Altova XMLSpy 2025 Enterprise Edition

Tutorial

Altova XMLSpy 2025 Enterprise Edition 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: 2019-2025

© 2019-2025 Altova GmbH

Inhaltsverzeichnis

XMLSpy Tutorial 5

1 XMLSpy-Benutzeroberfläche 6
- 1.1 Die Ansichten 7
- 1.2 Die Fenster 8
- 1.3 Menüs und Symbolleisten 10
- 1.4 Einstellungen für die Textansicht 13

2 XML-Schemas: Grundlagen 16
- 2.1 Anlegen einer neuen XML-Schema-Datei 16
- 2.2 Definieren von Namespaces 18
- 2.3 Definieren eines Content Model 19
- 2.4 Hinzufügen von Elementen mit Drag & Drop 24
- 2.5 Konfigurieren der Content Model-Ansicht 25
- 2.6 Fertigstellen des Schemas 27

3 XML-Schemas: komplexere Vorgänge 31
- 3.1 Arbeiten mit ComplexTypes und SimpleTypes 31
- 3.2 Referenzieren von globalen Elementen 39
- 3.3 Attribute und Attribut-Enumerationen 41

4 XML-Schemas: XMLSpy-Funktionen 44
- 4.1 Schemanavigation 44
- 4.2 Schema-Dokumentation 46

5 XML Dokumente 51
- 5.1 Erstellen eines neuen XML-Dokuments 52
- 5.2 Festlegen des Typs eines Elements 53
- 5.3 Eingabe von Daten in der Grid-Ansicht 56
- 5.4 Eingabe von Daten in der Textansicht 57
- 5.5 Validierung des Dokuments 62
- 5.6 Hinzufügen von Elementen und Attributen 66
- 5.7 Bearbeiten in der Tabellenansicht 68
- 5.8 Ändern des Schemas 71

6 XSLT-Transformationen ... 73
6.1 Zuweisen einer XSLT-Datei ... 73
6.2 Transformieren der XML-Datei ... 74
6.3 Ändern der XSL-Datei ... 75
7 Projektverwaltung ... 78
7.1 Vorteile von Projekten ... 78
7.2 Erstellen eines Projekts ... 78
8 Fertig! ... 81

Benutzeroberfläche und Umgebung **82**

Index **83**

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:

- XMLSpy-Benutzeroberfläche 6. Hier wird die grafische Benutzeroberfläche (GUI) der Applikation erläutert.
- Erstellen eines XML-Schemas 16. Sie lernen, wie man in der intuitiven Schema-Ansicht von XMLSpy ein XML-Schema erstellt, wie man mit Drag-and-Drop komplexe Inhaltsmodelle erstellt und wie man die Schema-Ansicht konfiguriert.
- Erstellen von complex und simpleTypes, globalen Elementreferenzen und Attributenumerationen mit Hilfe von Funktionen der Schema-Ansicht 31.
- Sie lernen, wie man in der Schema-Ansicht in Schemas navigiert 44 und wie man Dokumentation zu Schemas generiert 44.
- Erstellen eines XML-Dokuments 51. 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 73. Dabei müssen Sie eine XSLT-Datei zuweisen und die Transformation mit Hilfe des in XMLSpy integrierten XSLT-Prozessors durchführen.
- Arbeiten mit XMLSpy Projekten 78, 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:\Dokumente und Einstellungen\<Benutzer>\Eigene Dokumente\Altova\XMLSpy2025\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:\Programme\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

In diesem Abschnitt des Tutorials starten Sie XMLSpy und erhalten eine Einführung in die Benutzeroberfläche.

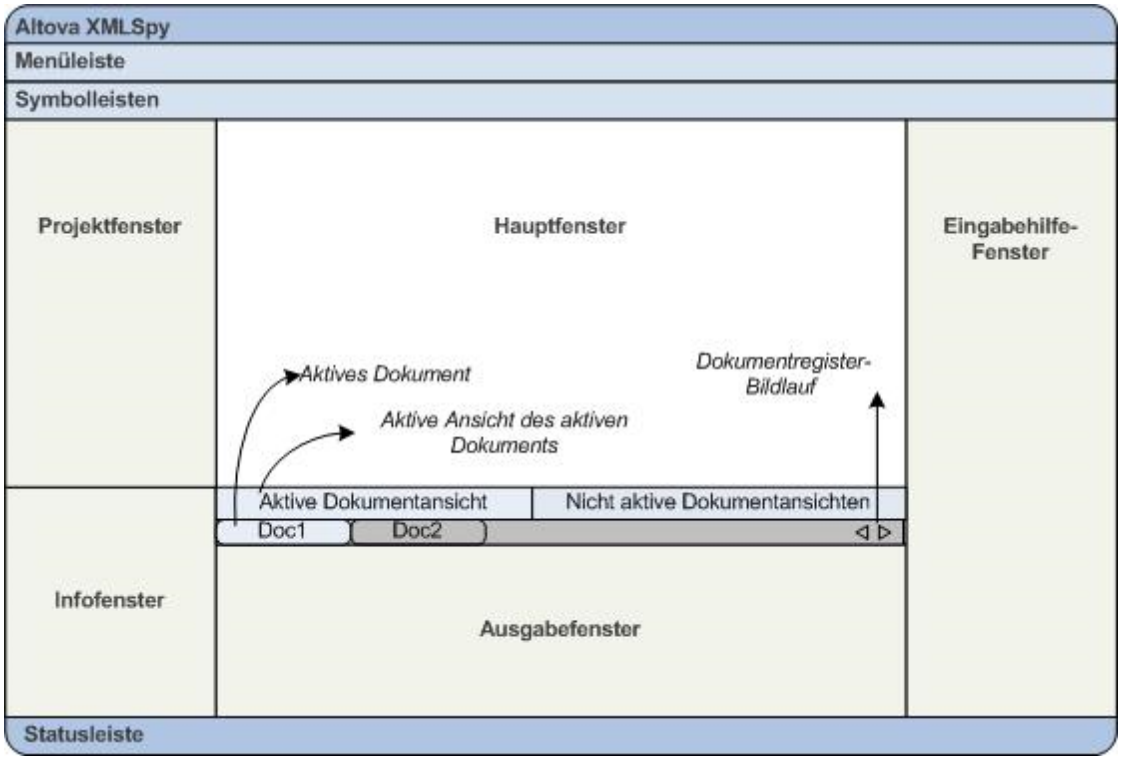
Starten von XMLSpy

Um XMLSpy zu starten, doppelklicken Sie auf das XMLSpy-Symbol auf Ihrem Desktop oder rufen Sie das XMLSpy-Programm über das Menü **Start | Alle Programme** auf. Daraufhin wird XMLSpy gestartet, ohne dass Dokumente auf der Benutzeroberfläche geöffnet werden. Öffnen Sie XMLSpy jetzt.

Übersicht über die Benutzeroberfläche

Standardmäßig ist die Benutzeroberfläche von XMLSpy in drei vertikale Bereiche gegliedert (*Abbildung unten*). Diese drei Bereiche bestehen von links nach rechts gesehen aus: (i) dem Projekt- und dem Info-Fenster; (ii) dem Hauptfenster und den Ausgabefenstern; und (iii) den Eingabehilfefenstern. Im Projektfenster sehen Sie das Beispielprojekt, das beim ersten Start von XMLSpy standardmäßig geöffnet wird.

Im Folgenden werden die wichtigsten Elemente der Benutzeroberfläche und die Funktionen der verschiedenen Komponenten übersichtsmäßig beschrieben. In den Unterabschnitten dieses ersten Teils des Tutorials wird die Benutzeroberfläche näher beschrieben.



Dokumentleiste im Hauptfenster: Wenn mehrere Dokumente geöffnet sind, werden die einzelnen Dokumente im Hauptfenster auf Registern der Dokumentleiste angezeigt (*siehe Abbildung*). Durch Klicken auf ein Register machen Sie das jeweilige Dokument zum aktiven Dokument. Durch Klicken auf die Pfeile auf der rechten Seite der Dokumentleiste können Sie einen Bildlauf durch die Dokumentregister durchführen. Öffnen Sie zwei oder mehr Dateien (z.B. aus dem Projekt "Examples"), um zu sehen, wie die Register verwendet werden.

 © 2019-2025 Altova GmbH

Dokumentbearbeitungsansichten: Das aktive Dokument kann in einer von mehreren möglichen Bearbeitungsansichten angezeigt werden. Zum Beispiel:

- Ein XML (`.xml`)-Dokument kann in der Text-, Grid-, Authentic- und Browser-Ansicht, nicht aber in anderen Ansichten wie der Schema-Ansicht angezeigt werden.
- Ein XM-Schema (`.xsd`)-Dokument kann dagegen in der Text-, Grid-, Schema- und Browser-Ansicht, nicht aber in der Authentic-Ansicht angezeigt werden.

Es stehen die folgenden Ansichten zur Verfügung: Textansicht, Grid-Ansicht, Schema-Ansicht, Authentic-Ansicht, Archivansicht und Browser-Ansicht.

Eingabehilfen: Die Eingabehilfen ändern sich je nach Art des gerade aktiven Dokuments (z.B. XML oder XSD oder CSS oder WSDL) und je nach der gerade aktiven Dokumentansicht (z.B. Textansicht oder Schema-Ansicht). Über die Eingabehilfen stehen kontextsensitive Bearbeitungsoptionen zur Verfügung, sodass Sie das aktive Dokument schnell und korrekt bearbeiten können.

1.1 Die Ansichten

In diesem Teil des Tutorials erfahren Sie, (i) wie man zwischen den Dokumentbearbeitungsansichten wechselt und (ii) wie man die Standardbearbeitungsansicht eines bestimmten Dokumenttyps ändert.

Wechseln zwischen Dokumentansichten

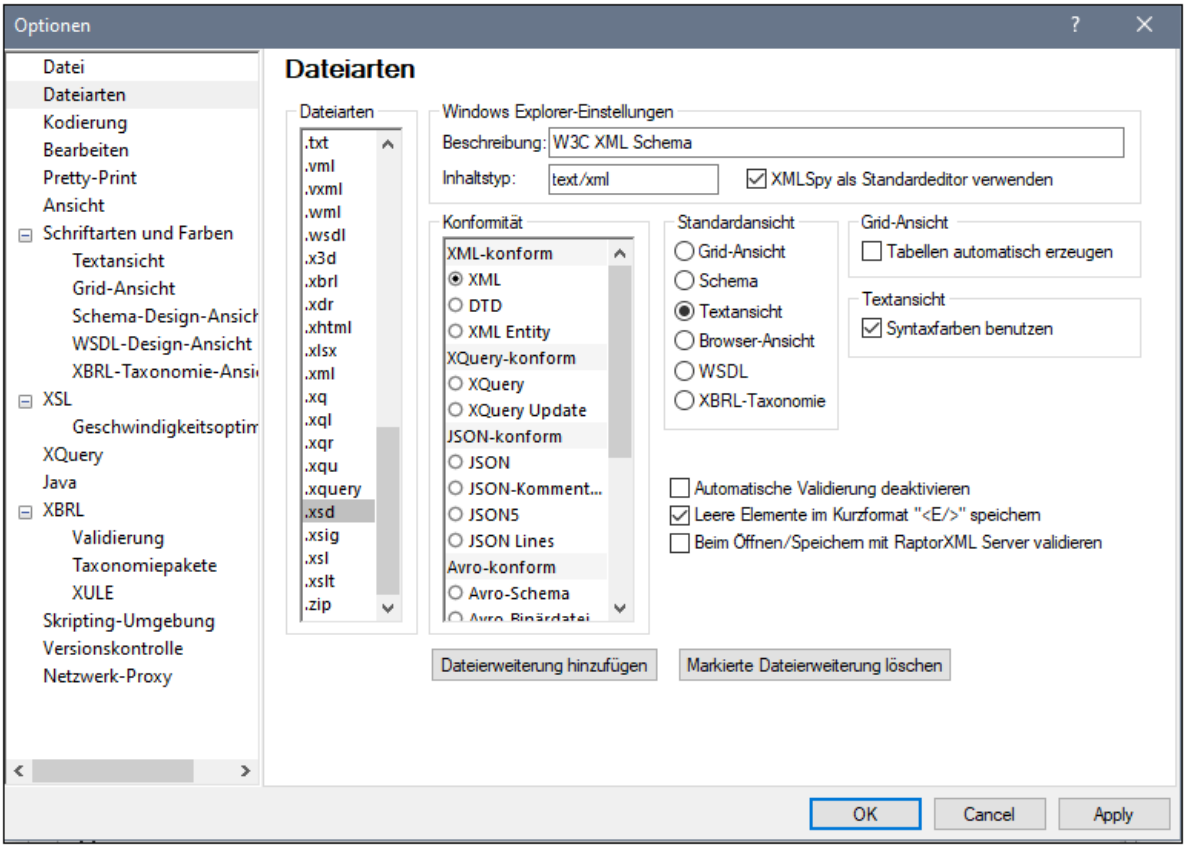
Wenn Sie ein Dokument öffnen, wird es in der für diesen Dokumenttyp definierten Standardansicht geöffnet. So öffnen Sie ein Dokument:

1. Wählen Sie den Befehl **Datei | Öffnen**.
2. Navigieren Sie zur Datei `AddressFirst.xsd` im Ordner `C:\Dokumente und Einstellungen\<Benutzer>\Eigene Dokumente\Altova\XMLSpy2025\Examples\Tutorial`, wählen Sie sie aus und klicken Sie auf **Öffnen**. Die Datei wird in der Schema-Ansicht geöffnet.
3. Durch Klicken auf die verschiedenen Ansichtsregister (Text, Grid, usw.) am unteren Rand des Hauptfensters können Sie zwischen den verschiedenen Ansichten wechseln. Sie können das XML-Schema-Dokument in der Text-, Grid-, Schema- und Browser-Ansicht anzeigen.
4. Sie können die Ansicht auch durch Auswahl der gewünschten Ansicht im Menü "Ansicht" wechseln. Versuchen Sie, die Ansicht des Dokuments `AddressFirst.xsd` über die Befehle des Menüs "Ansicht" zu wechseln.
5. Schließen Sie das Dokument (mit **Datei | Schließen**).

Ändern der Standardansicht eines Dokumenttyps

Alle Dokumente mit der Erweiterung `.xsd` werden standardmäßig in der Schema-Ansicht geöffnet. Sie können die Standardansicht, in der ein Dokument geöffnet wird, für jeden Dokumenttyp im Dialogfeld "Optionen" ändern. Nehmen Sie diese Einstellung nun für `.xsd`-Dokumente vor.

1. Wählen Sie den Befehl **Extras | Optionen** und wechseln Sie zum Abschnitt *Dateiarten* (*Abbildung unten*) .
2. Scrollen Sie im Bereich *Dateiarten* hinunter zum Eintrag `.xsd` und wählen Sie diesen aus (*siehe Abbildung*).
3. Wählen Sie im Bereich "Standardansicht" die Option "Textansicht" aus.



4. Klicken Sie auf **OK**.
5. Wählen Sie den Befehl **Datei | Öffnen** und öffnen Sie die Datei `AddressFirst.xsd`. Die Datei wird in der Textansicht geöffnet.
6. Wechseln Sie zur Schema-Ansicht, um die Datei in dieser Ansicht zu sehen und schließen Sie die Datei anschließend (**Datei | Schließen**).
7. Wechseln Sie zurück in das Dialogfeld "Optionen" (**Extras | Optionen**) und ändern Sie die Standardansicht von `.xsd`-Dateien im Abschnitt *Dateiarten* wieder zurück in Schema.

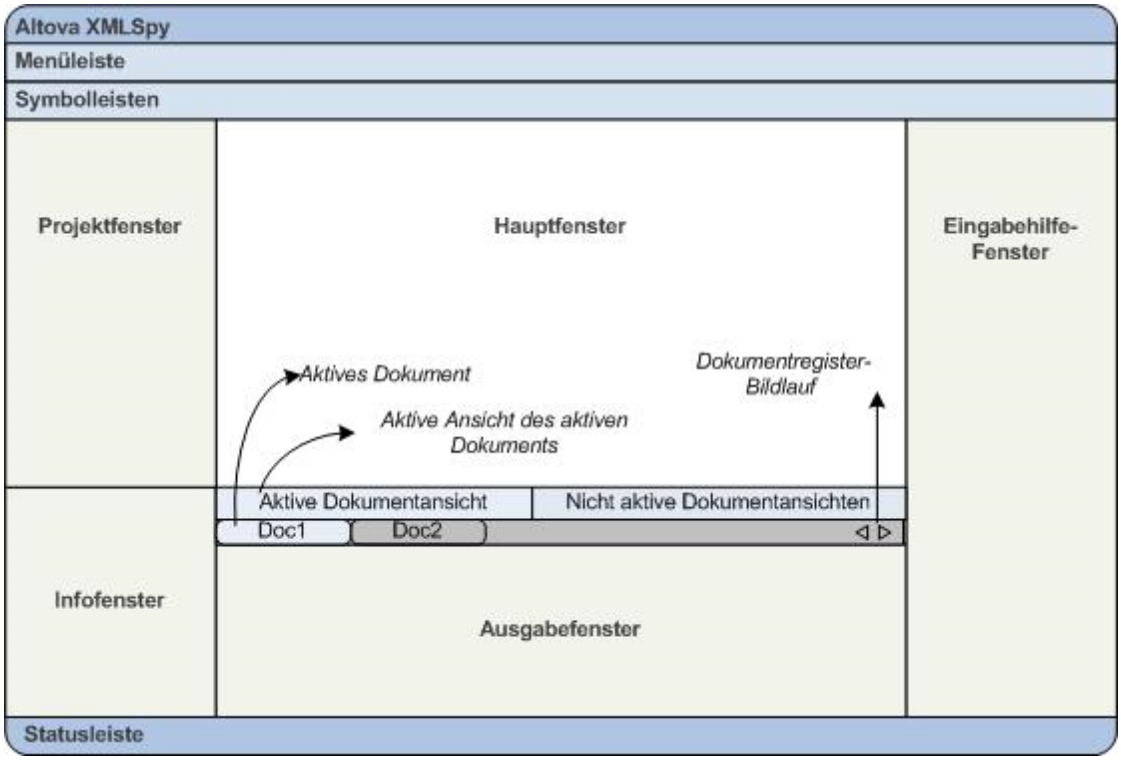
Anmerkung: Sie können im Dialogfeld "Optionen" im Abschnitt *Dateiarten* (*Abbildung oben*) die Standardansicht jeder der aufgelisteten Dateierweiterungen ändern. Über die Schaltfläche **Neue Dateierweiterung hinzufügen** können Sie eine neue Dateierweiterung zur Liste hinzufügen.

1.2 Die Fenster

Standardmäßig sind die Seitenleisten rund um das Hauptfenster angeordnet (*siehe Abbildung unten*) und werden in die folgenden Gruppen eingeteilt:

- Projektfenster
- Infofenster
- Eingabehilfen: (verschiedene, von der Art der aktiven Dokumentart abhängig)
- Ausgabefenster: Meldungen, XPath, XSL-Übersicht, In Dateien suchen, In Schemas suchen

 © 2019-2025 Altova GmbH



In diesem Abschnitt erfahren Sie, wie man Fenstergruppen ein- und ausblendet und Fenster am Bildschirm verschiebt. Dies ist nützlich, wenn Sie mehr Platz auf der Benutzeroberfläche benötigen.

Ein- und Ausblenden der Fenstergruppen

Die Fenstergruppen (Projektfenster, Info-Fenster, Eingabehilfen, Ausgabefenster) können über die entsprechenden Befehle im Menü **Fenster** ein- und ausgeblendet werden. Eine Fenstergruppe, die gerade angezeigt wird, kann auch durch Rechtsklick auf die Titelleiste und Auswahl des Befehls **Ausblenden** ausgeblendet werden. Eine ausgeblendete Fenstergruppe kann nur über das Menü **Fenster** wieder eingeblendet werden.

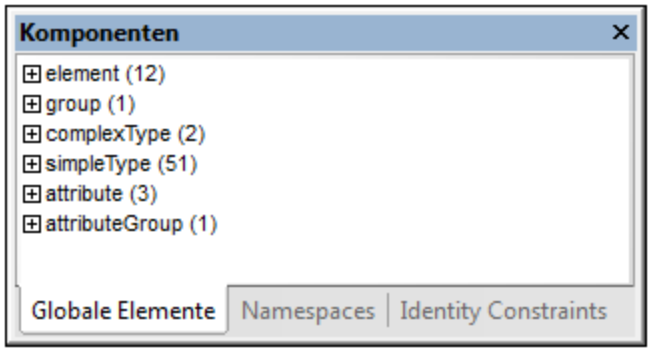
Öffnen Sie eine XML-Datei aus dem Ordner `C:\Dokumente und Einstellungen\<Benutzer>\Eigene Dokumente\Altova\XMLSpy2025\Examples\Tutorial` und üben Sie die Verwendung dieser einfachen Befehle, um sich damit vertraut zu machen. Nähere Informationen zum Anzeigen und Ausblenden von Fenstergruppen finden Sie im Abschnitt XMLSpy-Benutzeroberfläche.

Gespeicherter und gesicherter Status

XMLSpy macht standardmäßig alle fünf Sekunden ein Backup nicht gespeicherter Dokumente. Das Register der jeweiligen Datei am unteren Rand des Hauptfensters gibt mit Hilfe von Symbolen Aufschluss über den gespeicherten/nicht gespeicherten Zustand der Datei und Ihren Backup-Status. Da Ihnen diese Symbolewährend der Arbeit immer wieder unterkommen werden, sollten Sie deren Bedeutung kennen: Informationen zu diesen Symbolen finden Sie im Abschnitt Automatisches Backup von Dateien.

Verschieben von Fenstern auf dem Bildschirm

Ein einzelnes Fenster kann entweder frei schwebend innerhalb der Benutzeroberfläche oder darin angedockt angezeigt werden. Ein Fenster kann auch als Register in einer anderen Fenstergruppe angedockt werden. (*Erklärung zu Fenstergruppen siehe oben*). So sehen Sie z.B. in der Abbildung unten die Eingabehilfe "Komponenten" in der Schema-Ansicht. Sie hat drei mit Registern versehene Fenster: das Fenster "Gobale Elemente", das Fenster "Namespaces" und das Fenster "Identity Constraints".



Ein Fenster kann auf die folgenden Arten an- oder abgedockt werden:

- Doppelklicken Sie auf die Titelleiste des Fensters. Wenn das Fenster angedockt war, wird es nun frei schwebend angezeigt. Wenn es frei schwebend angezeigt wurde, wird es nun dort angedockt, wo es zuletzt angedockt war.
- Klicken Sie mit der rechten Maustaste auf die Titelleiste eines Fensters und wählen Sie den entsprechenden Befehl aus (**Abgedockt** oder **Angedockt**).
- Ziehen Sie das Fenster (an der Titelleiste als Ziehpunkt) aus seiner angedockten Position bis es frei schwebend angezeigt wird. Ziehen Sie ein frei schwebendes Fenster (an seiner Titelleiste) an die Stelle, an der es angedockt werden soll. Es erscheinen zwei Gruppen blauer Pfeile. Mit den äußeren vier Pfeilen können Sie das Fenster relativ zum Applikationsfenster andocken (am oberen, rechten, unteres oder linken Rand der Benutzeroberfläche). Mit den inneren Pfeilen können Sie das Fenster relativ zu dem Fenster andocken, über dem sich der Cursor gerade befindet. Wenn Sie ein Fenster über den Mittelpunkt der inneren Pfeile ziehen (oder auf die Titelleiste eines Fensters), wird es als Registerkarte in dem Fenster angezeigt, in das es gezogen wurde.

Um ein Fenster, das als Registerkarte angezeigt wird, frei schwebend anzuzeigen, doppelklicken Sie auf sein Register. Um ein als Register angezeigtes Fenster aus einer Gruppe derartiger Fenster herauszuziehen, ziehen Sie es am Register heraus.

Um das Verschieben von Fenstern zu üben, öffnen Sie eine XML-Schema-Datei aus dem Ordner `C:\Dokumente und Einstellungen\<Benutzer>\Eigene Dokumente\Altova\XMLSpy2025\Examples\Tutorial` und probieren Sie die oben beschriebenen Methoden bei geöffneter Schema-Ansicht aus, um sich damit vertraut zu machen.

1.3 Menüs und Symbolleisten

In diesem Abschnitt des Tutorials werden die Hauptfunktionen der Menüs und Symbolleisten von XMLSpy erläutert.

 © 2019-2025 Altova GmbH

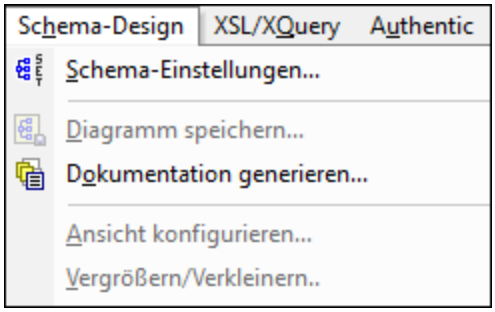
Menüs

Es gibt zwei Hauptleisten: (i) ein Standardmenü, das angezeigt wird, wenn kein Dokument geöffnet ist und (ii) das vollständige XMLSpy Applikationsmenü, das angezeigt wird, sobald ein Dokument geöffnet ist. Gehen Sie folgendermaßen vor:

1. Schließen Sie alle offenen Dokumente mit dem Menübefehl **Datei | Alles schließen**. Daraufhin wird das Standardmenü angezeigt.
2. Öffnen Sie die Datei `AddressFirst.xsd` durch Klicken auf ihren Namen in der Liste der zuletzt geöffneten Dateien ganz unten im Menü **Datei.** Sobald die Datei in der Schema-Ansicht geöffnet ist, ändert sich das Menü und Sie sehen das vollständige XMLSpy-Applikationsmenü.

