get-mime-content-disposition-param も参照してくたさい。

署名

set-mime-content-disposition(s:stream, disposition:string, filename:string="") ->
string

ジェネリックな署名

パラメーター

名前	型	説明	
S	stream	入カストリームを指定します。	
disposition	string	設定する"Content-Disposition" へり ーのdisposition 値	
filename	string	設定する"Content-Disposition" ヘッダ ーのfilename 値デフォルト ではこれは空 の文字列です。	

サンプル

次の式は Content-Disposition ヘッダーは 以下のように設定します: Content-Disposition: attachment; filename="GETMSG.edi":

set-mime-content-disposition(msg, "attachment", "GETMSG.edi")

13.10.22 set-mime-content-id

value に設定されている "Content-ID" へッダーを持つストリームを返します。set-mime-header 関数を使用して同じ結果を達成する場合、この関数において直接なアプローチを表します。

署名

set-mime-content-id(s:stream, value:string="") -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
S	stream	入カストリームを指定します。	
value	string	["] Content-Disposition」内の値を指定 します。	

サンプル

ストリームmsg内のContentID ヘッダーの値を<root.attachment>に設定することを前提とます。これを行うコよ次の式を使用します:

set-mime-content-id(msg, "<root.attachment>")

13.10.23 set-mime-header

ヘッダー キーが 値に設定されているストリームを返し、その他のヘッダーとエンテンソコよ影響はありません。一度に複数のヘッダーを変更する場合 set-mime-headers 関数を使用する必要かあります。

署名

set-mime-header(s:stream, key:string, value:string) -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
s	stream	入カストリームを指定します。	
key	string	設定するヘッダーのキーを指定します。	
value	string	設定するヘッダー値を指定します。	

サンプル

"Content-Type" ヘッダーをオーノーライドするコよ 次を使用します:

set-mime-header(s, "Content-Type", "text/plain; charset=iso-8859-1")

13.10.24 set-mime-headers

ヘッダーからのキー値ペアによるヘッダー引数を持つストリームを返します新しいヘッダーは同じ名前を持つ既存のヘッダーをシャドウします。

署名

set-mime-headers(s:stream, headers:list of (string, string)) -> stream

ジェネリックな署名

パラメーター

名前	型	説明	
s	stream	入カストリームを指定します。	
headers	list of (string, string)	設定されるへダーのノスト	

サンプル

"Content-Type" ヘッダーをオーバーライドするコよ 次を使用します:

set-mime-headers(s, list(("Content-Type", "text/plain; charset=iso-8859-1")))

14 ジョブの構成の例

このチャプター・コはステップがイステップのFlowForceジョブ構成サンプルが含まれています。下のテーブルは特定の関数の種類と各サンプル て説明されているトリガーと共にすべてのサンプルをリストしています。

/mi	説明される概念			
191	内蔵された関数	式関数	トリガー	
「 <u>Hello, W orld!」ジョブの作成ジョブ</u>	 /system/compute 		Web サービス	
<u>ノ やか存在するか確認する</u>	/system/shell/commandline/system/compute-string	content() stdout() trim()	Web サービス	
ファイルをコピーする	 /system/filesystem/copy 	list-files()	Web サービス	
<u>MapForce マピンブからショブを作成す</u> る	● MapForce マッピング		タイマー	
ジョブを他のジョブのステップとて使用する	 /system/filesystem/copy 			
ディレナリポーレグジョブの作成	・ MapForce マッピング ・ /system/filesystem/move		ファイルシステム	
<u>エラー処理をジョブに追加する</u>	 /system/shell/commandline /system/mail/send 	failed-step() error-message() exitcode() stdout() stderr() content() instance-id()	Web サービス	
<u>Webサービスとしてジョブを公開する</u>	• MapForce マッピング		Web サービス	
<u>JSON をFlowForce Web サービス</u> <u>へポストする方法</u>	 /system/filesystem/copy 	as-file() instance-id()	Web サービス	
ジョブの結果のキャッシュ	/system/shell/commandline/system/compute	stdout()	Web サービス	
<u>StyleVision 変換からションを作成する</u>	 StyleVision 変換 /system/compute /system/filesystem/copy 	results() nth() as-file()	タイマー	
RaptorXMLを使用してドキュメトを検 証する	/RaptorXML/valany		タイマー	
エラーログのある XML を検証する	 /RaptorXML/valxml-withxsd /system/compute /system/filesystem/copy 	failed-step() stdout() as-file()	Web サービス	
RaptorXMLを使用してXSLTを実行 する	• /RaptorXML/xsIt	list()		
<u>XML ファイルからPDF を生成</u>	 MapForce マッピング StyleVision 変換 /system/compute 	as-file() results() filename()	Web サービス	

14.1 "Hello, World!" ジョブの作成

この例では、テキスト "Hello, W orld!" をブラウザー内に出力する簡単なジョブの作成方法が説明されています。このテキストは、 FlowForce 式を使用して作成されます。ブラウザー内のレンをクトックすることにより、必要に応じてジョブをトリガーすることができます(つましり、ジョブはW eb サービスとして公開されます)。

必要条件

- 必要とれるライセス FlowForce Server
- FlowForce Web Server とFlowForce Server サービスか構成された<u>ネットワークアドレスとポート</u> でリッスンされていること。
- <u>コンテナー</u>の1つでパーミッションを有する FlowForce Server ユーザーアカウトを有していること(デフォルトでは、この例で使用されている/public コンテナーイコは認証されたユーザーがアクセスすることができます)。

ジョブの作成方法

- 1. FlowForce Server コンプオンレ /public コンテナーインナビゲートします。
- 2. 作成 | コンテナーの作成 をクリック て[Examples] とう名前の新規のコンテナーを作成します。

/public/Examples コンテナーかごのドキュメントで説明されるジョブの多くにより使用されています。他のコンテナーでジョブ を作成することも可能であすが、このドキュメント内の次に続くチュートリアルもフォローする必要があり/public/Examples コンテナーを作成することが奨励されます。

3. /public/Examples コンテナーで作成 | ジョブの作成をクリックして、ジョブのタイトルと説明を入力します。

以下にジョブを作成 /public				
ジョブ名:	HelloWorld			
ジョブの説明:	OUtputs the text "Hello, World!"			

4. ビルトイン関数 ①/system/compute を呼び出す新規の実行ステップを追加します。

実行ステップ				
	+			
	4	関数の実行 /system/compute		
		バラメーター: 条件式: <i>条件式</i>		
	=	- ステップの結果を割り当て名前 としてTO		

5. 「式」フィールドに、一重引用符で囲まれた、テキスト 'Hello, W orld' を入力します。このフィールドのエンテンソよ FlowForce Server 式を示します。

実行ステップ					
	•				
	4	関数の実行	/system/cor	npute	
		バラメーター:	条件式:	"Hello, World!"	
	=	ステップの結果	」 果を割り当て	名前	として T0

6. 実行の結果を「文字列」として宣言します。

実行の結果	
戻り値の型を以下と宣言する: 文字	列

7. HTTP を介してこのジョブを公開する... チェックボックスを選択し "HelloW orldService" をサービス名として入力します。詳細に関しては Web サービスをジョブとして公開を参照してくたさい。



- 8. 「資格情報」から、既存の資格情報の記録を選択、おけ、ローカルの資格情報を指定してくたさい。詳細に関しては資格情報を参照してくたさい。
- 9. [保存]をクルクします。

ジョブを実行する方法

文字列の植 "Hello, World!"を計算し、ジョブの結果として返すジョブの作成が完了しました。ジョブを実行するこよ、以下の内のしつを行ってくたさい

- ホームニ移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/HelloWorldService</u>をブラウザーのアドレスドーに入力します。このURLは FlowForce Server サービスがデンオルトのオストアドレスとポート名でリンスンする場合のみ機能します。 構成ページ内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してください。
- FlowForce Server の任意のホスト名 フィールドをセオアップページから設定するとHTTP を介してこのジョブを使用可

能にする チェックボックスの横の と ボタンをクリックして web サービスの呼び出しを直接ジョフ構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスこアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コログオンするために使用する同じ資格情報を

ジョブの構成の例

提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウトを使用してWebサービスにアクセスします。HTTP 認証を無効化するには、Webサービスを公 開し、Service - Use パーミッションをユーザーAnonymous に与えます。パーミッションのしくみを参照してください。

ジョブの実行に成功すると、ブラウザーは、ジョブの出力を表示します

Hello, World!

ジョブの実行に失敗するとブラウザーは、「サーイ、実行の失敗」メッセージを表示します。この場合、FlowForce Server 内の<u>ジョブのログ</u>をチェックして、エラーを確認してくたさい。ジョブのログの確認を参照してくたさい。

14.2 パスが存在するか確認する

この別では、(ファイルまけはディレクトリーの)、ながオペレーティングシステム上に存在するか通知するジョブを作成する方法が説明されています。この目的を達成するために、ビルトイン関数と式関数の組み合わせを使用する必要があります。ジョブをブラウザーからURL にアクセスすることにより、需要に応じてトリガーするために、Webサービスとして定義されます。ジョブは、パを引数として取り、FlowForce Serverが作動するオペレーティングシステム上に引数として与えたれたいなが存在するか通知する文字列を返します。

必要条件

- 必要とされるライセンス FlowForce Server
- FlowForce Web Server とFlowForce Server サービスか構成されたホトワークアドレスとポート でかえこされていること。
- <u>コンテナー</u>の」つでパーミッションを有する FlowForce Server ユーザーアカウトを有していること(デフォルトでは、この例で使用されている/public コンテナートコよ認証されたユーザーカアクセスすることができます)。

ジョブの作成方法

- FlowForce Server Iコングオンし、新しいジョブの作成のペーミッションか与えられているコンテナーに移動します。他のサンプルとの 一貫性を保づためここのチュートリアルでは/public/Examples コンテナーを使用します。このコンテナーが存在しない場合作 成│コンテナーの作成コマイドを使用して作成してください。
- 2. /public/Examples コンテナー内で作成をクリックしてジョブの作成を選択します。
- 3. ジョブの名前を追加し、(この例では、CheckPath)任意でショブの説明を追加します。

以下に	ジョブを作成 /public
ジョブ名:	CheckPath
ジョブの説明:	Checks if a path exists.

4. ジョブ入カハラメーターから、 きをクリックし、「ハマリハラメーターを以下のように追加します。

ジョ	ブ入カパラ	<u> </u>
名	前: path	型: ファイルまたはディレクトリとしての文字列 🔽 デフォルト: 🔸 説明: Specifies the path to be checked.
•		

5. ①/system/shell/commandline 関数を呼び出す新規の実行ステップを追加し、ファイルの存在を確認するシェルコマドを 入力します。このステップの結果を、下に示されるように宣言してくたさい、この例では、output と呼びます)。

実行ステップ ・					
	4	関数の実行	/system/shell/comma	ndline	
		バラメーター:	コマンド:	IF EXIST{path}(ECHO1)ELSE(ECHO0)	
			エラー時には中断:	+	
			作業用ディレクトリ:	•	
	=	ステップの 結界	表記のまた。 event	としてresult	

Windows では、パンカ存在する場合、シェルコマンドは「1」を出力し、パンカ存在しな、場合は、「0」を出力します。 FlowForce Server は、Unix システム上で作動しますが、コマンドを必要に応じて調整してくたさい。コマンドは FlowForce 式 {path} を埋め込んでいることに注意してくたさい。この式は、前のステップで定義されている入力パラメーターを 参照しています。

- 6. 「実行ステップ」の下の オタンをクリックし、「新規選択ステップ」を選択します。条件式として、 trim(content(stdout(output))) == '1' を入力します。この式は、次のネストされなっつ関数から構成されます :stdout、content、および trim。最初に、stdout 関数は、前のステップにはし返された結果の標準出力を取得します。そ して、content 関数は、標準出力を文字列に変換します。最後に、trim 関数は、先頭と末尾のスペース、改行、改行文 字を標準出力から削除します。結果は、等値演算子を使用して、「1」と比較されます。2つの値が等しいと、パンは存在します。 それ以外の場合、パンパは存在しません。
- 7. When 句の下に、以下に示されるように実行ステップを追加します。この実行ステップは、パンか存在する場合返される文字列の 値を構築するために「/system/compute-string 関数を呼び出します。値は、FlowForce 式 {path}を埋め込んでいる ことに注意してくたさい、この式は前のステップで定義された入力パラメーターを参照しています。

選択	
When	<pre>trim(content(stdout(output))) = = '1'</pre>
•	
4	関数の実行 /system/compute-string
	パラメーター: 条件式: Path {path} exists.

8. Otherwise 句の下に、以下に示されるように実行ステップを追加します。この実行ステップは、パンか存在しない場合返される 文字列の値を構築するために①/system/compute-string 関数を呼び出します。値は、FlowForce 式 {path}を埋め 込んでいることに注意してください。この式に前のステップで定義された入力パラメーターを参照しています。



9. 実行の結果で、戻り値の値を「文字列」として宣言します。

Ę	ミ行の結果		
	戻り値の型を以下と宣言する:	文字列	•

10. サービスから「このジョブをHTTP を介して利用可能にする」チェックボックスをクトックして選択し、CheckPathService をサービスの名前として入力します。詳細に関しては<u>Webサービスの公開</u>を参照してくたさい。

Se	ervice	
	Make this job available via HTTP at URL http://< <i>FlowForce server</i> >/service/	CheckPathService

- 11.「資格情報」から、既存の資格情報の記録を選択、おま、ローカルの資格情報を指定してくたさい。詳細に関しては資格情報を参照してくたさい。
- 12. [保存]をクリックします。

ジョブを実行する方法

ジョブを実行するコよ以下の内のしつを行ってくたさい

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/CheckPathService</u>をブラウザーのアドレスバーに入力します。このURL は FlowForce Server サービスがデフォルトのオストアドレスとポート名でリッスンする場合のみ機能します。 <u>構成ページ</u>内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してください。
- FlowForce Server の
 注意の
 ホスト名
 フィールドを
 セルアップページ
 から
 設定する
 とHTTP を
 介してこの
 ジョブを
 使用
 可

能にする チェックボックスの横の 🕒 ボタンをクトックして web サービスの呼び出しを直接ジョン構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server にログオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のオダの資格情報を提供することはテスト目的のオダです。生産のオダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウトを使用してWebサービスこアクセスします。HTTP認証を無効化するコよWebサービスを公開し、Service - Use パーミッションをユーザーAnonymous に与えます。パーミッションのしくみを参照してくたさい。

このジョブコお一数が存在し、ブラウザー内のWebサービスこアクセスすると引数を与えるようコプロンプトされます。

Parameters	
- 4 * 01	7
path *: C:\	
Cubmit	
Submit	

ジョブの実行に成功すると、ブラウザーはジョブの出力を表示します。例

Path C: $\langle exists. \rangle$

ジョブの実行に失敗するとブラウザーは、「サーバー実行の失敗」メッセージを表示します。この場合、FlowForce Server 内のジョブのログ をチェックして、エラーを確認してください。ジョブのログの確認を確認してください。

14.3 ファイルをコピーする

このサンプルは、FlowForce Server ジョブの助けを得てローカルファイルシステム上の複数のファイルをコピーする方法について説明しています。

C:\FlowForceExamples\CopyFiles\Sourceから新規のディレオリにC:\FlowForceExamples\CopyFiles\Target 全てのファイルをエピーすると仮定します。(UNIX システム上では、パンを適宜調整してくたさい)。この目的を達成するは、ディレオリ内の すべてのファイルを繰り返す「for-each」ステップを使用して、ループ内の各アイテムのために<u>/system/filesystem/copy</u> 関数を呼 び出します。

必要条件

- 必要とされるライセンス FlowForce Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ネットワークアドレスとポート</u> でルスンされていること。
- このジョブはC:\FlowForceExamples\CopyFiles\Source からディレオリC: \FlowForceExamples\CopyFiles\Target にファイルをユピーします。ジョブを作成する前にコーカルファイルシステム上にこれらのディレオ・小が作成されていることを確認してくたさい。また、ジョブをテストするためにノースディレオ・リにファイルが含まれていることを確認してくたさい。

ジョブの作成方法

FlowForce Server コングオノ、新しいショブの作成のペーミッションか与えられているコンテナーコ移動します。他のサンプルとの一貫性を 保つすっめここのチュートリアルでは/public/Examples コンテナーを使用します。このコンテナーが存在してい場合作成|コンテナーの作成コマンドを使用して作成してくたさい。

/public/Examples コンテナー内で新規のジョブを作成します。(例えば、「CopyFiles」などの、ジョブ名を入力し、オプションでジョブの 詳細を入力します。

	public	
以下にジョブを作成 /public		
ジョブ名:	CopyFiles	
ジョブの説明:	1つのディレクトリから他のディレクトリに複数のファイルをコピーします。	

リスト内のアイテムを繰り返すために、FlowForce Server はよ「for-each」実行ステップが搭載されています。このようなステップはシーケンス内の最後のアイテムを含むアイテムのシーケンス(リスト)を繰り返します。このサンプル内では、アイテムのシーケンスはノースディレクトリ 内のファイルのリストレゴンります。リクエストされたリストを作成するはよ「新規の実行ステップ」をクリックして、「関数の実行」の横に 「/system/compute」を入力します。下記の通り、ドロップダウノリストからこの、文を選択することもできます。

実	行スラ	゠ップ			
	+				
	▲ 関数	数の実行 /sys	stem/compute		▼ 🗗
	76E	ラメータ-	/ O system O		
	= 75	テップの新	名前 👌		
	新規の実行	テ ステップ	as2	^	
			🛅 filesystem		
美	行の統	課	🛅 ftp		
	戸り値の型	型を以下人	💼 mail		
		ECATC	🛅 maintenance		
64	-田の+		💼 shell		
杧	家のト	ーヤツ	🗊 abort		
	□ 結果	をキャッ	🗊 compute		

次に、式フィールド内に次の式を入力します

list-files("C:\FlowForceExamples\CopyFiles\Source*.*")

次に「このステップの結果を次に割り当てる」フィールドにリストの名前を入力します(この場合、名前は1ist)です)。これにより以降の ステップ内の新規のファイルのリストを簡単に参照することができます。実行ステップは以下のようてなります:

E>	ecu	ion Steps	
	+		
		xecute function //system/compute	
		Parameterry Everencion: Use (Use (Use Eleve Everenciae) Come Eleve and **)	
		anameters: Expression: (Ist-files(C:\FlowForceExamples\CopyFiles\source\",")	
	=	Assign this step's result to list as TO	
	(\bullet)		l

上記の式は<u>list-files</u> 式関数を呼び出します。関数はお|数(この場合、C:\Source*.*) としていなを取り、特定のいなでファイル(おこし、 デルノトリののオペズのファイルを選択するために、パマロオフイルドカード *.* かきまれていることに注意してくたさい。必要な場合、ワイルドカードを調節して、特定のファイル拡張子を選択することができます。例えば、 *.txt。FlowForce 内の式に関する詳細は、FlowForce 式を参照してくたさい。

実際の「for-each」繰り返しステップの作成を開始することができます。「新規のFor-Each ステップ」をクトックして、「シーケンス内」ボックス内に1istを入力します。(これは前の実行ステップで作成された1istを参照しています)。

Execution Steps				
A	Execute function /system/compute			
	Parameters: Expression: list-files('C:\FlowForceExamples\CopyFiles\source*.*')			
-	Assign this step's result to list as T0			
4	For each item in sequence list			

ヒント: [for-each]の[in シーケンス」ボックスに式をコピーし、この結果最初の実行ステップを削除することができます。

E	xecu	tion Step)S					
	4	Execute function /system/compute Parameters: Expression: list-files('C:\FlowForceExamples\CopyFiles\source*.*')						
	=	Assign this step's result to list as TO						
		For each item		in se	in sequence list			
		•						
		⊿ Execu	te funct	ion /system/file	syste	m/copy		
		Param	neters:	Source:		{item}		
				Target:		C:\FlowFo	orceExamples\CopyFiles\ta	arget
				Overwrite:		\checkmark		
				Abort on error:		+		
				Working directo	ory:	+		

上記のとおり、copy 関数が次の引数を使用して呼び出されます:

- ループ内のソースは現在のアイテム(ファイル)です。ソースボックス内で {item} を入力、おけ、 Set to ト ボタンをクトックして「アイテム」を選択します。

• 「上書き」オプションが有効化されていると、すなわち、ソースディレクトリ内で同じ名前を持つファイルは既存の場合、上書きされて います。上書きて回避するために、
・
ボタンタクトックしてくたさい。

