

Tutorial



Copyright © 1998–2009, Altova GmbH. All rights reserved. Use of this software is governed by and subject to an Altova software license agreement. XMLSpy, MapForce, StyleVision, SemanticWorks, SchemaAgent, UModel, DatabaseSpy, DiffDog, Authentic, AltovaXML, MissionKit, and ALTOVA as well as their logos are trademarks and/or registered trademarks of Altova GmbH.

XML, XSL, XHTML, and W3C are trademarks (registered in numerous countries) of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions, MIT, INRIA, and Keio. UNICODE and the Unicode Logo are trademarks of Unicode Inc. This software contains 3rd party copyrighted software or material that is protected by copyright and subject to other terms and conditions as detailed on the Altova website at http://www.altova.com/legal_3rdparty.html

ALTOVA®

Altova XMLSpy 2010 Tutorial

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.

Published: 2009

© 2009 Altova GmbH

Inhaltsverzeichnis

1	XMLSpy-Benutzeroberfläche	2
2	XML-Schemas	3
3	XML Dokumente	5
3.1	Erstellen eines neuen XML-Dokuments	6
3.2	Festlegen des Typs eines Elements	8
3.3	Eingabe von Daten in der Textansicht	10
3.4	Validierung des Dokuments	14
3.5	Hinzufügen von Elementen und Attributen	16
4	XSLT-Transformationen	17
4.1	Zuweisen einer XSLT-Datei	18
4.2	Transformieren der XML-Datei	19
4.3	Ändern der XSL-Datei	20
5	Projektverwaltung	22
5.1	Vorteile von Projekten	23
5.2	Erstellen eines Projekts	24
6	Fertig!	26
	Index	27

XMLSpy Tutorial

Mit der Ihnen vorliegenden Anleitung erhalten Sie einen umfassenden Einblick in XML. Sie finden darin Anleitungen zu den wichtigsten Aufgaben im Zusammenhang mit XML. Schritt für Schritt werden Sie mit einigen der wichtigsten Funktionen von XMLSpy vertraut.

Das Tutorial besteht aus den folgenden Teilen:

- Erstellen eines XML-Schemas. Enthält eine Einführung in die Arbeit mit XML-Schemas und den verschiedenen Ansichten, die in XMLSpy zum Anzeigen und Bearbeiten von XML-Schemas zur Verfügung stehen.
- [Erstellen eines XML-Dokuments](#). Sie lernen, wie man einem XML-Dokument ein Schema zuweist, wie man ein XML-Dokument in der Grid- und der Text-Ansicht bearbeitet und XML-Dokumente mit Hilfe des integrierten Validators von XMLSpy validiert.
- [Transformieren einer XML-Datei mittels XSLT-Stylesheets](#). Dabei müssen Sie eine XSLT-Datei zuweisen und die Transformation mit Hilfe des in XMLSpy integrierten XSLT-Prozessors durchführen.
- [Arbeiten mit XMLSpy Projekten](#), mit denen Sie Ihre XML-Dokumente ganz einfach organisieren können.

Installation und Konfiguration

In dieser Anleitung gehen wir davon aus, dass Sie XMLSpy erfolgreich auf Ihrem PC installiert haben, dass Sie einen kostenlosen Evaluierungs-Key erhalten haben oder dass Sie sich als User bereits registriert haben. Die Evaluierungsversion von XMLSpy ist voll funktionsfähig. Allerdings ist diese auf dreißig Tage begrenzt. Sie können danach eine reguläre Lizenz über unseren sicheren Webserver oder über einen Vertriebspartner erwerben.

Anleitung: Beispieldateien

Beispieldateien, die in der vorliegenden Anleitung angesprochen werden, finden Sie im Applikationsordner unter:

```
C:\Documents and Settings\\My Documents\Altova\XMLSpy\  
Examples\Tutorial
```

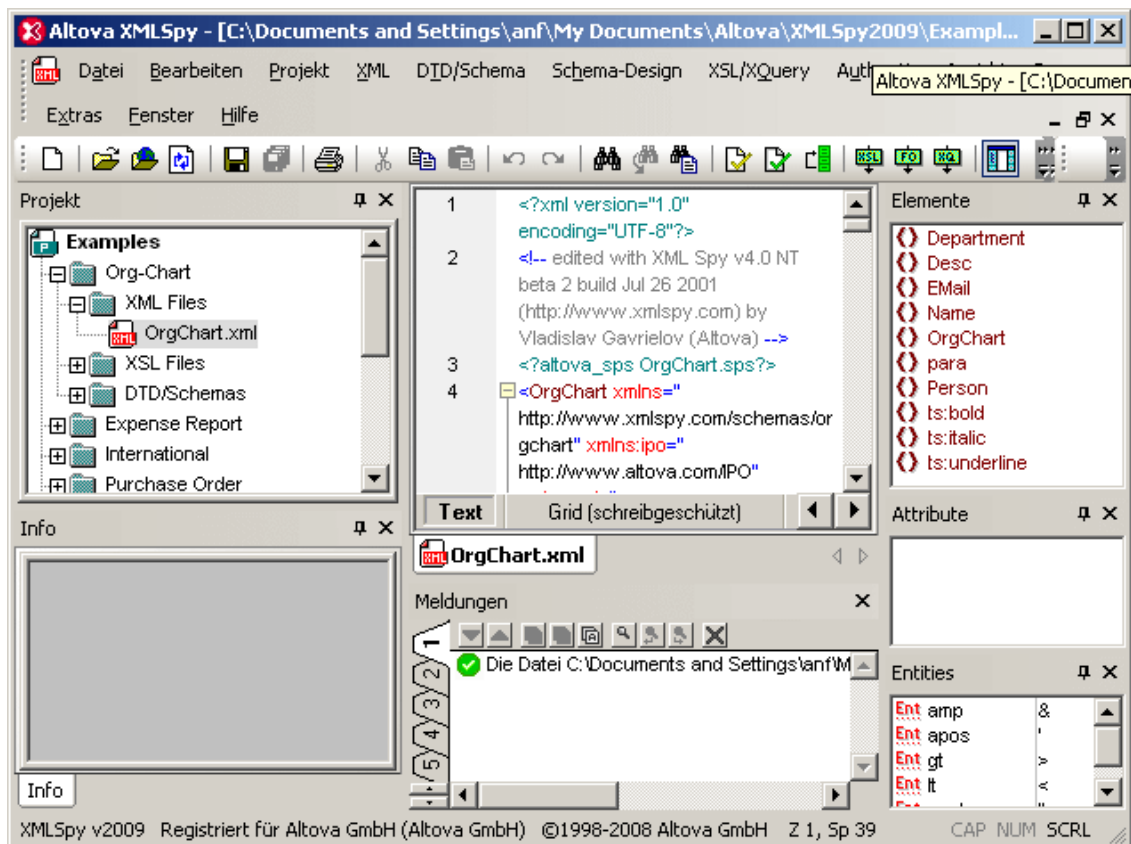
Im Verzeichnis `Examples` finden Sie viele XML-Dateien, mit denen Sie experimentieren können. Im Verzeichnis `Tutorial` sind alle Dateien abgelegt, die in dieser Anleitung verwendet werden.

Das Verzeichnis `Template` (**Vorlagen**) im Applikationsordner (normalerweise in `C:\Program Files\Altova`) enthält sämtliche XML-Vorlagen, die verwendet werden, wenn Sie die Menüoption **Datei | Neu** aufrufen. Diese Dateien enthalten alle nötigen Daten (Namespace und XML-Deklarationen) die Sie benötigen, um mit dem XML-Dokument arbeiten zu können.