Die Menüs sind in erster Linie nach Funktion geordnet. Ein Befehl in einem Menü ist nur dann aktiv, wenn er an der Stelle, an der sich der Cursor gerade befindet, ausgeführt werden kann oder, um in der aktuellen Ansicht des aktiven Dokuments eine Auswahl vornehmen zu können. Um zu sehen, unter welchen Umständen ein Menübefehl aktiv oder inaktiv ist, führen Sie die folgenden Schritte durch:

1. Klicken Sie auf das Menü **Schema-Design**. Beachten Sie, dass die Befehle **Diagramm speichern, Ansicht konfigurieren** und **Vergrößern/Verkleinern** deaktiviert sind (*Abbildung unten*).

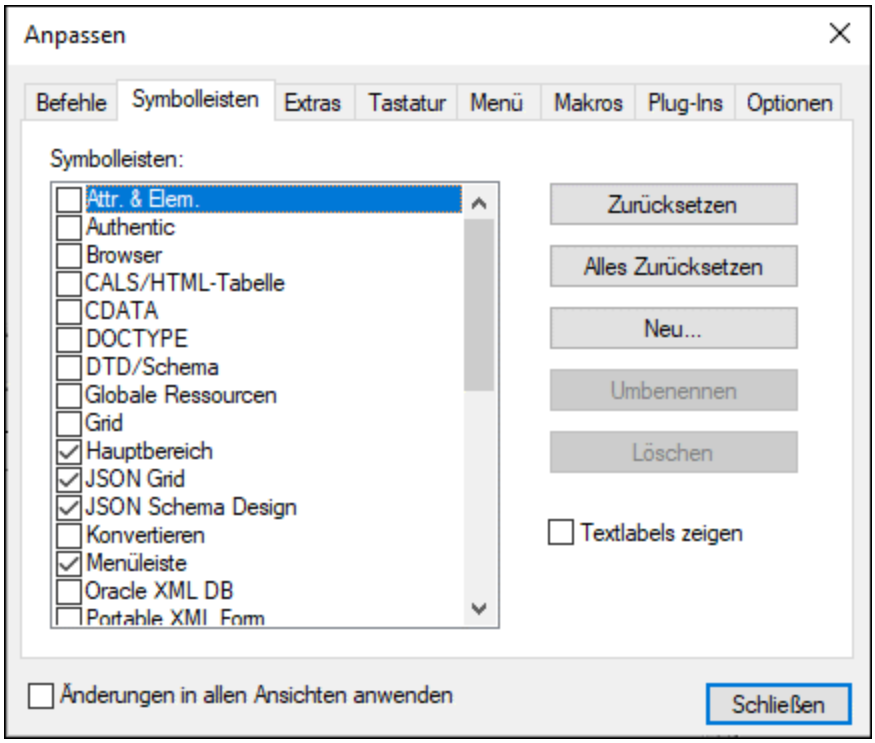


2. Klicken Sie in einen leeren Bereich außerhalb des Menüs, damit das Menü nicht mehr angezeigt wird. Klicken Sie anschließend auf die Schaltfläche **Diagramm anzeigen** links von der Elementkomponente. Daraufhin wechseln Sie in die Content Model-Ansicht der Schema-Ansicht (dies ist die zweite der beiden Schema-Ansichten; die erste ist die Schema-Übersicht). Wenn Sie nun das Menü "Schema-Design" öffnen, sehen Sie, dass die Befehle **Diagramm speichern, Ansicht konfigurieren** und **Vergrößern/Verkleinern** nun aktiv sind. Diese Befehle sind nur in der Content Model-Ansicht der Schema-Ansicht, nicht aber in dedr Schema-Übersicht der Schema-Ansicht aktiv. Beachten Sie außerdem, dass nur XML-Schema-Dateien in der Schema-Ansicht geöffnet werden können.
3. Eine XML-Schema-Datei ist gleichzeitig eine XML-Datei, daher wird sie in der Text- und Grid-Ansicht als XML-Datei angezeigt. In diesen Ansichten sind alle Menübefehle, die sich auf XML-Dateien anwenden lassen, aktiv. Vergleichen Sie die Befehle im Menü **Bearbeiten** in der Schema- und Text-Ansicht miteinander (ob sie aktiv sind oder nicht).
4. Vergleichen Sie als Nächstes die Befehle im Menü **XML | Einfügen** in der Text- und der Grid-Ansicht (ob sie aktiv sind oder nicht). Die Befehle in diesem Menü sind nur in der Grid-Ansicht aktiv.

Eine Beschreibung aller Menübefehle finden Sie im Abschnitt Menübefehle des Benutzerhandbuchs.

Symbolleisten

Die Anzeige der Symbolleisten variiert je nachdem, welche Ansicht aktiv ist. In der Standardeinstellung sind die richtigen Symbolleisten für die einzelnen Ansichten definiert. Daher unterscheiden sich die angezeigten Symbolleisten je nach Ansicht. Sie können die Symbolleisten allerdings im Dialogfeld "Anpassen" auf dem Register "Symbolleisten" (**Extras | Anpassen | Symbolleisten**, *Abbildung unten*) auch anpassen.



Üben Sie nun das Verschieben der Symbolleisten auf der Benutzeroberfläche. Klicken Sie auf den Ziehpunkt einer Symbolleiste und ziehen Sie die Symbolleiste an die gewünschte Stelle auf der Benutzeroberfläche. (Der Ziehpunkt wird durch die vertikale gepunktete Linie auf der linken Seite jeder Symbolleiste gekennzeichnet; *siehe Abbildung unten*).

Versuchen Sie, eine Symbolleist an die folgenden Stellen zu ziehen: (i) in eine andere Zeile im Symbolleistenbereich; (ii) nach links oder rechts von einer anderen Symbolleiste; (iii) in die Mitte des Hauptfensters; (iv) angedockt an die linke oder rechte Seite des Applikationsfensters (dazu muss der Ziehpunkt über den linken bzw. rechten Rand des Fensters platziert werden).

Schließen Sie anschließend die Datei `AddressFirst.xsd` wieder.

© 2019-2025 Altova GmbH

1.4 Einstellungen für die Textansicht

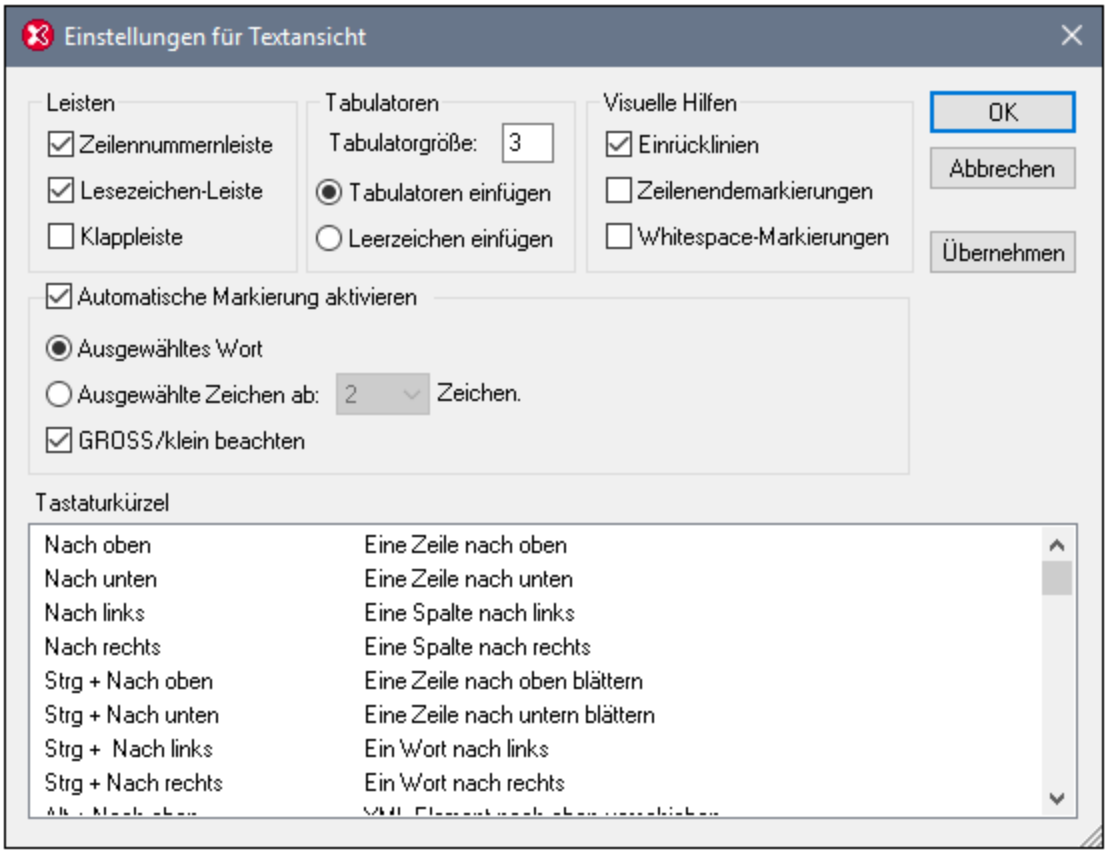
In diesem Abschnitt erfahren Sie, wie Sie die Pretty-Print Anzeige für ein Dokument konfigurieren und Lesezeichen bei der Bearbeitung verwenden. Mit Pretty-Print wird ein Dokument in der Textansicht so angezeigt, dass jede untergeordnete Ebene in der XML-Hierarchie weiter eingerückt ist als die darüberliegende Ebene (*siehe Abbildung unten*). Mit Hilfe von Lesezeichen können Sie schnell zu einer markierten Stelle zurückkehren.

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <Company>
3       <Address xsi:type="US-Address">
4           <Name>US dependency</Name>
5           <Street>Noble Ave.</Street>
6           <City>Dallas</City>
7           <Zip>04812</Zip>
8           <State>Texas</State>
9       </Address>
10      <Person Manager="true" Degree="BA" Programmer="false"
11          <First>Fred</First>
12          <Last>Smith</Last>
13          <PhoneExt>22</PhoneExt>
14          <Email>Smith@work.com</Email>
15      </Person>
16  </Company>
```

Pretty-Print-Anzeige

Für die Pretty-Print-Anzeige sind zwei Schritte erforderlich: (i) Konfiguration der Pretty-Print-Anzeige, (ii) Anwenden von Pretty-Print.

1. Öffnen Sie die Datei **`CompanyFirst.xml`**aus dem Ordner **`C:\Dokumente und Einstellungen\<Benutzer>\Eigene Dokumente\Altova\XMLSpy2025\Examples\Tutorial`**.
2. Wechseln Sie in die Textansicht, wenn diese nicht die Standardansicht von XML-Dokumenten ist.
3. Wählen Sie den Menübefehl **Ansicht | Einstellungen für Textansicht**, um das Dialogfeld "Einstelllungen für Textansicht (*Abbildung unten*) zu öffnen.



4. Verringern Sie die Tabulatorgröße im Bereich "Tabulatoren" auf 3. Belassen Sie die Standardeinstellung des Optionsfelds *Tabulatoren einfügen* unverändert. Mit dieser Einstellung wird die Pretty-Print-Einrückung als Tabulator (und nicht in Form von Leerzeichen) angezeigt, wobei jeder Tabulator eine Breite von drei Leerzeichen hat. Klicken Sie abschließend auf **OK**.
5. Wählen Sie den Menübefehl **Bearbeiten | Pretty-Print**. Dadurch wird Pretty-Print auf das Dokument angewendet. Daraufhin wird das Dokument mit der neuen Tabulatorbreite angezeigt.
6. Öffnen Sie das Dialogfeld "Einstellungen für Textansicht" (**Ansicht | Einstellungen für Textansicht**) wieder und aktivieren Sie im Bereich *Visuelle Hilfen* die Option "Zeilenendemarkierungen".
7. Gehen Sie in der Textansicht zum Ende einer Zeile und löschen Sie die Zeilenendemarkierung, sodass die nächste Zeile eine Zeile hinaufspringt.
8. Wechseln Sie in die Grid-Ansicht und wieder zurück in die Textansicht.
9. Wählen Sie den Menübefehl **Bearbeiten | Pretty-Print**. Das Dokument wird mit Pretty-Print angezeigt und die gelöschte Zeilenendemarkierung wird wiederhergestellt.

Anmerkung: Wenn Sie im Abschnitt "Pretty-Print" des Dialogfelds "Optionen" (**Extras | Optionen | Pretty-Print**) das Kontrollkästchen *Einrückungen verwenden* deaktivieren und dann ein Pretty-Print durchführen, beginnen alle Zeilen ohne Einrückung.

Lesezeichen

Lesezeichen werden in eine Lesezeichenleiste links von den zu markierenden Zeilen platziert. Sie können dann schnell von einem Lesezeichen zum nächsten in Ihrem Dokument und wieder zurück springen.

 © 2019-2025 Altova GmbH

1. Stellen Sie im Dialogfeld "Einstellungen für Textansicht" (**Ansicht | Einstellungen für Textansicht**, *Abbildung oben*) sicher, dass die Option "Lesezeichen-Leiste" im Bereich *Leisten* aktiviert ist. Klicken Sie abschließend auf **OK**.
2. Platzieren sie den Cursor in der Textansicht der Datei `CompanyFirst.xml` an eine beliebige Stelle innerhalb einer zu markierenden Zeile und wählen Sie dann den Menübefehl **Bearbeiten | Lesezeichen einfügen/löschen**. Daraufhin wird die Zeile mit einem blauen Lesezeichen in der Lesezeichenleiste versehen (*siehe Abbildung unten*).
3. Erstellen Sie, wie in Schritt 2 beschrieben, ein weiteres Lesezeichen in eine anderen Zeile.

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <Company>
3      <Address xsi:type="US-Address">
4         <Name>US dependency</Name>
5         <Street>Noble Ave.</Street>
6         <City>Dallas</City>
7         <Zip>04812</Zip>
8         <State>Texas</State>
9      </Address>
10     <Person Manager="true" Degree="BA" Programmer="false">
11        <First>Fred</First>
12        <Last>Smith</Last>
13        <PhoneExt>22</PhoneExt>
14        <Email>Smith@work.com</Email>
15     </Person>
16  </Company>
17
```

4. Drücken Sie **F2** (oder wählen Sie den Befehl **Bearbeiten | Zum nächsten Lesezeichen**), um zum nächsten Lesezeichen im Dokument zu gehen. Drücken Sie **Umschalt + F2** (oder wählen Sie den Befehl **Bearbeiten | Zum nächsten Lesezeichen**), um zum vorherigen Lesezeichen im Dokument zu gelangen. Wiederholen Sie einen oder beide Befehle so oft Sie wollen.
5. Setzen Sie den Cursor in eine der mit einem Lesezeichen versehenen Zeilen und wählen Sie den Menübefehl **Bearbeiten | Lesezeichen einfügen/löschen,** um das Lesezeichen zu löschen. Das Lesezeichen wird entfernt.
6. Speichern und schließen Sie die Datei. In der Datei werden keine Lesezeicheninformationen gespeichert. Öffnen Sie die Datei erneut, um dies zu überprüfen.

2 XML-Schemas: Grundlagen

Ein XML-Schema beschreibt die Struktur eines XML-Dokuments. Ein XML-Dokument kann anhand eines XML-Schemas 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.

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-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.

Ziel

In diesem Tutorial-Abschnitt lernen Sie, wie man XML-Schemas in der Schema-Ansicht bearbeitet. Dabei lernen Sie die folgenden Schritte:

- Erstellen einer neuen Schema-Datei
- Definieren von Namespaces für das Schema
- Definieren eines einfachen Inhaltsmodells (Content Model)
- Hinzufügen von Elementen zum Inhaltsmodell über Kontextmenüs und mittels Drag-and-Drop
- Konfigurieren der Content Model-Ansicht

Wenn Sie das einfache Schema fertig erstellt haben, können Sie mit dem nächsten Abschnitt des Tutorials 31, in dem Sie erfahren, wie Sie in XMLSpy mit den komplexeren Funktionalitäten von XML-Schema arbeiten, fortfahren. Auf diesen Abschnitt folgt ein Abschnitt über Schemanavigation und Dokumentation 44 in XMLSpy.

In diesem Abschnitt verwendete Befehle

Wir verwenden in diesem Abschnitt des Tutorials ausschließlich die Schema-Ansicht. Dabei werden die folgenden Befehle verwendet:

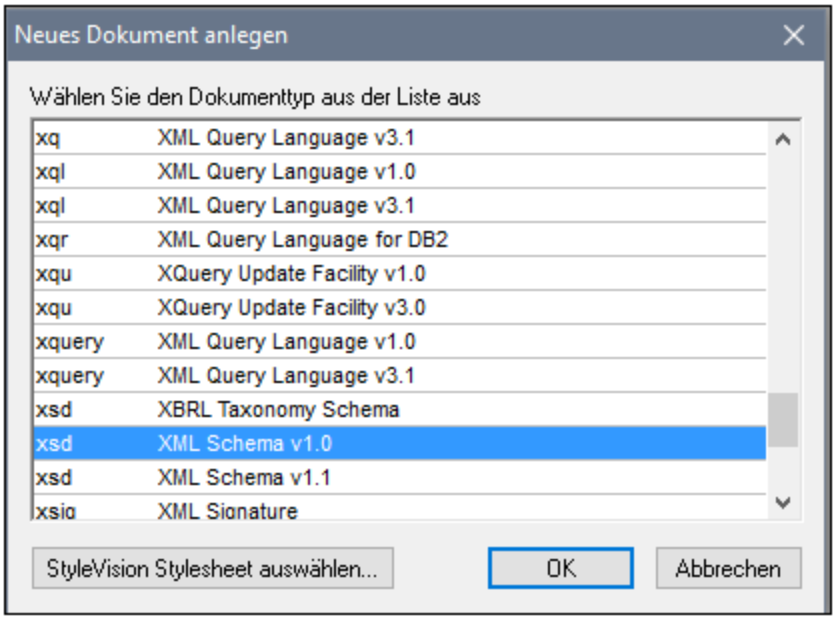
	Diagramm anzeigen (oder Content Model-Ansicht anzeigen). Dieses Symbol befindet sich links von alle globalen Komponenten in der Schema-Übersicht. Wenn Sie darauf klicken, wird das Content Model der dazugehörigen globalen Komponente angezeigt.

2.1 Anlegen einer neuen XML-Schema-Datei

Anlegen einer neuen XML-Schema-Datei

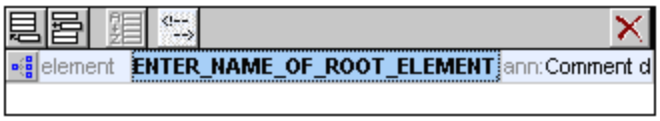
1. Wählen Sie die Menüoption **Datei | Neu**. Das Dialogfeld "Neues Dokument anlegen" wird angezeigt.

 © 2019-2025 Altova GmbH

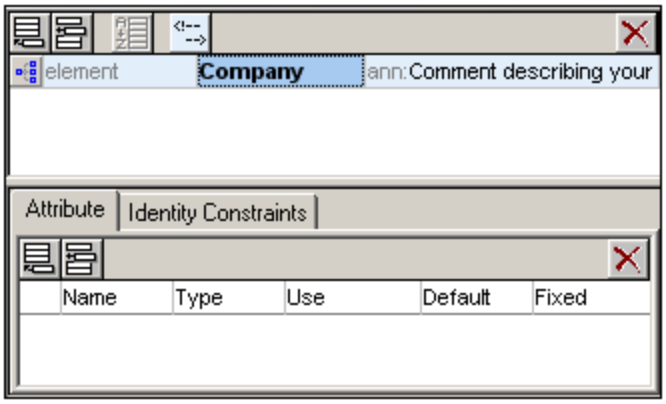


2. Wählen Sie im Dialogfeld den Eintrag *XSD (XML Schema v1.0)* (die Dokumentbeschreibung und die Liste im Fenster sind unter Umständen nicht mit denen in der Abbildung identisch). Bestätigen Sie mit **OK**. Eine leere Schema-Datei wird im Hauptfenster in der Schema-Ansicht angezeigt.
3. Klicken Sie in der Schema-Design-Symbolleiste auf die Schaltfläche **XSD 1.0** (*siehe Abbildung unten*), damit die Schema-Ansicht im XSD 1.0-Bearbeitungsmodus verwendet wird.

4. Sie werden aufgefordert, den Namen des Root-Elements einzugeben.



5. Doppelklicken Sie in das markierte Feld, und geben Sie `Company` ein. Bestätigen Sie mit der **Eingabetaste**. `Company` ist somit das "Root" Element dieses Schemas und gleichzeitig ein globales Element. Die im Hauptfenster angezeigte Ansicht (*siehe Abbildung unten*) nennt sich Schema-Übersicht und enthält eine Liste aller globalen Komponenten im oberen Bereich des Hauptfensters; die Attribute und Identity Constraints der ausgewählten globalen Komponente werden im unteren Fenster angezeigt. (Sie können das Content Model einzelner globaler Komponenten ansehen und bearbeiten, indem Sie auf das Symbol "Diagramm anzeigen" klicken, das links von der globalen Komponente zu sehen ist.)



6. Geben Sie im Annotationsfeld (`ann`) des Elements `Company` die Beschreibung für das Element ein, in diesem Fall `Root element`.
7. Klicken Sie auf die Menüoption **Datei | Speichern** und benennen Sie Ihr Schema (z.B. `AddressFirst.xsd`).

Die farbigen Kreissymbole auf dem Register der Datei geben Aufschluss über den Backup-Status der Datei. Informationen zu diesen Symbolen finden Sie im Abschnitt Automatisches Backup von Dateien.

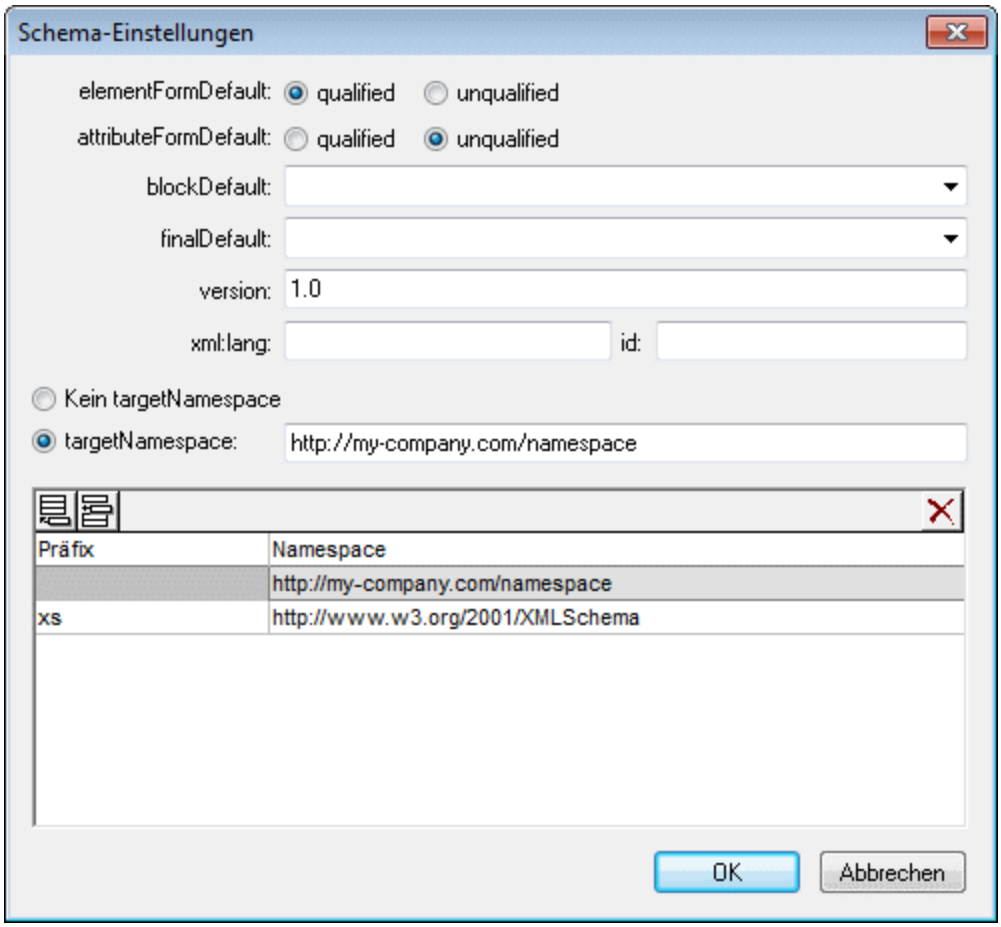
2.2 Definieren von Namespaces

XML Namespaces spielen eine wichtige Rolle in XML-Schemas und XML-Dokumenten. Ein XML-Schema muss den XML-Schema-Namespace referenzieren und kann optional auch einen Target Namespace für das XML-Instanzdokument definieren. Als Schema-Autor müssen Sie festlegen, wie Sie diese beiden Namespaces definieren (im Grunde mit Hilfe von Präfixen).