便宜上、このサンプル内では他の2つの引数は設定されていません。詳細に関しては、/system/filesystem/copy 関数を参照して くたさい。

作成されビジョブよ 必要とされるすべての処理ステップを含んで、ますが、トリガーはおご存在しません。ジョブを再帰的な時間の間隔でトリ ガーするコよ、タイマートリガーを使用することができます。<u>タイマートリガー</u>を参照してください。おコよ、変更のためこノースディレクトリをモニター し、ファイルシステムトリガーの方法を用いてジョブをトリガーすることもできます。ファイルシステムトリガーを参照してください。最後に、Webサ ービスの呼び出しとして、ジョブをオンデマドでトリガーすることができます。

このサンプルでは、ブラウザー内のURLをクトックして、ジョブをオンデマトドでトリガーします(実際には、ジョブがWebサービスとして呼び出 されます)。ジョブをWebサービスにするには「HTTPによりこのジョブを使用できるようにする」チェックボックスを選択し、Webサービ スの名前を入力します。

サービス

🗹 このジョブを HTTP を介して URL 上で利用可能にする http://<*FlowForce server*>/service/ CopyFileSerive

最後に、作動するためにショブは資格情報を必要とします。このために、「「資格情報」セクションで表示されている FlowForce Server ユ ーザー名と ペワード ではなく オペレーティングシステムのユーザー名と ペワードを入力してくたさい。ませよ、<u>資格情報の定義</u>で説明されて いるとおり、スタイドアロンの資格情報が作成されている場合、「既存の資格情報を選択」オプションを使用して選択することができます。

之	資格情報			
	資格情報によりジョブを実行:	◎ 既存の資格情報を選択:		
		◉ ローカルの 資格情報を定義: ユーザー名:	root	
		バスワード:	••••	

ジョブを実行する方法

ジョブをテストするコよ以下の内のしつを行ってくたさい

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/CopyFilesService</u>をブラウザーのアドレスバーに入力します。このURL は FlowForce Server サービスがデフォルトのオストアドレスとポート名でリンスンする場合のみ機能します。構成ページ内で他のオストとポート設定を定義した場合、アドレスを必要に応じて変更してくたさい。
- FlowForce Server の任意のホスト名フィールをセトアップページから設定するとHTTP を介してこのジョブを使用可

能にする チェックボックスの横の と ボタンをクリックして web サービスの呼び出しを直接ジョフ構成ページから実行することができます。それ以外の場合、このパタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コログオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウトを使用して W eb サービスにアクセスします。HTTP 認証を無効化するコは、W eb サービスを公 開し、Service - Use パーミッションをユーザー Anonymous に与えます。パーミッションのしくみを参照してくたさい。

実行に成功すると、ジョブはすべてのファイルをノースからターゲトディレクトリニピーします。それ以外の場合、「サービスの実行に失敗しました」エラーがブラウザーに表示されます。エラーか表示されると、ジョブのログをチェックしてください。詳細に関しては、ジョブログの雑認を参照してください。エラーの原因には、正確ではない資格情報、ファイルシステム上の正確ではない、いージンタンなどか含まれます。例えば、下記のとおり「上書き」チェックボックスが選択されていない場合、および、ターゲル・ディレクトリが既に同じ名前を持つファイルを含む場合、ジョブは失敗します。

Date	Message
2020-09-17 12:26:53	Starting instance 8.
2020-09-17 12:26:53	Starting job execution: job /public/Examples/CopyFiles in queue /public/Examples/CopyFiles
2020-09-17 12:26:53	Running instance 8 locally.
2020-09-17 12:26:53	Job /public/Examples/CopyFiles
2020-09-17 12:26:53	System function /system/compute
2020-09-17 12:26:53	For each item in <i>list</i>
2020-09-17 12:26:53	Iteration 1
2020-09-17 12:26:53	System function /system/filesystem/copy
2020-09-17 12:26:53	Executing FlowForce.copy with parameters: Source: "C:\FlowForceExamples\CopyFiles\source\invoices.txt",
2020-09-17 12:26:53	Step FlowForce.copy failed: Failed copying the file: The file exists.
2020-09-17 12:26:53	Job execution failed: job /public/Examples/CopyFiles in queue /public/Examples/CopyFiles

14.4 MapForce マッピングからジョブを作成する

この例では MapForce マピングから FlowForce Server ジョブを作成する方法が説明されています。最初に MapForce から FlowForce Server でもマピングファイルをデプロイします。マピングが FlowForce Server にデプロイすると、サーバージョブをこれから 作成します。ジョブは毎日特定の時間に実行されるように構成されています。

必要条件

- 必要とれるライセンス MapForce Enterprise おはProfessional edition、MapForce Server おはMapForce Server Advanced Edition、FlowForce Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ホルワークアドレスとポート</u> でルスンされていること。
- エンテナーの1つでパーミッションを有する FlowForce Server ユーザーアカウトを有していること(デフォルトでは、この例で使用されている/public コンテナーイコは認証されたユーザーがアクセスすることができます)。

使用されるデモファイル

このサンプルで使用されているマンピングファイルはCompletePO.mfdと呼ばれ、MapForceがインストールされているマシン上で以下の、なで見つけることができます: < Documents > \Altova \MapForce2021 \MapForceExamples。 MapForce を最初に起動する場合 [MapForceExamples] ディレクトリが作成されることに注意してくたさい(起動前には作成されません)。



CompletePO.mfd

上ので説明されているでもマンビングは3つのXML ファイルを入力として取り、出力として単一のXML ファイルを作成します。このサンプルで は入力 XML ファイルは自動的に FlowForce Server にデプロイされたパッケージに含まれます。<u>FlowForce Server にマンビングをデ</u> プロイするで説明される通り、他のマンピングはデプロイ前に追加の準備ステップを必要とする場合があります。

ジョブの作成

マピングのデプロイはMapForce がマピングに使用sあれるリソースを単一のパケージに整理し、HTTP(または構成されている場合 HTTPS)を介してFlowForce Server に送信することを意味します。

FlowForce Server にマッピングをデプロイする

- 1. MapForce 内で Complete PO.mfd ファイルを開きます。
- 2. お行っていない場合、マピングの変換言語をビルトインに設定します。
- 3. 「ファイル」メニューから「FlowForce Server にデプロイする」をクルクします。
- 4. サーバーとポート テキストボックス内にWeb 管理インターフェイスのサーバー名とポートを入力します(例えば FlowForce Web Server サービスがデフォルトのポートで同じマシントでのノスティングの場合では127.0.0.1 と8082 など)。異なるアドレスとポートが構成されている場合これらの値を変更してください。 <u>ホーワーケ設定の定義</u>を参照してください。
- 5. User とPassword テキストボックスにFlowForce Server ユーザー名と マワードを入力します。
- 6. ディレクトリをログインドロップダウンリストからを選択、おけよ < Default> オプションをそのままします。

デルクトリサービス統合が有効化されていると、ドメインユーザー名とパワードを入力し、ドメイン名をログインドロップダウンリストから選択します。詳細に関しては、ディレクトリサービス設定の変更を参照してくたさい。

𝔅 Deploy Mapping ×				
Enter the host name mapping.	e and port of a FlowForce Administration Inte	rface to de	ploy the current	
Server:	localhost ~	P <u>o</u> rt:	8082	
<u>U</u> ser:	root		Use SSL	
Password:	••••			

7. サンプル間の一貫性を保つためマンピングは/public/Examples コンテナーにデプロイされます。Browse をクトックしてコンテナ ーの 文を /public/Examples に変更します。前のサンプルに従っている場合 /public/Examples コンテナーは既存である へきです。それ以外の場合、下のダイアログボックス内のコンテナーの作成 をクトックして作成します。

😢 Choose Deployment Name	×
Server containers: Existing map	pings:
□··· (a) RaptorXML □··· (b) RaptorXMLXBRL □··· (b) public □··· (b) Examples •··· (b) system	
	~
Create Container Delete Container	D <u>e</u> lete
Name: CompletePO mapping	
	OK Cancel

8. [Web ブラウザーを開き新規ジョブを作成する] チェックボックスを選択します。

×
to deploy the current
yrt: 8082
Use SSL
\sim
Browse
Cancel
rt: 8082 Use SSL

9. [OK] をクリックしてマピングをデプロイレます。

デプロイが完了すると、FlowForce Server 管理インターフェイスはWeb ブラウザーを開き、部分的に記入されたショブのページが表示されます。マノピング関数自体は以前に指定済みのエレテナー・マノニ保存されます。これでデプロイの過程は完了しました。

ジョブの作成方法

上記の通りFlowForce Server にマンピングファイルをデプロイレた後、ブラウザーは部分的に入力されたショブページを表示します。入力 済みのプラメーターと共に最初の実行ステップが自動的に作成されます。

以	下にジョ	ョブを作り	成 /publi	C
ジョブ1 ジョブ0	名: Comple D説明:	etePO.job		
ジョブ ・ 実行	ブ入力パラ ステップ	メーター		
4	関数の実行	/public/CompleteP	O.mapping	
	バラメーター:	Customers:	(入力) 🔮 👍	
		Articles:	(入力) 🗐 🔒	
		ShortPO:	(入力) 🖪 😱	
		CompletePO:	(出力) 🝕 😱	
		Working-directo	ry: 🔶	
=	ステップの 結界	見を割り当て 名前	L	以下としてCompletePO

ジョブを関数のページ(/public/Examples/CompletePO.mapping)を開いてジョブを作成し、ジョブの作成をケルクします。

ジョブを構成する方法:

- 1. デフォルトのジョブ名を[CompletePO.job]か詳細な名前に変更します。例えば[GeneratePurchaseOrder]など。これ は任意のステップですが、名前が同じコンテナー内の他のジョブにより使用されている場合必要なステップです。
- 2. デフォルトで作成された最初の実行ステップを以下のように入力します:

実行関数	このフィールドは以前にデプロイされたマンピング関数を指しています。そのまましてくたさい。
<i>\`</i> ¬ \ ~	Customers、Articles, および ShortPO フィールドはショブン シケージ済みの対応する XML フィールドを含んています。

	CompletePO フィールドは出力ファイルの やを指定します。 デフォルト でよ CompletePO.xml です。 以下で説明されてい る通り やしば作業ディレクトリニズリして 相対的です。
	このサンプルでは、全ての入力と出力オプションをそのままご保持することができます。入力 と出力インスタンスの変更に関する詳細はジョブとして魔ピングを実行し変換するを参照 してくたさい。
	作業ディレクトリボック内に作業ディレクトリの、文を入力します。この例は、ジョブの作業ディレクトリとして C:\FlowForceExamples\Mapping を使用しています。
	作業デルトリとは、ジョブが入力ファイルを展開、おけは、出力ファイルを保存する場所 が必要な場合の、実行ステップにより必要とされる、ウメーターです。FlowForce Server は、おこ、作業デルノトリをステップの実行中に発生する相対・ジを解決するた めこ、使用します。作業デルノトリを与えるように問われた場合、FlowForce Server が実行されているオペレーティングシステム上の有効ないジを与えてくたさい。ス テップを作成中に、作業ディレノトリが与えられない場合、FlowForce Server は一 時的なディレノトリを使用します。
ステップの結果を割り当て	このフィールドはマッピングの結果に名前を与えます。このサンプルでは、このフィールドを空 白のままえします。

- トリガーから「新規タイマー」をクリックします。
 実行の横から、タイマーを毎日、1日毎に実行するように設定します。開始の横から、ジョブが開始される日時を選択します。例:

Tr	Triggers				
	Run	daily		 ✓ every 	1 day(s)
	Repeat	+			
	Start:	₫ 2020-09-17	- 0	12:55:00 💼	
	Expires:	+			
	Time zone:	Europe/Vienna		•	
	enabled	I			

5. 「資格情報」から、既存の資格情報の記録を選択、おけ、ローカルの資格情報を指定してくたさい。詳細に関しては資格情 報を参照してくたさい。

省	 昏格情報		
	資格情報によりジョブを実行:	◎ 既存の資格情報を選択:	
		◉ ローカルの 資格情報を定義: ユーザー名:	root
		パスワード:	••••

6. [保存]をクルクします。

ジョブを実行する方法

トリガー内で指定された日付で、FlowForce Server マピングショブを実行します。ショブの実行に成功すると結果として生成されたファイル(CompletePO.xml)はC:\FlowForceExamples\Mapping ディレクトリ内で使用できるようてなります。ジョブの実行が成功したかを確認するココミショブログを参照します。

14.5 ジョブを他のジョブのステップとして使用する

この例では、他のショブのステップとして定義されたショブを使用する方法が説明されています。このサンプルは、前に作成されたショブを必要とするため、サンプルをこのサンプルを完了する前に、MapForceマンビングからジョブを作成するを完了することが奨励されます。

MapForce マンピングからジョブを作成するサンプルから思い出せるかもしれませんが、GeneratePurchaseOrderは実行される都度、XML ファイルを一時フォルダーに生成します。このサンプルは以下を行う方法について説明しています。

- 1. マピングには生成されたファイルをピーするショブを作成します。このショブを CopyOutput と呼びます。
- 2. GeneratePurchaseOrder を変更し CopyOutput ジョブを追加実行ステップとして追加します。

必要条件

- 必要とれるライセンス MapForce Enterprise おはProfessional edition、MapForce Server おはMapForce Server Advanced Edition、およびFlowForce Server
- FlowForce Web Server とFlowForce Server サービスか構成された<u>やホワークアドレスとポート</u> でルスンされていること。
- <u>コンテナー</u>の一つへのペーミッションを持つFlowForce Server ユーザーアカウト (デフォルトでは、/public コンテナーはすべての認証済みのユーザーによりアクセスすることが可能です)。
- このサンプルで作成されたマングジョブはファイルを一つのディレクトリから他のディレクトリにコピーします。FlowForce Server が 作動するオペレーティングシステムでは、両方のディレクトリカ存在している必要があり、これらのディレクトリ内にファイルを作成する 権利を有していることを確認してくたさい、このサンプルはC:\FlowForceExamples\MappingとC: \FlowForceExamples\Archiveディレクトリを使用しています。
- MapForce マンプカらジョブを作成する内で説明されているステップを完了してください。

ジョブの作成方法

- 1. 「構成」をクリックし /public/Examples コンテナーゴ移動します。前のサンプルを試している場合 public/Examples コンテナーゴの作成 コマイを使用して作成してください。
- 2. 「作成」をクリックし、「ジョブの作成」を選択します。
- 3. ジョブの名前を入力します(このサンプルでは、CopyOutput、)。

Create job in /public/Examples	
Job name:	CopyOutput
Job description:	Copies the output of a mapping to another directory

4. 実行ステップの下で、次の設定を持つ最初の実行ステップを追加します:

Execution Steps									
	Execute function /system/filesystem/copy								
	Parameters:	Source:	CompletePO.xml						
	Target:		C:\FlowForceExamples\Archive\PurchaseOrder.xml						
		Overwrite:							
		Abort on error:	+						
		Working directory:	C:\FlowForceExamples\Mapping						

実行関数	/system/filesystem/copy 関数を参照します。				
<i>y</i> -ス	CompletePO.xml				
	作業ディレクトリノデメーターが設定されていないため、相対的ないでを使用します。				
<u> </u>	これはFlowForce Server が作動するオペレーティングシステムの既存のファイルませまディレクトリの、文である必要があります。このサンプルでは、コピーされるとファイルの名前を変更し、ファイル名をパ スコ以下のようご追加します:				
	C:\FlowForceExamples\Archive\PurchaseOrder.xml				
上書き	このチェックボックスを選択します。これは、FlowForce Server に保存先パマで検索された同じ名前を持つファイルを上書きするようこ命令します。				
エラー時には中断する	ノウメーターをそのままします。				
	このブール・ウメーターはショブが失敗した場合の戻り値を決定します。「エラー時には中断」が FALSE の場合、関数は、ブール値のFALSE を返します。「エラー時には中断」がTRUE の 場合、ジョブの実行は中断されます。デフォルトの値は、TRUE です。				
作業デルクトリ	トリ FlowForce はすべての相対的なファイル 文をこのディレクトリで探します。以下に設定します:				
	C:\FlowForceExamples\Mapping				

- 5. 「資格情報」から、既存の資格情報の記録を選択、おさ、ローカルの資格情報を指定してくたさい。詳細に関しては<u>資格情報</u>を参照してくたさい。
- 6. [保存]をクルクします。

既知かもしれませんが、作成されたショブコオーガーが存在しません。他のショブからこのショブを呼び出すため、トリガーは定義されていません。

CopyOutput ジョブを他のジョブのステップとして追加する方法

- 1. /public/Examples コンテナーをGeneratePurchaseOrder から開きます。
- 2. 実行ステップで、「新規実行ステップ」をクリックし、既存のステップの後に新規ステップを追加します。
- 3. 実行関数の横でCopyOutput ジョブの場所を参照してくたさい。実行ステップは以下のようにないます:

Execution Steps										
	4	Execute function /public/Examples/CompletePO.mapping								
		Parameters:	Customers: Articles: ShortPO: CompletePO:	(input) (input) (input) (output)		+ + +				
			Working-directory:			C:\FlowForceExamples\Mapping				
(=	Assign this step's result to name				as CompletePO				
	Execute function /public/Examples/CopyOutput									

- 4. タイムトリガーから「保存」をクリックします。
- 5. トリガー内に入力された日時から、FlowForce Server は、ジョブを実行し、CompletePO.xml ファイルを指定されたデル ケリニピーし、名前を PurchaseOrder.xml に変更します。ジョブの実行が成功したかを確認するコはジョブログを参照します。

14.6 ディレクトリ ポーリング ジョブを作成する

このサンプルはFlowForce Server 内で作成されたファイルシステムトリガーを使用してディレクトリでの変更を監視する方法について説明 しています。(ファイルシステムトリガーを参照してくたさい)。新規 XML ファイルめディレクトリに追加されるとFlowForce Server は XML ファイルを入力・デメーターとして取るマンピングジョブを実行します。マンピングジョブの出力は、アーカイブディレクトリに移動されます。

必要条件

- 必要とれるライセンス MapForce Enterprise おはProfessional edition、MapForce Server おはMapForce Server おはびFlowForce Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ホルワークアドレスとポート</u> でルスンされていること。
- <u>コンテナー</u>の一つへのペーミッションを持つFlowForce Server ユーザーアカウト (デフォルトでは、/public コンテナーはすべての認証済みのユーザーによりアクセスすることが可能です)。
- このサンプルで作成されたマンピングショブはファイルを一つのディレクトリから他のディレクトリニンピーします。FlowForce Server が 作動するオペレーティングシステムでは、両方のディレクトリカ存在している必要があり、これらのディレクトリ内にファイルを作成する 権利を有していることを確認してくたさい、このサンプルはC:\FlowForceExamples\DirPolling とC: \FlowForceExamples\Archive ディレクトリを使用しています。

使用されるデモファイル

- ShortApplicationInfo.mfd FlowForce Server ジョブを作成するMapForce マピング
- ApplicationsPage.xml マピングに入力とて与えられるXML インスタンスファイル。

MapForce がインストールされているマシン上の次のいてで両方のファイルを見つけることができます: **<Documents>、Altova、MapForce2021、MapForceExamples、**。

マピングのしくみ

このサンプルShortApplicationInfo.mfd)で使用されているMapForceマナングは以下のとおりです。FlowForce Server からの観点では、XML ファイルと入力とてマナングが取りXML ファイルを出力として生成することが重要です。

マピングはXML ファイル Applications Page.xml)を他のスキーマに変換し ShortInfo.xml とて保存します。一番上の接続を 確認するとマピングは簡単に理解することができます:ソース内の各 アイテム のために、ターゲト 内で情報 アイテムか作成されます。対応 する子アイテムから値をコピーするために他の接続は使用されます。特定の関係がある項目は点線で接続されています。 MapForce 内で は、この接続は、ソース優先」と呼ばれ、サブセクションが複合型コンテングを含むために使用されます。



ShortApplicationInfo.mfd

ソースとターゲット XML スキーマの名前はそれそれ SectionedPage と ShortInfo であることに注意してください。下に示されるように、 FlowForce ジョブは FlowForce Server にデプロイ後も同じ名前の入力と出力パラメーターを持ちます。

Flow Force Server にマンピングをデプロイする

FlowForce Server にデプロイする前にマピング ShortApplicationInfo.mfd は特別なな準備を必要とません。ソースとターゲオコンポーネントがXML ファイルのため、FlowForce Server にデプロイされる シケージに自動的に含まれます。

FlowForce にマピングをデプロイするコよ MapForce 内で開きメニューコマボ「ファイル Deploy to FlowForce Server にデ プロイする」を実行します。
🙁 Deploy Mappi	ing	×		
Enter the host nam mapping.	e and port of a FlowForce Administration Interface to deploy the c	urrent		
Server:	localhost v P <u>o</u> rt: 8082			
<u>U</u> ser:	root Use S	SSL		
Pass <u>w</u> ord:	••••			
Login:	<default> ~</default>			
Global Resources	on the Server			
Use <u>R</u> esource	es			
Resource Path:				
	The path must start with a slash character.			
Deploy As				
Path:	/public/Examples/ShortApplicationInfo.mapping	2		
The path must start with a slash character.				
S <u>a</u> ve mapping	g before deploying			
Open web br	owser to create new job			
	OK Can	cel		