1 XMLSpy-Benutzeroberfläche

Die Oberfläche von XMLSpy ist vertikal in drei Bereiche gegliedert. Im mittleren Bereich stehen verschiedene Ansichten für Ihr Dokument bereit. Die beiden seitlichen Bereiche enthalten Informationen, Editierhilfen und Funktionen zur Dateiverwaltung.

- Das **Projekt-** und das **Info-Fenster** befinden sich auf der linken Seite.
- Der mittlere Teil besteht aus dem **Hauptfenster**. Hier bearbeiten Sie Ihr XML-Dokument. Sie können zwischen verschiedenen Ansichten wählen: Textansicht, Grid-Ansicht, Schema-Ansicht, WSDL-Ansicht, Authentic-Ansicht und Browser-Ansicht. In der Standard Edition sind die Grid-Ansicht und die Schema-Ansicht schreibgeschützt; In der Enterprise und der Professional Edition können diese Ansichten auch bearbeitet werden. Diese Ansichten werden in den entsprechenden Abschnitten dazu näher beschrieben.
- Auf der rechten Seite des Bildschirms befinden sich drei **Eingabehilfen-Fenster**. Hier können Sie Elemente, Attribute oder Entities einfügen oder anhängen. Die verfügbaren Einträge ändern sich je nachdem, welche Auswahl in der Grid-Ansicht vorgenommen wurde, bzw. je nach Positionierung des Cursors in der XML-Datei.



Die Einzelheiten der Benutzeroberfläche werden im Laufe des Handbuchs näher beschrieben. Beachten Sie, dass die Oberfläche sich dynamisch je nach gewählter Ansicht ändert und je nachdem, welches Dokument im Hauptfenster aktiv ist.

2 XML-Schemas

Ein XML-Schema beschreibt die Struktur eines XML-Dokuments. Ein XML-Dokument kann gegen ein XML-Schema validiert werden, um zu überprüfen, ob es mit den im Schema definierten Vorgaben übereinstimmt und somit gültig ist. Falls dies nicht der Fall ist, ist es ungültig. Mit Hilfe von Schemas kann der Verfasser eines Dokuments die Struktur und den zulässigen Inhalt eines XML-Dokuments definieren und überprüfen, ob das XML-Dokument gültig ist.

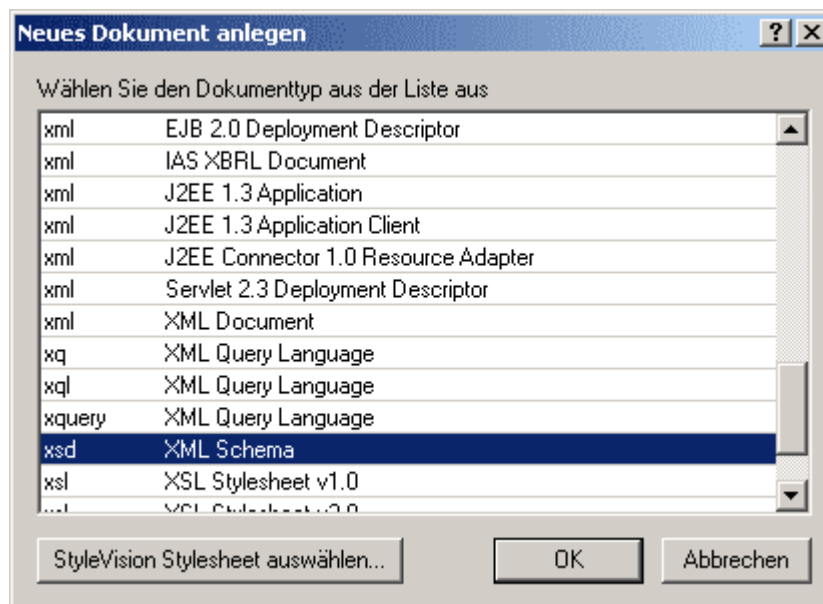
Schemabearbeitungsansichten in XMLSpy

Ein XML-Schema hat eine komplexe Struktur und Syntax. Da es sich bei einem XML-Schema selbst um ein XML-Dokument handelt, muss es den Regeln der XML-Schema-Spezifikation entsprechen. In XMLSpy können Sie in der Schema-Ansicht mit Hilfe von Drag & Drop-Techniken gültige XML-Schemas auf einfache Art erstellen. Das von Ihnen erzeugte XML-Schema-Dokument kann auch in der Text- und der Grid-Ansicht bearbeitet werden, viel einfacher lässt sich dies jedoch in der Schema-Ansicht bewerkstelligen. In der Standard Edition können XML-Schema-Dokumente in der Textansicht, der Schema-Ansicht und der Grid-Ansicht angezeigt, aber nur in der Textansicht bearbeitet werden. Die Bearbeitung in der Schema- und der Grid-Ansicht ist nur in der Enterprise und der Professional Edition möglich.

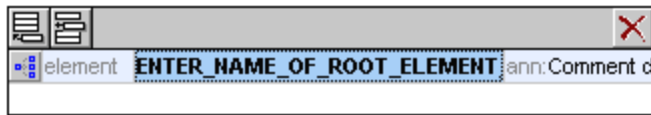
Erstellen eines neuen XML-Schema-Dokuments

Um eine neue XML-Schema-Datei in XMLSpy zu erstellen, müssen Sie zuerst XMLSpy starten und dann ein neues XML-Schema (.xsd)-Dokument erstellen. Erstellen Sie das Dokument wie folgt:

1. Wählen Sie die Menüoption **Datei | Neu**. Daraufhin wird das Dialogfeld "Neues Dokument anlegen" geöffnet.



2. Wählen Sie im Dialogfeld den Eintrag `xsd W3C Schema`. (Die Dokumentbeschreibung und die Liste im Fenster sind unter Umständen nicht mit denen in der Abbildung identisch) Bestätigen Sie mit OK. Eine leere Schema-Datei wird im Hauptfenster in der Schema-Ansicht angezeigt (*Abbildung unten*). In der Standard Edition können Sie XML-Schema-Dokumente in der Schema-Ansicht nicht bearbeiten. Sie müssen daher in die Textansicht wechseln, um das Dokument zu bearbeiten.



3. Das Schema, das Sie für den Rest dieses Tutorials verwenden werden, ist `AddressLast.xsd`. Es befindet sich im Ordner `C:\Documents and Settings\\My Documents\Altova\XMLSpy\Examples\Tutorial:` Öffnen Sie diese Datei und zeigen Sie sie in der Text- und der Schema-Ansicht an. Beachten Sie, dass Sie das Dokument in der Standard Edition in der Schema-Ansicht nicht bearbeiten können. Die Schema-Ansicht ist in der Enterprise und der Professional Edition eine Drag-and-Drop-Bearbeitungsansicht, in der Sie eine Übersicht der globalen Schemakomponenten bearbeiten und die einzelnen globalen Komponenten anschließend in einer separaten Ansicht (der Content Model-Ansicht dieser Komponente) bearbeiten können.

Die XML-Datei, die Sie im nächsten Teil des Tutorials erstellen, basiert auf dem Schema `AddressLast.xsd`, stellen Sie also sicher, dass Sie das mitinstallierte Schema `AddressLast.xsd` nicht bearbeiten.

3 XML Dokumente

In diesem Abschnitt lernen Sie:

Erstellen eines neuen XML Dokumentes in XMLSpy, Verwenden von verschiedenen Ansichten und Eingabehilfen für erleichtertes Bearbeiten und Validieren von Daten in XMLSpy.