Sie werden nun im XML-Schema, das Sie erstellen, einen Target Namespace für das XML-Instanzdokument definieren. (Die erforderliche Referenz zum XML Schema Namespace wird von XMLSpy automatisch erstellt, wenn Sie ein neues XML Schema-Dokument anlegen.) So erstellen Sie einen Target Namespace:

1. Wählen Sie die Menüoption **Schema-Design | Schema-Einstellungen**, um das Dialogfeld "Schema-Einstellungen" aufzurufen (*Abbildung unten*).

 © 2019-2025 Altova GmbH



2. Klicken Sie auf das *Target Namespace*-Optionsfeld und geben Sie `http://my-company.com/Namespace` ein. In XMLSpy wird der von Ihnen als Target Namespace eingegebene Namespace als Default Namespace des XML-Schema-Dokuments erstellt und in der Liste der Namespaces im unteren Bereich des Dialogfelds angezeigt.
3. Bestätigen Sie mit OK.

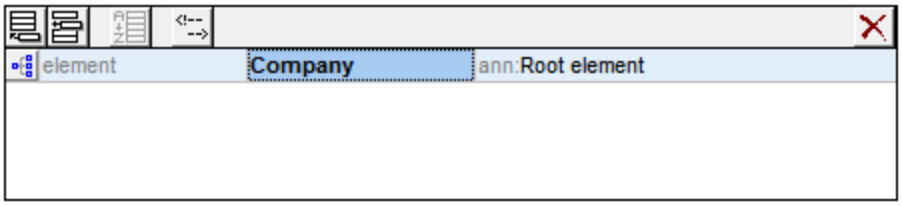
Anmerkung:

- Der XML Schema Namespace wird von XMLSpy automatisch erstellt und erhält das Präfix `xs:`.
- Wenn Sie das XML-Instanzdokument erstellen, muss es den im XML-Schema für das XML-Dokument definierten Target Namespace haben, um gültig zu sein.

2.3 Definieren eines Content Model

Sie haben in der Schema-Übersicht bereits ein globales Element mit dem Namen `Company` erstellt. Dieses Element soll ein einziges `Address`-Element und eine unbegrenzte Anzahl an `Person`-Elementen enthalten. Dies ist dann das Content Model von `Company`. Globale Elemente, die Content Models haben können, sind Elemente, complexTypes und Elementgruppen. In XMLSpy wird das Content Model einer globalen Komponente in der Content Model-Ansicht der Schema-Ansicht angezeigt. Um das Content Model einer globalen

Komponente anzeigen und editieren zu können, klicken Sie auf das Symbol "Diagramm anzeigen" links neben der globalen Komponente.

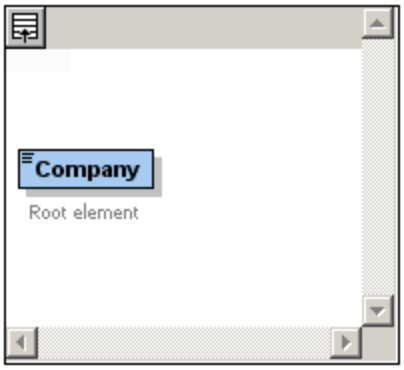


In diesem Abschnitt werden Sie das Content Model des Elements `Company` erstellen.

Erstellen eines einfachen Content Model

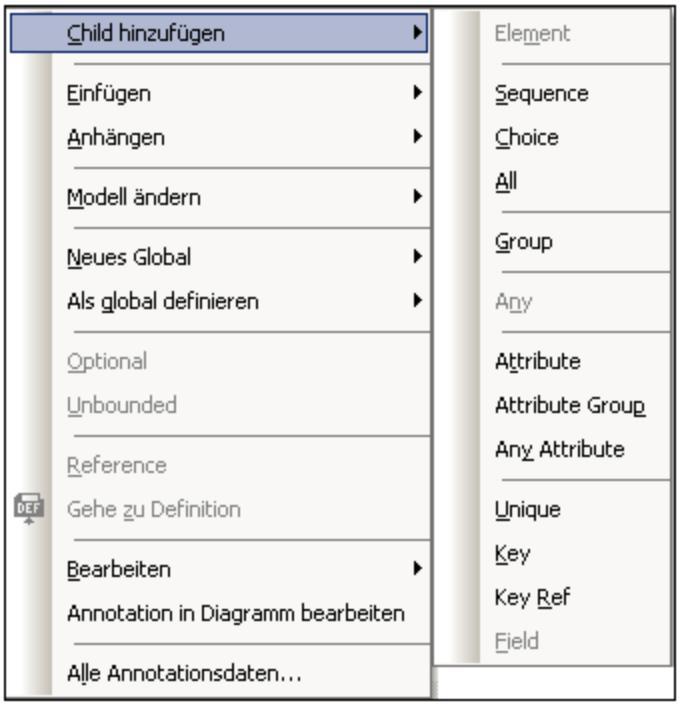
So erstellen Sie das Content Model des Elements `Company`:

1. Klicken Sie in der Schema-Übersicht auf das Symbol "Diagramm anzeigen" links neben dem Element `Company`. Hierdurch wird das Content Model des Elements `Company` angezeigt (*Abbildung unten*), das derzeit noch leer ist. (Alternativ dazu können Sie auch auf den Eintrag `Company` im Komponentenfenster doppelklicken, um das Content Model anzuzeigen.)

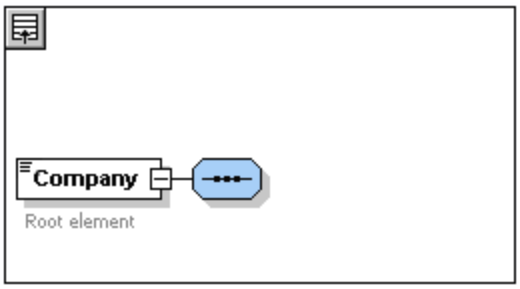


2. Ein Content Model besteht aus **Kompositoren** und **Komponenten**. Die Kompositoren definieren die Beziehung zwischen zwei Komponenten. Sie müssen nun im `Company` Content Model einen Child-Kompositor zum Element `Company` hinzufügen, um ein Child-Element hinzufügen zu können. Rechtsklicken Sie dazu auf das Element "Company" und wählen Sie im Kontextmenü die Option **Child hinzufügen | Sequence**. (Sequence, Choice und All sind die drei Kompositoren, die in einem Content Model verwendet werden können.)

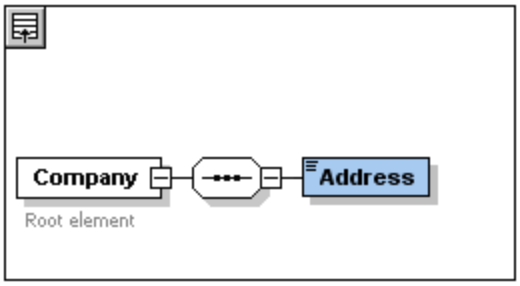
 © 2019-2025 Altova GmbH



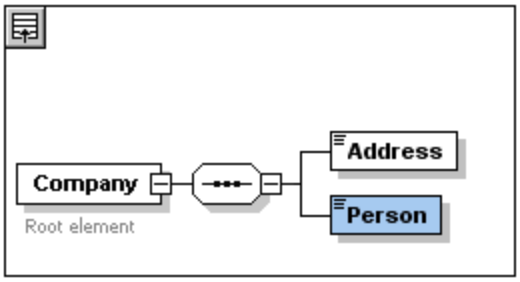
Daraufhin wird der Sequence-Kompositor eingefügt, der definiert, dass die Komponenten, die darauf folgen in der angegebenen Reihenfolge erscheinen müssen.



3. Rechtsklicken Sie auf den Sequence-Kompositor und wählen Sie **Child hinzufügen | Element**. Daraufhin wird ein unbenanntes Element hinzugefügt.
4. Benennen Sie das Feld `Address` und bestätigen Sie mit der Eingabetaste.

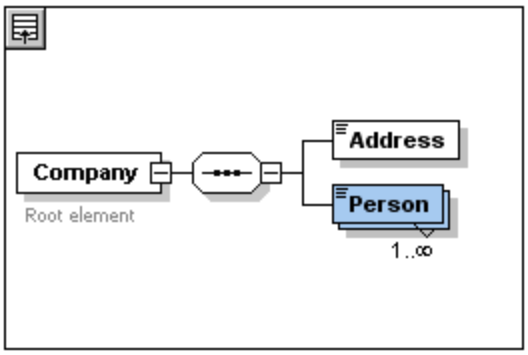


5. Rechtsklicken Sie erneut auf den Sequence-Kompositor und wählen Sie **Child hinzufügen | Element** und geben Sie der neuen Elementkomponente den Namen `Person`.



Sie haben nun ein Schema mit einem Eintrag für die Firmenadresse und einem Eintrag für einen Angestellten (`Person`) definiert. Wir müssen nun die Anzahl der `Person`-Elemente erhöhen.

6. Klicken Sie mit der rechten Maustaste auf das Element `Person` und wählen Sie **Unbounded** aus dem Kontextmenü. Unter dem Element `Person` wird nun die Anzahl der zulässigen Instanzen angezeigt: 1 bis unendlich.

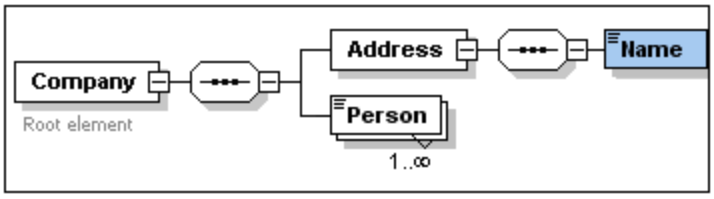


Sie können alternativ dazu in der Details-Eingabehilfe die Felder `minOcc` und `maxOcc` direkt editieren und die Anzahl der zulässigen Instanzen angeben, in diesem Fall 1 bzw. unbounded.

Hinzufügen von zusätzlichen Ebenen zur Struktur des Content Model

Bisher enthält das Content Model, das Sie erstellt haben, eine Ebene: eine Child-Ebene für das Element `company`, das die Elemente `Address` und `Person` enthält. Nun wollen wir den Inhalt des Elements `Address` definieren, und zwar die Subelemente `Name, Street` und `City`. Dies ist eine zweite Ebene. Wir müssen also zum Element `Address` einen Child-Kompositor hinzufügen und anschließend die Elementkomponenten selbst. Gehen Sie dabei folgendermaßen vor:

1. Klicken Sie mit der rechten Maustaste auf das Element `Address`, um das Kontextmenü zu öffnen, und wählen Sie **Child hinzufügen | Sequence**.
2. Klicken Sie mit der rechten Maustaste auf den **Sequence**-Kompositor und wählen Sie **Child hinzufügen | Element**. Benennen Sie die neu erstellte Elementkomponente mit "Name".



 © 2019-2025 Altova GmbH

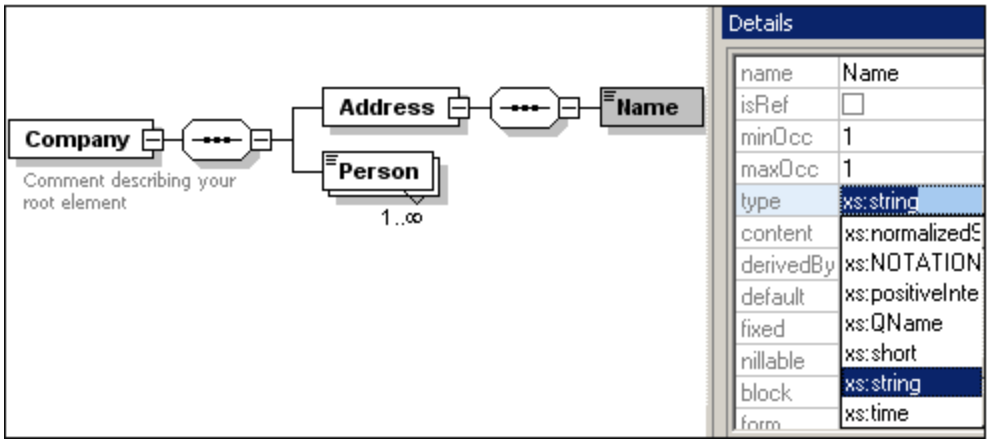
ComplexTypes, simpleTypes und XML Schema-Datentypen

Bisher haben wir noch keinen Elementtyp explizit definiert. Klicken Sie auf das Register **Text** um die Textansicht für Ihr Schema anzuzeigen (*siehe Liste unten*). Sie werden feststellen, dass jedes Mal, wenn ein Sequence-Kompositor eingefügt wurde, das Element `xs:sequence` im Element `xs:complexType` eingefügt wurde. Kurz gesagt, die Elemente `Company` und `Address` sind ComplexTypes, da sie Child-Elemente enthalten. Ein ComplexType ist ein Element, das Attribute oder Elemente enthält.

```
 <xs:element name="Company">
   <xs:annotation>
     <xs:documentation>Root element</xs:documentation>
   </xs:annotation>
   <xs:complexType>
     <xs:sequence>
       <xs:element name="Address">
         <xs:complexType>
           <xs:sequence>
             <xs:element name="Name"/>
           </xs:sequence>
         </xs:complexType>
       </xs:element>
       <xs:element name="Person"/>
     </xs:sequence>
   </xs:complexType>
</xs:element>
```

SimpleType-Elemente dagegen enthalten nur Text und haben keine Attribute. Beim Text kann es sich um Strings, Daten, Zahlen usw. handeln. Wir möchten, dass das Child-Element `Name` von `Address` nur Text enthält. Es ist ein SimpleType, dessen Textinhalt wir durch Restriktion auf einen String einschränken möchten. Dies lässt sich durch Verwendung des XML-Schema-Datentyps `xs:string` bewerkstelligen. Um das Element `Name` als diesen Datentyp zu definieren, gehen Sie folgendermaßen vor.

1. Klicken Sie auf das Register **Schema** um zur Schema/WSLD-Ansicht zurückzukehren.
2. Klicken Sie auf `Name`, um das Element auszuwählen.
3. Klicken Sie in der Details-Eingabehilfe auf das Listenfeld `type` in der mittleren Eingabehilfe und wählen Sie `xs:string` aus der Auswahlliste.



Beachten Sie, dass sowohl `minOcc` als auch `maxOcc` den Wert 1 hat, d.h. dieses Element hat nur eine Instanz.

In der Textdarstellung sieht das Element `Name` folgendermaßen aus:

```
<xs:element name="Name" type="xs:string"/>
```

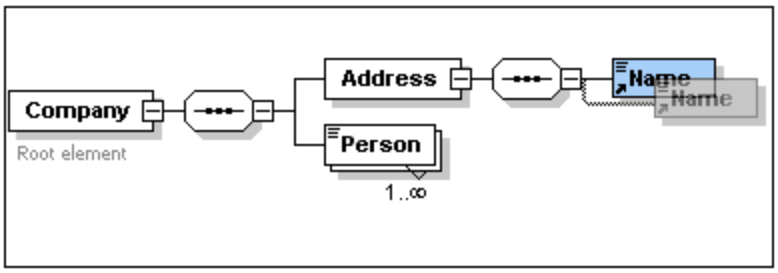
Anmerkung: Ein einfaches Type-Element kann jedem beliebigen von mehreren XML-Schema-Datentypen angehören. In all diesen Fällen erscheint das Symbol für Textinhalt nur im Elementfeld.

2.4 Hinzufügen von Elementen mit Drag & Drop

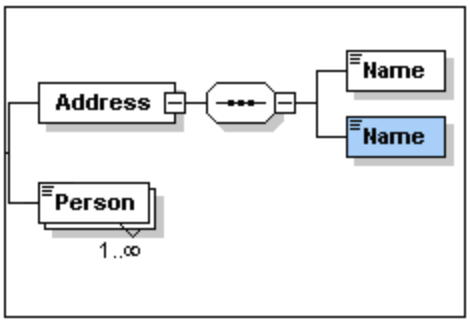
Sie haben Elemente bisher über das Kontextmenü hinzugefügt, das erscheint, wenn Sie mit der rechten Maustaste auf ein Element oder einen Kompositor klicken. Schneller lassen sich Elemente aber mit Drag & Drop erstellen. In diesem Abschnitt lernen Sie, wie man mit Drag & Drop mehr Elemente zur Definition des Elements Address hinzufügt und so diese Definition fertig stellt.

So stellen Sie de Definition des Elements Address mit Hilfe von Drag & Drop fertig:

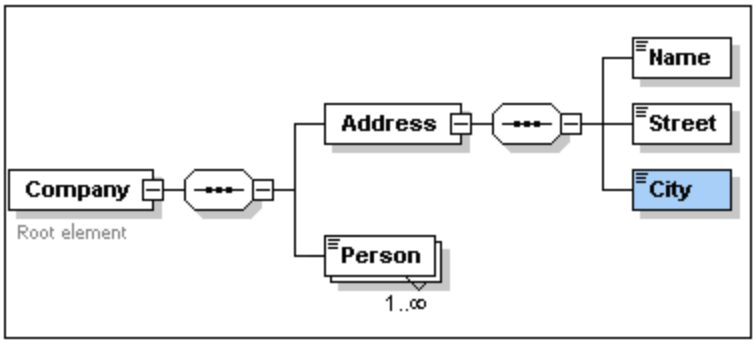
1. Klicken Sie auf das Element `Name` des Elements `Address`, halten Sie die Maustaste sowie die Strg-Taste gedrückt und ziehen Sie die Maus leicht nach unten. Die Markierung mit einem +Zeichen erscheint. Hierdurch wird angezeigt, dass das Element kopiert werden kann. Es erscheint eine Kopie des Elements sowie eine Konnektorlinie, sodass Sie sehen, wo das Element erstellt wird.



2. Lassen Sie nun die Maustaste und die STRG-Taste los um das neue Element in der `Address`-Sequenz anzulegen. Wurde das Element an der falschen Stelle angelegt, so ziehen Sie es einfach mit der Maus unter das Element `Name`.



3. Doppelklicken Sie in das Elementfeld und geben Sie `Street` ein, um den Namen des Elements zu ändern.
4. Wiederholen Sie obige Schritte um ein weiteres Element zu erzeugen. Benennen Sie dieses Feld um und geben Sie ihm den Namen `City`. Das Content Model sollte nun folgenden Inhalt haben:

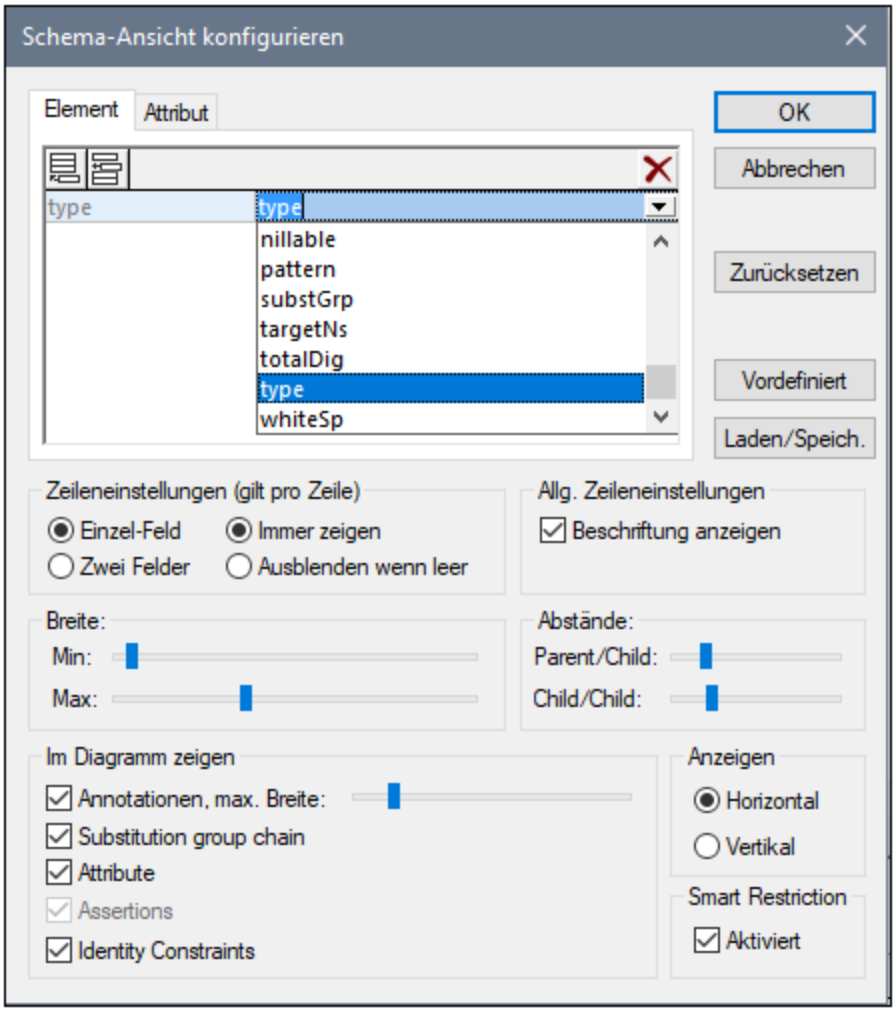


Das Element `Address` enthält eine Sequence mit einem Element `Name`, einem Element `Street` und einem Element `City`, und zwar in der obigen Reihenfolge.

2.5 Konfigurieren der Content Model-Ansicht

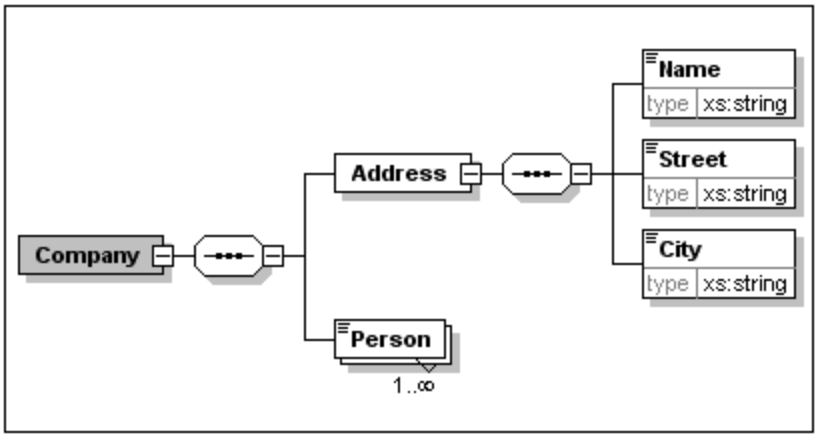
Sie können jetzt das Aussehen der Content Model-Ansicht konfigurieren. Wir wollen die Content Model-Ansicht so konfigurieren, dass der Typ eines Elements im jeweiligen Feld für das Element angezeigt wird. Gehen Sie dabei folgendermaßen vor:

1. Wählen Sie die Content Model-Ansicht einer Komponente aus (klicken Sie auf das Symbol Content Model-Ansicht).
2. Wenn in der Content Model-Ansicht die Menüoption **Schema-Design | Ansicht konfigurieren** aktiviert ist, wählen Sie den Befehl aus, um das Dialogfeld "Schema-Ansicht konfigurieren" aufzurufen (*Abbildung unten*).

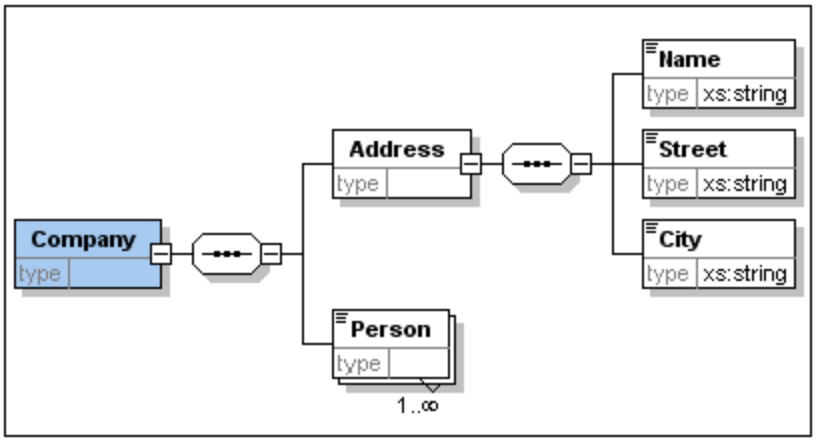


3. Klicken Sie auf dem Register *Element* auf das Symbol **Anhängen** und wählen Sie *type* (*siehe Abbildung*), um diese Eigenschaftsbeschreibungszeilen zu Elementfeldern hinzuzufügen.
4. Wählen Sie im Bereich *Zeileneinstellungen (Gilt pro Zeile)* die Option *Ausblenden wenn leer*, damit die Beschreibung des Datentyps im Elementfeld ausgeblendet wird, wenn das Element keinen Datentyp hat (z.B. wenn das Element ein ComplexType ist). Beachten Sie in der Abbildung unten, dass die Beschreibungszeile für den Typ bei den Elementen Name, Street und City aufscheint, da es sich dabei um SimpleTypes vom Typ xs:string handelt, nicht aber bei den ComplexType-Elementen, da die Option "Ausblenden wenn leer" aktiv ist.