FlowForce Server が異なるホストとポートで作動する場合、必要に応じて接続の詳細を変更してくたさい。<u>やトワーク設定の定義</u>を参照してくたさい。マピングの「マは、/public/Examples/ShortApplicationInfo.mapping であることに注意してくたさい。これは前のサンプルと一貫しています。

ディレクトリポーリングショブを作成する

FlowForce Server にマンングをデプロイ後、ブラウザーが開かれジョブ作成ページがロードされます。下に説明されているように最初の実行ステップは自動的に作成され、前にデプロイされたマンピング機能が呼び出されます。ターゲットコンポーネント(ShortInfo)と同じ名前が出力パラメーターに与えられていますが、ソース MapForce コンポーネント(SectionedPage)と同じ名前か数すえられていることに注意してください。

Ex	ecu +	ition Step	OS
	4	Execute func	tion /public/Examples/ShortApplicationInfo.mapping
		Parameters:	SectionedPage: (input) 😫 🕂
			ShortInfo: (output) 🗐 🕂
			Working-directory:
			·········

ジョブを次のように構成します

1. 作業ディレクトリボック内に作業ディレクトリの Sを入力します。この例は作業ディレクトリとしてC: 、FlowForceExamples\DirPollingを使用しています。

作業ディレオーリとは、ジョブが、スカファイルを展開、おけよ、出カファイルを保存する場所が必要な場合の、<u>実行ステップ</u>によい必要とされるパ ラメーターです。FlowForce Server は、おこ、作業ディレオーをステップの実行中に発生する相対パンを解決するために、使用します。 作業ディレオーを与えるように問われた場合、FlowForce Server が実行されているオペレーティングシステム上の有効なパンを与えてくだ さ、、ステップを作成中に、作業ディレオール、与えられない場合、FlowForce Server は一時的なディレオーを使用します。

2. トリガーから「新規 ファイルシステムトリガー」をクリックします。FlowForce Server は、入力、ウメーターご新規 triggerfile、ウメーターを自動的に追加することに注意してください。このエンテナーを次のステップで参照する必要があります。

ジョブ入力パラメーター				
名前: triggerfile 型: 文字列	このバラメーターは、ファイルシステム、ま ・ デフォルト: ・ 説明: たは、HTTPトリガーがトリガーを実行する ファイル名または、URIを受け取ります。:	1	•	٦
•				

- 3. 次のトリガーの値を設定します:
 - チェック 変更された日付
 - ファイルませまディレクトリ C:\FlowForceExamples\DirPolling*.xml
 - ポーレク間隔 60 秒

Triggers						
Check	Modified Date	of file or directory:	C:\FlowForceExamples\DirPolling*.xml	polling interval:	60	seconds.
Start:	+					
Expires:	+					
Time zone	Europe/Vienna	-				
🗹 enable	d					

4. 実行ステップでtriggerfile、デメーターをSectionedPage、デメーターイン対する入力値とていてします。これを行うコよ、

SectionedPage パラメーターの横の SectionedPage パラメーターの横の SectionedPage パラメーターの値は、{triggerfile} に変更されます。中かっこはFlowForce 式を表示し、削除される必要があります。

E	Execution Steps						
	+						
	Execute function /public/Examples/ShortApplicationInfo.mapping					icationInfo.mapping	
		Parameters:	SectionedPage:	(input)	욚	{triggerfile}	
			ShortInfo:	(output)	욚	+	
			Working-directory:			C:\FlowForceExamples\DirPolling	

これまでの構成により、ApplicationsPage.xml が作業ディノケリニピーされると、トリガーが実行されます。しかしなから、トリガーはフ イルドカード (*.xml) を使用するため、(ShortInfo.xml) を含む他のXML ファイルがディレケリ内で変更される場合でも実行されま す。これば期待される振る舞いではなくエラーを引き起こします。ですから、新規ディレケリから生成された出カファイルを削除する2番目のス テップを追加する必要があります。ますは、トリガーの名前を C:\FlowForceExamples\DirPolling\ApplicationsPage.xml に 名前を変更することができます(この場合、2番目のステップは必要ありません)。

新規ディレクトリに出力を移動するステップを追加するココントを行います

- 1. 前のステップのすく後に、新規実行ステップを追加します。
- 2. 次のようにステップを構成します(ソースと目的のフィールドは大文字と小文字を区別することに注意してくたさい)

実行関数	/system/filesystem/move 関数を参照します。
ソ -ス	ShortInfo.xml
	作業ディレクトリッテメーターが設定されていないため、相対的ないやを使用します。
保存先	これはFlowForce Server が作動するオペレーティングシステムの既存のファイルおさまディレクトリの、父である必要が決ます。以下に設定します:
	C:\FlowForceExamples\Archive
ターゲトの上書き	このチェックボックスを選択します。これは、FlowForce Server に保存先パスで検索された同じ名前を持つファイルを上書きするようこ命令します。
エ ラ 時 はおか断する	ノウメーターをそのままします。
	このブール・ラメーターはジョブが失敗した場合の戻り値を決定します。「エラー時には中断」が FALSE の場合、関数は、ブール値のFALSE を返します。「エラー時には中断」がTRUE の 場合、ジョブの実行は中断されます。デフォルトの値は、TRUE です。
作業ディレクトリ	FlowForce はすべての相対的なファイル マをこのディレナトリで探します。以下に設定します:
	C:\FlowForceExamples\DirPolling

Execute function /system/filesystem/move			
Parameters:	Source:	ShortInfo.xml	
Dest	Destination:	C:\FlowForceExamples\Archive	
	Overwrite target:	\square	
	Abort on error:	+	
	Working directory:	C:\FlowForceExamples\DirPolling	

最後にショブの実行に使用される使用中のオペレーティングシステムの資格情報を追加します

- 1. 「資格情報」から、既存の資格情報の記録を選択、おさ、ローカルの資格情報を指定してくたさい。詳細に関しては資格情報を参照してくたさい。
- 2. [保存]をクルクします。

ジョブを実行する方法

ApplicationsPage.xmlを作業デルクトリニピーしでショブをテストすることができます。これを行う際には、FlowForce Server はマピングショブを実行し、結果出力ファイルをアーカイブデルクトリニピーします。

ジョブの実行が成功したかを確認するコはジョブログを参照します。

14.7 エラー処理をジョブに追加する

このサンプルはショブコエラー処理を追加する方法について説明しています。このサンプルで使用されナジョブはディレクトリのコンテンンをリスト し、ブラウザーからWebサービスとして呼び出されます。ジョブの結果を処理するためにFlowForce Server をどのようご構成するかについ て学びます:

- ジョブの実行に成功すると、ブラウザー内でショブの出力が表示されます。
- 何らかの理由でショブが実行に失敗した場合、名前の挙げられた宛先に通知メールが送信されます。
- ジョブの実行が完了すると、実行の状態に関わらず、ローカルシステム上のファイルのジョブの内部 ID にエグされます。

FlowForce Server ではこの例では次の2つのエラー処理条件のために<u>保護されたブロック</u>を作成します:「エラー時」と「常に」(それ それ上記のシナリオの内のこを処理します)。

必要条件

- 必要とされるライセンス FlowForce Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ネルワークアドレスとポート</u> でルスンされていること。
- FlowForce Server メールの設定か構成済みであること。電子メールウメーターの設定を参照してくたさい。
- ・ <u>コンテナー</u>の一つへの、・ミッションを持つFlowForce Server ユーザーアカウトが必要です、デフォルトでは、/public コンテ ナーはすべての認証済みのユーザーによりアクセスすることが可能です)。

ビナ

• この例は、Windows パンとコマイを使用していますが、パンとコマイを必要に応じて変更すると、他のオペレーティングシステム 上でテストすることができます。

ジョブの作成方法

- 1. FlowForce Server が作動するマシン上で、ジョブ出力が保存されるディレクトリを作成します。このサンプルはC: **FlowForceExamples** ErrorHandling を使用しています。
- 2. FlowForce Server Web 管理インターフェイスコンゲイン /public/Examples コンテナーを開き、「ジョブの作成」をクトックします。前のサンプルゴ従っている場合 ublic/Examples コンテナーは既存であるへきです。それ以外の場合作成 | コンテナーの作成 コマンドを使用して作成します。
- 3. ジョブの作成をクリックして作成中のジョブの名前を入力します。例えば [ListDirectory] など。ジョブの説明は任意です。
- 4. ジョブ入カパラメーターで *** ボタンを押して型[文字列]の デメーターを追加します。ジョブランタイムでは、パラメーターはルストへの ディレクトリの やを提供します。このサンプルでは、パラメーターの名前は[nputDir] で後のステップで使用されます。
- 5. 「実行ステップ」で、「新規のエラー/成功処理ステップ」をクリックします。
- 6. 「エラー/成功処理を使用して実行」から、 📩 ボタンをクリックして、次の設定を持つ新規の実行ステップの追加を選択します:

実行関数	/system/shell/commandline 関数を参照します。
ᠴᢦᢞ	以下のシェルコマンドを入力します
	dir {inputDir}

	inputDir が以前に作成されたッテメーターの名前である個所名前は中かって、囲まれています。これはジョブのテンタイムで、コンテンンが動的にンテメーターの値を置き換えられるからです。詳細に関しては、文字列フィールド内に式を埋め込むを参照してくたさい。
エラー時には中断する	オプションをそのままします。詳細・理しては、 <u>/system/shell/commandline</u> 関数を参照してく たさい。
作業ディレクトリ	以前に作成された作業ディレケーリーのの ひを入力します。例えば、C: \FlowForceExamples\ErrorHandling

7. 「エラー時」条件から 💽 ポタンをクリックして、次の設定を持つ新規の実行ステップの追加を選択します:

実行関数	<u>/system/mail/send</u> 関数を参照します。
差出人	差出人の電子メールアドレスを入力します。例:flowforce@localhost。管理ページから電子メールの設定を構成した場合、このフィールドを空のままえします。
宛先	電子メールアドレスを入力します。
件名	通知電子メールの件名を次のようこ入力します: Job {instance-id()} has failed 現在の(失敗した) ジョブインスタンスの一意のID を取得する中かっこの間の部分はinstance-id 関数を呼び出す FlowForce 式です。
本文	以下を入力します: Exit Code: {string(exitcode(failed-step()))} Standard Error: {content(stderr(failed-step()))} Error message: {error-message(failed-step())} 中かつの間の部分はつのFlowForce 式です。これらの式はエラーのある出力を取得し、電子メールの本文となる文字列I-変換します: • failed-step 関数は、失敗したステップのresult を返します。exitcode、 stderr、おけまerror-message 関数への別数とて提供される更に使用しやすいようなるこれは抽象的な FlowForce 型です。 • exitcode 関数は終了コードが存在することを前提に数値と出て result から実際 のエラーコードを取得します。 • stderr 関数はストリームとて result からエラーの標準エラー出力を取得します。 • error-message 関数はFlowForce エラージッセージのテキストをログに表示されると おり「取得します。」 • storing 関数は返日の終了コードを文字列に変換します(これは電子メールの本文が string 関数はなトリームからのエラー出力を文字列に変換します。 • content 関数はストリームからのエラー出力を文字列に変換します。



- 8. 「新規のエラー/成功処理」をクトックして、「常に」を選択します。
- 9. 「常に」条件で 한 ボタをクトクレ、次の設定を持つ新規の実行ステップを追加します。

実行関数	① <u>/system/shell/commandline</u> 関数を参照します	
ᠴᢦᢣ	以下のシェルコマンドを入力します	
	<pre>echo {instance-id()} >> JobLog.txt</pre>	
	Windows 上では、このコマンドはジョブID を JobLog.txt と呼ばれるファイルに書き込みます。 ファイルニデータが含まれる場合、新規のテキストは既存のデータに追加されます。	
作業デルケリ	以前に作成された作業ディレケトリーの ひを入力します(C: \FlowForceExamples\ErrorHandling)。	
	このディレクトリは JobLog.txt ファイルへの ひを解決するために使用されます。	

この段階では、ジョブおり下のよう」ないます(異なる、スまけは、シェルロマイが使用されていないてとか仮定されます)。

エラー/成功処理にて実行				
+				
4	関数の実行 /system/shell/commandline			
	パラメーター:	コマンド:	dir/s	
		エラー時には中断:	(+)	
		作業用ディレクトリ:	C:\FlowForce	
=	ステップの結果	を割り当て名前	以下として結果	
•				
エラー	-時	✓ do		
+				
4	関数の実行 /sy	stem/mail/send		
	パラメーター:	差出人:	flowforce@localhost	
		宛先:	to@example.org	
		件名:	Job {instance-id() }has failed	
		本文:	Exit code:{string(exitcode(failed-step()))} Standard Error:{content(stderr(failed-step()))}	
		添付:	+	
		エラー時には中断:	+	
=	ステップの結果	を割り当て名前	以下としてブール値	
•				
常に		∼ do		
+				
4	関数の実行 /sy	stem/shell/commandli	ne	
	パラメーター:	コマンド:	echo{ instance-id ()}>>JobLog.txt	
		エラー時には中断:	+	
		作業用ディレクトリ:	C:\FlowForce	

10. ジョブをWebサービスにするには「HTTPによりこのジョブを使用できるようにする」チェックボックスを選択し、Webサービスの名前を入力します。例

ŀ	トリガー	
	実行	毎日 >> 毎 1 日
	繰り返し	毎 60 分 ⑧全日、または ○開始時間 ❹ 08:00:00 終了時間 ❹ 20:00:00 💼
	開始:	[™] 2019-06-02 ▼ ● 13:15:00 [™] [™] [™]
	有効期限:	+
	タイムゾーン	Europe/Berlin
	☑ が有効化る	されました

サービス名に注意してくたさい。Web サービスを呼び出す必要があります。

- 11. 「資格情報」から、既存の資格情報の記録を選択、ませよ、ローカルの資格情報を指定してくたさい。
- 12. [保存]をクルクします。

ジョブを実行する方法

この段階ではジョブ構成が完了しまた。Webサービスとしてこのジョブは公開されており以下の方法で実行することができます。

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/ListDirectoryService</u>をブラウザーのアドレスバーに入力します。このURL は FlowForce Server サービスがデフォルトのオストアドレスとポート名でリッスンする場合のみ機能します。 構成ページ内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してください。
- FlowForce Server の任意のホスト名 フィールドをセトアップページから設定するとHTTP を介してこのジョブを使用可能にする チェックボックスの横の ▶ ポタンをクリックして web サービスの呼び出しを直接ジョブ構成ページから実行することがで

龍にするナエックボックスの横の「ー」ボッシをクリックして web サービスの呼び出しを直接ション構成へーンから美行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server にログオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーに**Service - Use** パーミッション を与え、対応するユーザーアカウストを使用して W eb サービスにアクセスします。HTTP 認証を無効化するには、 W eb サービスを公 開し、**Service - Use** パーミッションをユーザーAnonymous に与えます。 パーミッションのしくみを参照してくたさい。

ジョブは デメーターを取り、ブラウザーから Webサービス こアクセスするよう ン デメーター値を提供するよう こプロンプト されます。

<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools	<u>H</u> elp	—		×
127	.0.0.1:4	1646/se	rvice/List[Directory: 🗙	+				
¢	$) \rightarrow$	G	0	127.0.0.1	:4646	/service/l	ListDirec	··· »	≡
	Paran nputI	neters Dir *:							
S	ubmit								

C:\のような有効なディレクトリを入力すると、例えば、ジョブは実行され結果がブラウザーに表示されます。

ジョブを実行する都度、ジョブインスタンスのID は前に構成された「常に〕条件に従いJobLog.txt ファイルのコンテンソニ追加されます。

[エラー時]条件で、[nputDir] パラメーターを意図的に例えば、存在しない かなど)他の無効な値に変更します。この場合ブラウザーはエ ラーを表示し FlowForce Server は[エラー時] パンドラーの受取人フィールド内で指定されているアドレスに電子メールを送信します。例 えば、電子メールは以下のようこなます:



以前に説明されている通りこのサンプルで使用されているエラー関数がすべてのジョブ構成のオンジェ値を返すことは保証されていません。 このオンジュ、電子メーリにより提供される詳細のレベルは上部構成と遭遇するエラーにより異なります。そして、終了コード、標準エラ ー、およびエラーメッセージ電子メーリフィーリドが常にテキストを含んでいるとは期待できません。エラーの原因として最も権限のある レファレンスはFlowForce Server log です。

14.8 Web サービスとしてジョブを公開する

この別では、Webサービスとしてのサンプルマメピングジョブを公表する方法が説明されています。MapForceを使用してにサンプルマメピン グはデザインされており、個人のレコードを補完するMicrosoftAccessデータベースからデータか読み取られます。マメピングはデータベース からい ラメータとして与えられる特定の文字で始まる姓を持つ個人のレコードのみが取得されます。(同じ、おけは、異なるマシン上で) MapForceからFlowForce Server へ既存のマメピングをデプロイレ、Webサービスへ変換する方法について学ぶことができます。この サンプルが終わると、Webサービスをブラウザーから呼び出すことができます。

必要条件

- 必要とれるライセンス MapForce Enterprise おはProfessional edition、MapForce Server おはMapForce Server Advanced Edition、FlowForce Server
- FlowForce Web Server とFlowForce Server サービスか構成された<u>や小ワークアドレスとポート</u> でルスンされていること。
- ・ <u>コンテナーの一つへの、トミッションを</u>持つ Flow Force Server ユーザーアカウト(デフォルトでは、/public コンテナーはすべての認証済みのユーザーによりアクセスすることが可能です)。

使用されるデモファイル

このサンプルは以下のファイルを使用しており、MapForceがインストールされているコンピューター上で次の、文を使用して見つけることができます: ..、Documents、Altova、MapForce2021、MapForceExamples.