Ziel

In diesem Abschnitt werden folgende Punkte behandelt:

- Erstellen eines neuen XML-Dokumentes basierend auf dem Schema `AddressLast.xsd`.
- Definition des Elementtyps, um für das Element bei der Validierung ein erweitertes Content Model für dieses Element bereitzustellen.
- Einfügen von Elementen und Attributen und Eingabe von Inhalt für diese in der Text-Ansicht mit Hilfe der intelligenten Eingabehilfen.
- Validieren des XML-Dokumentes.

In diesem Abschnitt verwendete Befehle



Datei | Neu erstellt eine neue XML-Datei.



Ansicht | Text wechselt in die Text-Ansicht.



F7 Zum Kontrollieren, ob das Dokument wohlgeformt ist.



F8 Validiert das XML-Dokument gegen das damit verknüpfte Schema bzw. die DTD.



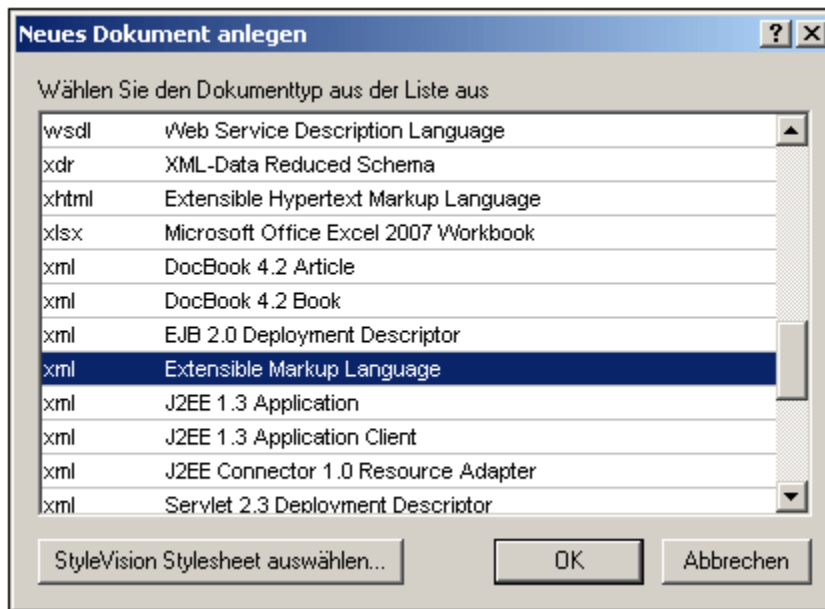
Öffnet die dazugehörige DTD oder XML-Schema-Datei.

3.1 Erstellen eines neuen XML-Dokuments

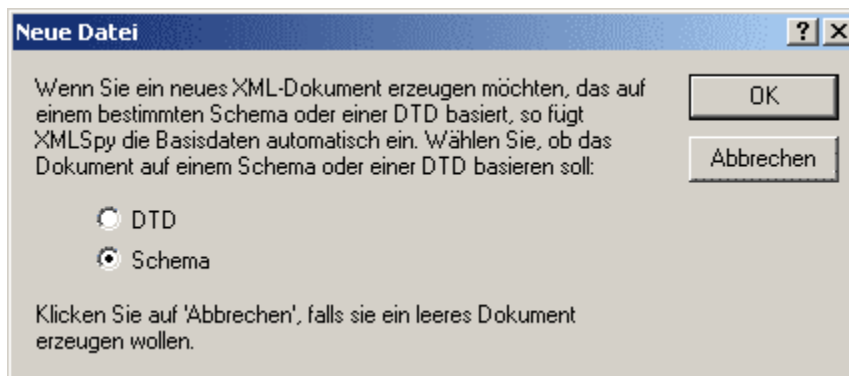
Beim Erstellen einer neuen XML-Datei in XMLSpy haben Sie die Option, dieser ein Schema (DTD oder XML-Schema) zugrunde zu legen oder auch nicht. In diesem Abschnitt werden wir eine neue Datei erstellen, die auf dem zuvor im Rahmen dieses Tutorials erzeugten Schema `AddressLast.xsd` basiert.

So erstellen Sie eine neue XML-Datei:

1. Wählen Sie die Menüoption **Datei | Neu**, um Dialogfeld "Neues Dokument anlegen" aufzurufen.



2. Wählen Sie im Dialogfeld den Eintrag `Extensible Markup Language` (oder einen generischen XML-Dokumenteintrag) und bestätigen Sie mit **OK**. Im nächsten Fenster werden Sie gefragt, ob das XML-Dokument auf einer DTD oder einem Schema basieren soll.



3. Wählen Sie das Optionsfeld "Schema" und bestätigen Sie mit OK. Sie werden nun aufgefordert, die Schema-Datei auszuwählen, auf der Ihr XML-Dokument basieren soll.
4. Wählen Sie "Durchsuchen" oder "Fenster", um die Schema-Datei zu suchen. Bei Klicken auf die Schaltfläche "Fenster" werden alle Dateien, die in XMLSpy und Projekten geöffnet sind, aufgelistet. Wählen Sie `AddressLast.xsd` (Pfad siehe

[Tutorial Einführung](#)) und bestätigen Sie Ihre Auswahl mit OK. Im Hauptfenster wird ein neues XML-Dokument geöffnet. Das Dokument enthält alle Hauptelemente, die im Schema definiert wurden. Beachten Sie die Struktur des Dokuments in der Textansicht.

5. Klicken Sie auf das Register **Grid**, um die Grid-Ansicht auszuwählen.
6. Beachten Sie in der Grid-Ansicht die Struktur des Dokuments. Wenn Sie auf ein Element klicken, wird die Auswahl auf dieses Element reduziert. Ihr Dokument sollte etwa folgendermaßen aussehen:

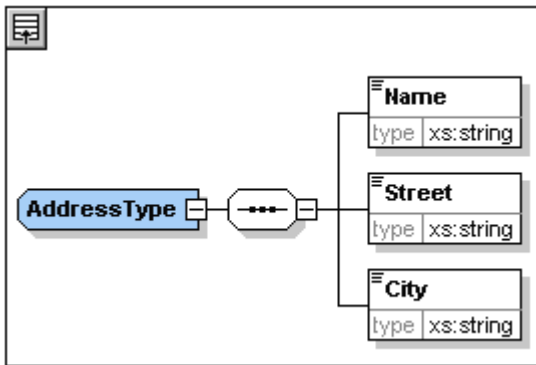
XML	
version	1.0
encoding	UTF-8
Company	
xmlns	http://my-company.com/namespace
xmlns:xsi	http://www.w3.org/2001/XMLSchema-
xsi:schemaLoca...	http://my-company.com/namespace AddressLast.xsd
Address	
Person Manager=	

7. Klicken Sie auf das Symbol neben Address, um die Child-Elemente von Address anzuzeigen. Ihr Dokument sollte aussehen wie dieses:

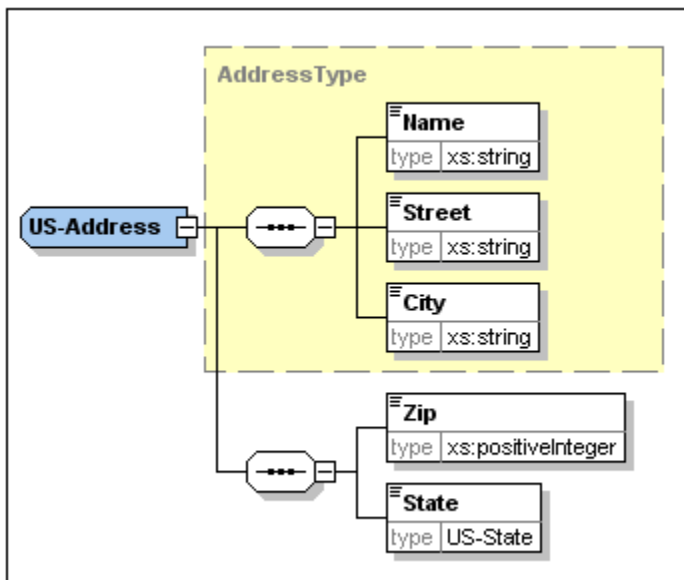
Company	
xmlns	http://my-company.com/namespace
xmlns:xsi	http://www.w3.org/2001/XMLSchema-insta
xsi:schema...	http://my-company.com/namespace AddressLast.xsd
Address	
	Name
	Street
	City
Person Manager=	

3.2 Festlegen des Typs eines Elements

Die in der Grid-Ansicht angezeigten Child-Elemente von `Address` sind die, welche für den globalen ComplexType `AddressType` (das Content Model dazu ist im XML-Schema `AddressLast.xsd` definiert, das Sie in der Abbildung unten sehen) definiert wurden.