 © 2019-2025 Altova GmbH



5. Aktivieren Sie im Bereich *Zeileneinstellungen (Gilt pro Zeile)* die Option *Immer zeigen*.
6. Klicken Sie auf **OK**, um die Änderungen zu speichern.



Beachten Sie, dass die Beschreibungszeile für den Datentyp immer angezeigt wird - selbst in Elementfeldern von ComplexTypes, in denen keine Werte in dieser Zeile angezeigt werden.

Beachten Sie die folgenden Punkte:

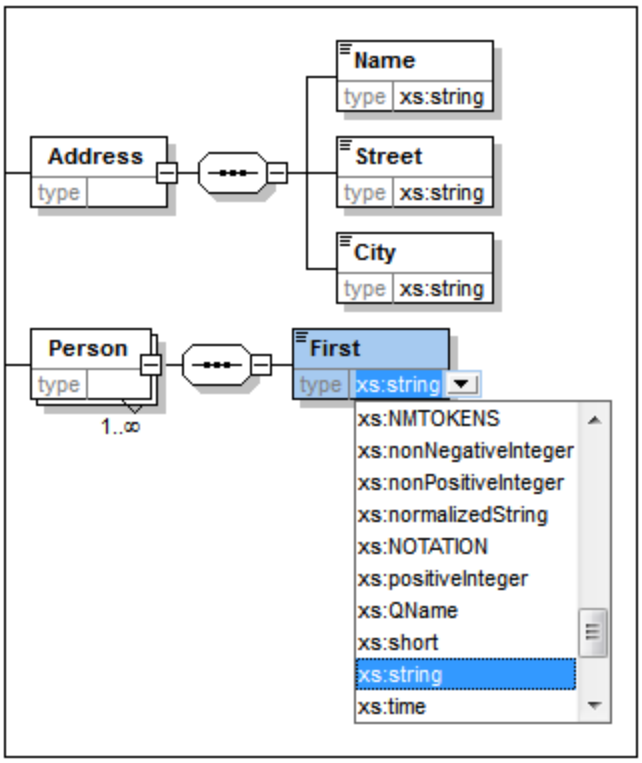
- Die Eigenschaftsbeschreibungszeilen können editiert werden. Werte, die Sie dort eingeben, werden Teil der Elementdefinition.
- Die Einstellungen, die Sie im Dialogfeld "Schema-Ansicht konfigurieren" vornehmen, gelten für die Ausgabe der Schema-Dokumentation ebenso wie für die Druckausgabe.

2.6 Fertigstellen des Schemas

Nachdem Sie nun den Inhalt des Elements **`Address`** definiert haben, müssen Sie den Inhalt des Elements `Person` definieren. Das Element **`Person`** soll die folgenden Child-Elemente vom Typ SimpleType enthalten: **`First`**, **`Last`**, **`Title`**, **`PhoneExt`** und **`Email`**. Alle diese Elemente müssen mit Ausnahme des Elements **`Title`** (das optional ist) obligatorisch sein, und alle Elemente müssen in der von Ihnen angegebenen Reihenfolge aufscheinen. Alle Elemente sollen vom Typ **`xs:string`** sein, mit Ausnahme von **`PhoneExt`**, das vom Typ **`xs:integer`** sein muss und maximal 2 Stellen haben darf.

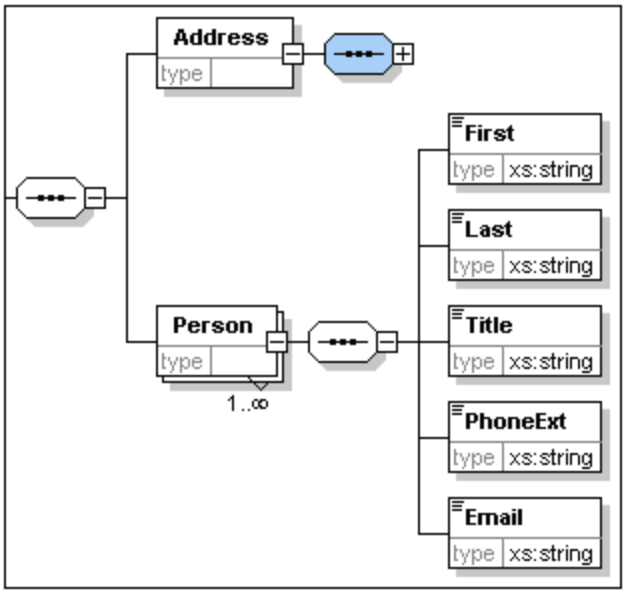
Um das Content Model für `Person` zu erstellen, gehen Sie folgendermaßen vor:

1. Rechtsklicken Sie auf das Element `Person`, um das Kontextmenü aufzurufen und wählen Sie **Child hinzufügen | Sequence**. Der Sequence-Kompositor wird eingefügt.
2. Klicken Sie mit der rechten Maustaste auf den Sequence-Kompositor und wählen Sie **Child hinzufügen | Element**.
3. Nennen Sie das Element `First` und betätigen Sie die **Tabulatortaste** Ihrer Tastatur. Sie gelangen automatisch in das Feld type.



4. Wählen Sie den Eintrag `xs:string` aus der Auswahlliste oder geben Sie ihn in das Feld *type* ein.
5. Erzeugen Sie weitere vier Elemente mittels der Drag & Drop Methode. Benennen Sie diese der Reihe nach `Last`, `Title`, `PhoneExt` und `Email`.

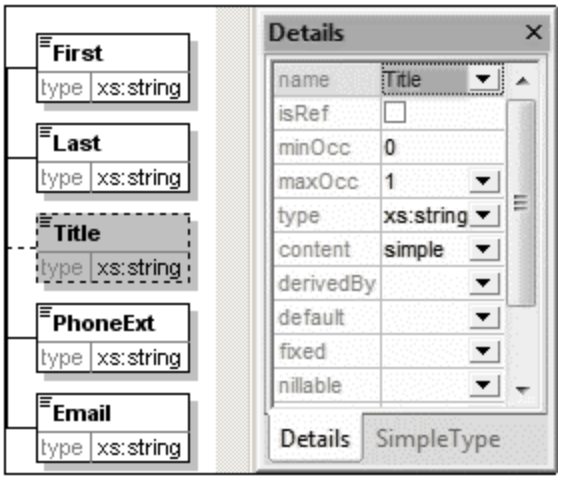
 © 2019-2025 Altova GmbH



Anmerkung: Sie können mehrere Elemente auswählen, indem Sie die **Strg**-Taste gedrückt halten, während Sie die Elemente anklicken. Auf diese Art können Sie mehrere Elemente gleichzeitig kopieren.

Ein Element optional machen

Rechtsklicken Sie auf das Element `Title` und wählen Sie **Optional** im Kontextmenü. Die Umrahmung des Elementes wechselt von durchgezogener Linie auf gestrichelt. Hierdurch wird angezeigt, dass dieses Element optional ist.

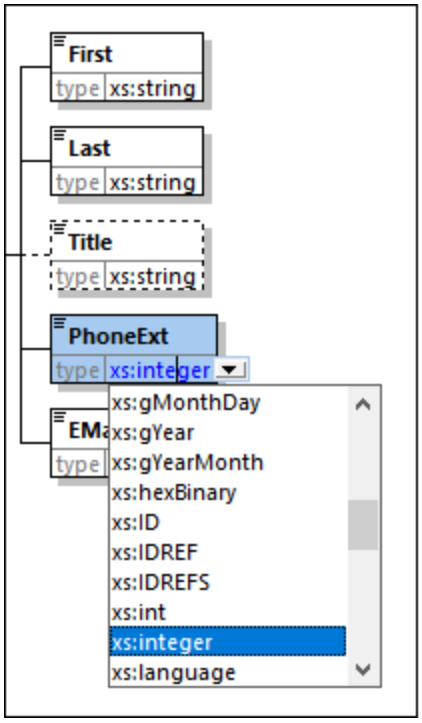


Sie sehen in der Details-Eingabehilfe, dass `minOcc=0` und `maxOcc=1` ist, d.h. das Element ist optional. Alternativ dazu können Sie das Element auch über das Kontextmenü optional machen und für `minOcc=0` einstellen, um das Element zu einem optionalen zu machen.

Den Inhalt eines Elements einschränken

Um das Element `PhoneExt` als Typ **`xs:integer`** mit maximal 2 Stellen zu definieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in das Feld type unter dem Element `PhoneExt` und wählen Sie den Eintrag **`xs:integer`** aus der Auswahlliste oder geben Sie ihn ein (*siehe Abbildung unten*).



2. Änderungen werden in der Facets-Eingabehilfe angezeigt. Doppelklicken Sie in der Facets-Eingabehilfe in das Feld `maxIncl`, geben Sie `99` ein und bestätigen Sie dies durch Drücken der **Eingabetaste**. Hierdurch wird definiert, dass Durchwahlnummern Werte von `0` bis inkl. `99` haben können.
3. Wählen Sie die Menüoption **Datei | Speichern** zum Speichern der Änderungen im Schema.

Anmerkung:

- Beim Auswählen eines XML-Schema-Datentyps vom Typ simpleType (z.B. `xs:string` oder `xs:date`) wird das Content Model automatisch in der Details-Eingabehilfe auf simple (`content = simple`) gesetzt.
- Beim Hinzufügen eines Kompositors zu einem Element (selection, choice oder all), wird das Inhaltsmodell automatisch auf complex (`content = complex`) gesetzt.
- Das verwendete Schema ist als `AddressFirst.xsd` im Ordner Tutorial Ihres XMLSpy-Anwendungsordners abgelegt.

 © 2019-2025 Altova GmbH

3 XML-Schemas: komplexere Vorgänge

Nachdem Sie nun ein einfaches Schema erstellt haben, können wir uns den komplizierteren Aspekten der Schema-Entwicklung widmen.

Ziel

In diesem Abschnitt werden folgende Punkte behandelt:

- Arbeiten mit complexTypes und simpleTypes 31, die später als die Typen von Schema-Elementen verwendet werden können.
- Erstellen von globalen Elementen 39 und Referenzieren dieser Elemente von anderen Elementen aus
- Erstellen von Attributen 41 und deren Eigenschaften, einschließlich Enumerationswerten.

Wir beginnen diesen Abschnitt mit dem einfachen Schema AddressFirst.xsd, das Sie im ersten Teil dieses Tutorials erstellt haben.

In diesem Abschnitt verwendete Befehle

In diesem Abschnitt des Tutorials wird ausschließlich die Schema-Ansicht verwendet. Dabei kommen die folgenden Befehle zum Einsatz:

	Diagramm anzeigen (oder Content Model-Ansicht anzeigen). Mit diesem Symbol, das in der Schema-Übersicht links von allen globalen Komponenten angezeigt wird, sehen Sie das Content Model der dazugehörigen globalen Komponente.
	Globale Elemente anzeigen. Dieses Symbol befindet sich in der linken oberen Ecke des Content Model. Mit diesem Befehl gelangen Sie zurück zur Schema-Übersicht, in der alle globalen Komponenten angezeigt werden.
	Anhängen. Mit diesem Symbol (in der linken oberen Ecke der Schema-Übersicht) können Sie eine globale Komponente hinzufügen.

3.1 Arbeiten mit ComplexTypes und SimpleTypes

Nachdem Sie ein Element definiert haben, möchten Sie es möglicherweise auch an einer anderen Stelle im Schema verwenden. Das könnte z.B. der Fall sein, wenn Sie ein Content Model für Adressen in den USA und Großbritannien definieren. Die beiden Adressformate haben einige Komponenten gemeinsam, wie z.B. die Komponenten Straße (street) und Stadt (city). Andere Komponenten unterscheiden sich wiederum voneinander. Daher empfiehlt es sich, in jedem Adress-Content Model (US und UK) die gemeinsamen Komponenten wiederzuverwenden und die beiden Content Models dann durch die länderspezifschen Komponenten zu ergänzen (wie z.B. ZIP in den USA und "Postal code" in UK). Wir können die gemeinsamen Komponenten zu diesem Zweck als globale complexTypes (oder alternativ dazu jede gemeinsame Komponente als globales Element) erstellen und den globalen complexTyp (oder die globalen Elemente) im Content Model der einzelnen Adresstypen wiederverwenden.

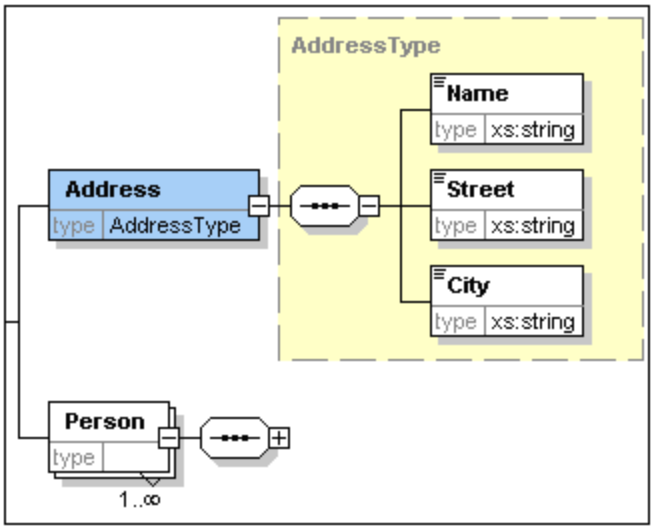
In diesem Abschnitt werden Sie mit globalen ComplexTypes arbeiten. Ein complexType ist eine Typdefinition für Elemente, die andere Elemente und/oder Attribute enthält. Wir werden zuerst einen ComplexType auf globaler Ebene erstellen, diesen anschließend in ein Content Model importieren und dann erweitern. Mehr über globale Elemente erfahren Sie weiter hinten in diesem Tutorial.

Erstellen eines globalen complex type

Das Basiselement Address, das wir zuvor definiert haben (und das die Elemente Name, Street und City enthält), kann in verschiedenen Adressformaten wiederverwendet werden. Wir wollen daher diese Elementdefinition als wiederverwendbaren ComplexType erstellen. Gehen Sie dazu folgendermaßen vor:

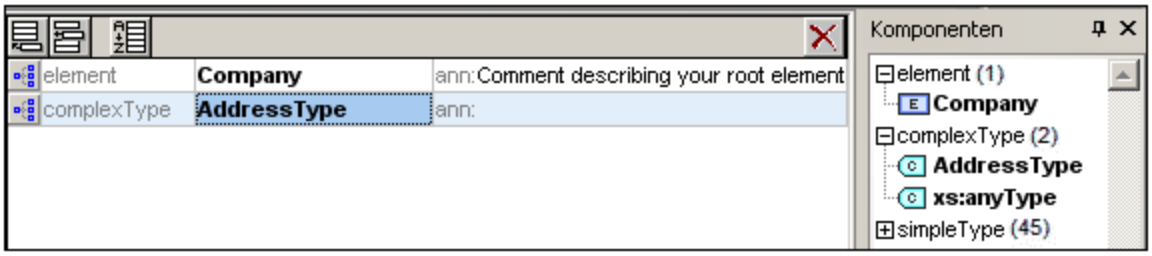
So erstellen Sie einen globales Element vom Typ Complex Type:

1. Klicken Sie in der Content Model-Ansicht mit der rechten Maustaste auf das Element **`Address`**.
2. Wählen Sie im Kontextmenü nun die Option **Als global definieren | Complex Type**. Es wird ein globaler Complex Type mit dem Namen **`AddressType`** erstellt, und dem Element **`Address`** im Content Model **`Company`** wird dieser Typ zugewiesen. Der Inhalt des Elements `Address` ist das Content Model von **`AddressType`**, das in einem gelb hinterlegten Kasten angezeigt wird. Beachten Sie, dass der Datentyp des Elements **`Address`** nun **`AddressType`** ist.

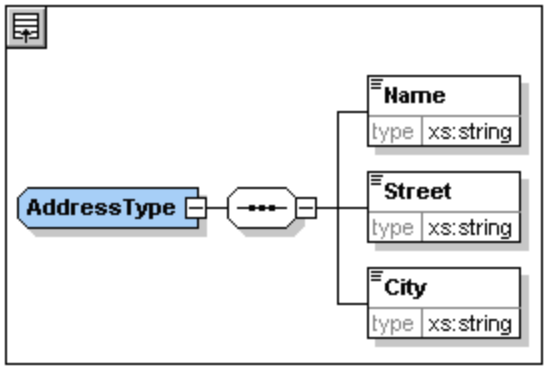


3. Klicken Sie auf das Symbol "Globale Elemente anzeigen" . Damit kehren Sie zur Schema-Übersicht zurück, in der nun alle globalen Komponenten angezeigt werden.
4. Klicken Sie in der Komponenten-Eingabehilfe auf das Erweiterungssymbol für die **`Element`**- und **`ComplexType`**-Einträge, um die jeweiligen Schema-Konstrukte zu sehen.In der Schema-Übersicht werden nun zwei globale Komponenten angezeigt: das Element **`Company`** und der ComplexType **`AddressType`**. Außerdem wird in der Komponenten-Eingabehilfe der ComplexType **`AddressType`** angezeigt.

 © 2019-2025 Altova GmbH



5. Klicken Sie auf das Symbol "Content Model-Ansicht" von **`AddressType`**, um dessen Content Model anzuzeigen (siehe Abbildung unten). Beachten Sie die Form des ComplexType-Behälters.

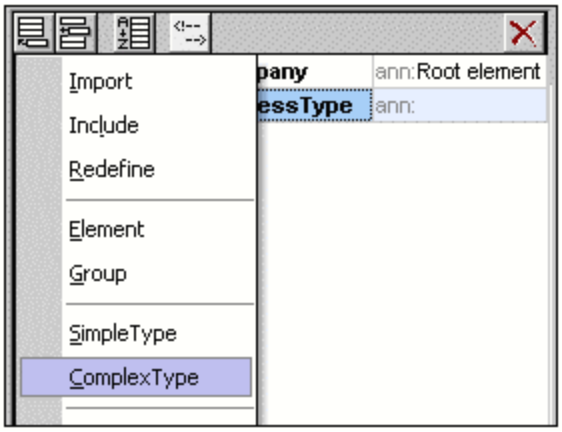


6. Klicken Sie auf das Symbol "Globale Elemente anzeigen", um zur Schema-Übersicht zurückzukehren.

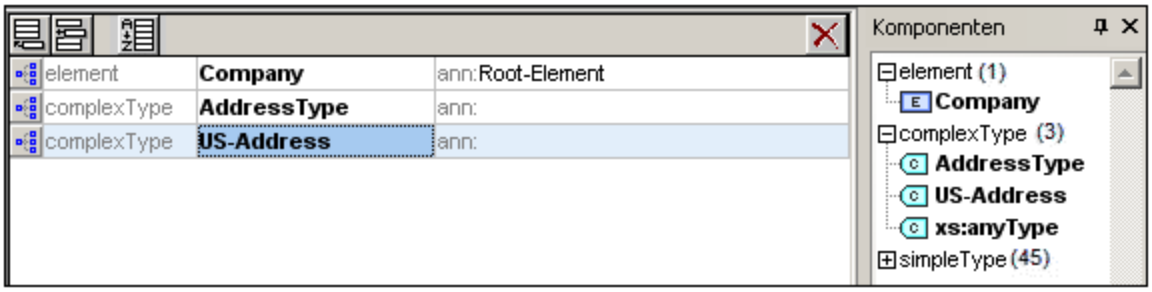
Erweitern einer "complex type" Definition

Nun werden wir die globale Komponente **`AddressType`** verwenden, um zwei Arten von länderspezifischen Adressen zu erstellen. Hierzu wird zuerst ein neuer complexType erzeugt, welcher auf der Komponente **`AddressType`** basiert. Anschließend wird diese Definition erweitert. Gehen Sie dazu folgendermaßen vor:

1. Wechseln Sie zur Schema-Übersicht (Wenn Sie sich in der Content Model-Ansicht befinden, klicken Sie auf "Globale Elemente anzeigen").
2. Klicken Sie auf das Symbol **Anhängen** (oben links im Komponentenfenster). Daraufhin wird das folgende Menü angezeigt:

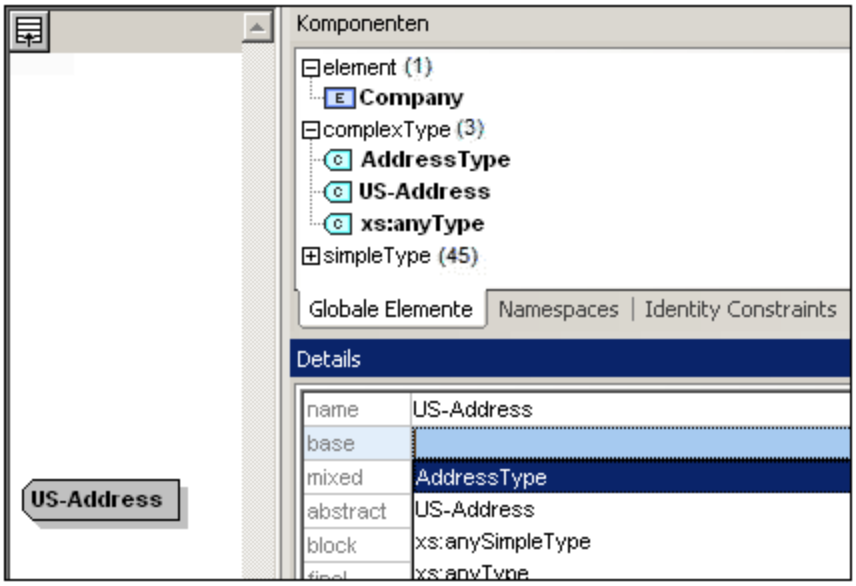


3. Wählen Sie im Menü **ComplexType** aus. Eine neue Leerzeile wird in der Komponentenliste angehängt, in der Sie den Namen der Komponente eingeben können.
4. Geben Sie **`US-Address`** ein und bestätigen Sie mit der **Eingabetaste**. (Sollten Sie den Bindestrich "-" beim Eingeben vergessen, und stattdessen ein Leerzeichen verwenden, wird der Name in Rot angezeigt. D.h., ein nicht erlaubtes Schriftzeichen wurde verwendet.)

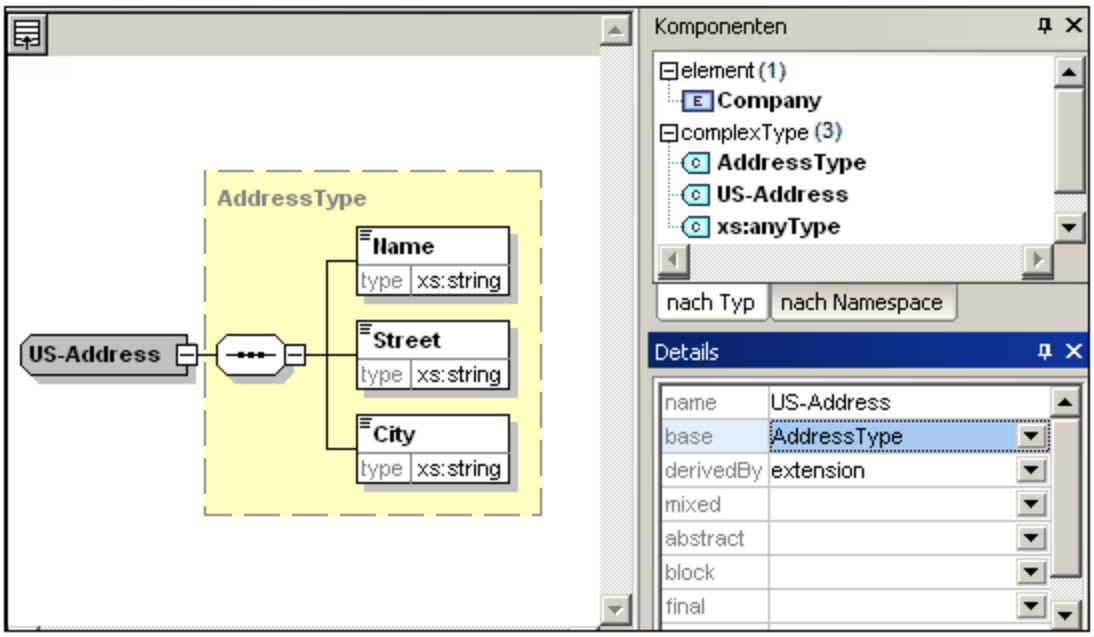


5. Klicken Sie auf das Symbol "Content Model-Ansicht" in der Zeile **`US-Address`**, um das Content Model des neuen ComplexType zu sehen. Das Content Model ist leer (*siehe Abbildung unten*).
6. Klicken Sie in der Details-Eingabehilfe auf das Listenfeld base und wählen Sie **`AddressType`**.