- **DB_PhoneList.mfd** (MapForce マピングデザインファイル)
- altova.mdb (マピングがデータを読み取る Microsoft Access データベース)。

異なるマシンへマピングをデプロイする準備

このマピングはデータベースファイルからデータを読み取るため、このセクションで説明されているとおり追加の構成がマピングのデプロイの前に 行われる必要があります。MapForceとFlowForce Server か同じコンピューター上にインストールされている場合、下記の「マピングの デプロイ」のセクションをスキップすることができます。

メモ 「ソースマシン」とう用語は、MapForceがインストールされているコンピューターを指しており、「ターゲトイマシン」はFlowForce Server がインストールされているマシンを指しています。

ターゲナマシンコマンプをデプロイする前に、以下を行ってください

- 1. ターゲトマンとでは、「FlowForce Web Server」サービスをすべてのインターフェイス おけよ ローカルホスト 以外の特定の IP アドレスとでリッスンするように構成します。 <u>ネットワープ設定の定義</u>を参照してくたさい。このサービスが正確に構成されている か、ブラウザーから次のURL にアクセスして確認することができます: http://<FlowForce Web Server><port>。指定さ れているアドレスとポートが受信する接続がファイアーウォールによりブロックされていたれにを確認してくたさい。
- 2. 上記の通り、このサンプルで作成されるショブはWebサービスとて使用できる必要があります。FlowForce内では、Webサービスとして公開されて、るショブへのノケエストは「FlowForce Web Server」サービスではなく「FlowForce Server」サービスとして公開されて、るショブへのノケエストは「FlowForce Web Server」サービスではなく「FlowForce Server」サービスにおり処理されます(しくみも参照してくたさい)。このため、Webサービスがローカルホストの外部のHTTPケライアントにアクセス可能な場合、「FlowForce Server」サービスはすべてのインターフェイス上で、おさは、ローカルホスト以外の特定のアドレスによりリッスンできるようご構成されている必要があります。このサービスが正確に構成されているか、ブラウザーから次のURLにアクセスして確認することができます: http://、FlowForce Server>くport>/service/。Webサービスとして公開されているすべてのジョブは(存在する場合)ブラウザーウィンドウ内に直接リンクとして表示されます。

デプロイする前に、マピングは、絶対パマの代わりに相対パマを使用するように再度構成される必要があります。

- 1. DB_PhoneList.mfd を MapForce 内でマピングを開き、マピングエリアを右クトックし、エンテキストメニューから「マッピン グの設定」を選択します。
- 2. 「生成されたコードでは絶対パスを使用する」チェックボックスをクリアします。

マッピング設定	×
マッピング出力 アプリケーション名(A): Mapping	
Java設定 ベースパッケージ名(B): com.mapforce	
ファイル パス設定 □ 生成されたコード内では絶対パスを使用する(M) □ ローカルのファイル システム上のファイル パス出力に対する Windows のパス規則を確認(E)	
出力ファイル設定 行の末尾(L): ユーザー レポートによる特権 ~ (ビルトイン実行、C#、Java、C++コード生成でサポートされます)	
XML スキーマのバージョン	
ウェブ サービス オペレーション設定 WSDL 定義(W): サービス:	
エンドポイント: オペレーション:	
OK キャンセル	

3. マピングを保存します。

Microsoft Access 封はSQLite などのファイルベースのデータベースはマナングと共にターゲトマシスにデプロイされません。このかれ Access データベースは手動でノースマンカらターゲトマシスにピーされる必要があります。ソースマシン上のデルノトリ.. \Documents\Altova\MapForce2021\MapForceExamples からaltova.mdb データベースファイルをエピーし、ターゲトマ シン上の空のデルノトリご張り付けます。このサンプルでは、ターゲナディレオトリは「C:¥FlowForceWorkingDir」です。このディレオトリ ば後にFlowForce ジョブで参照されます。

マピングはFlowForce Server にデプロイされる準備ができています。データベース接続を含むマピングのデプロイに関する情報は、サーバー実行のオンパマパングを準備する。

マピングのデプロイ

- 1. **DB_PhoneList.mfd**を MapForce 内で開きます。
- 2. 「ファイル」メニューで、「Flow Force Server にデプロイする」をクリックします。このサンプルの目的のために、マッピングはデフォルトの & (/public コンテナー)) にデプロイされます。異なるマシン上のFlowForce Server にデプロイする場合、「localhost 8082」 からサーバーアドレスとポートをFlowForce Server から構成されたものに変更してくたさい、上記を参照してくたさい。

🙁 マッピングのデプロ	r	×				
現在のマッピングをデプロイするため FlowForceWeb 管理インターフェースのホスト名とポート番号 を入力してください。						
サーバー <mark>(</mark> S):	localhost	ポート (0): 8082				
ユーザー <mark>(</mark> U):	root	□ SSL を使用				
パスワード <mark>(w)</mark> :	••••					
ログイン (L) :	<デフォルト>	~				
デプロイ						
パス(P):	/public/CompletePO.mapping	参照(B)				
	パスはスラッシュから開始する必要があります	0				
「デプロイ前にマ	ッピングを保存(A)					
🗹 Web ブラウザー	を開き新規のジョブを作成する					
		OK キャンセル				

FlowForce ジョブの作成方法

FlowForce Server にマンングがデプロイされ、ブラウザー内にショブ構成ページが表示されています(上記のダイアログボックスで「新規の ジョブを作成するためにWeb ブラウザーを開く」チェックボックスが選択されていることが前提です)。それ以外の場合、FlowForce Server Web 管理インターフェイスコログイン、(/public コンテナーである)前にデプロイされたマッピング関数を開き、「ジョブの作成」を クトックします。

ジョブを構成する方法:

1. 入力パラメーターから、型文字列の新規入力パラメーターを作成し、NamePrefixと名前をつけます。

ジョブ入力パラ	ラメーター		
◆ 名前:	型: 文字列	▼ デフォルト: 🔸 説明:	

- 2. 実行ステップで、NamePrefixの横の「設定」をクリックして、NamePrefixを選択します。マメピング・ラメーターの植 NamePrefixを前のステップで作成されたNamePrefix入力・ラメーターの値に設定します。
- 3. Working-directoryの横に「C:¥lowForceWorkingDir」を入力しますこれはAccess ファイルが以前によーされて いるディレケーと同じてある必要です)。

実行	ステップ			
•	関数の実行 /p	ublic/DB_PhoneList.mapping	• @	
	パラメーター:	NamePrefix: 🕅	(NamePrefix)	xs:string (オプション) セットする値 🕨 😰 💼
		PersonList: (出力) 锢 Working-directory:	+ C\FlowForceWorkingDir	文字列 (オプション) セットする値 🕨 😰 💼
=	ステップの結果	を割り当て名前	以下としてPersonList	

4. サービスから「このジョブをHTTP を介して URL 上で利用可能にする」チェックボックスをクリックして選択し、 GetPhoneList をサービスの名前として入力します。

ť	トービス	
	🗷 このジョブを HTTP を介して URL 上で利用可能にする http://< <i>FlowForce server>/</i> service/	GetPhoneList

5. 資格情報から、既存の資格情報の記録、おさよ、ローカルの資格情報を指定します。資格情報も参照します。

資	译格情 報		
	資格情報によりジョブを実行:	◎ 既存の資格情報を選択:	
		◉ ローカルの資格情報を定義: ユーザー名:	root
		パスワード:	••••

- メモ この資格情報をFlowForce Server Web 管理インターフェイスへのアクセスロ使用される資格情報特別してくたさい。また、こ コニューザーが入力する内容が作業ディレクトリからaltova.mdb データベースファイルにアクセスできるようご注意してくたさい。そ れ以外の場合、ジョブは成功せずに失敗します。
 - 6. [保存]をクリックします。

Web サービスの呼び出し

作成されたWebサービスを以下のようコ呼び出すことができます:

1. ([FlowForceServer]と[Port]を管理ページ内で構成された設定と置き換えます):

http(s)://[FlowForceServer]:[Port]/service/GetPhoneList

FlowForce Server の任意のホスト名フィールドをセオアップページから設定するとHTTPを介してこのジョブを使用可能にする

チェックボックスの横の 🎦 ボタンをクトックして web サービスの呼び出しを直接ジョブ構成ページから実行することができます。それ以外の場合、このポタンは表示されません。

- 2. 資格情報を与えるように問われると、FlowForce Server Web 管理インターフェイスにアクセスするために使用される資格情報を入力します。
- 3. Web サービスの デメーターを入力するように問われると、(「F」で始まる姓を持つすべての個人を抽出することを前提として) F を入力します。

Parameters
NamePrefix *: F
Submit

4. 「送信」をクリックします。FlowForce Server はションを処理し、結果を返します。



ジョブの実行に成功すると、ジョブの出力は直接ブラウザーに表示されます(おこ、作業ディレケトリ内で出力は生成されます C: 、FlowForceWorkingDir)。それ以外の場合、実行エラーが発生すると、詳細のためにジョブのログを参照してくたさい(以下を参照 ジョブのログを確認する)。

14.9 JSON を FlowForce Web サービスへポストする方法

このサンプルでは、HTTP リクエストボディ内でJSON データを持つPOST リクエストを受け入れるFlowForce Web サービスを作成する 方法が説明されています。第21こ、MapForceなどのクライアントからWeb サービスを呼び出す方法について説明されています。

このサンプルではWebサービスはJSONデーを受け入れるように構成されます。XML おけは他のエンテンを以下で示される方法に類似した方法でFlowForce Server によれ作成されたサービスにないすることができます。Webサービスは、JSONデーをを受け入れ、 更なる処理無しにコーカルで保存できるように簡素化されています。RaptorXML Server を使用して、JSONデーを検証、おけよ処 理するようごうで拡張することもできます。

このサンプルは、パラメーターとしてではなく、HTTP リクエストのボディ内にデータがポストされる場合を特別に説明しています。Webサービスをいうメーターを使用して呼び出すサンプルは関しては、Webサービスを公開するを参照してください。

必要条件

• 必要とされるライセンス FlowForce Server、MapForce Enterprise Edition。

コメント

FlowForce Server はW eb サービスを作成する素早い方法を提供します。MapForce Enterprise Edition はFlowForce Server により作成されるW eb サービスを呼び出すクライアトとして振る舞います。異なるクライアトを使用しても、同じ結果を得ることができます。

- FlowForce Web Server とFlowForce Server サービスか構成された<u>ネットワークアドレスとポート</u> でルスンされていること。
- <u>コンテナー</u>の一つへのペーミッションを持つFlowForce Server ユーザーアカウト (デフォルトでは、/public コンテナーはすべての認証済みのユーザーによりアクセスすることか可能です)。
- このジョブは、Webサービスにお受信された入力データをローカルの作業ディレオリに保存します C: 、 トFlowForceExamples、PostJson。このディレオリ(おけま 類似したディレオリ)がFlowForce Server が作動する マシン上で既存のマシンに存在し、オペレーティングシステムユーザーアカウストがこのディレオ・リに書き込むことのできる権利を有する 必要があります。

Flow Force ジョブの作成方法

FlowForce Server W eb 管理インターフェイスコエヴインし、/public/Examples コンテナーを開き、「ジョブの作成」をクルクします。 す。次に、名前を入力し、オプションで、作成するW eb サービスのナダの詳細を入力します。

メモ 前のサンプルに従っている場合public/Examples コンテナーは既存であるべきです。それ以外の場合作成 コンテナーの作成 コマンドを使用して作成します。



ジョブがPOSTデーダを任意のコンテンソとて扱うようにするコよ型ストリームのいつからメーダをが必要しています。パラメーターを作成

するコよ「ノラメーターの追加」 한 をクリックレ、ノラメーター名を入力します(このサンプルの場合「データ」) ストリームをデータ型として 選択します。

ÿ∃] ∙	ブ入力パラメータ	_		
名前:	data	型: ストリーム	\sim	説明: HTTP リクエストのボディ
•				

次に、新規の実行ステップを追加し、以下のように構成します

実行ステップ						
+	関数の実行 /sy	rstem/filesystem/copy				
	パラメーター:	ソース: ターゲット: 上書き: エラー時には中断: 作業用ディレクトリ:	{as-file(data)} file{instance-id()}.json + + C:\POST			

上記の実行ステップはFlowForce ビルトイン copy 関数を呼び出します。「ソース」 テキストボックス内に表示される式は、 as-file 式関数を使用してファイルするすっかにWeb サービスには受信された入力を変更します(入力・ウメーターは前のステップ内で、名前の付けられ

たデータであることを思しせしてくたさい。この式を自動的に取得するコよ「ソース」テキストボックスの横の Set to と ポタンをクリックし、「データ」を選択します。

「ターゲト」テキストボックスコンショブが呼び出される都度一意のファイル名を生成する式が含まれています。一意のファイル名を取得するこ は、FlowForce instance-id 式関数が呼び出されます。このため、JSON ファイル名は、「file35.json」のようてより、(FlowForce ジョブインスタンスのID に対応し)数値は各ジョブの呼び出しにより異なります。フルレスを入力することもできますが、このサンプルで示されてい るとおり、「作業ディレクトリ」・なか設定されていることが必須してよります。作業ディレクトリルなを設定すると、相対するファイル名が作業ディレク トリッマンマンして解消されます。

ディレクトリC:\FlowForceExamples\PostJSON(おけよ、アカ変更されている場合、上記に類似しているディレクトリ)が存在し、使用中のオペレーティングシステムユーザーアカウストか書き込む資格を有している必要かあります。

「サービス」から、「このジョブをHTTP から使用できるようにする」チェックボックスを選択し、「PostJsonService」を入力、おけよ、 Webサービスのために同様の名前を入力します。サービス名に注意してくたさい。Webサービスを呼び出す必要があります。

サードス 🗹 このジョブを HTTP を介して URL 上で利用可能にする http://*<FlowForce server>/s*ervice/ POST_JSON

「資格情報」から、既存の資格情報を選択、ませま ローカルの資格情報を指定します(<u>資格情報</u>も参照してくたさい。これらよ FlowForce Server が作動するオペレーティングシステム上のユーザーアカウトの資格情報である必要があります。

資	译格情報		
	資格情報によりジョブを実行:	◎ 既存の資格情報を選択:	
		◉ ローカルの 資格情報を定義: ユーザー名:	root
		パスワード:	••••

メモ この資格情報をFlowForce Server Web 管理インターフェイスへのアクセスコ使用される資格情報特別してくたさい。

[保存]をクリックします。新規のWebサービスをクライアントから呼び出す準備か整いました。

Web サービスをブラウザーから呼び出す方法

Webサービスを以下の方法でブラウザーから呼び出すことができます。

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクトックします。次に情報列内に表示されているショブの URLをクトックします。
- <u>http://127.0.0.1:4646/service/PostJsonService</u>をブラウザーのボレス・トースカします。このURLはFlowForce Server サービスカデフォルトのオストボレスとポート名でリッスンする場合のみ機能します。構成ページ内で他のオストとポート設定を定義した場合、アドレスを必要に応じて変更してくたさい。
- FlowForce Server の
 注意の
 ホスト名
 フィールドを
 セトアップページ
 から
 設定する
 とHTTP を
 介してこの
 ジョブを
 使用
 可

能にする チェックボックスの横の と ボターをクトックして web サービスの呼び出しを直接ショフ構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コングオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーに**Service - Use** パーミッション を与え、対応するユーザーアカウストを使用して W eb サービスにアクセスします。HTTP 認証を無効化するこよ W eb サービスを公 開し、**Service - Use** パーミッションをユーザーAnonymous に与えます。<u>パーミッションのしくみ</u>を参照してください。

ジョブお ラメーター敏江ストリームを期待するように構成されているためブラウザー内にン ラメーター値を入力するようにプロレプトされます。「参照」をクリックして、 POST リクエスト内に提供される JSON ファイルを選択します。

Parameters	
data *:	参照
Submit	

[送信]をクリックすると、FlowForce Server はショブを処理し、ブラウザーイコンパンズを出力します。

ジョブの実行に成功すると、ブラウザーは「true」を表示し、JSON ファイルが作業ディレケトリC:\FlowForceExamples\PostJson に保存されます。それ以外の場合、実行エラーが発生すると、詳細のためコエウを参照してくたさい。ジョブログのビューを参照してくたさい。

MapForce からWeb サービスを呼び出す方法

Webサービスを、MapForce Enterprise Edition などのブラウザー以外のクライアントから呼び出すことができます。

- 1. 新規のマンピングを作成するために「ファイル」メニューから「新規作成」をクリックします。
- 2. 「出力」メニューでビルトイン実行エンジン」をクリックします。
- 3. 「挿入」メニューカらWeb サービス関数」をクリックします。Web サービス 呼び出し 設定ダイアログボックスか 開かれます。
- 4. 「手動」をクトックし、POST をリクエストメットドとして選択し、URL ボックス内でWeb サービスのURL を入力します。ブラウ ザーからWeb サービスをテストするナックに使用されたURL と同じです。

😢 Web サービス 呼び出	し設定	×
サービス定義 〇 WSDL(W) ④ 手動(M)	WADL ファイルからインポート(I) URL からインポート(U)	
メソッドのリクエスト(M): 接続設定 URL(U): タイムアウト(T):	POST	

5. [HTTP セキューティ設定]の横の編集 ボタンをクリックし 動的な認証 チェックボックスを選択します。これにより、マンングの実行 時資格情報をマンピングへの入力・ランターとして、インタラクティブは提供することができます。 資格情報の使用 オプション はす る詳細については、マンピング関数内の資格情報を参照してくたさい。このダイアログボックス内にユーザー名と マワードを直接入力 数にとは下位互換性のためこのみっサポートされていますが、 奨励はされていません。

😢 HTTP Security 🗄	安定
HTTPS (TLS)	
これらの設定は <mark>[h</mark> t	ttps:] URL のみに適用されます。現在の URL は HTTPS を使用しません。
🗹 サーバーの証明	月書を確認する(S)
□ホスト名と調	証明書およびリクエストの不一致を許可する(H)
□クライアントの証明	書(C): 選択(E)
HTTP 承認	
ユーザー名 (U):	someuser
パスワード(P):	•••••
□動的な認証(ユーザー名 & パスワードはマッピングにより提供されます)(Y)
□ プリエンプティブ	認証(R)
	OK キャンセル

6. 「OK」をクリックして、ダイアログボックスを閉じます。マッピングは以下のようこなります:



7. 「挿入 | 出力の挿入」メニューコマンドを選択し、マンピング 3個の入力、 デメーターを追加します。 最初の2つはそれぞれユーザ 一名と マワードを提供します。 3番目は JSON データを適用します。



8. 上の入力コンポーネトのそれぞれをダブルクトック、マッピングのプレビューのオッジコ使用されるデザイン時の実行値を入力します。 最初の2つの、デメーターのオッジコンサービスニアクセスするオッジコン要とされるユーザー名と、なワードを入力します。これらはマッピング を実行するオッジコン要でセキューティ上の理由からマッピングファイルコ保存することは奨励されません。JSON デーダを提供するパ ラメーターのオッジュ、デザイン時にこのマッピングを実行するオッジュ、以下に示される通り、使用するオッジニサンプルJSON デーダを 入力します。

🙁 入力の作	F成 ×
名前(N):	input
データ型 (D)	: string ~
6	☑入力は必須(Ⅰ)
デザイン時	実行
☑値を指	;定(S)
值(V): {	"title":"Product A"}
	OK キャンセル

- メモ ここで示されているサンプルJSON データは、デモのためでとても短いです。MapForce Server がマピングを実行すると、 JSON データを入力・ウメーターとして、実際のJSON ファイルからのマピング「提供することができます。
 - 9. 「挿入|出力の挿入」メニューコマンドを選択し、マソピングの出力を追加します。

₩ input	POST http://localhos	t:4646/service/POST_JSON		It result
🗟 default input 🔁	Þ戸 区 リクエスト	□□□ レスポンス	Þ	▶ result
	🔉 🔤 🖾 Body 🔚		Þ	
		∃ === ^y ダ-		
		Body 🔚	Þ	

10. charset-encode とmime-entity 関数をライブラリウィボウからドラッグし下に示されるようにすべての接続を作成します。 「入力 | 定数」メニューコマボを選択し、2つの定数を追加する必要があります。



上記のマピング内では、単純型コンポーネントの方法でJSON 入力がマピングに提供されます。charset-encode とmimeentity 関数は、HTTP リクエストのボデを準備する MapForce ビルトイン関数です。Web サービスにより返されるステータスコードは マピングにより返される結果にマップされます。

HTTP リクエストのボデを構成されていない方法で上記のようご準備することは、POST リクエスト内にデータを送信する方法の一つです。JSON とXML 構造のために、「Web Service 呼び出しの設定」 ダイアログボックス内のノクエストのJSON おけは XML スキーマを入力することができます。この場合、サービスエンポーネントのボディは、リクエストのJSON/XML 構造をベースリエ、マノピングの入力 (コネクタ)を提供します。

出力 ダブをクトックして、MapForce を使用してマッピングを実行することができます。エラーカ発生すると、メッセージンイドウに表示されます。デバッグするコよ、FlowForce Serverログを確認する必要かある場合かあります(POST リクエストかサーバーに達していることか前提とされています)。それ以外の場合、実行に成功すると、以下か発生します:

- 1. HTTP ステータスコード "200" が出力 ペインに表示されます。
- 2. サーイ 測で、提出された JSON データはファイルに書き込まれ C:\FlowForceExamples\PostJson ディレケトリに保存されます。

おっ、MapForce Server を使用してマピングを実行、およ、FlowForce Server にデプロイすることができます。おう、Web サービスにすることもできます。詳細に関しては、MapForce <u>https://www.altova.com/ja/documentation</u> ドキュメノトを参照してくけさい。

14.10 ジョブの結果をキャッシュする

この例では、(<u>キャッシュプロデューサー</u>として参照される)ショブの結果をキャッシュし、(<u>キャッシュコンシューマー</u>として参照される)他のショブ内 で使用する方法が説明されています。両方のショブは以下の振る舞いをするWebサービスとして公開されます:

- キャッシュプロデューサーWebサービスが呼び出されると、再帰的にディレケトリのコンテンンがリストされ、キャッシュを作成ませま更新し、ブラウザー内に結果を出力します。
- キャッシュコンシューマーWebサービスが呼び出されると、キャッシュプロデューサーサービスにより作成されたキャッシュを読み取り、ブラウザー内に結果を出力します。

目的は、両方のショブの実行にかかすま時間を比較し、2番目のショブの実行がキャッシュされたデータを使用することにより、1番目のショブの実行より、ためを確認することです。

必要条件

- 必要とされるライセンス FlowForce Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ネットワークアドレスとポート</u> でルスンされていること。
- メモ この例は、Windows パンとコマンドを使用していますが、パンとコマンドを必要に応じて変更すると、他のオペレーティングシステム 上でテストすることができます。

ジョブを構成する方法

- 1. 「構成」をクリックし /public/Examples コンテナーに移動します。前のサンプルゴ従っている場合 public/Examples コンテ ナーは既存であるごをです。それ以外の場合作成 | コンテナーの作成 コマンドを使用して作成します。
- 2. 「作成」をクトックし、「ジョブの作成」を選択します。
- 3. ジョブ名前ボックス内に Cached Result を入力します。
- 4. 実行ステップで、次の設定を持つ新しい実効ステップを追加します:

実行関数	/system/shell/commandline 関数を参照します。	
بر کت	以下のシェルコマードを入力します	
	dir /s	
	Windows でにのコマドは <u>作業ディノナリのコンテンルを再帰的に</u> リストしてます、次の設定参照)。	
作業デルオリ	FlowForce Server が作動するマンン上のディレクトリーの値を設定します	
	C:\	
	異なるディレケーを使用する場合、ディレケールのエレテンを再帰的にしてトするために少なくとも 20-30 秒かかる大きさのディレケーを選択してくたさい。	
ステップの結果を割り当て	次のステップ内の実行ステップによし返される値を参照する必要があり、名前が必要にないます。このサンプルのスコープのためにdirをこのフィールドの値として入力します。	

5. 実行ステップで、次の設定を持つ新しい実効ステップを追加します。

実行関数	/system/compute 関数。
式	次のFlowForce Server 式を入力します:
	stdout(dir)
	stdout 関数は前の実行ステップには返される生結果をデータのストリームに変換します(ステップ結果関数を参照してくたさい)。

6. [実行結果]で戻り型を stream に設定します。既知かもしれませんが、ジョブの最後の実行ステップにより返されたデータ型と同じ型に設定します。ジョブは以下のようようはます:

Exect	ution Ste	os			
•	Execute function /system/shell/commandline			▼ 🖉	
	Parameters:	Command: Abort on error:	dir /s		
		Working directory:	C:\		
-	Assign this st	tep's result to dir		as result	
-	Execute func	tion /system/comput	te		• 🖻
	Parameters:	Expression: stdou	ıt(dir)		
=	Assign this st	tep's result to name		as TO	
new	v Execution step	new Choose step	new For-each step	new Error/Success handling step	new Postpone step
Execu	ution Res	ult			
Decl	are return type	as: stream	•		

- 7. キャッシュの結果から結果のキャッシュチェックボックスを選択します。
- 8. 新規のキャッシュコンスーマージョブを自動作成 チェックボックスを選択し DirectoryListingCachedService を W eb サービスの名前として入力します。

Ca	ching Result		
	Cache the result	Cache is used whenever this job is called from another job.	
	 Add a time t Create a job If "Initiated b In case of job 	rigger to create and refresh the cached result or check "Initiated by consumer" option below. that will call this one and will benefit from the cache. ry consumer" option is chosen then add Refresh or Purge Cache timers to prevent cache entries to bec o input parameters present set "Initiated by consumer" option and set "Maximum number of cache en	ome too old. tries" to expected number of
	Initiated by con	sumer	
	Maximum number o	of cache entries: 1	
	Cache consumer job	<u>/public/Examples/CachedResult.cached</u> is available via HTTP at URL:	
	http://< <i>Flowf</i>	Force server>/service/ DirectoryListingCachedService	Delete consumer job
	new Refresh Cache	timer new Purge Cache timer	

9. サービスから「このジョブをHTTP を介して利用可能にする」チェックボックスをクリックして選択し、 DirectoryListingService をサービスの名前として入力します。



- 10.「資格情報」から、既存の資格情報の記録を選択、ませよローカルの資格情報を指定してくたさい(以下を参照:<u>資格情</u> <u>報</u>)。
- 11.「保存」をクリックします。

ジョブを実行する方法

この段階では、キャッシュプロデューサーとキャッシュコンスーマージョブの両方の構成が完了されました。非キャッシュサービス (DirectoryListingService)の プォーマンスをブラウザー内でテストするコお以下のアプローチル ずれかを使用してジョブを実行しま す:

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/DirectoryListingService</u>をブラウザーのアドレスバーに入力します。このURLは FlowForce Server サービスがデフォルトのオストアドレスとポート名でリッスンする場合のみ機能します。 構成ページ内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してください。
- FlowForce Server の任意のホスト名 フィールをセトアップページから設定するとHTTP を介してこのジョブを使用可 キリーオス チャックガックの使の ト ボケ ケイルク チャット サードスの取り出 たきまごうつびまびょうかごまたすることがで

能にする チェックボックスの横の と ボターをクトックして web サービスの呼び出しを直接ショフ構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コログオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウストを使用してWebサービスにアクセスします。HTTP 認証を無効化するには、Webサービスを公 開し、Service - Use パーミッションをユーザーAnonymous に与えます。パーミッションのしくみを参照してください。 C:¥ディレケーリのコンテンンを再帰的にリストするように構成されているため、完了するまでに数分かかる可能性があります。ションを完了するまでかかって時間を確認するために、ションログを参照してくたさい。ションのログを確認するを参照してくたさい。

キャッシュ顧客サービス(DirectoryListingCachedService)の ひォーマンスをテスト するは enter

http://127.0.0.1:4646/service/DirectoryListingCachedServiceをプラウザーのアドレスドーに入力します(おけまれなト名とポートが異なる方法で構成されている場合、同等のURLを入力します)。このサービスは、既存のディンケトリノストの代わりにキャッシュされたデータを使用するので、短い時間で完了することが期待されます。

14.11 StyleVision 変換からジョブを作成する

この別では、StyleVision 変換からFlowForce Server ジョブを作成する方法が説明されています。ジョブは、以下の3つのステップから構成されています。

- 1. 最初のステップは、最初にStyleVision 変換を実行します。
- 2. 変換は、複数のストリームの配列を返すため、2番目のステップは、FlowForce Server 式を使用して変換によれた 複数のファイルビアクセスします。
- 3. 3番目のステップは、ファイルをアーカイブフォルダーイニュピーします。

必要条件

- 必要とれるライセンス Style Vision Enterprise おは Professional エディション、 Style Vision Server、 Flow Force Server
- FlowForce Web Server とFlowForce Server サービスが構成された<u>ネルワークアドレスとポート</u> でルスンされていること。
- <u>コンテナー</u>の1つでパーミッションを有する FlowForce Server ユーザーアカウトを有していること(デフォルトでは、この例で使用されている/public コンテナートコよ認証されたユーザーがアクセスすることができます)。
- このサンプルで作成されすショブはファイルを一つのディレケトリから他のディレケトリニコピーします。FlowForce Server が作動する オペレーティングシステムでは、両方のディレケトリが存在している必要があり、これらのディレケトリ内にファイルを作成する権利を有 していることを確認してくたさい、このサンプルは以下のディレケトリを使用しています。
 - C:\FlowForceExamples\GenerateHtml は全ての処理が行われ、相対的ないなか解決されるジョブの作業 ディレケトリです。
 - C:\FlowForceExamples\Archive はジョブにより生成されたHTML ファイルがピーされる保存先のディレクトリです。

使用されるデモファイル

StyleVision Power Stylesheet(.sps) ファイルはXML ファイルを処理し、HTML を含む複数のフォーマトで出力を生成します。これはAutoCalc.sps と呼ばれ StyleVision [Examples] サンプルプロジェクト Examples > Basics > AutoCalc.sps で使用することができます StyleVision サンプルプロジェクトを StyleVision で開くには、Examples をプロジェクト メニューカら開きます。

Style Vision 変換をを Flow Force Server にデプロイする方法

最初にStyleVision からFlowForce Server へデモ.sps ファイルをデプロイレます。sps ファイルのデプロイとはHTTP (おけ構成されている場合 HTTPS)を介して StyleVision か変換にしようされたリノースをオブジェクトに整理し、FlowForce Server にのなすることを意味します。FlowForce Server に変換がデプロイされると、サーィ・ジョブを作成することができます。

StyleVision 変換をデプロイする方法:

- 1. StyleVision 内でAutoCalc.sps ファイルを開きます。
- 2. 「ファイル」メニューから、「Flow Force (Server) にデプロイする」をクリックします。このオプションが無効化されているとデザインタブが選択されていることを確認してくたさい。変換をPXF ファイルとして保存するようにプロンプトされると、デフォルトの設定をそのままし、、「OK」をクリックします。

Deploy Transformation				
Enter the host name and port of a FlowForce Administration Interface to deploy the current transformation.				
Server:	127.0.0.1 ~	Port:	8082	
<u>U</u> ser:	root		Use SSL	
Pass <u>w</u> ord:				
Login:	<default></default>	\sim		
Deploy As				
Path:	/public/AutoCalc.transformation		Browse	
	The path must start with a slash character.			
S <u>a</u> ve design o	hanges before deploying			
Open web browser to create new job				
		OK	Cancel	

- 3. サーバーとポート テキストボックス内にWeb 管理イクターフェイスのサーバー名とポートを入力します(例えば、FlowForce Web Server サービスがデフォルトのポートで同じマシン上でのリスティングの場合では127.0.0.1 と8082 など)。異なるアド レスとポートが構成されている場合これらの値を変更してくたさい。<u>ネットワーク設定の定義</u>を参照してくたさい。
- 4. User とPassword テキストボックスにFlowForce Server ユーザー名と アワードを入力します。
- 5. ディレクトリをログインドロップダウノノストからを選択、おけよ < Default>オプションをそのままします。

デルノオ・リサービス統合が有効化されていると、ドメインユーザー名と、アフードを入力し、ドメイン名をログインドロップダウン リストから選択します。詳細に関しては、デルノオ・リサービス設定の変更を参照してくたさい。

6. パステキストボックスは変換がデプロイされるデフォルトの 父を表示します。他のサンプルとの一貫性のように参照 をクリックし、パ スを /public/Examples/AutoCalc.transformation に変更します。前のサンプルゴ従ってしる場 合 /public/Examples コンテナーイは既存であるべきです。それ以外の場合、下のダイアログボックス内のコンテナーの作成 を クリックして作成します:

(1) Choose Deployment Name		×
Server containers:	Existing transformations:	
Occumentation Maintenance AptorXML Sensitive public Examples system		~
		~
Create Container	D <u>e</u> lete	
Name: AutoCalc.transformation		
	OK Cancel	

7. [OK] をクリックし「変換をデプロイする」ダイアログボックスの新規のジョブを作成するためにWeb ブラウザーを開く チェックボックスを選択します。

(3) Deploy Transformation X					
Enter the host name and port of a FlowForce Administration Interface to deploy the current transformation.					
Server:	127.0.0.1 ~	P <u>o</u> rt:	8082		
<u>U</u> ser:	root		Use SSL		
Pass <u>w</u> ord:	•••••				
Login:	Directly	~			
Deploy As					
Path:	/public/Examples/AutoCalc.transformation		Browse		
	The path must start with a slash character.				
Save design changes before deploying Open web browser to create new job					
		ОК	Cancel		

8. [OK] をクトックして変換をデプロイします。

デプロイが完了すると、FlowForce Server 管理インターフェイスはWeb ブラウザーを開き、部分的に記入されたジョブのページが表示されます。変換関数自体は以前に指定済みのエレテナー・パス保存されます。これでデプロイの過程は完了しまた。

ジョブの作成方法

上記の通りFlowForce Server に.sps ファイルをデプロイレた後、ブラウザーは部分的に入力されたショブページを表示します。入力済 みのプラメーターと共に最初の実行ステップが自動的に作成されます。

Exect	ution Step	DS .		
	Execute funct	tion /public/Examples/AutoC	alc.trai	nsformation
	Parameters:	InputXml:	•{	altova://packagedfile/Data.xml
		OutHtml:	<mark>.₽</mark> HTHL	+
		GenerateHtmlOutputAsMim	e:	(+)
		OutRtf:	₽ RTF	(+)
		OutFo:	₽ F0	•
		OutPdf:	₽DF	•
		OutDocx:	. 0008	•
		Working-directory:		
=	Assign this st	ep's result to name		as ReturnTypeDocx, ReturnTypeHtml, ReturnTypeFo, ReturnTypePdf,

ジョブを関数のページ(/public/Examples/AutoCalc.transformation)を開いてショブを作成し、ジョブの作成をクルクします。

ジョブを構成する方法:

1. デフォルトのジョブ名を[AutoCalc.transformaton.job]から詳細な名前に変更します。例えば[GenerateHtml]など。これ は任意のステップですが、名前が同じコンテナー内の他のジョブにより使用されている場合必要なステップです。

Create job in /public/Examples				
Job name:	GenerateHtml			
Job description:	Converts XML data to HTML.			

2. デフォルトで作成された最初の実行ステップを以下のように入力します。

実行関数	このフィールドは以前にデプロイされているStyleVision変換関数を指してます。そのまましてくたい。
᠂ᡔᢣ᠆ᡒ᠆	InputXml フィールドはジョブ(Data.xml) に シケージ済みの XML ファイルを含んで します。この例のスコープのために デメーター値そのまま こすることができます。 入力と出力 インスタンスの変更に関する詳細は ジョブとしてマンピングを実行し変換するを参照してく たさい。
	AutoCalc.html を出力ファイルとて宣言するためにOutHtml の横の ・ ボタン をクトックします。
	GenerateHtmlOutputAsMime はこのサンプルでは設置されていません。このオプ ションは通常 FlowForce Server を使用して送信された電子メールメッセーンの HTML 本文を準備するために必要とされます。詳細に関しては <u>send-mime</u> 関数を 参照してくたさい。
	オプション OutRtf、OutFo、OutPdf、OutDocx は対応するフォーマトで(RTF、 FO、PDF、Docx)変換出力を宣言します。これらの出力は現在のサンプル内で使用 されていない サム、変更されないままで保持されます。
	作業ディレクトリボックスでは、FlowForce Server かジョブの出力を保存する必要 かあるディレクトリーへの ゆを入力してくたさい、この例は、ジョブの作業ディレクトリとして C: \FlowForceExamples\GenerateHtml を使用しています。
ステップの結果を割り当て	出力を入力します。これは明示的に最初の実行ステップによし返された結果へという名前 [output]を与えます。この結果を後に参照する必要があります。

3. 「新規実行ステップ」をクリックし以下のように構成します:

実行関数	/system/compute 関数。
式	次のFlowForce Server 式を入力します:

	<pre>as-file(nth(results(output), 0))</pre>
	この式は、FlowForce Server に以下を行うように命令します。
	1. 宣言済みのoutput によ返された配列を取得するために式関数の結果を 呼び出します。
	2. 配列内の最初のカイアムを取得するシュー目数 nth 「配列をついます。 配列インデックスは、ゼロベースであるため、関数 nth の2番目の引数とての を使用します。
	3. ファイルとて宣言するためご値をas-file 関数についます。
ステップの結果を割り当て	html_fileを入力します。これによりFlowForce Server にステップにより返された結果にhtml_file とう名前が付けられるように命令します。この結果を後に参照する必要があります。

4. 「新規実行ステップ」をクリックし、ステップを以下のように構成します:

実行関数	<u>/system/filesystem/copy</u> 関数を参照します。
у - д	Set to トをクリックして html_file を選択します。
<u></u> ም-ሥ	C:\FlowForceExamples\Archive\AutoCalc.html
上書き	「上書き」チェックボックスを選択します。
作業デルオリ	C:\FlowForceExamples\GenerateHtml

この段階では、ジョブページの実行ステップセクションは以下のようごなります

E	kecu (+)	ition Step	DS			
	4	Execute function /public/Examples/AutoCalc.transformation				
		Parameters:	InputXml:	e[8	altova://packagedfile/Data.xml	
			OutHtml:	HTHL	AutoCalc.html	
			GenerateHtmlOutpu	tAsMime:	+	
			OutRtf:	<mark>.</mark> ₽TF	+	
			OutFo:	₽ FO	+	
			OutPdf:	₽DF	+	
			OutDocx:	0 0008	+	
			Working-directory:		C:\FlowForceExamples\GenerateHtml	
	-	Assign this st	ep's result to output		as ReturnTypeDocx, ReturnTypeHtml,	
	4	Execute funct	tion /system/compute	e		
		Parameters:	Expression: as-file	e(nth(results(o	utput), 0))	
	-	Assign this st	ep's result to html_fil	e	as TO	
	•			,		
	4	Execute funct	ion /system/filesyste	m/copy		
		Parameters:	Source:	{html_file}		
			Target:	C:\FlowForce	Examples\Archive\AutoCalc.html	
			Overwrite:	✓		
			Abort on error:	+		
			Working directory:	C:\FlowForce	Examples\GenerateHtml	

5. トリガーから「新規タイマー」をクリックします。 6. 実行の横から、タイマーを毎日、1日毎に実行するように設定します。開始の横から、ジョブが開始される日時を選択します。例:

Tr	riggers			
	Run	daily	✓ every 1	day(s)
	Repeat	+		
	Start:	ti 2020-10-21 🔻 🛛	14:34:00 💼	
	Expires:	+		
	Time zone:	Europe/Vienna	•	
	🗹 enabled	l		

- 9. 「資格情報」から、既存の資格情報の記録を選択、おけ、ローカルの資格情報を指定してくたさい。詳細に関しては<u>資格情報</u>を参照してくたさい。
- 10. [保存]をクルクします。

ジョブを実行する方法

トリガー内で指定された日付で、FlowForce Server は、StyleVision 変換ジョブを実行します。ジョブの実行に成功すると AutoCalc.html ファイルを C:\FlowForceExamples\Archive ディレクトリ内で使用できるようてなります。ジョブの実行が成功した かを確認するコよジョブログを参照します。

14.12 RaptorXML を使用してドキュメントを検証する

この例では、XML スキーマファイルを検証するジョブの作成する方法が説明されています。条件付きエラー処理を使用せず、検証結果をカ スタムログファイルませまプラウザーに書き込まれ、サーめ、このサンプルはファイルを検証する最も簡単な方法を説明しています。検証結果は FlowForce Server ログ内でのみ使用することができます。更に複雑な検証ジョブのサンプルはエラーログを使用して XML を検証するを 参照してくたさい。

このサンプルで使用される検証ジョブは Raptor XML Server のvalany 関数を呼び出します。valany 関数は型をベースドキュメントを検証します。唯一の必須の、デメーターとして検証するオンガニアイルを取り、XML ファイル、XML スキーマ、DTD スキーマ、および他のファイル型を検証するオンガニ使用することができます。Raptor XML 関数に関しては Raptor XML ドキュメントを参照してくたさい (https://www.altova.com/ja/documentation)。

必要条件

- 必要とされるライセンス FlowForce Server、RaptorXML(おけはRaptorXML+XBRL) Server。
- FlowForce Web Server とFlowForce Server サービスか構成された<u>ネットワークアドレスとポート</u> でルスンされているこ
- <u>コンテナー</u>の1つでパーミッションを有する Flow Force Server ユーザーアカウトを有していること(デフォルトでは、この例で使用されている/public コンテナートコは認証されたユーザーがアクセスすることができます)。

使用されるデモファイル

と

address.xsd ファイルを検証するサンプルジョブは Raptor XML Server インストールフォルダー内の次の タロオタはす: C:\Program Files\Altova\Raptor XML Server 2021\examples\address.xsd.