Wir möchten jedoch einen spezifischen US- oder UK-Adresstyp verwenden anstelle des generischen Adresstyps. Sie werden sich erinnern, dass wir im Schema `AddressLast.xsd` globale ComplexTypes für `US-Address` und `UK-Address` erstellt haben, indem wir den ComplexType `AddressType` erweitert haben. Unten sehen Sie das Content Model von `US-Address`.



Um festzulegen, dass das Element `Address` im XML-Dokument einem der erweiterten `Address`-Typen entsprechen muss (`US-Address` oder `UK-Address`) und nicht dem generischen `AddressType`, müssen wir den erforderlichen erweiterten ComplexType als ein Attribut des Elements `Address` definieren.

Gehen Sie dabei folgendermaßen vor. Geben Sie im XML-Dokument am Element `Address` ein `xsi:type` Attribut mit einem Wert `US-Address` ein (Abbildung unten).

```
<Address xsi:type="US-Address">
  <Name>US dependency</Name>
  <Street>Noble Ave.</Street>
  <City>Dallas</City>
  <Zip>04812</Zip>
  <State>Texas</State>
</Address>
```

Sie können nun Daten für das Element `Address` eingeben. Geben Sie die in der Abbildung oben gezeigten Werte ein. Löschen Sie anschließend das Element `Person` (Es wird im nächsten Abschnitt des Tutorials hinzugefügt).

Bitte beachten Sie: Das vorgesetzte `xsi` ermöglicht die Verwendung von speziellen XML-Schema-spezifischen Befehlen in Ihrer XML-Dokumentinstanz. Beachten Sie, dass der Namespace für das Präfix `xsi` automatisch zum Dokumentelement hinzugefügt wurde, als Sie Ihrer XML-Datei ein Schema zugewiesen haben. Sie haben soeben einen Typ für das Element `Address` festgelegt. Nähere Informationen dazu finden Sie in der [XML Schema-Spezifikation](#).

3.3 Eingabe von Daten in der Textansicht

Die Textansicht eignet sich aufgrund ihrer intelligenten Editierfunktionen für DTD/XML-Schema ideal zum Editieren der eigentlichen Daten und des Markup-Codes von XML-Dateien.

Funktionen zum Bearbeiten der Struktur

Die Textansicht bietet eine Reihe von Funktionen zur Bearbeitung der Struktur, die das Editieren größerer Textabschnitte vereinfachen. Dazu zählen die folgenden Funktionen, die Sie durch Anklicken des entsprechenden Symbols aktivieren bzw. deaktivieren können.



Aktiviert/deaktiviert die Zeilennummerierung



Aktiviert/deaktiviert die Klappleiste



Aktiviert/deaktiviert die Lesezeichenfunktion



Fügt Lesezeichen ein und entfernt diese



Aktiviert/deaktiviert die Einrücklinien

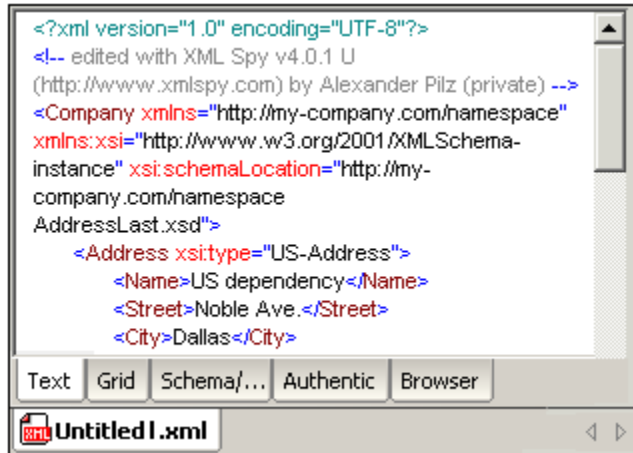
In der unten gezeigten Abbildung sehen Sie die Textansicht der XML-Datei, die sich in Bearbeitung befindet. Alle Sonderfunktionen der Textansicht sind aktiviert. Aus Gründen der Übersichtlichkeit werden diese Funktionen (Zeilennummerierung, Einrücklinien usw.) in weiterer Folge in dieser Anleitung in Textansichten nicht mehr angezeigt. Nähere Informationen zur Textansicht finden Sie im Benutzerhandbuch.

```
1      <?xml version="1.0" encoding="UTF-8"?>
2      =<Company xmlns="http://my-company.com/narr
3      C:\PROGRA~1\Altova\XMLSPY2004\Examples\
4      =<Address xsi:type="US-Address">
5          .....
6          <Name>US dependency</Name>
7          <Street>Noble Ave</Street>
8          <City>Dallas</City>
          </Address>
```

Bearbeitung in der Textansicht

In diesem Abschnitt werden wir Daten in der Textansicht eingeben und bearbeiten, um die Funktionen der Textansicht zu erläutern. Gehen Sie folgendermaßen vor:

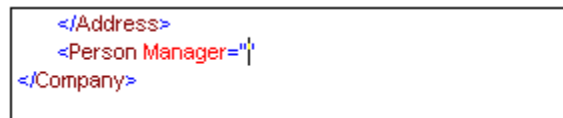
1. Wählen Sie die Menüoption **Ansicht | Text** oder klicken Sie auf das Register Text. Sie sehen nun das XML-Dokument in seiner Textform mit der Syntaxfärbung.



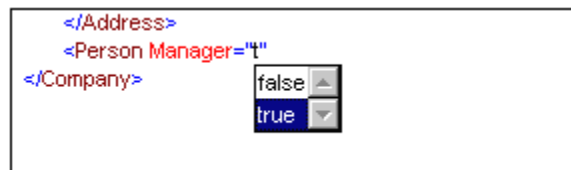
2. Setzen Sie den Cursor hinter den End-Tag des Elements "Address" und drücken Sie die Eingabetaste um eine neue Zeile einzufügen.
3. Geben Sie ein "Kleiner als..." Zeichen (<) an dieser Stelle ein. Eine Auswahlliste aller an dieser Stelle (laut Schema) zulässigen Elemente erscheint. Da an dieser Stelle nur das Element `person` zulässig ist, wird nur dieses eine Element in der Liste angezeigt.



4. Wählen Sie den Eintrag `Person` aus. Das Element `person` wird mit seinem Attribut `Manager` eingefügt, wobei sich der Cursor innerhalb des Wertefelds des Attributs `Manager` befindet.