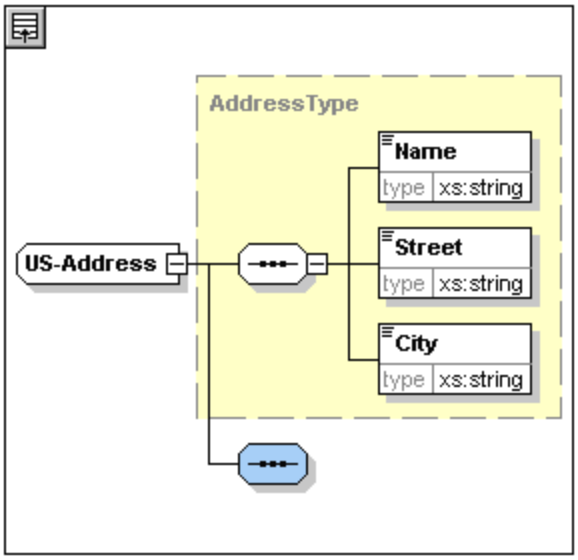
 © 2019-2025 Altova GmbH



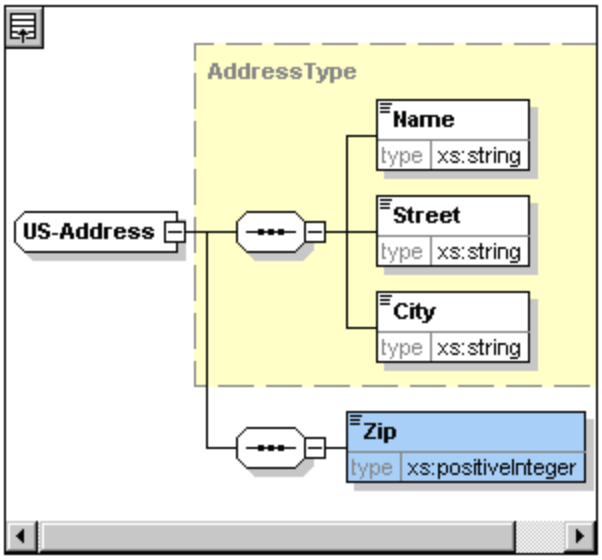
In der Content Model-Ansicht wird nun das Content Model **`AddressType`** als Content Model von US-Address (*siehe Abbildung unten*) angezeigt.



7. Wir können nun das Content Model des ComplexType **`US-Address`** um ein ZIP Code-Element erweitern. Klicken Sie dazu mit der rechten Maustaste auf das Element US-Address und wählen Sie **Child hinzufügen | Sequence**. Ein neuer Sequence-Kompositor wird außerhalb des Kastens AddressType (*Abbildung unten*) angezeigt; d.h., es handelt sich um die Erweiterung zu einem Element.



8. Klicken Sie mit der rechten Maustaste auf den neuen Sequence-Kompositor und wählen Sie **Child hinzufügen | Element**.
9. Benennen Sie das neu erstellte Element mit **`Zip`** und drücken Sie die **Tabulatortaste**. Daraufhin wird der Cursor in das Wertefeld der Typbeschreibungszeile gesetzt.
10. Wählen Sie in der Auswahlliste **`xs:positiveInteger`** aus und bestätigen Sie mit der **Eingabetaste**.



Sie haben nun einen ComplexType namens **`US-Address`**, der auf dem ComplexType `AddressType` basiert und diesen um einen ZIP Code (Postleitzahl) erweitert.

Globale SimpleTypes

Genau wie der ComplexType **`US-Address`** auf dem ComplexType **`AddressType`** basiert, kann ein Element auch auf einem SimpleType basieren. Der Vorteil ist derselbe wie bei globalen ComplexTypes: Der SimpleType kann wiederverwendet werden. Dazu muss der SimpleType zuerst als globales Element definiert werden. In diesem Tutorial werden wir ein Content Model für US-Bundesstaaten in einem SimpleType definieren. Dieser SimpleType wird als Basis für ein weiteres Element verwendet.

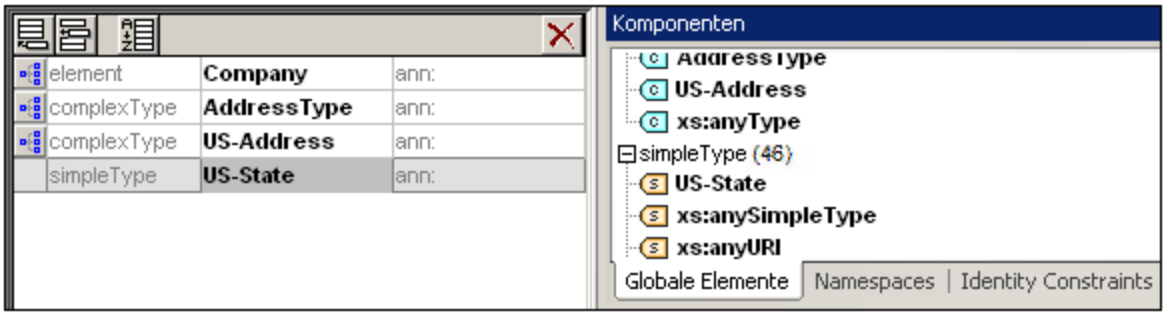
 © 2019-2025 Altova GmbH

Erstellen eines globalen SimpleType

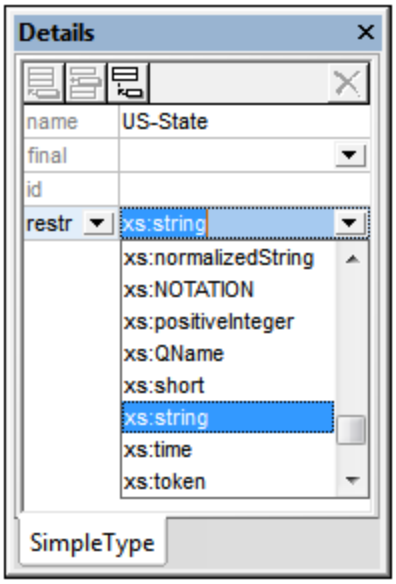
Um einen globalen SimpleType zu erstellen, müssen Sie einen neuen SimpleType an die Liste der globalen Komponenten anhängen, diesen benennen und seinen Datentyp definieren.

So erstellen Sie einen globalen SimpleType:

1. Wechseln Sie in die Schema-Übersicht (Klicken Sie auf "Globale Elemente anzeigen" , wenn Sie sich in der Content Model-Ansicht befinden).
2. Klicken Sie auf das Symbol **Anhängen** und wählen Sie im Kontextmenü **Simple Type**.
3. Geben Sie `US-State` als Elementname ein.
4. Bestätigen Sie mit der **Eingabetaste**. Der SimpleType `US-State` wird erstellt und in der Komponenten-Eingabehilfe in die Liste der SimpleTypes aufgenommen (Klicken Sie auf das Erweiterungssymbol des SimpleType-Eintrags, um es zu sehen).



5. Platzieren Sie in der Details-Eingabehilfe (*Abbildung unten*) den Cursor in das Wertefeld von **`restr`** und geben Sie **`xs:string`** ein oder wählen Sie **`xs:string`** in der Dropdown-Liste im Wertefeld **`restr`** aus.

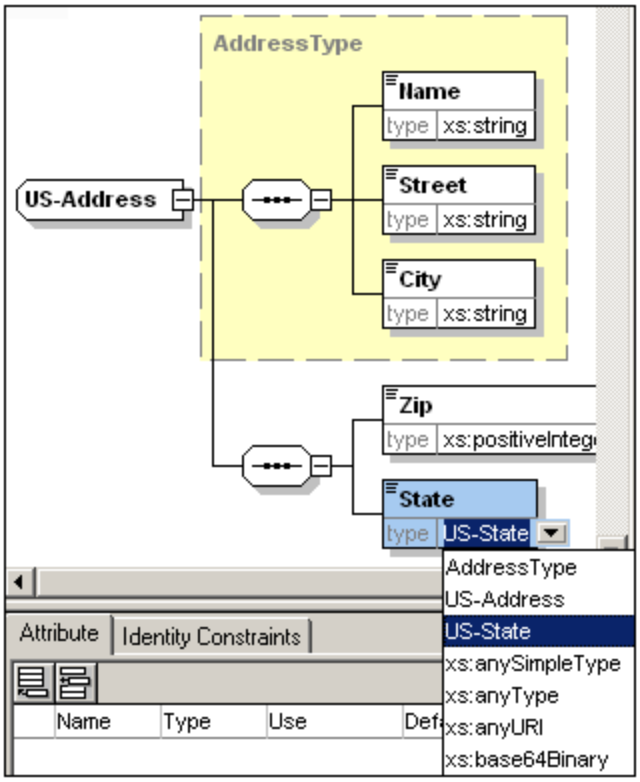


Damit wird ein SimpleType namens **`US-State`** vom Datentyp **`xs:string`** erstellt. Diese globale Komponente kann nun im Content Model von **`US-Address`** verwendet werden.

Verwendung eines globalen SimpleType in einem Content Model

Ein globaler SimpleType kann in einem Content Model dazu verwendet werden, den Typ einer Komponente zu definieren. Wir werden **`US-State`** verwenden, um im Content Model von **`US-Address`** ein Element namens **`State`** zu definieren. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie in der Schema-Übersicht auf das Symbol "Content Model-Ansicht" von **`US-Address`**.
2. Klicken Sie mit der rechten Maustaste auf den unteren Sequence-Kompositor und wählen Sie **Child hinzufügen | Element**.
3. Geben Sie `State` als Elementnamen ein.
4. Drücken Sie die **Tabulatortaste**, um den Cursor in das Wertefeld der Typbeschreibungszeile zu setzen.
5. Wählen Sie in der Dropdown-Liste den Eintrag **`US-State`** aus.



Das Element State basiert nun auf dem SimpleType **`US-State`**.

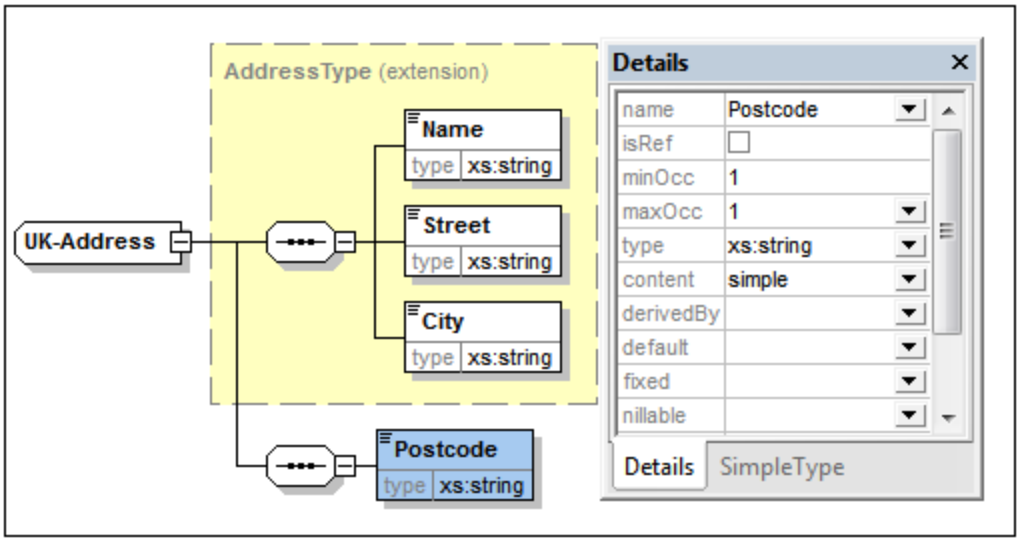
Erstellen eines zweiten ComplexType auf Basis von AddressType

Wir erstellen nun einen globalen ComplexType für Adressen in Großbritannien. Der ComplexType basiert auf dem Element `AddressType` und wird erweitert, bis er dem englischen Adressformat entspricht.

Gehen Sie folgendermaßen vor:

© 2019-2025 Altova GmbH

1. Erstellen Sie in der Schema-Übersicht auf Basis von `AddressType` einen globalen ComplexType mit dem Namen `UK-Address`. (`base=AddressType`).
2. Hängen Sie in der Content Model-Ansicht von `UK-Address` das neue Element `Postcode` an und weisen Sie ihm den Typ `xs:string` zu. bDas Content Model von UK-Address sollte nach erfolgreicher Erstellung folgendermaßen aussehen:

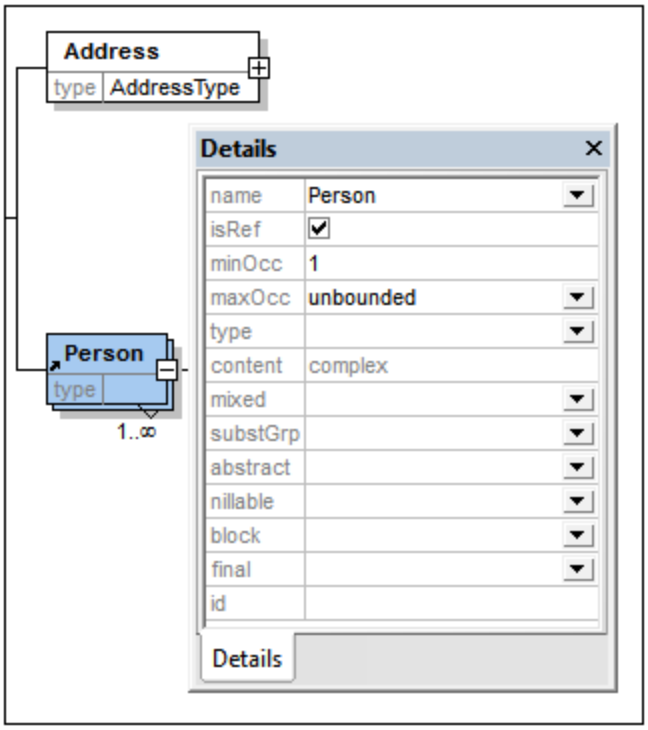


Anmerkung: In diesem Abschnitt haben Sie globale Simple und ComplexTypes erstellt und anschließend in Content Model-Definitionen verwendet. Der Vorteil von globalen Typen liegt darin, dass diese in mehreren Definitionen wiederverwendet werden können.

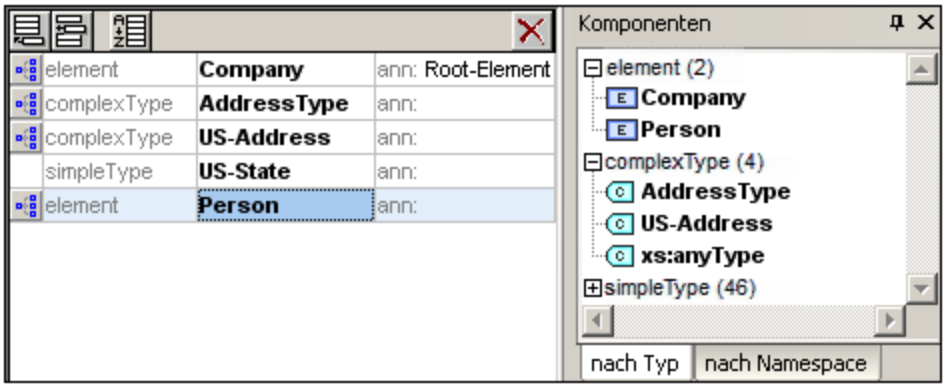
3.2 Referenzieren von globalen Elementen

In diesem Abschnitt werden wir das lokal definierte Element Person als globales Element definieren und es dann vom Element Company aus referenzieren.

1. Wechseln Sie in die Schema-Übersicht (Globale Elemente anzeigen).
2. Klicken Sie auf das Symbol "Diagramm anzeigen" in der Zeile Company.
3. Klicken Sie mit der rechten Maustaste auf das Element Person und wählen Sie Als global definieren | Element. Ein kleines "link"-Symbol wird in das Element Person eingefügt, das anzeigt, dass das Element das als global deklarierte Element Person referenziert. Das Kontrollkästchen `isRef` in der Details Eingabehilfe ist nun auf aktiv gesetzt.

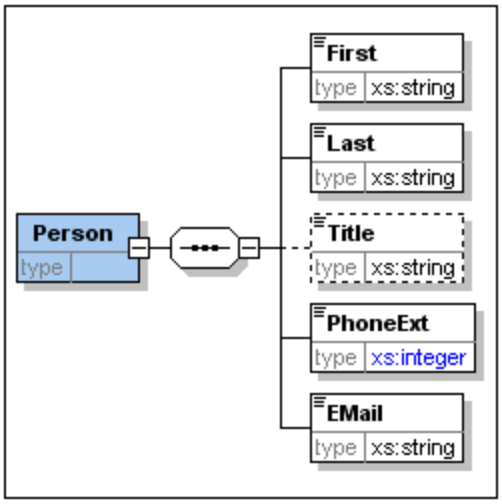


4. Klicken Sie auf das Symbol "Globale Elemente anzeigen" um zur Schema-Übersicht zurückzukehren. Das Element Person wird nun als globales Element aufgelistet und erscheint auch in der Komponenten-Eingabehilfe.



5. Doppelklicken Sie in der Komponenten-Eingabehilfe auf das Element Person, um das Content Model des globalen Elements Person anzuzeigen.

 © 2019-2025 Altova GmbH



Bitte beachten Sie, dass das Kästchen für das globale Element kein Link-Symbol hat, da es sich hierbei um das referenzierte Element und nicht das referenzierende Element handelt. Nur das referenzierende Element ist mit einem Link-Symbol gekennzeichnet.

Anmerkung:

- Ein Element, das ein globales Element referenziert, muss denselben Namen haben, wie das globale Element, das es referenziert.
- Globale Deklarationen geben keine Auskunft über die Positionierung der Elemente innerhalb des XML-Dokumentes. Sie bestimmen nur den Inhalt des jeweiligen Elementes. Nur wenn eine globale Deklaration von einer anderen Komponente aus referenziert wird, wird ihre Position im XML-Dokument definiert.

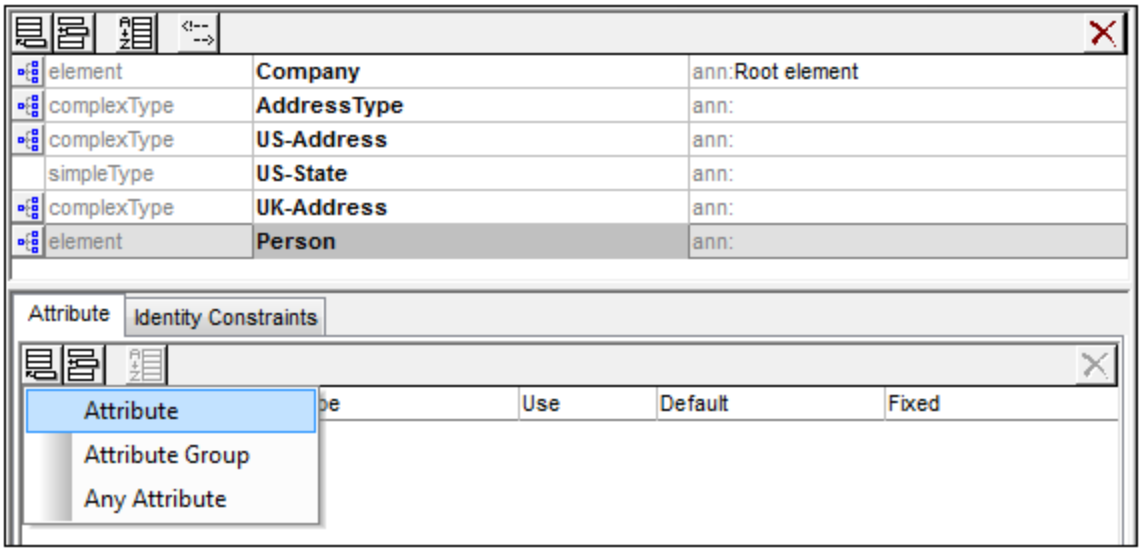
Ein global deklariertes Element kann an mehreren Stellen wiederverwendet werden. Es unterscheidet sich insofern von einem global deklarierten ComplexType, als sein Content Model nicht geändert werden kann, ohne dass auch das globale Element selbst geändert wird. Wenn Sie das Content Model eines Elements ändern, das ein globales Element referenziert, so wird auch das Content Model des globalen Elements geändert und damit auch das Content Model aller anderen Elemente, die das globale Element referenzieren.

3.3 Attribute und Attribut-Enumerationen

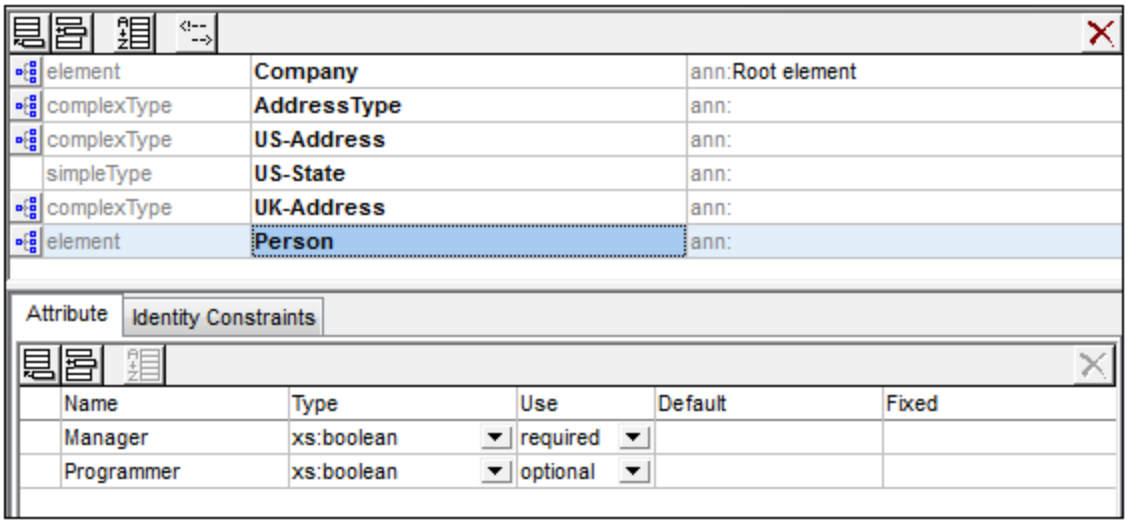
In diesem Abschnitt lernen Sie, wie man Attribute und Enumerationen für Attribute erstellt.

Definieren von Element-Attributen

1. Klicken Sie in der Schema-Übersicht auf das Element `Person` um es aktiv zu setzen.
2. Klicken Sie auf das Symbol "Anhängen" , (oben links im Fenster "Attribute/Identity Constraints", das den unteren Bereich des Schema-Übersichtsfensters einnimmt) und wählen Sie den Eintrag "Attribute".



3. Geben Sie Manager in das Namensfeld ein.
4. Wählen Sie in der Dropdown-Liste Type den Eintrag `xs:boolean`.
5. Wählen Sie in der Dropdown-Liste Use den Eintrag required.



6. Erstellen Sie nun auf dieselbe Art ein Attribut Programmer mit einem Type=`xs:boolean` und Use=`optional`.

Definieren von Enumerationen für Attribute

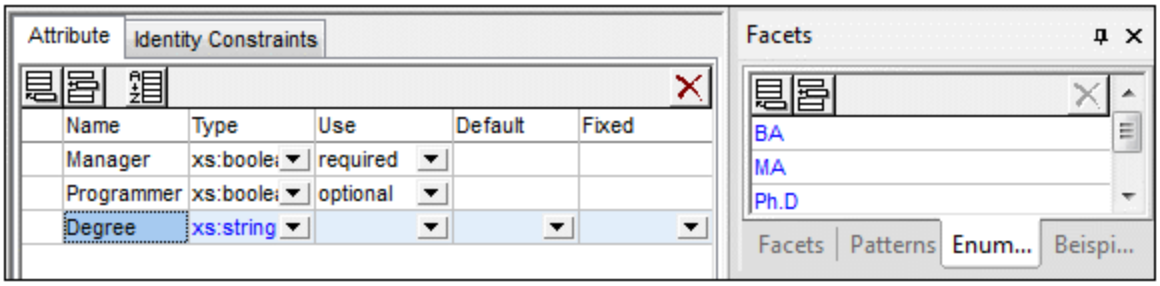
Enumerationen sind Werte, die bei einem bestimmten Attribut zulässig sind. Wenn der Wert des Attributs im XML-Instanzdokument nicht in einer im XML-Schema definierten Enumeration enthalten ist, ist das Dokument ungültig. Wir werden Enumerationen für das Attribut Degree des Elements `Person` erstellen.

Gehen Sie folgendermaßen vor:

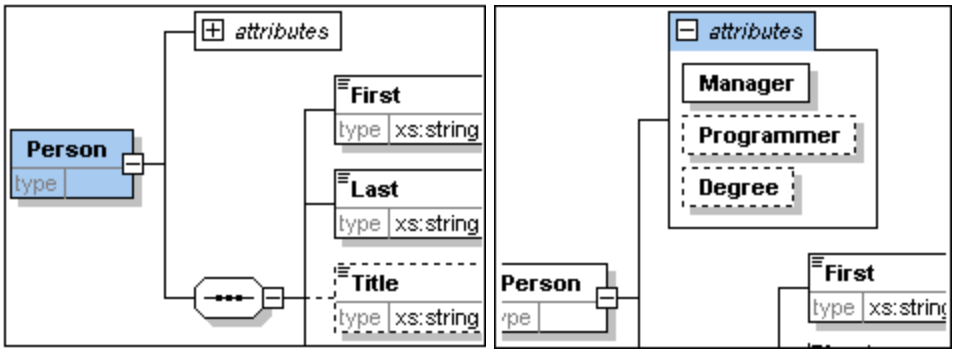
1. Klicken Sie in der Schemaübersicht auf das Element `Person`, um es zum aktiven zu machen.