RaptorXML Server を他の異なるフォルダーにインストールしている以外は32-ビット RaptorXML Server を作動する64-ビット Windows での ぷはC:\Program Files (x86)\Altova\RaptorXMLServer2021\examples\address.xsd です。

その他のXML スキーマファイルも使用することができます。

ジョブの作成

- 1. FlowForce Server W eb 管理イクターフェイスコングイン /public/Examples コンテナーを開きます。前のサンプルご従っ ている場合 public/Examples コンテナーは既存であるくきです。 それ以外の場合 作成 | コンテナーの作成 コマンドを使用 して作成します。
- 2. [ジョブの作成]をクリックします。次に名前を入力し、任意でショブの詳細を任意で入力します。このサンプルは [ValidateSchema] をジョブ名として使用しています。
- 3. [新規の実行ステップ]をクリックします。
- 4. [実行関数]の横で/RaptorXML/valany 関数を参照します。必須の、ラメーターファイルは拡張されたフィールドとて表示されていることに注意してください。

Execution S	teps				
∡ Execute	function /RaptorXML/valany	ion /RaptorXML/valany			
Paramet	ers: Working directory:	+			
	Error Format: ?	(+)			
	File: ?				
	Error Limit: ?	+			
	Info Limit: ?	•			
	Verbose: ?	•			
	Warning Limit: ?	•			
	XML User Catalogs: ?	•			
	Import Strategy: ?	•			
	Mapping Strategy: ?	•			
	xsi:schemaLocation Strategy: ?	+			

- メモ valany 関数はRaptorXML コンテナーの下に直接存在し 特定のRaptorXML リノース マ対応するコンテナー内にも存在します。例えば、 2021、2つの違い 当時る詳細は ジェネト・クセリース固有の RaptorXML 関数を参照してくたさい。
 - 5. ファイルテキストボックス内に検証するスキーマファイルへの Sを入力します。例えば C:\Program Files\Altova\RaptorXMLServer2021\examples\address.xsd など。
 - 6. トリガーから新規のタイマーをクリックして詳細の特定の時刻にションを実行するトリガーを作成します。詳細に関してはタイマート リガーを参照してくたさい。
 - 7. 「資格情報」から、既存の資格情報の記録を選択、おさ、ローカルの資格情報を指定してくたさい。詳細に関しては<u>資格情</u> <u>報</u>を参照してくたさい。
 - 8. [保存]をクルルします。

ジョブを実行する方法

トリガー内で指定された日付と時刻でショブは作動します。ショブの実行が成功したかを確認するコは<u>ショブログ</u>を参照します。具体的コよ <u>
ヘノスタンスログ</u> ページ内で以下のようなエトリが成功した検証を示しています:

file:///C:/Program%20Files/Altova/RaptorXMLServer2021/examples/address.xsd: runtime="0ms"
result="0K" cmd="valxsd"

ファイルが検証したい場合、ジョブの実行は失敗したと考えられ、(少なくとも1つのステップが失敗したため)エラーがログ内にしポートされます。 この場合、ログされたエトリは検証エラーの詳細と共にresult="Fail"を表示します。
14.13 エラーログのある XML を検証する

この別では、XML ファイルをスキーマーン対して検証するショブの作成する方法か説明されています。ショブが何らかの理由で失敗した場合、 詳細がログファイルに書き込まれます。検証のためこ FlowForce Server の管理下で作動する RaptorXML Server のvalxmlwithxsd 関数を使用します。エラーのログに関してですが、この別で説明されている技術は RaptorXML Server に依存するものでけな く、他のジョブの型に適用することもできます。

メモ RaptorXML Server 関数は RaptorXML Server がインストールされた後に Flow Force Server 内で使用できるような います。

このサンプルマはジョブはWebサービスとして定義されており、ブラウザーからURLにアクセスして、オンデマトでトリガーすることができます。 他の例で説明されているようこタイマー(まけはファイルシステム)トリガーイごうブを追加することもできます。同じジョブは・リガーとWebサ ービスの組み合わせを追加することもできます。このようしして、ジョブは、Webサービス呼び出された場合、トリガーのルールで定義されたの みではなく、オンデマトドで実行されることも可能です。

必要条件

- 必要とれるライセンズ FlowForce Server、RaptorXML(おはRaptorXML+XBRL) Server。
- FlowForce Web Server とFlowForce Server サービスか構成された<u>やホワークアドレスとポート</u> でルスンされていること。
- 使用中のFlowForce Server ユーザーアカウトはの内のコンテナー(デフォルトでは、認証済みのユーザーカアクセスする ことのできるこのサンプルで使用されている/public コンテナー)へのトミッションか与えられています。
- このサンプル内で作成されるジョブは作動する都度ログファイルを生成します。ですから、FlowForce Server が作動するオペレ ーティングシステムでディレクトリ内にファイルを作成する権利を持つ必要が設ます。(この)例では、C: \FlowForceExamples\ValidateXml ディレクトリを使用します)。

使用されるデモファイル

この例で検証されたXML ファイルは、RaptorXML Server インストールフォルダー内の次の いこみます: C:\Program Files\Altova\RaptorXMLServer2021\examples\NanonullOrg.xml.

RaptorXML Server を他の異なるフォルダーニインストールしている以外は32-ビット RaptorXML Server を作動する64-ビット Windows での ぬは C:\Program Files (x86)\Altova\RaptorXMLServer2021\examples\NanonullOrg.xml です。

他のXML ファイルを検証のために使用することができます。

ジョブの作成方法

- FlowForce Server W eb 管理化ターフェイスコングイン /public/Examples コンテナーを開きます。前のサンプルゴ従っ ている場合 public/Examples コンテナーは既存であるくきです。それ以外の場合 作成 | コンテナーの作成 コマンドを使用 して作成します。
- 2. [ジョブの作成]をクリックします。次に名前を入力し、任意でショブの詳細を任意で入力します。このサンプルは [ValidateSchema] をジョブ名として使用しています。

Create job in /public/Examples						
Job name: Job description:	ValidateXml Validates an XML file. If the job fails due to any reason, the error message is written to a log file.					

3. ジョブ入カノラメーターで オタンを押して新規の ラメータ型 [ファイルとしての文字列]を作成します。例

Job Input Parameters			
Name: inputFile	Type: string as file	✓ Default: + Description	Path to the XML file to validate.

4. 「実行ステップ」の下の 🛨 ボタンをクトックし、「新規選択ステップ」を選択します。

E	xecu (+)	tion Steps				
		Execute with error/success	handling - on error, retry 0 times	•	۲	
		۲				
		On error 🗸	do	1.	۲	
		•				
		new error/success handler				

5. 「エラー/成功処理を使用して実行」から、 💽 ボタンをクトックして、次の設定を持つ新規の実行ステップの追加を選択します:

実行関数	/RaptorXML/valxml-withxsd 関数を参照します。
ノウメーター	XML ファイル・デメーターの横から
	inputFile ジョブ入カ・デメーターを選択します。

6. 「エラー時」条件から、 🛨 ボタンをクリックして、次の設定を持つ新規の実行ステップの追加を選択します:

実行関数	/system/compute 関数。
<i>\`</i> ¬ \ ~	式の値を以下に設定してくたさい
	<pre>as-file(stdout(failed-step()))</pre>
	中かっこの間の部分はしつのFlowForce 式です。この式は出力を受け取り、ストリーム に変換し、ディスク上のファイルにも一度書き込みます:
	 failed-step 関数は、失敗したステップのresult を返します。 exitcode、stderr、おはerror-message 関数への別数とて提供される更に使用しやすいようよるこれは抽象的な FlowForce 型です。 stdout 関数はストリームとて result からの標準出力を取得します。 as-file 関数はストリームからファイルを作成します。パンは次のステップで指定されます。
ステップの結果を割り当て	このショブの結果を一意に識別することのできる値を入力します。例えば、MyResult など。これを行うことにより、この値を変数として宣言し、次でステップで使用することができ

ます。

7. 以下の設定を使用して 한 ボタンをクトックして前のステップの後に新規の実行ステップを追加します:

実行関数	/system/filesystem/copy 関数を参照します。
᠈᠂ᡨᢣ᠆᠋ᡒ᠆	ソース・デメーターの横から MyResult 変数を選択します。
	「ターゲット」、 ラメーターの横に、 ログが保存される、 ひを入力します(この例では、 、 穴は C:\FlowForceExamples\ValidateXml\error.log です)。
	上書き、ウメーターの横のチェックボックスを選択します。ジョブが実行される都度、ログファ イルは生成されます。これにより、ログファイルが既存であっても、ジョブが失敗することはあり ません。
	Working Directory / ラメーターをC:\FlowForceExamples\ValidateXml 設定します。

ジョブのエラー時」ブランチは以下のようないます

On er	ror	🗙 do		
	For Execute funct Parameters: Assign this st Execute funct Parameters:	 do tion /system/comput Expression: as-file rep's result to MyResult tion /system/filesyste Source: Target: Overwrite: Abort on error: Working directory: 	e e(stdout(failed-step())) ult as T0 em/copy {MyResult} error.log • ChEleusEesesEusenelee)//stidet=Yeel	• @
		working directory:	C:\FlowForceExamples\validateXml	

- 8. サービスから「このジョブをHTTP を介して URL 上で利用可能にする」 チェックボックスをクトックして選択し、 ValidateXmlService をサービスの名前として入力します。
- 9. 「資格情報」から、既存の資格情報の記録を選択、おけ、ローカルの資格情報を指定してくたさい。詳細に関しては資格情報を参照してくたさい。
- 10. [保存]をクリックします。

ジョブを実行する方法

ジョブを実行するコよ以下の内のしつを行ってくたさい

• ホームに移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。

- <u>http://127.0.0.1:4646/service/ValidateXmlService</u>をブラウザーのアドレスバーに入力します。このURLは FlowForce Server サービスがデンオルトのオストアドレスとポート名でリンスンする場合のみ機能します。 構成ページ内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してくたさい。
- FlowForce Server の任意のホスト名 フィールドをセトアップページから設定するとHTTP を介してこのジョブを使用可

能にする チェックボックスの横の と ボシをクリックして web サービスの呼び出しを直接ジョフ構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コングオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーに**Service - Use** パーミッション を与え、対応するユーザーアカウトを使用して W eb サービスにアクセスします。HTTP 認証を無効化するはよ W eb サービスを公 開し、**Service - Use** パーミッションをユーザー Anonymous に与えます。<u>パーミッションの人</u>みを参照してください。

ジョブが ラメーターを入力として期待するように構成されているために、ブラウザーは検証される XML ファイルへの ひを入力可能なフォームを表示します。

<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools	<u>H</u> elp		—		×
loca	lhost:4	4646/se	rvice/Vali	dateXmIS 🗙	+					
¢) →	G	۵	000	calhost	:4646/service/ValidateXmlService	•••	111	0	≡
	Paran nputF	neters File *:							 	
S	ubmit									

XML ファイル ウをテキストボックスに入力します (例えば C:\Program Files\Altova\RaptorXMLServer2021\examples\NanonullOrg.xml) そして Submit をクリックします。

ジョブの実行に成功すると、(つまり、終了コード「0」を返します)ブラウザーは、ジョブの標準の出力を表示します。

file:///C:/Program%20Files/Altova/RaptorXMLServer/examples/NanonullOrg.xml:
runtime="16ms" result="OK"

(例えば、有効でないいな、検証エラー、などによるジョブが「0」以外の終了コードを返すと、ブラウザーは、「サービスの実行の失敗」メッセージを表示し、出力がC:\FlowForceExamples\ValidateXml\error.log ファイルに書き込まれます。ログファイルが生成されない場合、エラーを識別するために、ジョブログをチェックします。以下の可能性が考えられます。例えば、/system/filesystem/copy 関数は、ターゲトノマへ書き込むためのペーミッションが不足しているために、失敗したなど。

14.14 RaptorXML を使用して XSLT を実行する

このサンプルはFlowForce Server 管理下で作動する RaptorXML Server (おけは RaptorXML+XBRL Server)を使用して XSLT 変換を行います。ジョブは RaptorXML Server のxslt 関数を呼び出します。 FlowForce Server 構成ページからジョブを構成する際にxslt 関数に デメーターを提供する方法があります。

- 1. テキストボックス内にキー値ペア(ノテメーター名と値)を入力する。
- 2. FlowForce Server 式を入力する。

これらの方法の詳細は以下で説明されています。

必要条件

- 必要とれるライセンス FlowForce Server、RaptorXML(おはRaptorXML+XBRL) Server。
- FlowForce Web Server とFlowForce Server サービスか構成されたホートワークアドレスとポート でリスンされていること。
- 使用中のFlowForce Server ユーザーアカウトはの内のI つのユンテナー(デフォルト では、認証済みのユーザーカアクセスする ことのできるこのサンプルで使用されている/public コンテナー)への トミッションか与えられています。
- このサンプルで作成されたショブは入力 XML ファイルを処理する XSLT スタイルシートを実行します。両方のファイルはオペレー ティングシステム上のディレケトリニ存在する必要があり、このディレケトリ内のファイルを読み込む、おけはファイルに書き込む権利が必要です。このサンプルは C:\FlowForceExamples\RunXslt を使用しています。

デモファイル

下て説明されるジョブは books.xml と呼ばれるファイルとっつの必須、ウメーター[year] と[genre] を入力として取る transformation.xslt と呼ばれる XSLT スタイルシートを実行します。ファイルの実際のコンテンソは下のコードリスティングで表示されてい ます。ジョブ内でこれらのファイルを使用する ゴよ 示されているファイル名と共に両方のコードリスティングを C: \FlowForceExamples\RunXslt directory に保存します。

XSLT スタイルシートは以下のようてなります。

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"</pre>
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:fn="http://www.w3.org/2005/xpath-
functions" exclude-result-prefixes="xs fn">
   <xsl:output method="xml" encoding="UTF-8" byte-order-mark="no" indent="yes"/>
   <xsl:param name="year" as="xs:string" required="yes"/>
   <xsl:param name="genre" as="xs:string" required="yes"/>
   <xsl:template match="/">
      <library>
         <xsl:attribute name="xsi:noNamespaceSchemaLocation"</pre>
namespace="http://www.w3.org/2001/XMLSchema-instance" select="'library.xsd'"/>
         <last updated>
            <xsl:sequence select="xs:string(fn:current-dateTime())"/>
         </last updated>
         <xsl:for-each select="(./books/book)[(fn:string(year) &gt; $year)]">
            <publication>
               <xsl:for-each select="@id">
                  \langle id \rangle
                      <xsl:sequence select="fn:string(.)"/>
                  </id>
               </xsl:for-each>
               <author>
```

```
<xsl:sequence select="fn:string(author)"/>
               </author>
               <title>
                  <xsl:sequence select="fn:string(title)"/>
               </title>
               <genre>
                  <xsl:sequence select="$genre"/>
               </genre>
               <publish year>
                  <xsl:sequence select="xs:string(xs:integer(fn:string(year)))"/>
               </publish year>
            </publication>
         </xsl:for-each>
      </library>
   </xsl:template>
</xsl:stylesheet>
```

transformation.xslt

入力 XML ファイルは以下のよう」なります:

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books.xsd">
   <book id="1">
      <author>Mark Twain</author>
      <title>The Adventures of Tom Sawyer</title>
      <category>Fiction</category>
      <year>1876</year>
   </book>
   <book id="2">
      <author>Franz Kafka</author>
      <title>The Metamorphosis</title>
      <category>Fiction</category>
      <year>1912</year>
   </book>
   <book id="3">
      <author>Herman Melville</author>
      <title>Moby Dick</title>
      <category>Fiction</category>
      <year>1851</year>
   </book>
   <book id="4">
      <author>Miguel de Cervantes</author>
      <title>Don Quixote</title>
      <category>Fiction</category>
      <year>1605</year>
   </book>
</books>
```

books.xml

ジョブの作成方法

XSLT 変換などのFlowForce Server ジョブを以下のように作成することができます:

- FlowForce Server W eb 管理化ターフェイスコングイン /public/Examples コンテナーを開きます。前のサンプルゴ従っ ている場合 public/Examples コンテナーは既存であるくきです。それ以外の場合 作成 | コンテナーの作成 コマンドを使用 して作成します。
- 2. [ジョブの作成]をクリックします。次に名前を入力し、任意でジョブの詳細を任意で入力します。このサンプルは[RunXslt]をジョ ブ名として使用しています。
- 3. [新規の実行ステップ]をクリックします。
- 4. [実行関数]の横で/RaptorXML/xslt 関数を参照します。

E	xecu +	ution Step	S		
		Execute funct	tion /RaptorXML/xslt		▼ 🛃
		Parameters:	Working directory:	+	
			Error Format: ?	+	
			XSLT File: ?		
			Disable Chart Extension: ?	•	

- メモ xslt 関数はRaptorXML コンテナーの下に直接存在し、特定のRaptorXML リリースに対応するコンテナー内にも存在します。例えば、「2021」。2つの違いは関する詳細はジェネルックとリース固有のRaptorXML 関数を参照してくたさい。
 - 5. 作業ディレクトリィラメーターをC:\FlowForceExamples\RunXsl に設定します。
 - 6. XSLT File パラメーターを transformation.xslt に設定します。このファイルは作業ディレクトリ内に存在する必要かあます。
 - 7. XSLT Input パラメーターを books.xml に設定します。このファイルも作業ディレクトリ内に存在する必要があります。
 - 8. パラメーター パラメーターを以下のように設定します。

a. ノラメーターの横の 한 ボタンをクリックします。これにより洛ノ・ラメーター名と値をそれそれ追加できるページ内のサブセクション が展開されます。

Parameters: ?	•
Enable streaming serialization: ?	+

b. 追加する必要のあるそれぞれの新規/ラメーターのために いた ボタンをクトックします。このサンプル内のXSLTを実行するため に、以下のようこ ・ラメーターを入力する必要があります:

4	•						as sequence of (string, string) (optional)	Set to 🕨 🗃 💼
		year	Set to 🕨	'1900'	Set to 🕨	1 🕹 💼		
		genre	Set to 🕨	'Fiction'	Set to 🕨	1.		
	•							

XSLT パラメーターはジョブにキー値ペアとして提供されます。パラメーター名と値は個別のボックス内に入力されることに注意してくたさい。また、パラメーター値におり用符で囲まれています。

- 9. サービスから「このジョブをHTTP を介して URL 上で利用可能にする」 チェックボックスをクリックして選択し RunXsltService をサービスの名前として入力します。
- 10.「資格情報」から、既存の資格情報の記録を選択、おさ、ローカルの資格情報を指定してくたさい。詳細に関しては<u>資格情</u> 報を参照してくたさい。

11. [保存]をクリックします。

これはジョブの構成の部分を完了します。

XSLT パラメーターを式として与える方法

上記のジョブ構成ではxslt 関数にテキストボックスを使用してパテメーターが提供されました。これを行うために2番目の方法が数据す。 FlowForce Server 式 パテメーター テキストボックスに入力します。例:

Named Template Entry Point: ?	+
Primary Output: ?	•
Parameters: ?	list(("year","'1900'"),("genre","'Fiction'"))
Enable streaming serialization: ?	+

この2番目のアプローチを使用して パラメーターの横の。Set to Mail をクリックし <Expression> をクリックします。解析エラーを回避する たかに式の型を慎重に入力してくたさい。式は list 式関数を呼び出し、2つのキー値ペアを作成します。キー値ペアでは、キーは デメーター 名で、値は デメーターの値です。 重要な点は デメーター値が単一の引用符で囲まれてい る点です。

テキストレイアナトに切り替えるゴは Set to > パラメーターの横をクトックして < Value>をクトックします。

ジョブを実行する方法

ジョブを実行するコよ以下の内のしつを行ってくたさい

- ホームニ移動し全てのアクティブなトリガーとサービスを表示するをクリックします。次に「情報列内に表示されているジョブの URLをクリックします。
- <u>http://127.0.0.1:4646/service/RunXsltService</u>をブラウザーのアドレスバーに入力します。このURLはFlowForce Server サービスがデフォルトのオストアドレスとポート名でリッスンする場合のみ機能します。構成ページ内で他のオストとポート設定を定義した場合、アドレスを必要に応じて変更してくたさい。
- FlowForce Server の任意のホスト名 フィールドをセトア・プページから設定するとHTTP を介してこのジョブを使用可

能にする チェックボックスの横の と ボタンをクリックして web サービスの呼び出しを直接ジョン構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server にログオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウストを使用してWebサービスにアクセスします。HTTP 認証を無効化するには、Webサービスを公 開し、Service - Use パーミッションをユーザーAnonymous に与えます。パーミッションのしくみを参照してください。

ジョブの実行に成功すると、ブラウザーはジョブの出力を表示します。例



ジョブの実行に失敗するとブラウザーは、「サーバー実行の失敗」メッセージを表示します。この場合、FlowForce Server 内の<u>ジョブのログ</u>をチェックして、エラーを確認してください、ジョブのログの確認を参照してください。

14.15 複数の PDF を複数の XML から生成する

このサンプルでは、複数のXML ファイルを入力として取り、出力複数のPDF ファイルを返す FlowForce Server ジョブの作成の方法について説明されています。FlowForce Server ジョブは、MapForce Server (複数のソースXML ファイルからXML 出力を生成するために)とStyleVision Server (XML 出力をPDF に変換するために)を呼び出します。

このサンプルは、MapForce マピングとStyleVision 変換に関する基本的な理解を必要とます。StyleVision とMapForce の 初心者の場合は、MapForce とStyleVision ドキュントの チュートリアルリチャプターを最初に読むことが奨励されています。

- <u>クイックスタートチュートリアル(MapForce)</u>
- <u>クイックスタートチュートリアル(StyleVision)</u>

必要条件

- 必要とされるライセス
 - MapForce Enterprise おけま Professional エディションのインストール。このソールによりスキーマから他のスキーマに XML ファイルを変換するマンピング変換(.mfd ファイル)をデザインすることかできます。
 - MapForce Server おはMapForce Server Advanced Edition。このソールによりサードートでマナピングを ジョブとして実行することができます。
 - StyleVision Enterprise おさま Professional エディション。このソールにより XML ファイルから PDF ファイル 変換するスタイルシート (.sps ファイル) をデザインすることができます。
 - o StyleVision Server.このソールによりサーバー上で変換をショブとして実行することができます。
 - FlowForce Server このソールは上記の変換を計画通り、ませオンデマ・ドショブとして実行し、入力を必要な場合は変更する方法を提供します。
- ・ FlowForce Web Server とFlowForce Server サービスか構成された<u>ネットワークアドレスとポート</u> でルスンされていること。
- 使用中のFlowForce Server ユーザーアカウトはの内の つのユンテナー(デフォルトでは 認証済みのユーザーカアクセスする ことのできるこのサンプルで使用されている/public コンテナー)への キミッション か与えられています。