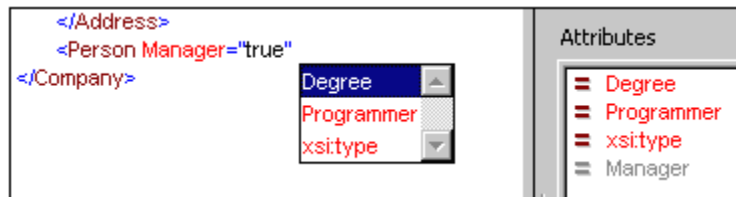
5. Wählen Sie in der Dropdown-Liste des Attributs `Manager` den Eintrag `true` aus.



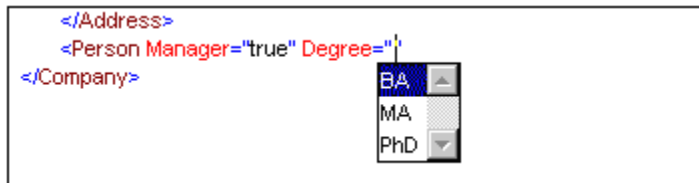
Nach Betätigen der Eingabetaste wird der Wert `true` an der Cursorposition eingefügt.

6. Setzen Sie den Cursor an das Ende der Zeile (oder betätigen Sie die "Ende"-Taste Ihrer Tastatur). Drücken Sie die Leerzeilentaste einmal. Daraufhin wird eine Dropdown-Liste geöffnet, die diesmal eine Liste von an dieser Stelle zulässigen

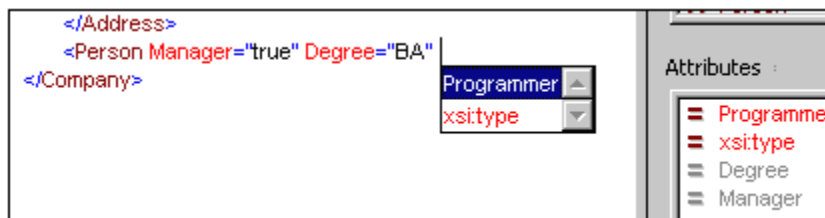
Attributen enthält. In der Attribut-Eingabehilfe sind verfügbare Attribute rot markiert. Das Attribut "Manager" ist ausgegraut, da es bereits verwendet wurde.



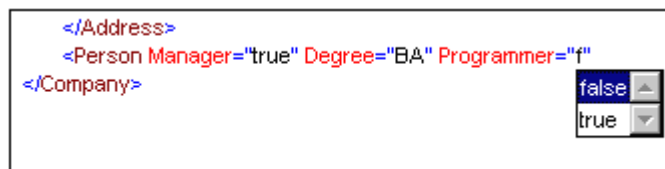
- Wählen Sie `Degree` durch Betätigen des 'Nach unten'-Pfeiles und drücken Sie die Eingabetaste. Eine weitere Liste wird geöffnet. Hier können Sie eine der vordefinierten Enumerationen auswählen (`BA`, `MA` oder `PhD`).



- Wählen Sie `BA` aus der Liste, drücken Sie die Eingabetaste und setzen Sie den Cursor durch Drücken der Ende-Taste an das Ende der Zeile. Drücken Sie die Leerzeilentaste. Nun sind `Manager` und `Degree` in der Attribut-Eingabehilfe ausgegraut.



- Wählen Sie `Programmer` aus der Liste und drücken Sie die Eingabetaste.



- Geben Sie den Buchstaben "f" ein und drücken Sie die Eingabetaste.
- Setzen Sie den Cursor ans Ende der Zeile, und geben Sie das "größer als..." Zeichen (>) ein. XMLSpy fügt automatisch alle erforderlichen Child-Elemente von `Person` ein. (Beachten Sie, dass das optionale Element `Title` nicht eingefügt wird). Jedes Element hat einen Start- und einen End-Tag aber keinen Inhalt.

```

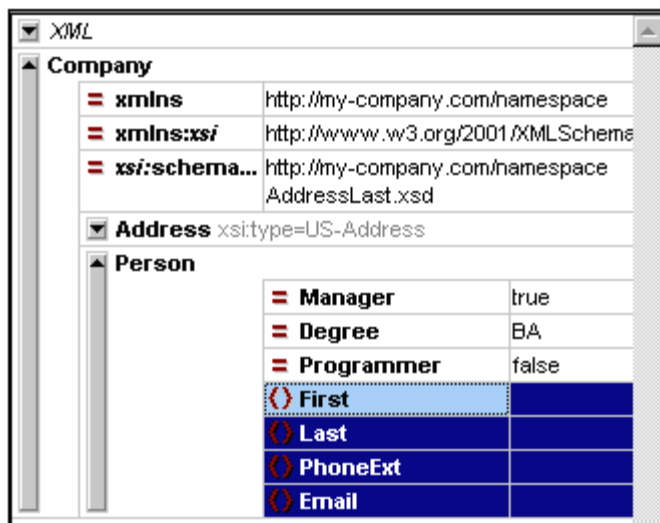
<?xml version="1.0" encoding="UTF-8"?>
<Company xmlns="http://my-company.com/namespace"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://my-company.com/namespace
AddressLast.xsd">
  <Address xsi:type="US-Address">
    <Name>US dependency</Name>
    <Street>Noble Ave</Street>
    <City>Dallas</City>
  </Address>
  <Person Manager="true" Degree="BA" Programmer="false">
    <First></First>
    <Last></Last>
    <PhoneExt></PhoneExt>
    <Email></Email>
  </Person>
</Company>

```

Sie könnten nun die Personendaten hier eingeben, doch wollen wir nun in die Grid-Ansicht wechseln, damit Sie sehen, wie einfach sich Ansichten beim Bearbeiten eines Dokuments wechseln lassen.

Wechseln zur Grid-Ansicht

Um zur Grid-Ansicht zu wechseln, wählen Sie die Menüoption **Ansicht | Grid-Ansicht** oder klicken Sie auf das Register **Grid**. Die neu hinzugefügten Child-Elemente von `person` erscheinen markiert.



Wir wollen das Dokument nun validieren und etwaige Fehler korrigieren.

3.4 Validierung des Dokuments

XMLSpy bietet zwei Funktionen zur Überprüfung des XML-Dokuments:


- Wohlgeformtheitprüfung
- Validierung

Wenn ein Fehler bei einer dieser beiden Überprüfungen gefunden wird, muss das Dokument entsprechend geändert werden.

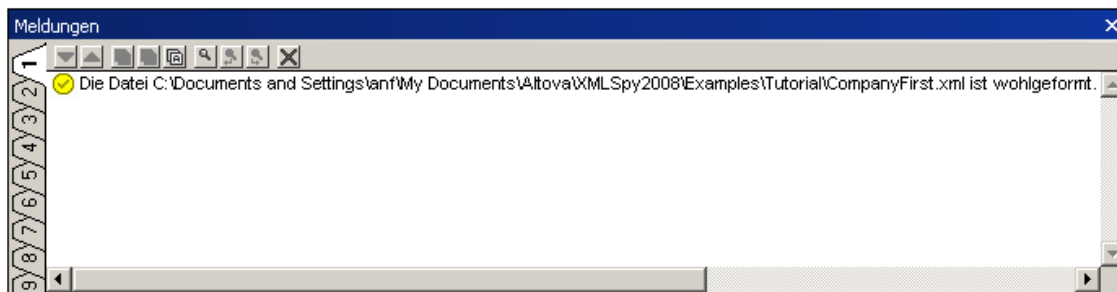
Überprüfung der Wohlgeformtheit

Ein XML-Dokument ist wohlgeformt, wenn es zu jedem Start-Tag einen End-Tag gibt, Elemente korrekt verschachtelt sind und keine Zeichen fehlen oder an der falschen Stelle stehen (z.B. eine Entity ohne das Semikolon usw.).