 © 2019-2025 Altova GmbH

2. Klicken Sie auf das Anhängen-Symbol oben links im Attribut-Fenster und wählen Sie den Eintrag "Attribute".
3. Geben Sie Degree als Attributsnamen ein und wählen Sie `xs:string` als Attribut-Type.
4. Klicken Sie in der Facets-Eingabehilfe auf das Register Enumerations (siehe Abbildung unten).



5. Klicken Sie auf dem Register "Enumerations" auf das Anhängen-Symbol .
6. Geben Sie `BA` (Bachelor of Arts) ein und bestätigen Sie mit der Eingabetaste.
7. Mit derselben Methode geben Sie zwei weitere Werte in die Enumerations-Liste ein: `MA` und `PhD`.
8. Klicken Sie auf das Symbol "Content Model Ansicht" des Elements `Person`.



Die zuvor definierten Attribute werden in der Content Model-Ansicht angezeigt. Wenn Sie auf das Erweiterungssymbol klicken, werden alle für dieses Element definierten Attribute angezeigt. Sie den Anzeigemodus und das Register "Attribute" ein- und ausblenden, indem Sie die Menüoption **Schema-Design | Ansicht konfigurieren** wählen und das Kontrollkästchen für Attribute aktivieren bzw. deaktivieren.

9. Klicken Sie auf das Symbol "Globale Elemente anzeigen" , um zur Schema-Übersicht zurückzukehren.

Speichern des fertigen Schemas

Anmerkung: Bevor Sie Ihre Schema-Datei speichern, sollten Sie zuerst die vorhandene Datei `AddressLast.xsd`, die mit XMLSpy mitgeliefert wird, umbenennen (z.B. in `AddressLast_original.xsd`), um sie nicht zu überschreiben.

Speichern Sie das fertige Schema unter einem Namen Ihrer Wahl (**Datei | Speichern unter**). Wir empfehlen Ihnen, es unter dem Namen `AddressLast.xsd` zu speichern, da die XML-Datei, die Sie im nächsten Teil des Tutorials erstellen werden, auf dem Schema `AddressLast.xsd` basieren wird.

4 XML-Schemas: XMLSpy-Funktionen

Nach Fertigstellung Ihres XML-Schemas sollten Sie sich nun mit einigen Navigationskürzeln 44 vertraut machen und etwas über die Schemadokumentation 46, die Sie in XMLSpy generieren können, erfahren. Diese Themen werden in den folgenden Unterabschnitten erörtert.

In diesem Abschnitt verwendete Befehle

Wir verwenden in diesem Abschnitt des Tutorials ausschließlich die Schema-Ansicht. Dabei werden die folgenden Befehle verwendet:

	Diagramm anzeigen (oder Content Model-Ansicht anzeigen). Mit diesem Symbol, das in der Schema-Übersicht links von allen globalen Komponenten angezeigt wird, sehen Sie das Content Model der dazugehörigen globalen Komponente.

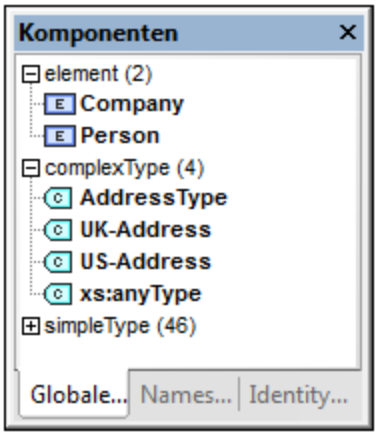
4.1 Schemanavigation

In diesem Abschnitt wird beschrieben, wie Sie effizient innerhalb der Schema-Ansicht navigieren können. Probieren Sie diese Navigationsverfahren am besten aus, um sich damit vertraut zu machen.

Anzeigen des Content Model einer globalen Komponente

Globale Komponenten, die ein Content Model haben können, sind ComplexTypes, Elemente und Elementgruppen. Die Content Model-Ansicht dieser Komponenten lässt sich auf folgende Arten aufrufen:

- Klicken Sie in der Schema-Übersicht auf das Symbol **Diagramm anzeigen** links vom Namen der Komponente.
- Doppelklicken Sie entweder in der Schema-Übersicht oder in der Content Model-Ansicht in der Komponenten-Eingabehilfe auf das Element, den ComplexType oder die Elementgruppe (*siehe Abbildung unten*). Daraufhin wird der Inhalt des Content Model dieser Komponente angezeigt.



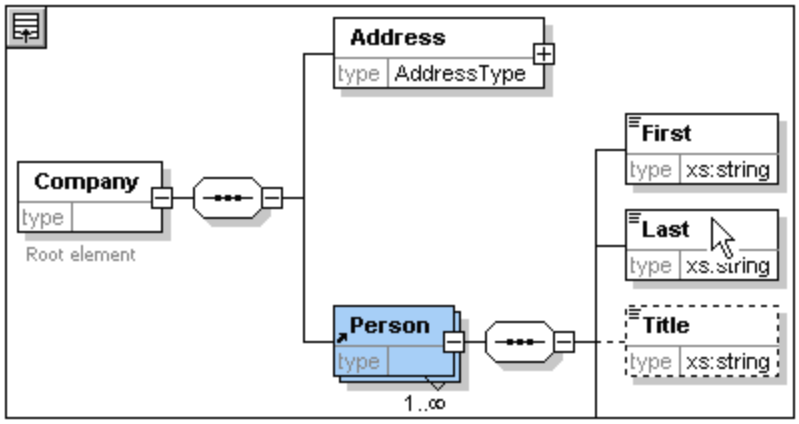
Wenn Sie in der Komponenten-Eingabehilfe auf eine der anderen globalen Komponenten (SimpleType, Attribut, Attributgruppe) doppelklicken, erscheint diese Komponente in der Schema-Übersicht markiert (da eine solche Komponente kein Content Model hätte).

 © 2019-2025 Altova GmbH

Das Anzeigen der Komponenten durch Doppelklick funktioniert in der Komponenten-Eingabehilfe sowohl auf dem Register "Globale Elemente" als auch auf dem Register "Namespaces".

Aufrufen der Definition eines globalen Elements von einem referenzierenden Element aus

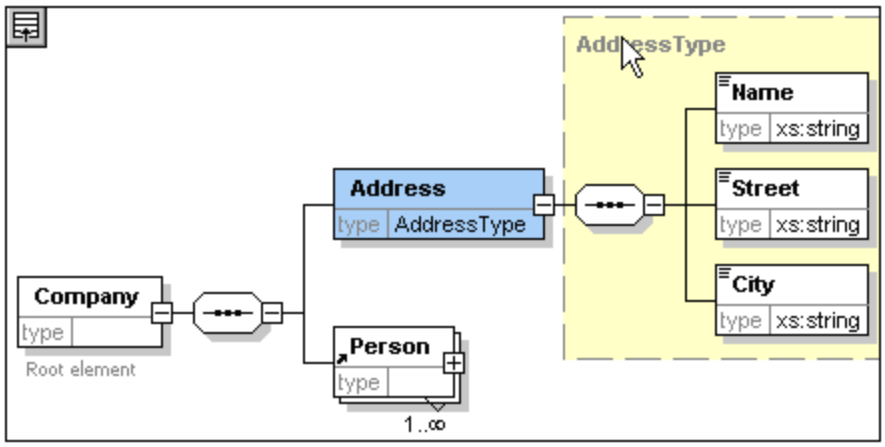
Wenn ein Content Model ein Element enthält, das ein globales Element referenziert, können Sie das Content Model dieses globalen Elements oder das seiner darin enthaltenen Komponenten direkt aufrufen, indem Sie die **Strg**-Taste gedrückt halten, während Sie auf das gewünschte Element doppelklicken.



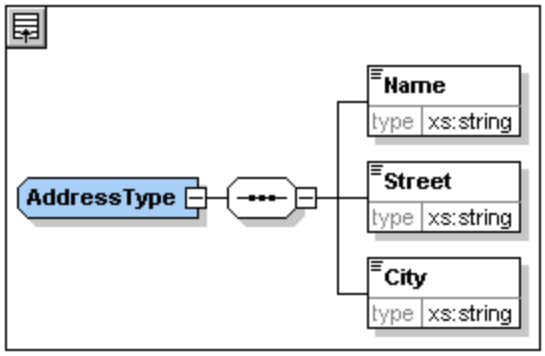
Wenn das Element `Last` markiert ist, werden alle seine Eigenschaften sofort in den entsprechenden Eingabehilfen und Informationsfenstern angezeigt.

Aufrufen der Definition eines ComplexType

ComplexTypes werden oft als Elementtyp in einem Content Model verwendet. Um die Definition eines ComplexType direkt von einem Content Model aus aufzurufen, doppelklicken Sie im gelben Kästchen auf den **Namen** des ComplexType (*siehe Mauszeiger in der Abbildung unten*).



Auf diese Art gelangen Sie zur Content Model-Ansicht des ComplexType.



Anmerkung: Wie auch bei referenzierten globalen Elementen können Sie direkt zu einem Element innerhalb der ComplexType-Definition gehen, indem Sie die **Strg**-Taste gedrückt halten und auf das gewünschte Element im Content Model, das den ComplexType enthält, doppelklicken.

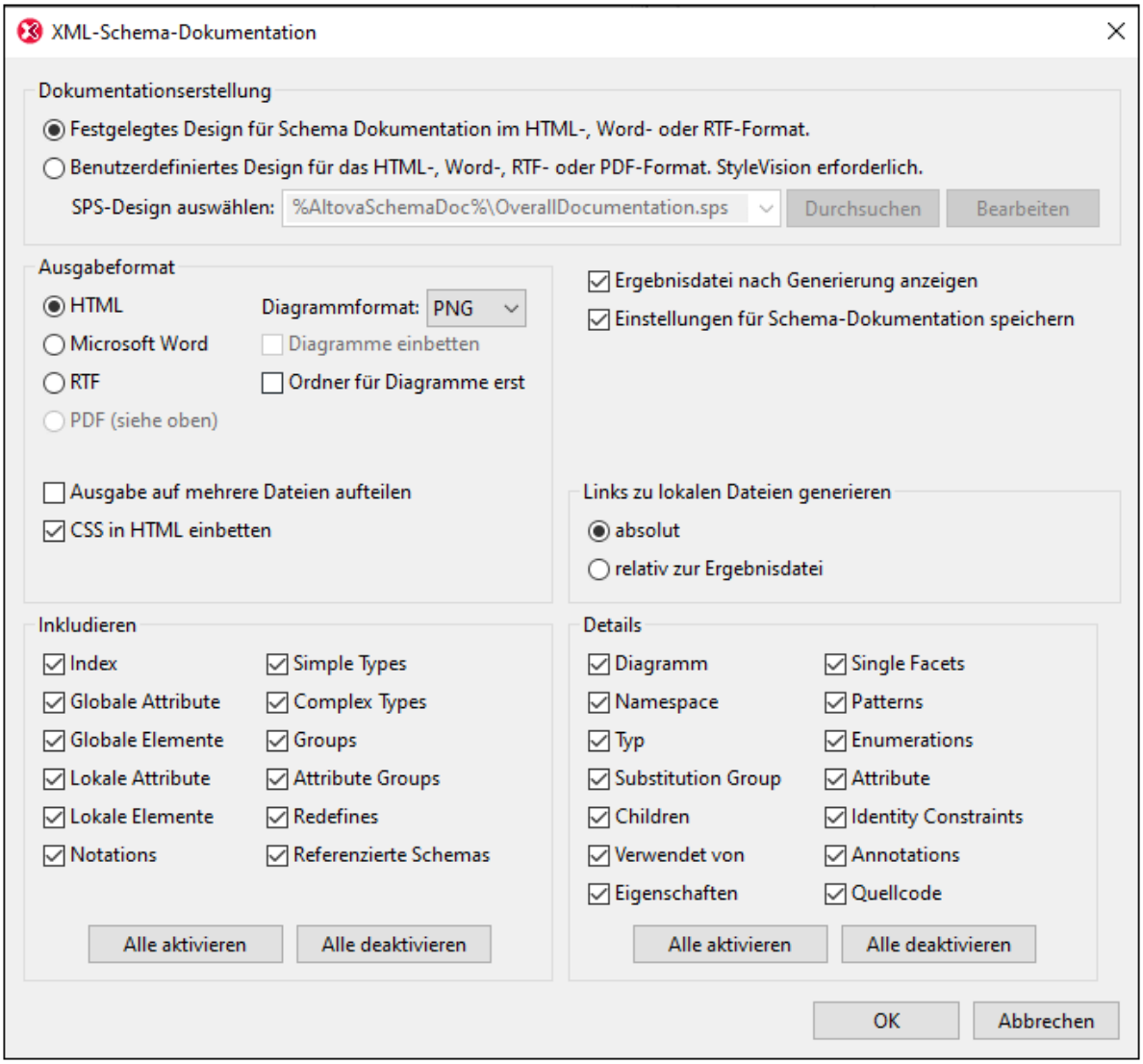
4.2 Schema-Dokumentation

XMLSpy verfügt über Funktionen zur Erstellung einer detaillierten Dokumentation von XML-Schemas im HTML- und MS Word-Format. Sie können die Komponenten und die Genauigkeit der Dokumentation auswählen. Miteinander in Zusammenhang stehende Komponenten werden im HTML- und im Word-Dokument durch Hyperlinks verbunden. Um Dokumentation im MS Word-Format zu generieren, muss MS Word auf Ihrem Computer (oder Netzwerk) installiert sein.

Sie werden in diesem Abschnitt Dokumentation für das XML-Schema `AddressLast.xsd` generieren. Gehen Sie dazu folgendermaßen vor:

1. Wählen Sie die Menüoption **Schema-Design | Dokumentation generieren,** um das Dialogfeld "Schema-Dokumentation" aufzurufen.

 © 2019-2025 Altova GmbH



2. Wählen Sie das Ausgabeformat HTML und bestätigen Sie mit OK.
3. Wählen Sie im Dialogfeld "Speichern unter" das Verzeichnis und geben Sie einen Namen für die HTML-Datei ein (z.B. `AddressLast.html`). Bestätigen Sie mit **Speichern**. Das HTML-Dokument wird in der Browser-Ansicht von XMLSpy angezeigt. Wenn Sie auf einen Link klicken, gelangen Sie zur damit verknüpften Komponente.

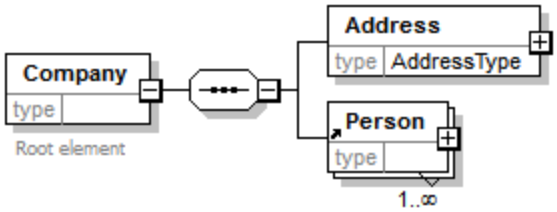
Schema **AddressLast.xsd**

schema location: C:\Users\alU\Documents\Altova\XMLSpy2013\Examples\Tutorial\AddressLast.xsd
attributeFormDefault: **unqualified**
elementFormDefault: **qualified**
targetNamespace: **http://my-company.com/namespace**

Elements Complex types Simple types
Company AddressType US-State
Person UK-Address
US-Address

element **Company**

diagram



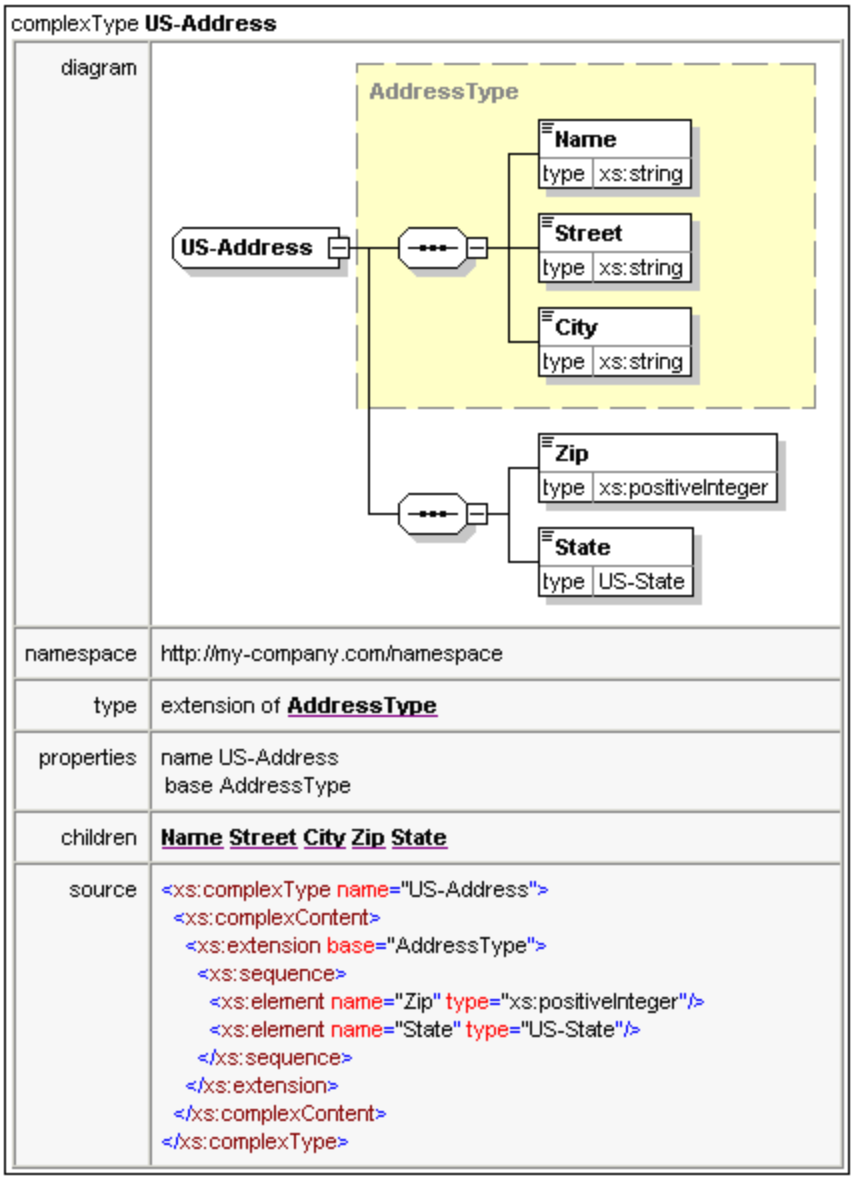
namespace	http://my-company.com/namespace
properties	content complex
children	Address Person
annotation	documentation Root element