使用されるデモファイル

このサンプルは、次の、次でみつけることのできるサンプルファイルを使用しています: <Documents>\Altova\MapForce2021\MapForceExamples。

- MultipleInputToMultipleOutputFiles.mfd (MapForce マメピングファイル)
- PersonListWithGrouping.sps (StyleVision 変換 ファイル)
- Nanonull-Branch.xml、Nanonull-HQ.xml (入力 XML ファイル)

MapForce マピングのしくみ

下に表示されるとおり、マピングは、ソースコンポーネト(Altova_Hierarchical)、ターゲトコンポーネト(PersonList)、出力に 書き込むかの他の文字列をビルドするかがご使用されると多種のMapForce ビルトル、関数から構成されています。



MultipleInputToMultipleOutputFiles.mfd

マピングはディノケリ < Documents>\Altova\MapForce2021\MapForceExamples から Nanonull-」とう名前のXML ファイルを入力とて取ります。この項目をノース MapForce エンボーネント設定内で定義することができます(MapForce 内では、下に表 示されるようこ Altova_Hierarchical エンポーネントのヘッダーを右クリックし、エンテキストメニューから「Properties」を選択します)。 入力ファイルは、アステリスクが「フイルドカードの役割を果たす、Nanonull-*.xml に設定されています。文字通り、入力は 「Nanonull-」から開始する、xml 拡張子を持つファイルです。

😢 Component Settings			×
Component name: Altova_Hierarchical			
Schema <u>Fi</u> le			
Altova_Hierarchical.xsd	Bro <u>w</u> se	Edi <u>t</u>	
Input XML File			
Nanonull-*.xml	Browse	Edit	

ターゲトコンポーネント、PersonList は、ソースXML ファイルのファイル名をベースにしてファイル名を動的に生成するように構成されて します。これは、コンポーネントの上の「File/String」ボタンを右クトックし、「マッピングにより与えられる動的ファイル名を使用する」メ ニューオプションを選択することに定義することができます。「ファイルくdynamic>」ノードへの接続は、新規ファイルはノース内の各値の ために作成されます。remove-folder 関数は、ソース・マンカンマナルダー名無しの、ファイル名のみを取得することを意味します。これは Persons-<Source filename> などの文字列をビルドするトップ concat 関数に対する値として、マンシれます。

2番目の concat 関数は、マンピングルスへの完全なルマが後に続く、Generated by Altova... などの文字列をビルドします。結果は ターゲナ XML ファイル内のコメントとして書き込まれます。 3番目のconcat 関数は文字列をビルドするために、ソースからマップされる個人のレコードの個数を示す count 関数の出力を使用します。結果はターゲット XML ファイル内のコメトとして書き込まれます。

ターゲオ Person ノードへの接続は、ターゲオへのノースから個人のデータをエピーします。マップされる必要のある「Person」の各子要素のための個別の接続が存在します。

更に、ターゲトコンポーネトで生成された出力を各 XML に生成されたファイルのためにPDF に変換するように構成されています。ターゲトコンポーネトの、ダーを右クリンプ、Properties を選択し、StyleVision .sps スタイルシート への相対パンを指定する StyleVision Power スタイルシート ファイルテキストボックスを確認してくたさい。後者は、XML からPDF への実際の変換を実行しまず、下で更に詳しく説明されています。。

PersonListWithGrouping.sps	Browse Edit	
Enable input processing optimizations bas	sed on min/maxOccurs	
✓ Enable input processing optimizations base ✓ Save all file paths relative to MFD file	sed on min/maxOccurs	

このマピングの出力を直接 MapForce 内でルビューするコよ マピングペインの下の出力 タブをクルクします。StyleVision 変換の PDF 結果をルビューするコよ PDF タブをクルクします。出力ペイン内で複数のXML(おけよPDF)が出力されます。以下を参照し てください

•	•	Preview 1 of 2	C:\Users\ \Documents\Altova\MapForce2017\MapForceExamples\Persons-Nanonull-Branch.xml
1		xml version="1.0"</th <th>encoding="UTF-8"?></th>	encoding="UTF-8"?>
2		Generated by Alt</th <th>ova MapForce (http://www.altova.com/mapforce) using C:\Users\ \Documents\Altova\MapForce2017</th>	ova MapForce (http://www.altova.com/mapforce) using C:\Users\ \Documents\Altova\MapForce2017
3	Ę	PersonList xsi:noNa	<pre>imespaceSchemaLocation = "PersonList.xsd" xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"></pre>
4		6 Persons map</th <th>ped from input file C:\Users\ \Documents\Altova\MapForce2017\MapForceExamples\Nanonull-Branc</th>	ped from input file C:\Users\ \Documents\Altova\MapForce2017\MapForceExamples\Nanonull-Branc
5	4	Person role="Of	fice Manager*>
9	9	Person role="Accession" Accession	counts Receivable*>
13	9	Person role="PR	& Marketing Manager US">
17	9	Person role="IT I	Manager'> []
21	9	Person role="Su	pport Engineer"> []
25	9	Person role="Su	pport Engineer"> []
29	L		
30			

この段階では、(デフォルトでは、MapForce は一時ファイルを作成するためディスクに2つの出力 XML ファイルの1つを保存することが奨励されます。StyleVision内で StyleVisionパワースタイルシートを開きテストする場合ファイルは、サンプル(作業 XML)としての役割を果たします、次のセクションを参照してくたさい、出力ファイルを保存するこは、「出力」タブをクトックして、「出力」メニューから、「出力ファ イルの保存」をクトックします。

Style Vision 変換について

StyleVision を作動し、PersonListWithGrouping.sps 変換 ファイルを開きます。このファイルは、MapForce マピングで説明 されている同じディノケーと同じファイルで、ターゲナ MapForce エノポーネントによ参照されています。



PersonListWithGrouping.sps

上記のStyleVision .sps スタイルシートは、単一のXMLをノースとして表示しており、PDF を作成します。PDF ドキュメントは、見出し(「h2」)、紹介パラグラス、動的に作成されたテーブル、終わりのパラグラスには構成されています。見出しと紹介パラグラスは、静的テキストを含んでしますか、終わりのパラグラスは、ラノビングタブで指定されているとおり、ソース XML ファイルのノードから作成されています。

Style Vision でこの変換を直接プレビューするコよ下のステップゴビレます:

1. 「デザイン概要」ペイン内で、「作業 XML」 🔮 をクトックします。

Design Overview	- 	×
∇ Sources		<u>^</u>
🚰 \$XML (main)	0	
Schema PersonLi	0	
Working XML	0	
Template XML	E.	Edit File in XMLSpy
XML Signature Disabled		
add new Source	Ľ.	Assign Working XML File
∇ Modules	~	
add new Module	\sim	Unassign Working XML File
∇ CSS Files		
add new CSS File		

- 2. 「作業 XML ファイルの割り当て」を選択して、MapForce から保存された XML 出力ファイルを参照します、前のセクションを参照してくたさい。
- 3. PDF タブをクトックします。

.sps スタイルシートは、実際の名前、おゴは、ソースXML ファイルのオリジイン対して未知であり、(指定されたXML スキーマーン準拠する限 り入力として与えられたXML ファイルを処理し、 PDF を作成します。このスタイルシートを自動化し、複数のPDF ファイルを生成するようご構成するコよ、下に示されるようこ、 FlowForce Server にデプロイされる必要があります。

Flow Force Server にファイルをデプロイする

このサンプル内で使用される MapForce マピングとStyle Vision 変換の目的について説明されまたが、MapForce マピングと Style Vision スタイルシートのデザインに関する詳細に関しては、これらの製品のドキュメートを参照してくたさい (https://www.altova.com/documentation.html)。

自動化を可能にするために、両方のファイルは、FlowForce Server にデプロイされる必要かあります。上の「必要条件」セクションで指定されているとおり、FlowForce Server にライセンスか与えられており、MapForce Server とStyleVision Server が FlowForce Server の管理下で実行されている必要かあります。Windows 上では、ライセンスのステータスをチェックするための verifylicense コマイドを使用することができます。ライセンスか見つからは、いまけは、無効な場合、他のオペレーティングシステムでは、 ジョブ実行に失敗すると、エラーか表示されます。

StyleVision スタイルシートをFlowForce Server にデプロイする

- 1. 「ファイル」メニューから、「Flow Force (Server) にデプロイする」をクトックします。(このコマイトが灰色表示されている場合 デザイン タブに最初に切い替えます。)
- 2. ファイルをPXF (Portable XML Form) フォーマナとしてデザインファイルを保存するようにプロンプトされると「OK」をクリックします。

StyleVision	×
FlowForce ヘデザインをデプロイするには、デザインを PXF フォー ットで保存する必要があります。	-マ
ОК ‡ т	ンセル

3. デプロイされオッシケージ内に含まれるようコファイルを選択するようコプロンプトされると、デフォルトの設定を保留してくたさい。このサンプル内で、PDFのみが生成されますが、他の出力を含む出力で保存すると、後でHTMLとRTFなどの追加フォーマナで出力することができます。

🕒 Portable XML Form (PXF) の構成	×
Portable XML Form のコンテンツを構成	
デザイン時ファイル その他のファイル	_
パスならびにファイル名が StyleVision により認識されているイメージ ファイルなどのファイルは、自動的に PX ァイルに埋め込むことができます。 PXF ファイルに埋め込むファイルを選択してください。	FJ
グローバル構成 SPS ファイルの埋込み	V
イメージの埋込み	
XSLT ファイルの埋込み	
SPSモジュールの埋込み	
Authentic カスタムホタンを埋め込み フナーフソーフ cymu (メイン)	
スキーマ ファイルの埋込み	
作業用 XML ファイルの埋込み	
一時 XML ファイルの埋込み	
XSLT ファイルを生成し、XMLSpy や Authentic Desktop にて 以下を作成	
T HTML(H)	
Vord 2007+(W)	
L PXF ファイルを保存した後は、デザイン概要からこのダイアログにアクセスして、オプションの変更を行うことができ	きます。
ОК ++>	セル //

- 4. プロンプトされると、FlowForce Web Server への接続の詳細を入力します。簡素化のために、下のイメージ内では、変換は プレーンHTTPを使用してローカルズンのポート8082 にデプロイされていますが、FlowForce Web Server がこのような接続を受け入れるように構成されていると前提して、リモートアドレスを指定して、SSL-暗号化された接続を使用してファイルをデプロイすることもできます。<u>ネルワーク設定の定義</u>を参照っしてくたとい。ユーザーク、次ワードの値はレート FlowForce アカウトのために下て説明されています。しかしなから指定されている。マスニデーダを書き込むメーミッションカあることが前提で、その他のFlowForce ユーザーアカウトも使用されている場合かあります。このサンプルでは、ジョブの構成は異なるステップであり、後に説明されているため、「新規ジョブを作成するため」ごプラウザーを開く」チェックボックスは、意図的にチェックされていません。
- 5. 他のサンプルとの一貫性のために、ターゲナノス/public/Examples/PersonListWithGrouping.transformation を使用することが奨励されます。

い 変換のデプロイ	, ,			X
このマッピングをデプロ ださい。	コイする FlowForce 管理イ	ンターフェイスの木	スト名とポートを	入力してく
サーバー(S):	127.0.0.1		ポート (o) :	8082
ユーザー <mark>(</mark> U):	root			SSLを使用
パスワ <i>ー</i> ド <mark>(w)</mark> :	••••			
ログイン <mark>(L)</mark> :	<デフォルト >		-	
デプロイ				
パス (P):	/public/PersonListWithGr	ouping.transforma	ation 参	照(B)
	パスはスラッシュから開始	する必要があります	す 。	
✓デプロイ前にテ	"ザインの変更点を保存(A −を聞き新たな◇海づを作F) 芯する (1)		
			ОК	キャンセル

MapForce マピングをFlowForce Server にデプロイする

• 「ファイル」メニューから、「Flow Force Server にデプロイする」をクリックします。Style Vision のために上て説明されている とお、接続の詳細を記入します。また他のサンプルと一貫して、ターゲトパ ス/public/Examples/MultipleInputToMultipleOutputFiles.mapping が使用されることが奨励されます。

😢 マッピングのき	デプロイ			×
現在のマッピングをき 番号を入力してくだ	デプロイするため FlowForce さい。	eWeb 管理インタ、	-フェースのホスト	名とポート
サーバー (S):	10.100.61.20		ポ−Ւ (o) :	8082
ユ <i>ー</i> ザー (U):	root]		SSLを使用
パスワ <i>ー</i> ド(w):	••••]		
ログイン <mark>(L)</mark> :	ディレクトリ		•	
デプロイー				
パス <mark>(</mark> P):	/public/MultipleInputToM	IultipleOutputFiles	.mapping 参	照(B)
	パスはスラッシュから開始	する必要がありま	す。	
デプロイ前に	マッピングを保存(A)			
🔲 ウェブ ブラウサ	「ーを開き新たなジョブを作	成する <mark>(</mark>)		
			ОК	キャンセル

FlowForce Server にエグインすると、ファイルのデプロイロ成功すると、指定されたFlowForce エンテナート対応するエントリカ表示されます(この場合、 "/public/Examples"):

ホーム 構成 ログ 管理 ヘルプ	
	'/public/MultipleInputToMultipleOutpu
C / C public	
□ 名前	
MultipleInputToMultipleOutputFiles.mapping	
PersonListWithGrouping.transformation	
作成▼オブジェクトのインポート 選択	されたオブジェクトの移動または名前の変更をする

上記のエトリは、おこジョブではないことご注意してください。実際のジョブが作成される FlowForce 関数を作成する必要があます。

Flow Force ジョブの作成と構成

MapForce マピングとStyleVision 変換がFlowForce Server にデプロイされました。これらよ 必要とするションを作成するために使用されます。方法は以下のとおりです:

1. FlowForce /public/Examples コンテナーニナビゲートし、以前にデプロイされた男数 MultipleInputToMultipleOutputFiles.mapping をクルクします。MapForce マピングのノースコンポーネントは FlowForce 関数につけする入力パラメーター オンつびます。おこ、マピングにより処理されるインスタンス XML ファイルロン ひされ るデフォルトの値が存在します。この値は、必要に応じて後に上書きすることができます。「作業ディレクトリ」パラメーター か自動的 にFlowForce により追加されています。次のステップでこの、ラメーターの役割が次のセグションで説明されています。

以下に関数 MultipleInputT /public	oMultipleO	utputFiles.mapping を作成
関数入力パラメーター		
名前: Altova_Hierarchical 🚦 型: 文字列	デフォルト: C:/Users/	v/Documents/Altova/MapForce2018/MapForceExamples/Nanonull-*.xml
名前: Working-directory 型: ディレクトリとしての文字列	() デフォルト:	
「MapForce' バージョン '2018' により関数が実行されます。		
ジョブの作成		

- 2. [ジョブの作成]をクトックします。
- 3. 名前を入力し、任意でショブの詳細を任意で入力します。
- 4. ジョブの「実行ステップ」部分を以下のように構成します。

実	行	ステップ
	+	
	4	関数の実行 /public/MultipleInputToMultipleOutputFiles.mapping 🔹 🖻
		パラメーター: Altova_Hierarchical: 引 🖣
		Working-directory: C:\FlowForceWorkingDir
	=	ステップの結果を割り当て output 以下としてPersonList
	•	
	4	For each item シーケンスで results(output)
		•
		▲ 関数の実行 /system/compute
		AS-The(nem)
		= ステップの結果を割り当て file 以下としてTO
		•
		▲ 関数の実行 /public/PersonListWithGrouping.transformation 🔹 🗗
		パラメーター: InputXml: •file}
		OutHtml:
		GenerateHtmlOutputAcMime:
		OutPdf: {filename(file)}.pdf
		OutDocx:
		Working-directory: C:\FlowForceWorkingDir
		= ステップの結果を割り当て 名前 以下としてReturn TypeDocx, Return TypeHtml, Return TypeF
	=	ステップの結果を割り当て 名前
	新規	の実行ステップ 新規の選択ステップ 新規の For-each ステップ 新規エラー/成功処理ステップ

5. 「サービス」から「HTTP を使用してこのジョブを使用する」チェックボックスを選択し、オンデマイでトリガーされるWebサービスの 名前を入力します。例えば GeneratePdfsService か挙げたれます。ジョブを予定して実行する おは、ファイルシステムト リガーとして実行するは、適切なトリガーを設定してくたさい(トリガーの管理を参照してくたさい)。

サービス 🗹 このジョブを HTTP を介して URL 上で利用可能にする http://<FlowForce server>/service/ GenerateMultiplePDFs

6. 「資格情報」にオペレーティングシステムユーザーアカウトのユーザー名と マワードを入力します(ジョブは このユーザーとして実行されます)。FlowForce Web 管理インターフェイスの マワードと マワードを区別してくたさい(<u>資格情報</u>を参照してくたさい)。



7. [保存]をクルクします。

ジョブの作動を理解するコよジョブの「実行ステップ」セクションに注目してくたさい。最初の実行ステップは前にデプロイされたマッピングを呼び出します。作業ディレオリ内で[Nanonull-] から始まる XML ファイルを検索します。このサンプルでは、作業ディレオリはC: \FlowForceExamples\GeneratePdfs です。

最初の実行ステップにより返された出力は、マッピングにより返されたデーダを表しています。次のステップ内で参照できるように、outputと明示的に名前が付けられています。

ジョブの2番目のステップは「for-each」ステップです。「for-each」ステップがFlowForce 式 results (output) を使用して、(マッ ビングによ返された出力である最初のステップによ返されたデータにアクセスします。具体的には、式は、前のステップによ返された引数 出力として取る関数 results () を呼び出します。ステップ結果関数を参照してくたさい。FlowForce 式の基本的な紹介に関しては FlowForce 式 を参照してくたさい。

「for-each」 ステップは 2つの小さ 実行ステップに お構成されて ます:

- 1. 最初のステップは、マピング出力を通常 file と呼ばれる)。実際のファイルに変換する/system/compute ビルトクと関数 を呼び出します。重要な点は、results (output) 式の出力は、ファイルではなく、ストリームである点です。(このサンプル同様)マピングが複数の出力を返す場合、マピング出力は、ストリームのシーケンスを表示しています。この理由のため、 FlowForce 式 関数 (as-file) が反復される)現在のストリームを実際のファイルに変換するために使用されます。
- 2. 2番目のステップは前にStyleVision 変換によりデプロイされた反復される各ストリームを呼び出します。具体的には、反復ごと に、StyleVision Server が呼び出され、XML ファイルが入力として与えられ、PDF ファイルは、出力として返されます。 FlowForce 式 {filename(file)}.pdf は、実際のPDF ファイル名をディスク上に生成します。このオペレーションは 「作業ディレクトリ」パラメーターにお指定された作業ディレクトリ内で実行されます。
- メモ このサンプルでは、同じ作業ディレケトリは、マピングを実行し、Style Vision 変換を実行する両方のステップで使用されています。一部の場合、ファイル名の競合ませばショブの実行エラーを回避するために個別に作業ディレケトルを指定する必要があります。

ジョブを実行する方法

ジョブのナックに入力デーダを準備するコはNanonull-Branch.xmlとNanonull-HQ.xmlを <Documents>\Altova\MapForce2021\MapForceExamples から作業ディレクトリ(C: \FlowForceExamples\GeneratePdfs) にことーします。このようして、ジョブが実行される際にデータか読みとるオュウロごうプの最初のステップはファイルの入力を取得します。

ジョブを実行するコよ以下の内のつを行ってくたさい

- ホームに移動し全てのアクティブなトリガーとサービスを表示するをクルクします。次に1情報列内に表示されているジョブの URLをクルクします。
- <u>http://127.0.0.1:4646/service/GeneratePdfsService</u>をブラウザーのアドレスバーに入力します。このURL は FlowForce Server サービスがデンオルトのオストアドレスとポート名でリンスンする場合のみ機能します。 構成ページ内で他のホ ストとポート設定を定義した場合、アドレスを必要に応じて変更してください。

- FlowForce Server の任意のホスト名 フィールドをセトア・プルら設定するとHTTP を介してこのジョブを使用可
 - 能にする チェックボックスの横の 🕑 ボタンをクリックして web サービスの呼び出しを直接ジョブ構成ページから実行することができます。それ以外の場合、このボタンは表示されません。

Webサービスにアクセスする際に資格情報をプロンプトされた場合、FlowForce Server コングオンするために使用する同じ資格情報を提供してくたさい。

FlowForce Server ユーザーにHTTP 認証のナダの資格情報を提供することはテスト目的のナダです。生産のナダに新規の FlowForce ユーザーを作成することが奨励されており、ジョブが存在するコンテナー上でこのユーザーにService - Use パーミッション を与え、対応するユーザーアカウストを使用して W eb サービスにアクセスします。HTTP 認証を無効化するには W eb サービスを公 開し、Service - Use パーミッションをユーザーAnonymous に与えます。パーミッションのしくみを参照してください。