Sie können die Wohlgeformtheit eines Dokuments in jeder Bearbeitungsansicht überprüfen. Wählen Sie zum Prüfen der Wohlgeformtheit die Menüoption **XML | Wohlgeformtheit prüfen**

oder drücken Sie die Taste **F7** oder klicken Sie auf . Am unteren Rand des Hauptfensters erhalten Sie im Fenster "Meldungen" eine Meldung, welche anzeigt, ob das Dokument wohlgeformt ist.

Beachten Sie, dass das Validierungsfenster neun Ausgaberegister enthält. Das Ergebnis der Validierung wird immer auf dem aktuellen Register angezeigt. Sie können daher die Wohlgeformtheit für eine Schemadatei auf Register 1 überprüfen und das Ergebnis aufbewahren, indem Sie zu Register 2 wechseln, bevor Sie das nächste Schema-Dokument validieren (andernfalls wird Register 1 durch das Validierungsergebnis überschrieben.)




Bitte beachten Sie: Bei der Wohlgeformtheitsprüfung nicht geprüft wird, ob die Struktur der XML-Datei schemakonform ist. Dies wird bei der Validierung der Datei überprüft.

Überprüfung der Gültigkeit (Validierung)

Ein XML-Dokument ist gemäß einem Schema gültig, wenn es von Struktur und Inhalt mit den Schema-Spezifikationen übereinstimmt.

Um die Gültigkeit Ihres XML-Dokuments zu überprüfen, wählen Sie zuerst die Text-Ansicht aus und wählen Sie anschließend die Menüoption **XML | Validieren** oder drücken Sie die Taste **F8**

oder klicken Sie auf . Im Fenster "Meldungen" wird eine Fehlermeldung angezeigt, dass die Datei nicht gültig ist. Nach dem Element `City` in `Address` werden obligatorische Elemente erwartet. Wenn Sie Ihr Schema überprüfen, werden Sie feststellen, dass der ComplexType `US-Address` (dem aufgrund von `xsi:type` attribute dieses `Address`-Element entsprechen muss) ein Content Model hat, in dem auf das Element `City` ein Element `zip` und ein Element `State` folgen muss.

Beheben von Validierungsfehlern

Das Element, welches den Fehler verursacht hat, in diesem Fall das Element `city`, wird markiert angezeigt.

Bitte beachten Sie hierzu auch die Eingabehilfe (oben rechts), bei welcher ein Ausrufungszeichen "!" vor dem Element `zip` anzeigt, dass es sich hierbei um ein **obligatorisches** Element handelt.

Um den Validierungsfehler zu beheben, gehen Sie folgendermaßen vor:

1. Platzieren Sie den Cursor hinter das Element `city` und doppelklicken Sie in der Element-Eingabehilfe auf das Element `zip`.
2. Stellen Sie sicher, dass sich der Cursor zwischen dem Start- und dem End-Tag des Elements `zip` befindet, geben Sie eine Postleitzahl (Zip) ein - (z.B. 04812) und bestätigen Sie mit der Eingabetaste. Die Element-Eingabehilfe zeigt nun an, dass das Element `state` obligatorisch ist (ein Ausrufezeichen ist dem Element vorangestellt).
3. Platzieren Sie den Cursor hinter das Element `zip` und doppelklicken Sie in der Element-Eingabehilfe auf das Element `state`.

3.5 Hinzufügen von Elementen und Attributen

Momentan gibt es im Dokument nur ein `Person`-Element.

So fügen Sie ein neues Element `Person` hinzu:

1. Platzieren Sie den Cursor hinter das Element `Person`, das wir bereits erstellt haben.
2. Drücken Sie die Eingabetaste. Daraufhin wird eine neue Zeile erstellt, wobei der Cursor an den Beginn der neuen Zeile gesetzt wird. Beachten Sie, dass das Element `Person` nun in der Element-Eingabehilfe zur Verfügung steht.
3. Doppelklicken Sie in der Element-Eingabehilfe auf das Element `Person`. Ein neues `Person`-Element sowie alle obligatorischen Child-Elemente werden angehängt (siehe *Abbildung unten*). Beachten Sie, dass das optionale Child-Element `Title` nicht eingefügt wird.

```
<Person Manager="true" Degree="BA" Programmer="false">
  <First>Fred</First>
  <Last>Smith</Last>
  <PhoneExt>22</PhoneExt>
  <Email>Smith@work.com</Email>
</Person>
<Person Manager="">
  <First></First>
  <Last></Last>
  <PhoneExt></PhoneExt>
  <Email></Email>
</Person>
```

4. Drücken Sie **F12** um in der Grid-Ansicht zum neuen Element `Person` zu wechseln.
5. Klicken Sie auf das Attribut `Manager` im neuen Element `Person`. Beachten Sie die vorhandenen Attribute in der Attribut-Eingabehilfe. Der Eintrag `Manager` ist ausgegraut, da er bereits verwendet wurde. Werfen Sie auch einen Blick auf das Info-Fenster, in dem nun Informationen über das Attribut `Manager` angezeigt werden. Es handelt sich hierbei um ein obligatorisches Attribut, weswegen es hinzugefügt wurde. Das Attribut `Programmer` wurde nicht hinzugefügt.
6. Wechseln Sie in die Textansicht. Doppelklicken Sie in der Attribut-Eingabehilfe auf den Eintrag `Programmer`. Hierdurch wird ein leeres `Programmer`-Attribut hinter dem `Manager`-Attribut eingefügt. Das Attribut `Programmer` ist nun in der Attribut-Eingabehilfe ausgegraut.

Wählen Sie die Menüoption **Datei | Speichern unter...** und speichern Sie die Datei als `CompanyLast.xml`. (Vergessen Sie nicht, die mit XMLSpy installierte Originaldatei `CompanyLast.xml` umzubenennen, z.B. in `CompanyLast_orig.xml`).

Bitte beachten Sie: Die mitinstallierte Datei `CompanyLast.xml` befindet sich im Ordner `Tutorial`.

4 XSLT-Transformationen

Ihr Ziel in diesem Abschnitt

Erstellen einer HTML-Datei von einer XML-Datei mittels eines XSLT Stylesheet zum Transformieren der XML-Datei. Beachten Sie, dass die XML-Datei bei einer "Transformation" nicht geändert wird. Es wird nur eine neue Ausgabedatei erzeugt. Das Wort "Transformation" ist eine Konvention.

Methode:

Gehen Sie beim Transformieren folgendermaßen vor:

- Weisen Sie dem XML-Dokument eine vordefinierte XSL-Datei, nämlich `company.xsl` zu.
- Führen Sie die Transformation mit Hilfe eines der beiden in XMLSpy integrierten Altova XSLT-Prozessoren durch (*siehe Hinweis unten*).

In diesem Abschnitt verwendete Befehle

Die folgenden XMLSpy-Befehle werden in diesem Abschnitt verwendet:



XSL/XQuery | XSL zuweisen weist dem aktiven XML-Dokument eine XSL-Datei zu.



XSL/XQuery | Gehe zu XSL öffnet die vom aktiven XML-Dokument referenzierte XSL-Datei.