source

```
<xs:element name="Company">
  <xs:annotation>
    <xs:documentation>Root element</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Address" type="AddressType"/>
      <xs:element ref="Person" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Das oben abgebildete Diagramm stellt die **erste Seite** der Schema-Dokumentation im HTML-Format dar. Falls Komponenten anderer Schemas miteinbezogen wurden, werden diese ebenfalls dokumentiert.

 © 2019-2025 Altova GmbH



Das oben abgebildete Diagramm stellt dar, wie ComplexTypes dokumentiert werden.

element **US-Address/Zip**

diagram	**Zip** type xs:positiveInteger
namespace	http://my-company.com/namespace
type	**xs:positiveInteger**
properties	name Zip isRef 0 content simple
source	<xs:element name="Zip" type="xs:positiveInteger"/>

element **US-Address/State**

diagram	**State** type US-State
namespace	http://my-company.com/namespace
type	**US-State**
properties	name State isRef 0 content simple
source	<xs:element name="State" type="US-State"/>

simpleType **US-State**

namespace	http://my-company.com/namespace
type	**xs:string**
properties	name US-State
used by	element **US-Address/State**
source	<xs:simpleType name="US-State"> <xs:restriction base="xs:string"/> </xs:simpleType>

Das oben abgebildete Diagramm zeigt an, wie Elemente und SimpleTypes dokumentiert werden.

Sie können nun die MS Word-Ausgabeoption ausprobieren. Das Word-Dokument wird in MS Word geöffnet. Um Hyperlinks im MS Word-Dokument zu verwenden, halten Sie die **Strg**-Taste gedrückt, während Sie auf den Link klicken.

© 2019-2025 Altova GmbH

5 XML Dokumente

In diesem Abschnitt lernen Sie, wie Sie ein neues XML-Dokument in XMLSpy erstellen und damit arbeiten. Außerdem erfahren Sie, wie Sie mit den verschiedenen intelligenten Bearbeitungsfunktionen von XMLSpy arbeiten.

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 Grid- und der Text-Ansicht mit Hilfe der intelligenten Eingabehilfen.
- Kopieren von XML-Daten von XMLSpy in ein MS Excel-Blatt, Hinzufügen neuer Daten in MS Excel und Zurückkopieren der geänderten Daten in XMLSpy. Diese Funktion steht in der Tabellenansicht der Grid-Ansicht zur Verfügung.
- Sortieren von XML-Elementen mit Hilfe der Sortierfunktion der Tabellenansicht.
- Validieren des XML-Dokuments.
- Ändern des Schemas, so dass dreistellige Telefondurchwahlnummern eingegeben werden können.

In diesem Abschnitt verwendete Befehle

In diesem Abschnitt des Tutorials arbeiten Sie hauptsächlich mit der Grid- und der Textansicht und in einem Abschnitt mit der Schema-Ansicht. Dabei werden die folgenden Befehle verwendet:

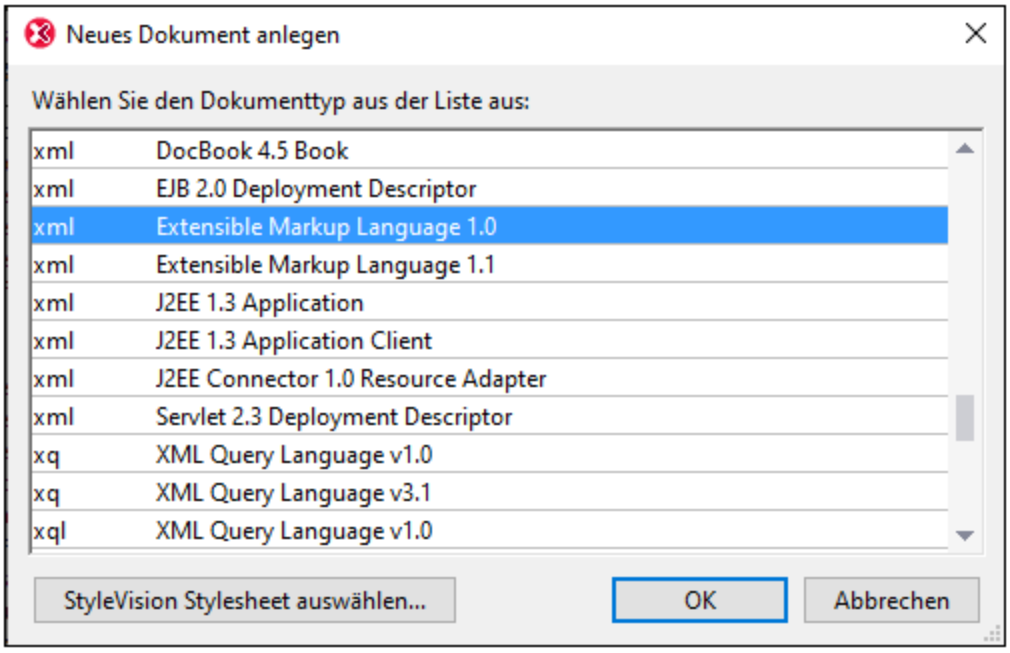
	Datei | Neu. Erstellt eine neue Art von XML-Datei.
	Ansicht | Text. Wechselt in die Textansicht.
	Ansicht | Grid-Ansicht. Wechselt in die Grid-Ansicht.
	XML | Tabelle | Als Tabelle anzeigen. Zeigt Mehrfachinstanzen eines einzelnen Elementtyps auf einer einzelnen hierarchischen Ebene als Tabelle an. Diese Ansicht des Elements nennt sich Tabellenansicht. Das Symbol dient zum Wechseln zwischen der Tabellen,- und der normalen Grid-Ansicht.
	F7. Überprüft, ob das Dokument wohlgeformt ist.
	F8. Validiert das XML-Dokument anhand des damit verknüpften Schemas bzw. der DTD.
	Öffnet die dazugehörige DTD oder XML-Schema-Datei.

5.1 Erstellen eines neuen XML-Dokuments

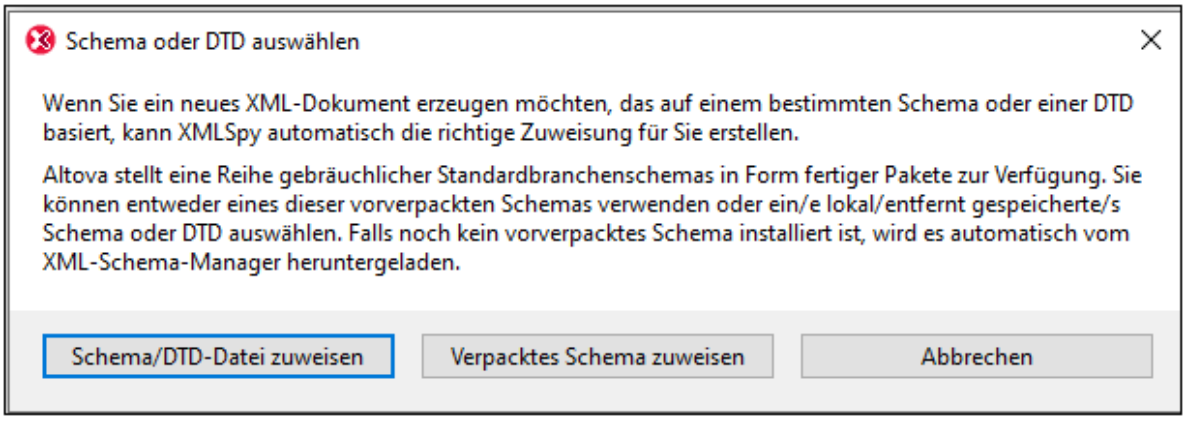
Beim Erstellen einer neuen XML-Datei in XMLSpy haben Sie die Option, dieser ein Schema (DTD oder XML-Schema) zugrunde zu legen oder auch nicht. In diesem Abschnitt werden wir eine neue Datei erstellen, die auf dem zuvor im Rahmen dieses Tutorials erzeugten Schema **`AddressLast.xsd`** basiert.

So erstellen Sie eine neue XML-Datei:

1. Wählen Sie die Menüoption **Datei | Neu**. Das Dialogfeld *Neues Dokument anlegen* wird angezeigt.



2. Wählen Sie *Extensible Markup Language 1.0* und bestätigen Sie mit **OK**. Daraufhin wird das Dialogfeld "Schema oder DTD auswählen" angezeigt.

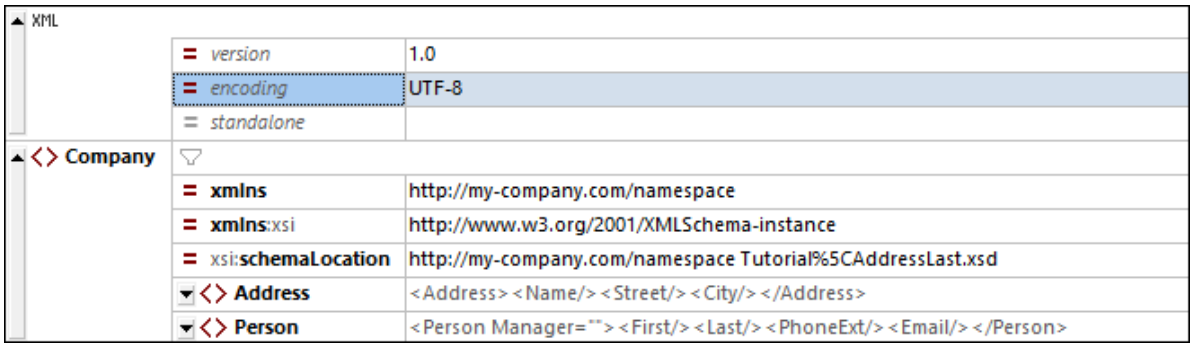


3. Klicken Sie auf **Schema/DTD-Datei zuweisen**.
4. Verwenden Sie im daraufhin angezeigten Dialogfeld entweder die Schaltfläche **Durchsuchen** oder die **Fenster**-Schaltfläche, um die Schema-Datei zu suchen. (Über die **Fenster**-Schaltfläche werden alle derzeit in XMLSpy geöffneten Dateien angezeigt). Wählen Sie **`AddressLast.xsd`** (Pfad siehe Tutorial

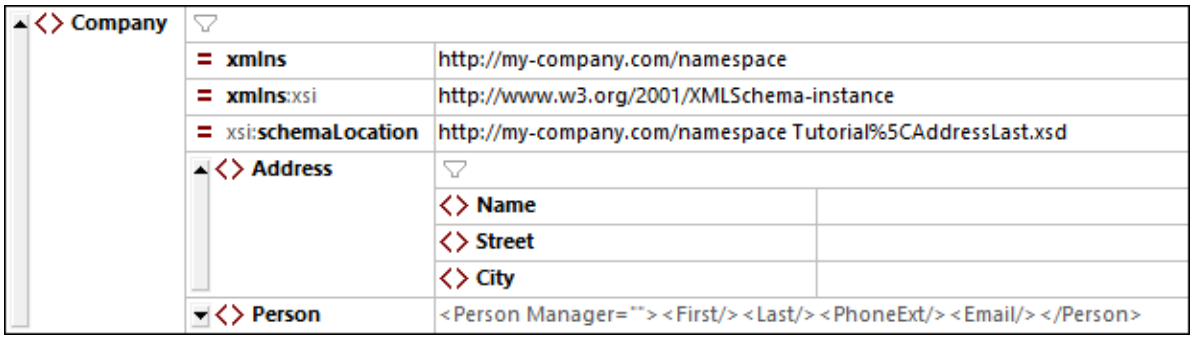
 © 2019-2025 Altova GmbH

Einführung[5]) 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.

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:



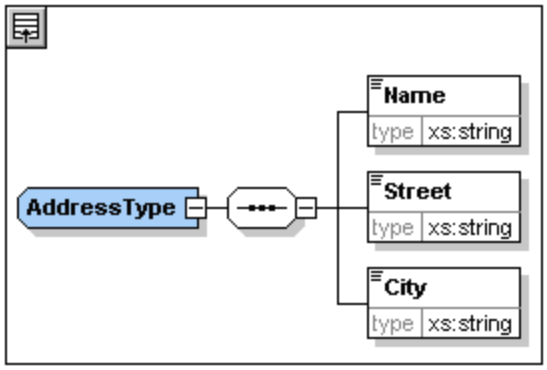
7. Klicken Sie auf das Symbol neben **`Address`**, um die Child-Elemente von **`Address`** anzuzeigen. Ihr Dokument sollte aussehen wie dieses:



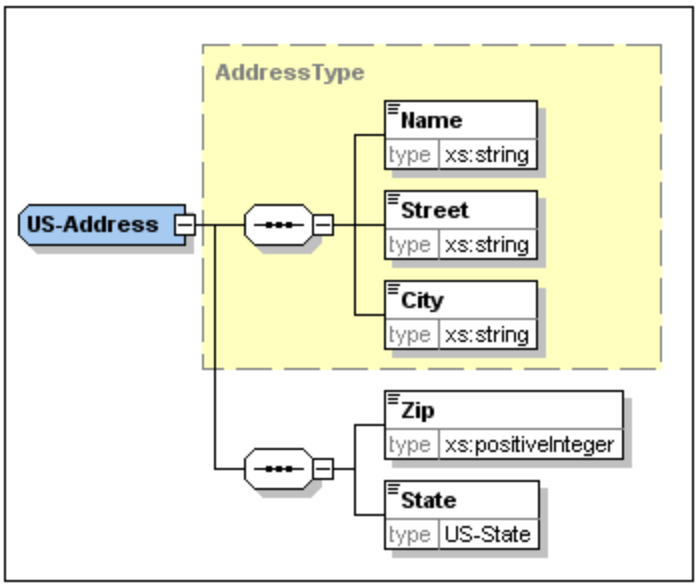
8. Wählen Sie die Menüoption **Datei | Speichern** und speichern Sie sie im Ordner **`Tutorial`**. Geben Sie Ihrem XML-Dokument einen passenden Namen (z.B. **`CompanyFirst.xml`**). Beachten Sie, dass es im Ordner **`Tutorial`** bereits eine fertige Datei mit dem Namen **`CompanyFirst.xml`** gibt, d.h. Sie sollten diese vorher umbenennen.

5.2 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 anstelle des generischen Adresstyps einen spezifischen US- oder UK-Adresstyp verwenden. 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**.

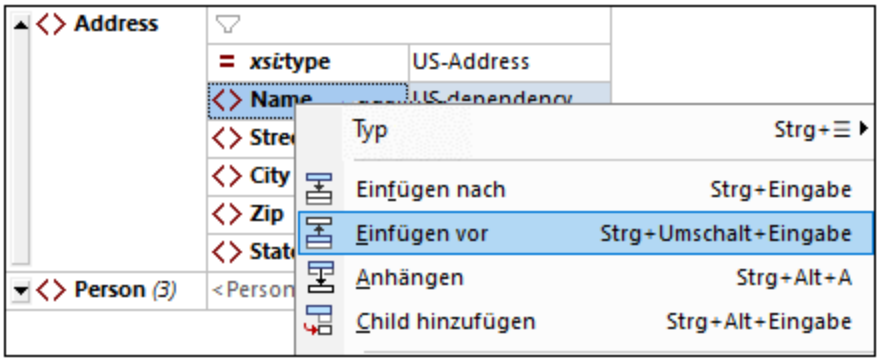


Um festzulegen, dass das Element **Address** im XML-Dokument einem der erweiterten **Address**-Typen entsprechen muss (**US-Address** oder **UK-Address**) und nicht dem generischen **AddressType**, müssen wir den erforderlichen erweiterten ComplexType als ein Attribut des Elements **Address** definieren.

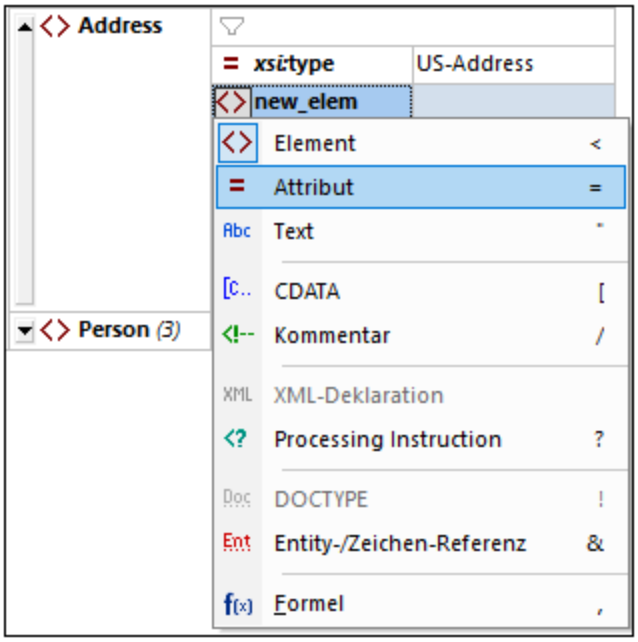
Dieses Attribut des Elements **Address** wird folgendermaßen hinzugefügt:

1. Klicken Sie im XML-Dokument mit der rechten Maustaste auf das Element `Name` und wählen Sie im Kontextmenü **Einfügen vor** (*siehe Abbildung unten*).

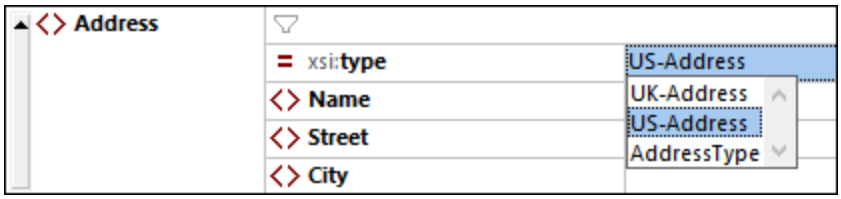
 © 2019-2025 Altova GmbH



2. Daraufhin wird oberhalb des Elements **`Name`** ein neuer Element-Node namens **`new_elem`** hinzugefügt (*siehe Abbildung unten*). Klicken Sie links vom Namen des Node auf den Elementtyp und wählen Sie im daraufhin angezeigten Menü (*Abbildung unten*) den Node-Typ *Attribut* aus. Daraufhin ändert sich der Node-Typ in den Node-Typ *Attribut*; der Name bleibt jedoch weiterhin **`new_elem`**.



3. Doppelklicken Sie auf den Node-Namen und wählen Sie in der daraufhin angezeigten Eingabehilfe **`xsi:type`**.
4. Drücken Sie die **Tabulatortaste**, um zum Wertefeld des Attributs zu gelangen. Daraufhin wird ein Popup-Menü mit den verfügbaren **`xsi:type`**-Werten angezeigt (*siehe Abbildung unten*). Diese Werte sind die für das Element **`Address`** im Schema definierten complexTypes.



5. Wählen Sie als Wert des Attributs **`xsi:type US-Address`** aus.

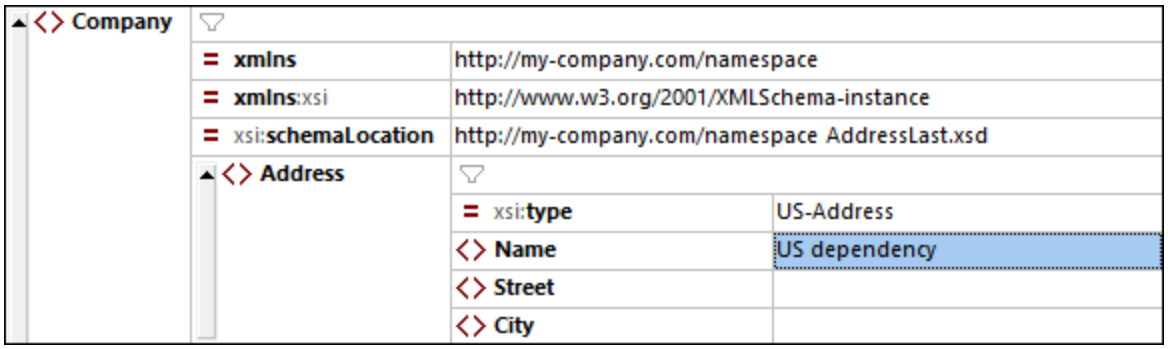
Anmerkung: Das vorgesetzte **xsi:** ermöglicht die Verwendung von speziellen XML-Schema-spezifischen Befehlen in Ihrer XML-Dokumentinstanz. Beachten Sie, dass der Namespace für das Präfix **xsi.** automatisch zum Dokumentelement hinzugefügt wurde, als Sie Ihrer XML-Datei ein Schema zugewiesen haben.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

5.3 Eingabe von Daten in der Grid-Ansicht

Sie können jetzt Daten in Ihr XML-Dokument eingeben.

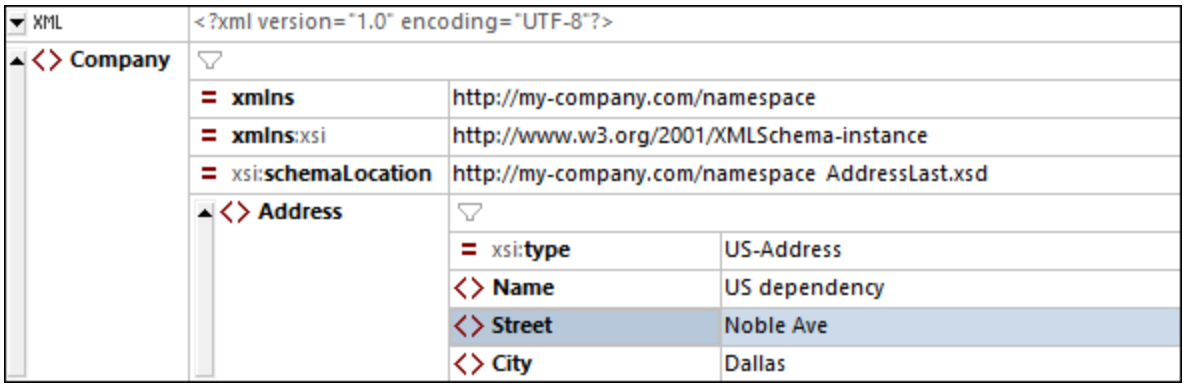
Gehen Sie folgendermaßen vor:

1. Doppelklicken Sie in das Wertefeld `Name` (oder verwenden Sie die Pfeiltasten) und geben Sie **US dependency** ein. Bestätigen Sie mit der **Eingabetaste**.



2. Geben Sie auf dieselbe Art einen **Street** und **City** Namen ein (z.B. **Noble Ave.** und **Dallas**).

3. Klicken Sie auf das Element **Person** und drücken Sie die **Entf**-Taste, um es zu löschen. (Wir werden es im nächsten Abschnitt des Tutorials wieder hinzufügen). Daraufhin erscheint nun das gesamte Element **Address** markiert.
4. Klicken Sie auf ein beliebiges Child-Element des Elements **Address**, um die Auswahl für alle Child-Elemente von **Address** mit Ausnahme des gewählten Elements aufzuheben. Ihr XML-Dokument sollte folgendermaßen aussehen:



 © 2019-2025 Altova GmbH

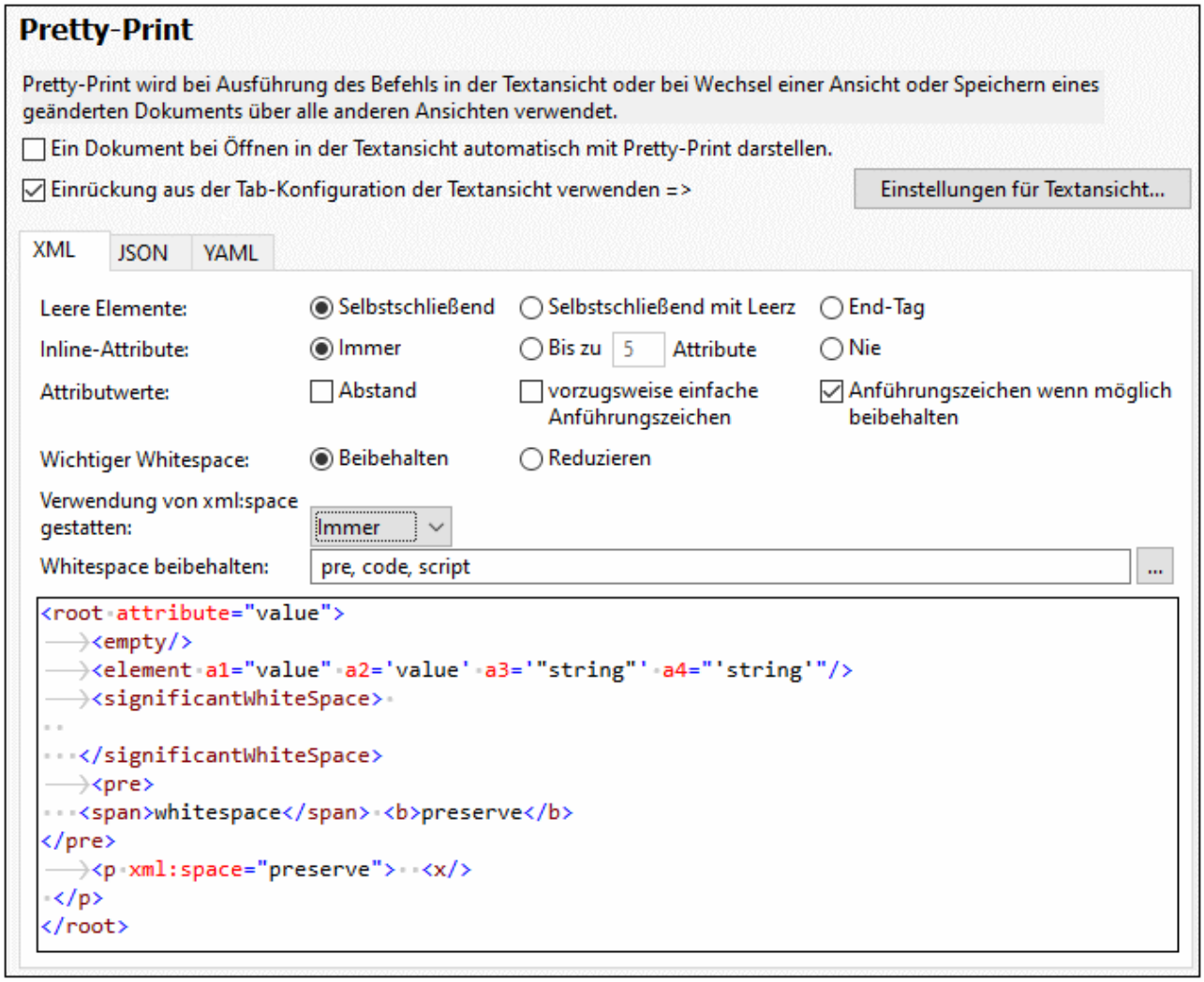
5.4 Eingabe von Daten in der Textansicht

In der Textansicht werden die Daten und der Markup-Code der XML-Dateien in einem übersichtlich strukturierten Layout angezeigt, und gemäß dem jeweiligen Schema werden intelligente Bearbeitungsfunktionen bereitgestellt.

Dokument-Layout

Das Dokument-Layout der Textansicht wird an zwei Stellen definiert:

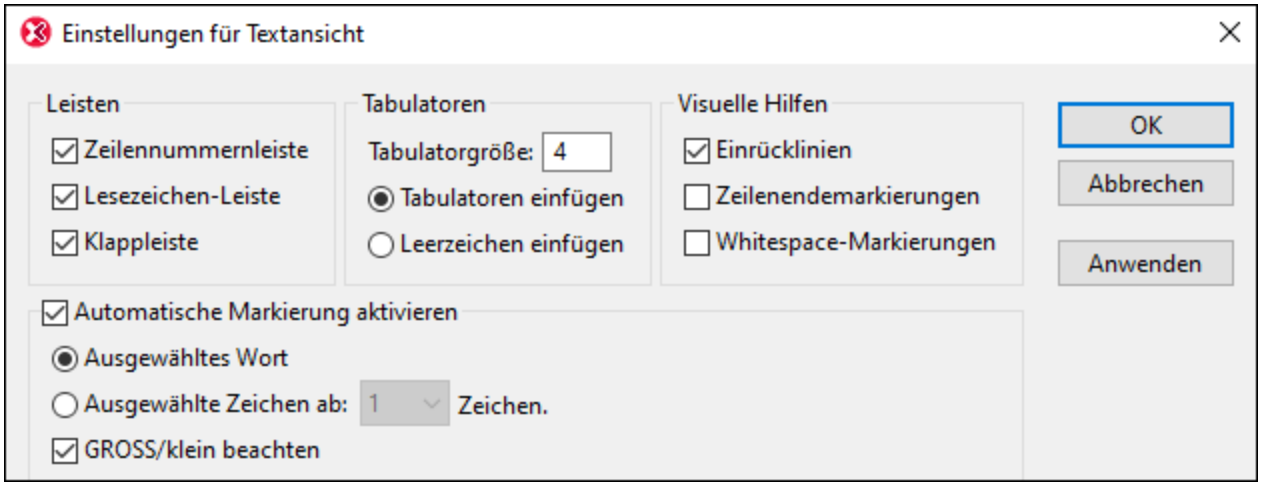
Pretty-Print-Optionen
Diese Einstellungen befinden sich im Abschnitt Pretty-Print des Dialogfelds "Optionen" (*Abbildung unten*). Wenn Sie eine Option definieren, sehen Sie die Auswirkung davon sofort unten im Vorschaubereich. Konfigurieren Sie die Pretty-Print-Optionen nach Bedarf. Während Sie das Dokument in der Textansicht bearbeiten, kann es vorkommen, dass das Layout des Dokuments unstrukturiert erscheint, vor allem, wenn Sie Textblöcke kopiert haben. Um ein sauberes und hierarchisches Layout zu erhalten, klicken Sie einfach auf **Bearbeiten | Pretty-Print**.



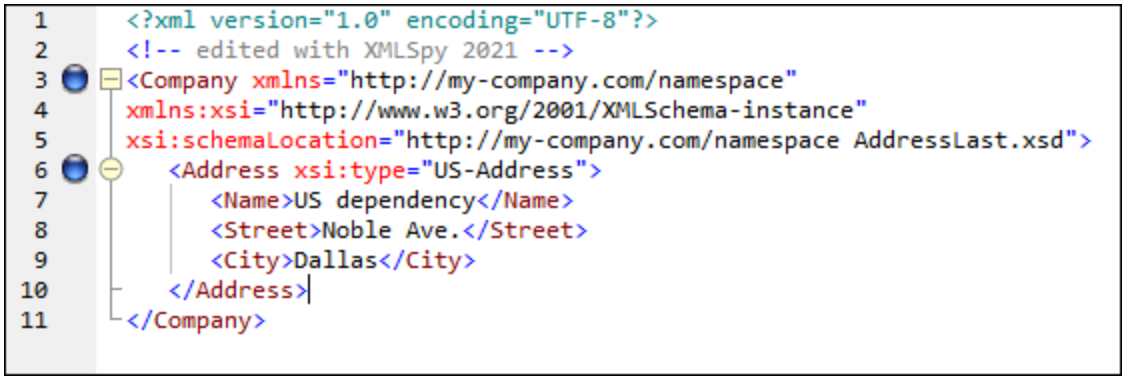
Einstellungen für Textansicht

Das Dialogfeld "Einstellungen für Textansicht (*Abbildung unten*) bietet nicht nur zusätzliche Layout-Optionen, sondern dient auch zum Ein-/Ausschalten nützlicher Funktionalitäten der Textansicht, wie z.B. Zeilennummerierung und Klappleisten. Das Dialogfeld "Einstellungen für Textansicht" wird mit dem Befehl **Ansicht | Einstellungen für Textansicht** aufgerufen.

© 2019-2025 Altova GmbH



In der Abbildung unten sehen Sie die aktuelle XML-Datei in der Textansicht. Die einzelnen Elemente der Textansicht sind gemäß den Einstellungen im Dialogfeld oben eingeblendet.



Auf der linke Seite befinden sich drei Leisten: (i) die Zeilennummernleiste, (ii) die Lesezeichenleiste (die zwei blaue Lesezeichen enthält) und (iii) die Klappleiste (über die Sie XML-Elemente ein- und ausklappen können). Einrücklinien sind die hellgrauen senkrechten Linien, die die Einrückung von Tags auf derselben hierarchischen Ebene anzeigen. Zusätzliche visuelle Hilfen sind Zeilenendemarkierungen und Whitespace-Markierungen, die im Bereich *Visuelle Hilfen* (*siehe Abbildung oben*) aktiviert oder deaktiviert werden können.

Anmerkung: Die Textansicht-spezifischen Funktionen "Pretty-Print" und "Lesezeichen" wurden bereits weiter oben im Abschnitt Einstellungen für Textansicht 13 dieses Tutorials behandelt.

Bearbeitung in der Textansicht

In diesem Abschnitt werden wir Daten in der Textansicht eingeben und bearbeiten, um die Funktionen der Textansicht zu erläutern.

Anmerkung: Da die Funktion *Bei Bearbeitung validieren* standardmäßig aktiv ist, werden alle bei der Bearbeitung entstandenen Validierungsfehler sofort gekennzeichnet. Die Fehlermeldung(en) dazu werden im Fenster "Meldungen" angezeigt. Ignorieren Sie diese Fehler und Meldungen für den Moment. Wenn Sie die Validierung um Hintergrund stört, können Sie die Funktion *Bei Bearbeitung validieren* in den

Validierungseinstellungen des Dialogfelds "Optionen" abschalten. Beachten Sie, dass Sie Ihr Dokument jederzeit validieren können (Beschreibung im nächsten Abschnitt 62 dieses Tutorials).

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.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy 2021 -->
<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

</Company>
```

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.

```
      </Address>
      <Person Manager="">
</Company>                false
                          true
```

6. Setzen Sie den Cursor an das Ende der Zeile (oder betätigen Sie die "**Ende**"-Taste Ihrer Tastatur). Drücken Sie die Leerzeichentaste einmal. Daraufhin wird eine Dropdown-Liste geöffnet, die 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.

```
      </Address>                                  Attributes
      <Person Manager="true"
</Company>                Degree                  Degree
                          Programmer              Programmer
                          xsi:type                xsi:type
                                                  Manager
```

7. 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**). (Enumerationen sind Werte, die gemäß einem XML-Schema zulässig sind).