ジョブニ成功すると作業ディレオリC:\FlowForceExamples\GeneratePdfs内にジョブこと生成されたPDFファイルが表示されます。PDFファイルをブラウザーに出力することができなす。なブラウザーゴジョブを出力することができません」が実行が成功した場合で も表示されます。した場合、ブラウザーは、「サービスの実行に失敗しました」メッセージを表示します。この場合ジョブのエラーログを参照して ください、問題をトラブルシューティングする場合、このページの上にコストされている必要条件全てをもう一度検証する必要があるかもしれません。

15 用語

用語のセクションはFlowForce Server に関する用語のリストを含みます。

15.1 C

キャッシュコンシューマー

使用可能な場合キャッシュからデータを取得するショズ つまり、キャッシュを「消費する」 ジョブ)。キャッシュの結果が使用できない場合、コンシューマーはションを実行し、実際のショブの結果を取得します。

キャッシュプロデューサー

データを使用してキャッシュを作成するジョブ(つまし、キャッシュを「作成する」ジョブ)。

選択ステプ

型「選択ステップ」のステップにより、どの条件下で他のショブステップが実行されるかを定義することができます。選択ステップはお以下の構造 かあります:

When {some expression} Execute (some step) Otherwise Execute (some other step)

各「選択ステップ」の下では、他の選択ステップサン条件)をネストすることができます。例

When {expression} When {expression} Execute (step) Otherwise Execute (step) Otherwise Execute (step)

定義することのできる条件ステップの数量に制限はありません。

When / Otherwise ペアの中では、FlowForce Server は、true てある条件のみを実行します。他の条件は無視されます。

クラスター

クラスターはインスタンスか使用できない場合、ジョブを並列して実行、ませよ、ジョブを再配布するために通信する Flow Force Server の 複数のサービスインスタンスを表します。クラスターはつの「マスター」と複数の「ワーカー」によい構成されています。

コンテナー

名前か意味するように、コンテナーは、シケージされたデータです。FlowForce Server では、コンテナーは、オペレーシングシステム上のフォ ルダーと大まかは比較することができます。コンテナーは以下を含むことができます:ジョブ、資格情報、関数と他のコンテナー。コンテナーで、パ <u>ーミッション</u>を設定することに以、コンテナー内のデータを閲覧およびデータにアクセスする権利を管理することができます。データをコンテナー別 に整理し、対応する、トミッションを各コンテナーのためにセットアップすることは、よいセキュリティ対策*と*なるでしょう。

資格情報

資格情報オブジェケトは、認証に関する情報を保管します。これはFlowForce Server ジョブが作動する箇所でのオペレーティングシステム上のユーザーアカウトに関連するユーザー名と「スワードの組み合わせですがHTTP おけはFTP 資格情報 おけはOAuth セキュリティの詳細のセナ であることもできます。

15.2 E

エラー/成功の処理ステップ

ジョブのステップに失敗すると、FlowForce Server は、ジョブを中断します。型「エラー/成功の処理ステップ」のステップを使用して、ジョブを完全に完了する前に、クリーンアップアグションを実行します(以下を参照: ステップのエラーの処理)。

実行の結果

FlowForce Server 内では、ステップの実行はステップの実行後に返される内容を定義します(例えば、ファイル、おさま、テキスト)。ジョ ブと作業する際、ステップの実行の結果を特定のデータ型(文字列まさはブール値)になるように、おさな破棄されるように、明示的に宣言する ことができます。通常、結果を他のジョブで使用する場合、おさま、結果をキャッシュする場合、実行の結果を宣言する必要があります。

実行ステップ

型「実行ステップ」のステップにより特定のFlowForce 関数を実行することができます。FlowForce Server により提供される MapForce マメピングをデプロイする、おけま変換を行う、他のショブのステップを実行する<u>ビルトイン関数</u>を含む関数を使用することができます。

15.3 F

For-Each ステップ

型「For-Each ステップ」のステップにより、シーケンス中を反復し、実行ステップを無制限に繰り返すことができます(例えば、ディレクトリ内のファイルのリスト)。「for-each」ステップの構成は以下の通りです:

For each item in sequence {sequence expression}
 Execute (step)

FlowForce はシーケンス式の全てのアイテムをループして完了するまでステップを実行します。

関数

ジョブのエレテキストでは、関数は、FlowForce Server によい理解される、ターゲットファイルシステム上のオペレーションを実行するすめの 命令です。関数は以下であることができます:

- ビルインFlowForce 関数(以下を参照 ビルイン関数)
- StyleVision 変換
- MapForce マピング
- ジョブの実行ステップ

関数の多くは入力、デメーターを有します。必須インプト、デメーターは、ステップを成功裏に実行するため呼び出し側により与えられる必要 かあります。

15.4 I

入力パラメーター

FlowForce Server ジョブのエンテキストで、入力、ラメーターは、どのジョブが特定の方法で実行されるかを表したジョブに与えられた任意の情報を意味します。ジョブの入力、ラメーターは、関数引数にプログラム言語の観点から類似しています。入力、ラメーターは夏数引数にプログラム言語の観点から類似しています。入力、ラメーターは夏種の型(例えば、ファイルおけまディレクトリルファレンス、テキスト、番号、ブール値、その他)であることができます。一部のジョブの型に対しては、FlowForce Server は、入力、デメーターを自動的に作成します。

15.5 J

ジョブ

ジョブとはFlowForce Server 内のコア概念です。この概念サーイ、一上で実行されるはタスクまけはタスクのシーケンスを表します。ジョブは、 電子メールの送信などにつのステップから構成されるジョブであることができます。しかしながら、複数のアケンコンを実行し結果(例えば、ファイ ルを他のジョブの、ウメーターとして、やするジョブを作成することもできます。ジョブは入力、ウメーター、ステップ、トリガーおよび他の設定から構成されます。

ジョブインスタンス

ジョブイレスタレスはジョブとは異なります。FlowForce ジョブをジョブ構成ページから構成すると、ジョブ構成を作成することができます。ジョブ のナダの定義済みのトリガーの必要条件が適用されると、ジョブの実行か開始されます。ジョブイレスタレスはジョブと関連する実行キューにより 定義されているとおりクラスター内で再配布されます。ジョブイレスタレスは常に単一のクラスターメンバー上で全体で作動します。

15.6 M

マスター

ジョブニとトリガーされた条件を継続的に評価し、FlowForceサービスインターフェイスを提供する「マスター」はFlowForce Server イン スタンスです。マスターは同じクラスター内のフーカーマンンを考慮し、ジョブインスタンスを処理する、更に、おけよ、代わりに)、ジョブインスタンス を割り当てるように構成されている可能性があります。 用語

15.7 P

パマードポルシー

FlowForce Servers は、管理者が複雑なユーザー ペワードを管理する ペワードポリシーを使用します。 パマフードポリシーとは、有効になるためにコーザー ペワードが必要とする最低条件です (例えば、最低 N 文字のなど)。

パーミッション

パーミッションは、ユーザーのエンテナーへのアクセスを管理します。特権と同様、パーミッションは、ユーザーとロールに与えられることができます。ですから、ユーザーがロールの一部である場合、ロールに与えられたパーミッションは、ユーザーイミも自動的に適用されます。

デフォルトで、コンテナーに設定されたパトミッションは、親コンテナーから継承されます。例えば、コンテナーA には、子コンテナーB かあると します。コンテナーA にアクセスするパトミッションを持つユーザーは、コンテナーB にもアクセスすることのできるデフォルトのパトミッションを持つ ことっています。しかしなから、管理者は、コンテナーの階層の全てのレベルで、全てのユーザーおよびロールのパトミッションを再定義することが できます。

特権

特権は、FlowForce Server 内でユーザーが行うことのできる内容を定義します、例えば、自身のパワフト、設定、ユーザーとロールの読み 取り、ジョブの停止、など)。特権は、パーミッションはエレテナーへのユーザーアクセスを管理し、特権はグロー・シルこFlowForce Server 全体に対して効果がある点で<u>パーミッション</u>と異なります。次の簡単なルールは、特権とパーミッションの違いを理解する助けてよります。特権 はグロー・シルであり、パーミッションはローカルです。

パーミッションは類似して、特権は個々のユーザーとユールに割り当てることができます。ですから、ユーザーがFlowForce Server にエグオンすると、有効な特権はお以下により決定されます。

- a)直接割り当てられた特権
- b)ユーザがシンドーであるロールに割り当てられた特権

15.8 Q

+1-

実行キューはジョブの「プロセッサー」です。これはジョブインスタンスかどのようご実行されるかを管理します。実行するためはは、各ジョブはター ゲト実行キュー(書い当てられます。キューは(キュー(書い当てられているすべてのジョブの)ジョブインスタンスが一度に実行される数、実 行間の遅延を管理します。デフォルトで、キュー設定はジョブニメしてローカルですが、キューを複数のジョブによりシェアされるスタンドアロンオブ ジェケトとして定義することもできます。キューはジョブニメしてローカルであること、おけよ、複数のジョブで共有されることができます。複数のジョ ブが同じキュー(こ書い当てられている場合、実行のためにそのキューは共有されています。

更に、適切な(特権とは異なるコンテナート・ミッションを有する場合のみキューを確認、おけよ、キュービジョブを割り当てすることができま す。具体的には、キューを作成するナッシュン・ザーが「実行キューを定義する「特権を持つ必要があります、特権のしくみを参照してくたさい。 更に、(特権とは異なる)適切ないトミッションがある場合のみユーザーはキューを確認、おけよキュービジョブを割り当てることができます。 パー ミッションのしくみを参照してくたさい。これは、キュービジョブを割り当てできることを意味します。キューへのアクセスを制限するけは、キューか定 義されている個所にコンテナーをナビゲートし、コンテナーの、トミッションをロールauthenticated のナッジゴ「キュー-アクセス無し」に変更 します。次に、パーミッション「キューー使用」を特定のロール、おけよ、必要なユーザーに割り当てます。詳細に関しては、かいしに、コンテナ ーへのアクセスを制限するを参照してくたさい。

RaptorXML Server

(略して RaptorXML と呼ばれる) Altova RaptorXML Server は パラレルコンピューティング環境と最新の標準のさめに最適化された Altova の第3世代の、高速 XML プロセッサーです。高水準なクロスプラナトフォームの機能性のためにデザインされたエンジンは、XML の 高速な処理を行うために、現在のマルチコアコンピュータの利点を活用しています。RaptorXML には2つのエディションが存在します:(i) RaptorXML Server と(ii) RaptorXML+XBRL Server。RaptorXML+XBRL Server エディションは、XML に加え、 XBRL (eXtensible Business Reporting Language) ドキュメトの検証と処理へのサポートを含んでいます。

RaptorXML+XBRL Server

RaptorXML+XBRL Server は、検証へのサポートを含み、XML に加え XBRL (eXtensible Business Reporting Language)、ドキュメートを処理する RaptorXML Server の特別なエディションです。

ロール

ロールはビジネスニーズをベースしたセキュノティの強化を手助けする特権のセルです。標準的なロールをベースしたセキュノティイコよ 最低2つのロールが含まれます:管理者と一般ユーザー。各ロールは、ロールに与えられた特権により定義されています。例えば、管理者は、 自身のおよび他のユーザーのペワードを変更することができますが、一般ユーザーは自身のペワードのみしか変更することができません。ユー ザーイコロールを割り当て、必要に応じて、ユーザーのロールを取り消すことができます。

15.10 S

ステップ

ステップは、FlowForce Server ジョブが実際に行う内容を定義します(例えば、ファイルの判除、MapForce マピングの実行、おさま 電子メールの送信など)。最も簡単な形式としては、ステップは失敗おされ成功した成果を持つオペレーションで、実行されるために<u>関数</u>を 必要とします(以下を参照:<u>実行ステップ</u>)。しかしながら、ループとして他のステップを条件付きで実行する方法を与えるステップも存在しま す(以下を参照:<u>選択ステップ</u>、エラー/成功の処理ステップ および<u>For Each ステップ</u>)。同じジョブ内で必要な数だけステップを作成 し、実行される順序を設定することができます。

15.11 T

トリガー

ジョブを作成する際、トリガーとしてジョブをトリガーする条件(おけは基準)を指定する必要があります。 FlowForce Server は、定義されたトリガーを監視し、トリガーの条件が満たされるとショブを実行します。

15.12 U

ユーザー

ユーザーとはFlowForce Server コングオムジョンを構成、おとはMapForce おとはStyleVision 変換をデプロイレ、FlowForce Server を管理する個人を指します。ユーザーか使用することのできるアグションは以下には影響されません

- a)割り当てられたパ<u>トミッション</u>おけま<u>特権</u> b)ユーザーがメンバーであるロールに割り当てられている<u>パーミッション</u>と特権

15.13 W

Web 管理インターフェイス

FlowForce Server W eb 管理インターフェイスは、サーバーを管理し、ジョブを構成する FlowForce Server のフロントエイです。 W eb 管理インターフェイスにフトレスとポートの構成のW eb のブラウザーからアクセスすることができます。

Worker

ローカルジョブを実行する代わりにマスターインスタンスと通信するように構成されている Flow Force Server インスタンスです。ワーカーはマスターFlow Force Server か割り当てているジョブのみを実行することができます。

作業ディレクトリ

作業ディレクトリとは、ジョブが入力ファイルを展開、おけよ、出力ファイルを保存する場所が必要な場合の、実行ステップによい必要とされるパ ラメーターです。FlowForce Server は、おこ、作業ディレクトリをステップの実行中に発生する相対パンを解決するために、使用します。 作業ディレクトリを与えるようご問われた場合、FlowForce Server が実行されているオペレーティングシステム上の有効なパンを与えてくださ、 、ステップを作成中に、作業ディレクトリか与えられない場合、FlowForce Server は一時的なディレクトリを使用します。

インデックス

A

Active Directory. FlowForce Server との統合, 83 Altova LicenseServer. 新しいマシンに移行,47 AS2. Altova 製品との統合, 287 FlowForce Server と使用, 282 データの受信,308 データの送信,304 パートナーの構成,297 メッセージ交換,313 暗号化, 282, 292 暗号化の解除,292 暗号化の設定,297 概念, 292 概要, 282 署名, 282, 292 署名の設定,297 証明書の構成,293 制限,282 AS2 サービス, パーミッションの処理,308 リクエストの処理.308 公開アクセスのための構成,308 作成, 308

С

Container, definition, 563 Credentials, definition, 563

Ε

Execution Step, definition, 572

F

FlowForce Server. Web 管理インターフェイス, 24 アーキテクチャ、22 アプリケーションデータフォルダー,94 コマンドライン インターフェイス, 334 システム必要条件,28 ジョブの基本概念, 19 セットアップページ,50 ネットワークアドレスとポートの設定,54,57 のローカライズ,96 の言語の変更.351 の構成.49 ビルトイン関数,356 ローカライズ、343 ログオン,26 基本セキュリティ概念, 21 最新のバージョンに移行する,348 新しいマシンに移行,47 新規機能, 15 用語,562 FlowForce Server Function, definition, 565 FlowForce Web Server, ネットワークアドレスとポートの設定, 54, 57 For-each ステップ, 定義,565 FTP. 資格情報, 142, 146

Η

HTTP, リクエストボディのリミットを増やす,57 HTTP トリガー, の概要,179 HTTS 接続, FlowForce が次を受け入れるように構成する,54,57
I

INI ファイル, の構成, 57 Input Parameters, definition, 566

J

Job, definition, 567

LDAP, FlowForce Server との統合との統合, 83 Linux, でサービスを開始する, 91 でサービスを停止する, 91 信頼できるサーバー証明書, 68

Μ

Mac,
サービスの停止,92
でサービスを開始する,92
信頼できるサーバー証明書,68
MapForce Server,
ジョブからの呼び出し,550
新しいマシンに移行,47
統合,233,243
MDN,
AS2 プロセス内,292
MobileTogether Server,
新しいマシンに移行,47

0

OAuth 2.0,

(C) 2015-2021 Altova GmbH

資格情報, 142, 145, 146 認証, 246

Ρ

PDF ファイル, 生成, 550 Permission, definition, 569 Privilege, definition, 569

R

RaptorXML Server, 定義, 571 RaptorXML 関数, とジョブを作成する, 539, 541 RaptorXML(XBRL) Server, 新しいマシンに移行, 47 RaptorXML+XBRL Server, 定義, 571

S

SSL, 接続の暗号化, 54, 57
SSL 暗号化,
FlowForce Web サーバーのために を有効化する方法, 76
FlowForce サーバーのために を有効化する方法, 78
秘密キーの暗号化, 75
秘密キーの必要条件, 75
Step,
definition, 572
Style Vision Server,
ジョブからの呼び出し, 550
新しいマシンに移行, 47
統合, 233, 243

Т

Trigger, Definition, 573 triggerfile パラメーター, 目的, 150

U

UNC, syntax in paths, 191 パス内の構文, 174

W

Web サービス、 としてジョブを公開する、181、515 認証、181
Web 管理インターフェイス、 定義、575
Windows、 でサービスを開始する、93 ネットワークパスへの参照、356 以下でサービスを停止する、93 信頼できるサーバー証明書、68
Windows ドメインユーザー、 FlowForce Server にインポートする、114

Ζ

```
アイテム,
FlowForce データ型として, 230
アプリケーションプール, 279
インストール,
インストールフォルダー (Linux), 33
インストールフォルダー (Mac), 36
インストールフォルダー (Windows), 46
システム必要条件, 28
現在インストールされている製品を確認する (Linux), 33
エラー/成功の処理ステップ.
```

定義,564 エラーの処理, ジョブ内, 158, 162 とジョブを作成する,509 エラーログ、 とジョブを作成する,541 キー/値ペア. XSLT シートにパスする、545 キャッシュ コンシューマー, 定義,563 キャッシュプロデューサー, 定義,563 グローバルリソース. FlowForce Server 内での使用, 267 コマンドラインインターフェイス, ヘルプの取得,338 コンシューマー, 定義,563 コンテナー. /public へのアクセスを制限する, 136 のパーミッションを閲覧する, 133 の概要,128 パーミッションの設定, 135 移動,130 作成,130 名前の変更,130 ジョブ. MapForce マッピングから作成する,493 StyleVision 変換から作成, 531 キュー設定の定義,188 コピー, 149 ステータスのレファレンス, 211 のログを確認する,215 の結果のキャッシュ.527 の結果をキャッシュ,185 の戻り型を宣言する.170 ファイルからインポート, 193 ファイルヘエクスポート, 193 作成,148 資格情報, 142, 146 実行のモニタリング,207,210 他の FlowForce Server インスタンスにエクスポート, 193 停止, 214 複製.149 ステップ. ジョブ内での使用の例,500 作成,152 内でのエラー処理,158

タイマートリガー, の概要,176 ツールファイル, オプション, 279 ディレクトリ ポーリング ジョブ. の例,503 データ型. FlowForce 内. 230 デフォルトのタイムゾーン, の構成,81 トリガー, 作成,175 削除,175 パーミッション. のリスト, 125 の概要,125 パスワードポリシー, の概要,139 作成,139 定義,569 ファイルシステムのトリガー, の概要,177 ホスト名. FlowForce Server のための設定, 54 FlowForce Web Server のための設定, 54 ユーザー, ビルドインのロール, 115 作成,113 名前の変更,116 リソース, 作成,267 編集,267 ルートユーザー, のパスワードのリセット, 118 ロール. ビルドインのロール,115 ユーザーに割り当て,117 ロールの割り当て,117 作成,114 定義,571 名前の変更,116 ログ. サイズの削減.86 のジョブインスタンス, 217 設定.86 暗号化, AS2 プロセス内, 292 延期されたステップ,

サンプル、165 の概要,165 環境変数. 設定,280 関数. 定義,565 結果.230 FlowForce データ型として, 230 再試行カウントの設定. ジョブ構成オプションとして,160 作業ディレクトリ, 使用法, 243 定義,575 資格情報. 追加, 144 式.419 実行の結果. 定義,564 署名. AS2 プロセス内, 292 新しいマシンに移行, Altova LicenseServer, 47 FlowForce Server, 47 MapForce Server, 47 MobileTogether Server, 47 RaptorXML(XBRL) Server, 47 StyleVision Server, 47 選択ステップ, 定義,563 電子メールパラメーター, の構成,82 電子証明書. Linux 上で信頼できる, 68 Mac 上で信頼する.68 Windows 上での管理,72 Windows 上で信頼できる, 68 特権. に関するレポートを閲覧,122 のリスト, 119 の概要,119 継承,119 入力パラメーター, ビルトイン、150 削除,150 追加.150 認証, HTTP, 26, 181 Windows ドメイン, 26, 181

(C) 2015–2021 Altova GmbH