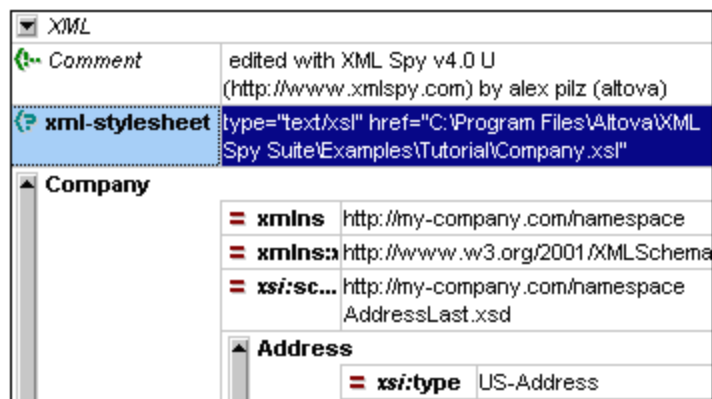
XSL/XQuery | XSL-Transformation (F10) oder die Symbolleistenschaltfläche transformiert das aktive XML-Dokument mit Hilfe des diesem zugewiesenen XSL Stylesheet. Wenn keine XSL-Datei zugewiesen wurde, werden Sie beim Aufrufen dieses Befehls aufgefordert, eine zuzuweisen.

Bitte beachten Sie: XMLSpy verfügt über zwei integrierte XSLT-Prozessoren, den Altova XSLT 1.0-Prozessor und den Altova XSLT 2.0-Prozessor. Der Altova XSLT 1.0-Prozessor dient zum Transformieren von XSLT 1.0 Stylesheets, während der Altova XSLT 2.0-Prozessor für die Verarbeitung von XSLT 2.0 Stylesheets verwendet wird. Der richtige Prozessor wird von XMLSpy automatisch auf Basis des Versionsattributs im Element `xsl:stylesheet` oder `xsl:transform` ausgewählt. In diesem Tutorial verwenden wir für die Transformation XSLT 1.0 Stylesheets. Bei Aufruf des Befehls **XSL-Transformation** wird bei diesen Stylesheets automatisch der Altova XSLT 1.0-Prozessor verwendet.

4.1 Zuweisen einer XSLT-Datei

So weisen Sie der Datei `CompanyLast.xml` eine XSLT-Datei zu:

1. Klicken Sie im Hauptfenster auf das Register `CompanyLast.xml`, um es aktiv zu setzen und wechseln Sie in die Textansicht.
2. Wählen Sie die Menüoption **XSL/XQuery | XSL zuweisen**.
3. Klicken Sie auf die Durchsuchen-Schaltfläche und wählen Sie im Ordner `Tutorial` die Datei `Company.xsl` aus. Sie können im Dialogfeld die Option "Pfad relativ zu `CompanyLast.xml` machen", aktivieren, um den Pfad zur XSL-Datei (im XML-Dokument) relativ zu machen.
4. Klicken Sie auf OK um dem XML-Dokument die XSL-Datei zuzuweisen.
5. Wechseln Sie in die Grid-Ansicht, um die Zuweisung zu sehen (*Abbildung unten*).




Im XML-Dokument wird eine `XML-stylesheet` Verarbeitungsanweisung eingefügt, die die XSL-Datei referenziert. Wenn Sie das Kontrollkästchen "Pfad relativ zu `CompanyLast.xml` machen" aktiviert haben, ist der Pfad relativ; andernfalls ist er absolut (wie in der obigen Abbildung).

4.2 Transformieren der XML-Datei

Um das XML-Dokument mit Hilfe der zugewiesenen XSL-Datei zu transformieren, gehen Sie folgendermaßen vor:

1. Stellen Sie sicher, dass die XML-Datei das aktive Dokument ist.
2. Wählen Sie die Menüoption **XSL/XQuery | XSL-Transformation (F10)** oder klicken Sie

auf das Symbol , um die Transformation mittels des in dem XML-Dokument referenzierten XSL Stylesheet zu starten. (Da die Datei `Company.xml` ein XSLT 1.0-Dokument ist, wird für die Transformation automatisch der integrierte Altova XSLT 1.0-Prozessor ausgewählt). Das Ausgabedokument wird in der Browser-Ansicht angezeigt. Es hat den Namen `XSL Output.html`. (Wenn die HTML-Ausgabedatei nicht generiert wird, stellen Sie sicher, dass die Standarddateierweiterung im Dialogfeld "Optionen" (**Extras | Optionen**) auf dem Register "XSL" auf `.html` gesetzt wurde.) Die Firmendaten werden in einem Block links oben angezeigt, während die Personendaten unterhalb davon in Tabellenform dargestellt werden.



Your Company

Name: US dependency
Street: Noble Ave
City: Dallas
State: Texas
Zip: 04812

First	Last	Ext.	E-Mail	Manager	Degree
Alfred	Aldrich	33	Aldrich@work.com	false	MA
Colin	Coletti	444	Coletti@work.com	true	Ph.D
Fred	Smith	22	Smith@work.com	true	BA

Text | Browser

CompanyLast.xml | AddressLast.xsd | XSL Output.html

Bitte beachten Sie: Falls in der Ausgabedatei nur eine Tabellenkopfzeile aber keine Tabellendaten zu sehen sind, stellen Sie bitte sicher, dass Sie den Target Namespace für Ihr Schema wie am Beginn des Tutorials unter Definieren von Namespaces beschrieben, definiert haben. Der Namespace muss in allen drei Dateien (Schema, XML und XSL) **identisch** sein.

4.3 Ändern der XSL-Datei

Sie können die Ausgabe der Datei ändern, indem Sie die zugrunde liegende XSL-Datei bearbeiten. In diesem Fall soll die Hintergrundfarbe der Tabelle von "Lime (Hellgrün)" in "Yellow (Gelb)" geändert werden.

Gehen Sie folgendermaßen vor:

1. Klicken Sie auf das Register `CompanyLast.xml` um es aktiv zu setzen und stellen Sie sicher, dass Sie sich in der Grid-Ansicht befinden.
2. Wählen Sie die Menüoption **XSL/XQuery | Gehe zu XSL**.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xmlns:my="http://my-company.com/namespace">

<xsl:template match="/">
  <html>
    <head> <title>Your company</title></head>
    <body>
      <h1><center>Your Company</center></h1>
      <xsl:apply-templates select="//my:Address"/>
      <table border="1" bgcolor="lime">
        <thead align="center">
          <td><strong>First</strong></td>
          <td><strong>Last</strong></td>
          <td><strong>Ext.</strong></td>
```

Mit diesem Befehl wird die Datei `Company.XSL` geöffnet, die in der XML-Datei referenziert wird.

3. Gehen Sie zur Zeile `<table border="1" bgcolor="lime">` und ändern Sie den Eintrag von `bgcolor="lime"` in `bgcolor="yellow"`.

```
<h1><center>Your Company</center></h1>
<xsl:apply-templates select="//my:Address"/>
<table border="1" bgcolor="yellow">
  <thead align="center">
    <td><strong>First</strong></td>
    <td><strong>Last</strong></td>
```

4. Wählen Sie die Menüoption **Datei | Speichern**, um die Änderungen an der XSL-Datei zu speichern.
5. Klicken Sie auf das Register `CompanyLast.xml` um die XML-Datei aktiv zu setzen, und wählen Sie **XSL/XQuery | XSL-Transformation** oder drücken Sie die Funktionstaste **F10**. In der Browser-Ansicht von XMLSpy erscheint eine neue Datei namens `XSL Output.html`. Die Hintergrundfarbe der Tabelle ist gelb.

Your Company

Name: US dependency
Street: Noble Ave
City: Dallas
State: Texas
Zip: 04812

First	Last	Ext.	E-Mail	Manager	Degree
Alfred	Aldrich	33	Aldrich@work.com	false	MA
Colin	Coletti	444	Coletti@work.com	true	Ph.D
Fred	Smith	22	Smith@work.com	true	BA

6. Wählen Sie die Menüoption **Datei | Speichern** und benennen Sie die Datei `Company.html`.

5 Projektverwaltung

In diesem Abschnitt werden Sie mit den Projektverwaltungsfunktionen von XMLSpy vertraut gemacht. Nachdem Sie die Vorteile der Gliederung von XML-Dateien in Projekten kennengelernt haben, werden wir die soeben erstellten Dateien in einem einfachen Projekt strukturieren.

5.1 Vorteile von Projekten

Im Folgenden sind die Vorteile der Strukturierung von XML-Dateien in Form von Projekten aufgelistet.

- Dateien und URLs können durch gemeinsame Erweiterungen oder andere Kriterien in Ordner gruppiert werden.
- Batchprozesse können bestimmten Ordnern oder dem ganzen Projekt zugewiesen werden.
- DTD / XML-Schemas können bestimmten Ordnern zugewiesen werden, um eine sofortige Validierung der darin befindlichen Dateien zu ermöglichen.
- XSL-Dateien können bestimmten Ordnern zugewiesen werden, um eine sofortige Transformation der darin befindlichen Dateien zu erlauben.
- Es können Zielordner für die XSL-Transformationsdateien definiert werden.

Alle Einstellungen können unter **Projekt | Projekteigenschaften...** definiert werden. Im nächsten Abschnitt werden Sie über das Menü "Projekt" ein Projekt erstellen.

Des Weiteren stehen die folgenden Projektfunktionen zur Verfügung:

- XML-Dateien können unter Versionskontrolle gestellt werden (**Projekt | Versionskontrolle | Zu Versionskontrolle hinzufügen...**). Weitere Informationen erhalten Sie im Abschnitt Versionskontrolle der Online-Hilfe.
- Persönliche, netzinterne und Webordner können in Projekte eingefügt werden, wodurch die Batchvalidierung ermöglicht wird.

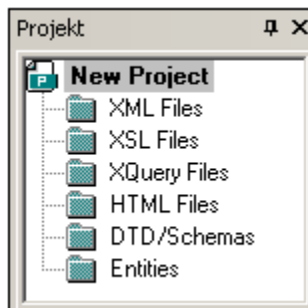
5.2 Erstellen eines Projekts

Im Hauptfenster ist nun eine Reihe von Tutorial-Dateien offen. Sie können diese Dateien in einem Tutorial-Projekt gliedern. Zuerst müssen Sie ein neues Projekt erstellen und anschließend die Tutorial-Dateien zu den jeweiligen Unterordnern des Projekts hinzufügen.

Erstellen eines einfachen Projekts

Um ein einfaches Projekt zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Menüoption **Projekt | Neu**. Im Projektfenster wird ein Projektordner mit dem Namen `New Project` erstellt. Das neue Projekt enthält leere Ordner für typische XML-Dateikategorien in einem Projekt (siehe Abbildung unten).



2. Klicken Sie auf das Register `CompanyLast.xml` um diese Datei im Hauptfenster aktiv zu setzen.
3. Wählen Sie die Menüoption **Projekt | Aktive und verwandte Dateien hinzufügen**. Es werden zwei Dateien zum Projekt hinzugefügt: `CompanyLast.xml` und `AddressLast.xsd`. Beachten Sie, dass durch Verarbeitungsanweisungen referenzierte Dateien (wie z.B. XSLT-Dateien) nicht als verwandte Dateien gelten.
4. Wählen Sie die Menüoption **Projekt | Projekt speichern** und speichern Sie das Projekt unter dem Namen `Tutorial`.

Hinzufügen von Dateien zu einem Projekt

Sie können auch andere Dateien zum Projekt hinzufügen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie auf eine beliebige offene XML-Datei (mit der Erweiterung `.xml`) mit Ausnahme von `CompanyLast.xml`, um diese XML-Datei zur aktiven zu machen. (Wenn keine andere XML-Datei geöffnet ist, öffnen Sie eine oder erstellen Sie eine neue XML-Datei).
2. Wählen Sie die Menüoption **Projekt | Aktive Datei hinzufügen**. Die XML-Datei wird anhand ihres Dateityps (`.xml`) in den Ordner XML Files hinzugefügt.
3. Fügen Sie nun auf dieselbe Art eine HTML-Datei und eine XSD-Datei (z.B. `Company.html` und `AddressFirst.xsd`) zum Projekt hinzu. Diese Dateien werden zum Ordner HTML Files bzw. DTD/Schemas hinzugefügt.
4. Speichern Sie das Projekt, entweder durch Auswahl der Menüoption **Projekt | Speichern** oder durch Auswahl einer beliebigen Datei oder eines Ordners im Projektfenster und Klicken auf das Symbol "Speichern" in der Symbolleiste (oder **Datei | Speichern**).

Bitte beachten Sie: Alternativ dazu können Sie mit der rechten Maustaste auf einen Projektordner klicken und den Befehl "Aktive Datei hinzufügen" auswählen, um die aktive Datei zu diesem bestimmten Ordner hinzuzufügen.

Andere nützliche Befehle

Hier sind einige weitere gebräuchliche Projektbefehle aufgelistet:

- Um einen neuen Ordner zu einem Projekt hinzuzufügen, wählen Sie die Menüoption **Projekt | Projektordner in Projekt einfügen** und fügen Sie den Namen des Projektordners ein.
- Um einen Ordner aus einem Projekt zu löschen, rechtsklicken Sie auf den Ordner und wählen Sie im Kontextmenü den Befehl **Löschen**.
Um eine Datei aus einem Projekt zu löschen, wählen Sie die Datei aus und drücken Sie die **Entf**-Taste.

6 **Fertig!**

Herzlichen Glückwunsch!

Wir hoffen, dass Ihnen diese Anleitung geholfen hat, um die Grundzüge von XMLSpy kennen zu lernen. Weitere Informationen finden Sie in der kontextsensitiven Online-Hilfe. Oder drucken Sie die PDF-Version des Tutorials aus, die sich im Anwendungsordner von XMLSpy befindet (tutorial.pdf).

Index

B

Beispieldateien,
Tutorial, 1

E

Eingabehilfen, 2
in Grid-Ansicht, 16

Elementtyp,
in XML-Dokument festlegen, 8

Erstellen,
Neues XML Dokument, 6

F

Fenster,
Übersicht, 2

G

Grid-Ansicht,
Attribut / Element anhängen, 16
Eingabehilfen verwenden, 16

H

Hauptfenster, 2

I

Info,
Fenster, 2

Infofenster, 2

N

Neues XML-Dokument,
erstellen, 6

P

Projekte in XMLSpy,
erstellen, 24
Vorteile, 23

Projektfenster, 2

Projektverwaltung in XMLSpy, 22

T

Textansicht,
Bearbeiten in, 10

Tutorial,
Beispieldateien, 1
Ziele, 1

type,
Erweiterung im XML-Dokument, 8

U

Übersicht, 2

V

Validieren,
XML-Dokument, 14

Vorlagen,
Ordner, 1

W

Wohlgeformtheitsprüfung,

Wohlgeformtheitsprüfung,

bei XML-Dokument, 14

X**XML,**

Dokument, 5

Neues Dokument, 6

Symbole, 5

XML Schema Tutorial, 3**XML-Dokument,**

in Textansicht bearbeiten, 10

xsi:type,

Verwendung, 8

XSLT,

in XMLSpy ändern, 20

XSL-Transformation,

siehe XSLT, 17

XSLT-Transformation,

in XMLSpy, 19

Tutorial, 17

XSLT-Datei zuweisen, 18