 © 2019-2025 Altova GmbH

```
    </Address>
    <Person Manager="true" Degree="">
</Company>
```

8. Wählen Sie **BA** durch Betätigen des 'Nach unten'-Pfeiles und bestätigen Sie die Auswahl mit der **Eingabetaste**. Setzen Sie den Cursor anschließend an das Ende der Zeile (über die **Ende**-Taste Ihrer Tastatur). Drücken Sie die Leerzeichentaste einmal. Nun sind **Manager** und **Degree** in der Attribut-Eingabehilfe ausgegraut.

```
    </Address>
    <Person Manager="true" Degree="BA"
</Company>
```

9. Wählen Sie **Programmer** durch Betätigen des 'Nach unten'-Pfeiles und drücken Sie die **Eingabetaste**.

```
    </Address>
    <Person Manager="true" Degree="BA" Programmer="f"
</Company>
```

10. Geben Sie den Buchstaben **"f"** ein und drücken Sie die **Eingabetaste**.
11. Setzen Sie den Cursor (mit der **Ende**-Taste) 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"?>
<!-- edited with XMLSpy 2021 -->
<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**. Sehen Sie selbst, wie die neu hinzugefügten Child-Nodes von `Person` angezeigt werden.

XML	<?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	<Address xsi:type="US-Address"><Name>US dependency</Name>
	Person	
		Manager: true
		Degree: BA
		Programmer: false
		First
		Last
		PhoneExt
		Email

Wir wollen das Dokument nun validieren und etwaige Fehler korrigieren.

5.5 Validierung des Dokuments

XMLSpy bietet zwei wichtige Funktionen zur Überprüfung des XML-Dokuments:

- eine Wohlgeformtheitsprüfung
- eine Validierung

	Wohlgeformtheit prüfen
	XML validieren

Da die Funktion *Bei Bearbeitung validieren* standardmäßig aktiv ist, werden alle bei der Bearbeitung entstandenen Validierungsfehler sofort gekennzeichnet. Die Fehlermeldung(en) dazu werden im Fenster "Meldungen" angezeigt. Wenn Sie die Validierung um Hintergrund stört, können Sie die Funktion *Bei Bearbeitung validieren* in den Validierungseinstellungen des Dialogfelds "Optionen" abschalten. Falls Sie dies tun, beachten Sie, dass Sie Wohlgeformtheitsprüfungen und Validierungen jederzeit durch Aufruf des entsprechenden Befehls über das Menü "XML" durchführen können. In diesem Abschnitt des Tutorials wird beschrieben, wie Sie diese Überprüfungen durchführen.

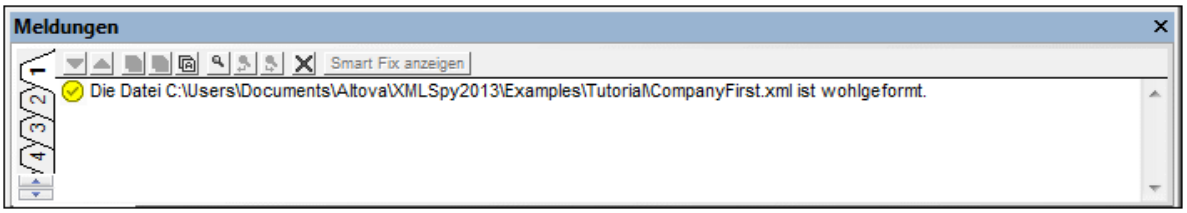
Überprüfung der Wohlgeformtheit

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). Sie können in jeder Bearbeitungsansicht eine Wohlgeformtheitsprüfung durchführen. Überprüfen Sie Ihr Dokument folgendermaßen:

1. Wählen Sie die Textansicht.
2. Wählen Sie die Menüoption **XML | Wohlgeformtheit prüfen** oder drücken Sie die Taste **F7**. (Alternativ dazu können Sie in der Symbolleiste auf die entsprechende Schaltfläche dafür klicken). Am

© 2019-2025 Altova GmbH

unteren Rand des Hauptfensters erhalten Sie im Fenster "Meldungen" eine Meldung, welche anzeigt, ob das Dokument wohlgeformt ist.



Beachten Sie, dass die Ausgabe des Fensters Meldungen" über neun Register verfügt. Das Ergebnis der Aktion wird immer auf dem aktiven Register angezeigt. So könnten Sie etwa die Wohlgeformtheit auf dem Register 1 prüfen und zum Register 2 wechseln, um eine Valdierung durchzuführen. Wenn Sie das Register nicht wechseln, wird das vorherige Ergebnis auf dem aktiven Register durch das neue Ergebnis überschrieben.

Anmerkung: Bei der Wohlgeformtheitsprüfung wird nicht geprüft, ob das XML-Dokument schemakonform ist. Dies wird bei der Validierung der Datei überprüft.

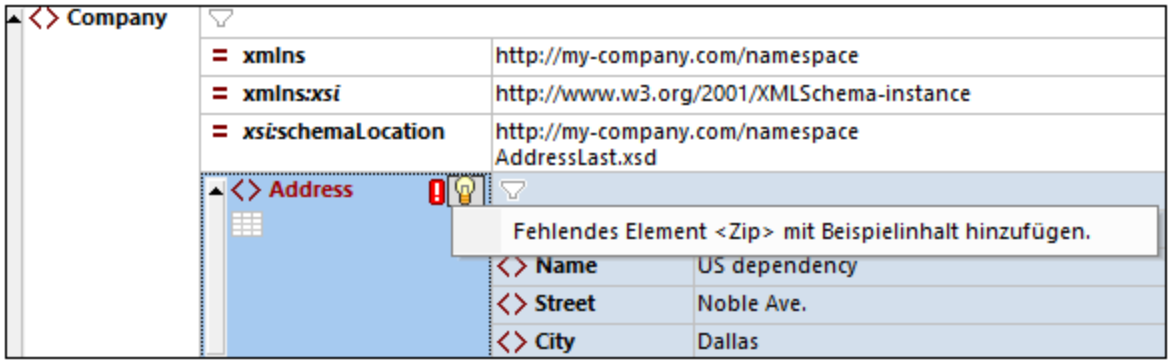
Überprüfung der Gültigkeit (Validierung)

Ein XML-Dokument ist gemäß einem Schema gültig, wenn mit der im Schema definierten Struktur und dem Inhalt übereinstimmt. Sie können in jeder Bearbeitungsansicht eine Validierung durchführen. Validieren Sie Ihr Dokument folgendermaßen:

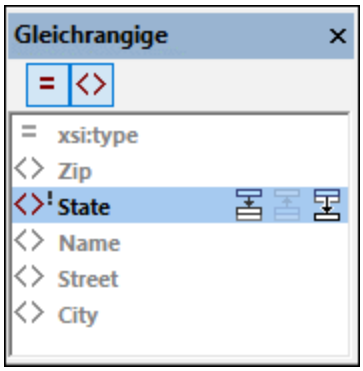
1. Wählen Sie die Grid-Ansicht.
2. Wählen Sie die Menüoption **XML | Validieren** oder drücken Sie die Taste **F8**. (Alternativ dazu können Sie in der Symbolleiste auf die entsprechende Schaltfläche dafür klicken). Im Fenster "Meldungen" wird eine Fehlermeldung angezeigt, dass die Datei nicht gültig 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 des `xsi:type`-Attributs 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

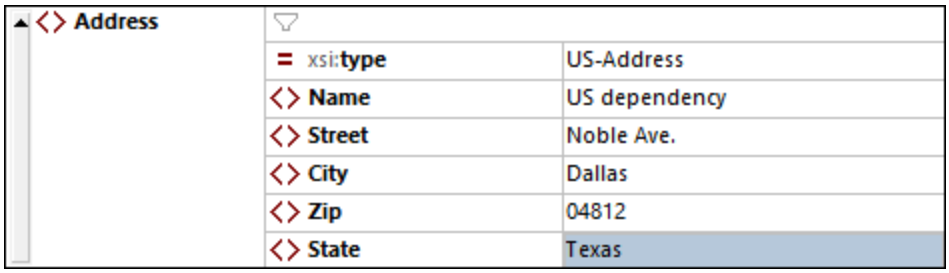
Die Stelle, an der das Dokument ungültig wird, erscheint rot markiert. Gleichzeitig werden hier ein Fehlersymobl und ein Smart Fix-Symbol angezeigt. Das ungültige Element ist in diesem Fall `Address`. Wenn Sie auf das Smart Fix-Symbol klicken, sehen Sie eine Popup-Meldung mit ungefähr folgendem Inhalt: *Fehlendes Element <Zip> mit Beispielinhalt hinzufügen.* Wenn Sie das Schema überprüfen, wird sich herausstellen, dass auf das Element `Address/City` das obligatorische Element `Zip` folgen muss. Um dies zu überprüfen, wählen Sie das Element `City` aus und sehen Sie sich die Eingabehilfe "Gleichrangige" an. Sie werden feststellen, dass vor dem Element `Zip` ein Ausrufezeichen steht, welches darauf hinweist, dass das Element im aktuellen Kontext obligatorisch ist.



Klicken Sie nun auf Smart Fix (*siehe Abbildung oben*). Daraufhin wird das Element **`Zip`** mit Beispielinhalt, durch den das Element gültig wird, angezeigt. Geben Sie die korrekte Postleitzahl (**`Zip`**) ein (z.B. **`04812`** für Dallas). Sehen Sie sich nun wieder die Eingabehilfe "Gleichrangige" an. Sie sehen darin nun, dass das Element **`State`** obligatorisch ist (es steht ein Ausrufezeichen davor). Wenn Sie das Element **`State`** auswählen, werden die Eingabehilfeoptionen, die dafür zur Verfügung stehen, aktiv (*Siehe Abbildung unten*). Dies sind die Aktionen, um das Element **`State`** nach dem aktuell im Hauptfenster ausgewählten Element (in diesem Fall **`City`**) einzufügen oder **`State`** nach allen gleichrangigen Elementen von **`City`** anzuhängen.



Da beide Aktionen in diesem Fall denselben Effekt haben, wählen Sie eine der beiden Aktionen aus. Daraufhin wird ein **`State`**-Element nach **`City`** hinzugefügt. Doppelklicken Sie in das Inhaltsfeld von **`State`** und geben Sie als Namen des Bundesstaats **`Texas`** ein (*Abbildung unten*). Beachten Sie, dass die Eingabehilfe "Gleichrangige" nun nur ausgegraute Elemente enthält. Dies weist darauf hin, dass keine weiteren obligatorischen Elemente vorhanden sind, die hinzugefügt werden müssen.



Fertigstellen des Dokuments und erneutes Validieren

Stellen Sie nun das Dokument fertig (geben Sie die Daten für das Element `Person` ein), bevor Sie es erneut validieren.

Gehen Sie folgendermaßen vor:

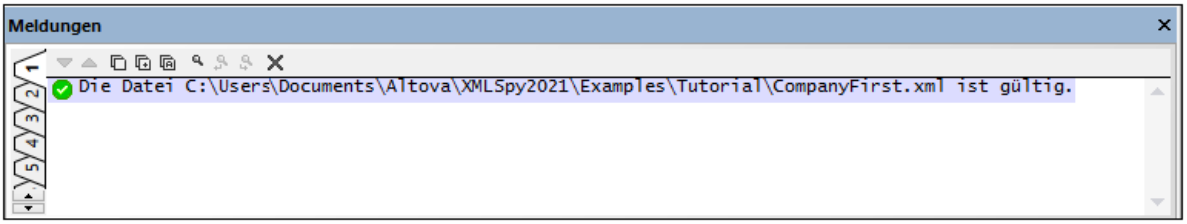
1. Klicken Sie auf das Wertefeld des Elements **`First`** und geben Sie den Vornamen der Person ein - z.B. `Fred`, bestätigen Sie mit der **Eingabetaste**.

Person		
	Manager	true
	Degree	BA
	Programmer	false
	First	Fred
	Last	
	PhoneExt	
	Email	

2. Geben Sie nun auf dieselbe Art Daten für alle Child-Elemente von **`Person`**, also **`Last`**, **`PhoneExt`** und **`Email`** ein. Sie können sich mit der Tabulatortaste durch die Zellen bewegen. Beachten Sie, dass der Wert von **`PhoneExt`** eine Ganzzahl mit einem Maximalwert von **`99`** sein muss (da dies der gültige Wertebereich von **`PhoneExt`** ist, den Sie im Schema definiert haben). Ihr XML-Dokument sollte in der Grid-Ansicht nun so aussehen:

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
		Street	Noble Ave.
		City	Dallas
	Person		
		Manager	true
		Degree	BA
		Programmer	false
		First	Fred
		Last	Smith
		PhoneExt	22
		Email	Smith@work.com

3. Klicken Sie auf die Schaltfläche , um das Dokument neuerlich zu prüfen. Diesmal erhalten Sie im Fenster "Meldungen" die Nachricht ", dass die Datei gültig ist. Das XML-Dokument ist nun gemäß dem Schema gültig.



4. Wählen Sie die Menüoption **Datei | Speichern**.

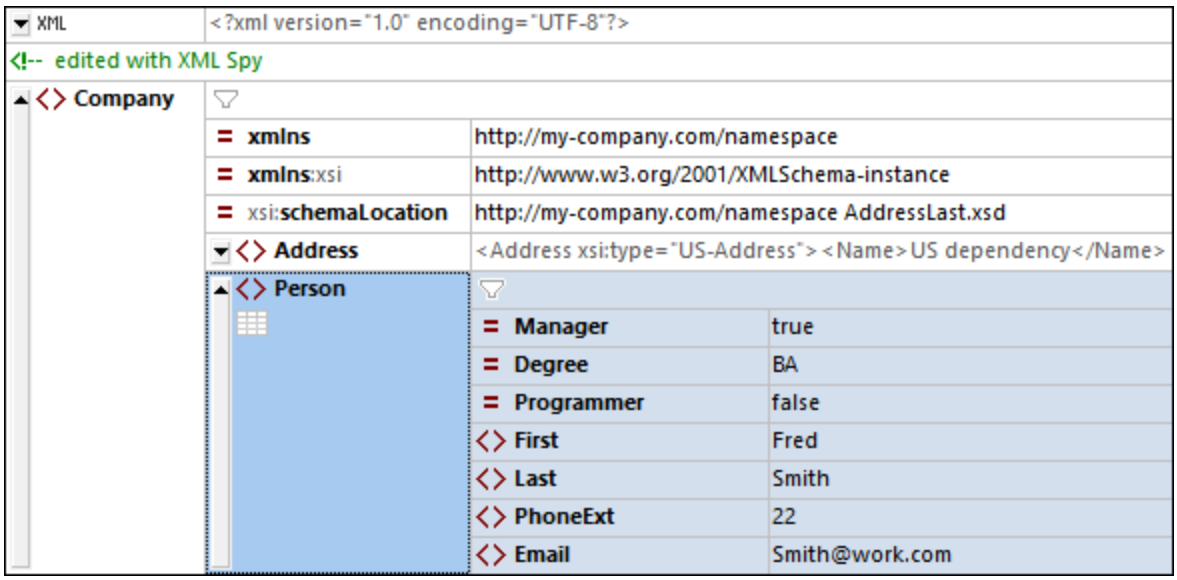
Anmerkung: Ein XML-Dokument kann auch gespeichert werden wenn es nicht gültig ist. Beim Speichern werden Sie gefragt, ob Sie das nicht korrekte Dokument speichern wollen. Mit **Trotzdem speichern** wird das Dokument in seinem aktuellen, ungültigen Status gespeichert.

5.6 Hinzufügen von Elementen und Attributen

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

Um ein neues `Person`-Element hinzuzufügen, gehen Sie folgendermaßen vor:

1. Klicken Sie auf die graue Bildlaufleiste links neben dem Element `Address`, um die `Address`-Elemente auszublenden und mehr Platz in der Ansicht zu schaffen.
2. Wählen Sie das gesamte Element `Person` aus, indem Sie in der Grid-Ansicht auf oder unter den Namen des Elements `Person` klicken. Beachten Sie, dass das Element `Person` nun in der Eingabehilfe "Gleichrangige" verfügbar ist.



3. Wählen Sie das `Person`-Element in der Eingabehilfe "Gleichrangige" aus und klicken Sie entweder auf **Einfügen nach** oder **Anhängen**. Daraufhin wird ein neues `Person`-Element angehängt (*siehe Abbildung unten*).

 © 2019-2025 Altova GmbH

Person <1>		
	Manager	true
	Degree	BA
	Programmer	false
	First	Fred
	Last	Smith
	PhoneExt	22
	Email	Smith@work.com
Person <2>		

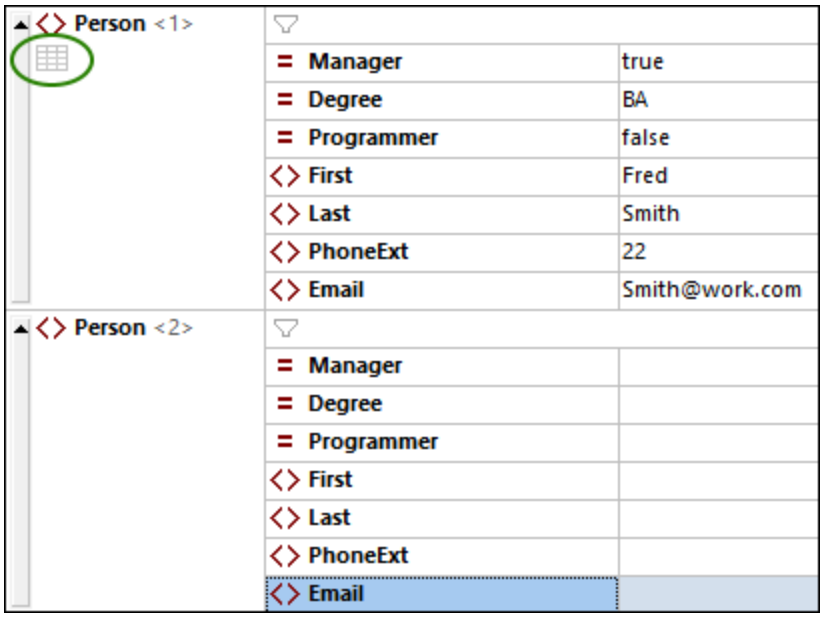
4. Wenn das Element **Person** ausgewählt ist, werden die Child-Attribute und -Elemente dieses Elements in der Eingabehilfe "Children" angezeigt. Doppelklicken Sie auf Attribute und Elemente, um dieselben Child-Nodes wie für das erste **Person**-Element hinzuzufügen. Wenn der Fokus im Hauptfenster vom Element **Person** zu einem hinzugefügten Child-Element geändert wird, können Sie auf eine von zwei Arten Zusätzliche Diagrammarten Children des Elements **Person** hinzufügen: (i) Bringen Sie das Element **Person** in den Fokus (indem Sie es auswählen) und ein neues Child aus der Eingabehilfe "Children" hinzufügen; (ii) Fügen Sie, während sich der Fokus auf dem hinzugefügten Child-Element befindet, aus der Eingabehilfe "Gleichrangige" ein gleichrangiges Child-Element hinzu. Child-Nodes von **Person**, die bereits hinzugefügt wurden, erscheinen in beiden Eingabehilfen ausgegraut.

Person <1>		
	Manager	true
	Degree	BA
	Programmer	false
	First	Fred
	Last	Smith
	PhoneExt	22
	Email	Smith@work.com
Person <2>		
	Manager	
	Degree	
	Programmer	
	First	
	Last	
	PhoneExt	
	Email	

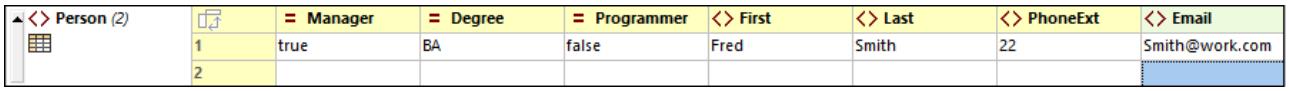
Sie könnten in der normalen Grid-Ansicht Inhalt für die Child-Nodes des Elements **Person** hinzufügen, wechseln wir jedoch nun in die Tabellenansicht der Grid-Ansicht, da sich diese besser zum Editieren einer Struktur mit mehreren Instanzen wie z.B. bei **Person** eignet.

5.7 Bearbeiten in der Tabellenansicht

Die Grid-Ansicht enthält eine spezielle Ansicht namens Tabellenansicht, die sich zum Bearbeiten von Elementen mit mehreren Instanzen eignet. So hat etwa das Element `Person` mehrere Instanzen (*siehe Abbildung unten*) und kann daher als Tabelle angezeigt werden. Um ein solches Element als Tabelle anzuzeigen, klicken Sie auf das **Tabellenansicht**-Symbol der ersten Instanz des Elements. In der Abbildung unten ist das **Tabellenansicht**-Symbol des `Person`-Elements grün umrandet. (Wählen Sie alternativ dazu den Menübefehl **XML | Als Tabelle anzeigen** oder klicken Sie auf die Symbolleisten-Schaltfläche in der Grid-Ansichts-Symbolleiste).



Wenn Sie auf das Tabellenansicht-Symbol klicken, wird das Element `Person` als Tabelle angezeigt. Die Child-Nodes des Elements (seine Attribute und Elemente) werden dabei als Spalten angezeigt, während die einzelnen `Person`-Elemente als Zeilen angezeigt werden (*siehe Abbildung unten*).



Vorteile der Tabellenansicht
Die Tabellenansicht bietet die folgenden Vorteile:

- Sie können eine Spaltenüberschrift mit der Maus verschieben, um ganze Spalten relativ zueinander zu verschieben. Im eigentlichen XML-Dokument verschiebt sich dabei die relative Position von Child-Nodes aller Element-Instanzen (die den Zeilen der Tabelle entsprechen).
- Tabellen - und die Element-Instanzen, die diesen entsprechen - können nach dem Inhalt jeder beliebigen Spalte in aufsteigender und absteigender Reihenfolge sortiert werden. Verwenden Sie dazu den Menübefehl **XML | Aufsteigend sortieren** oder **Absteigend sortieren**.
- Mit Hilfe der Befehle im Menü **XML** können rasch zusätzliche Zeilen (d.h. Element-Instanzen) angehängt oder eingefügt werden. Der Vorteil davon ist, dass nicht nur ein neues Element, sondern alle seine durch die Spalten der Tabelle repräsentierten Children hinzugefügt werden.
- Sie können *strukturierte Daten* von und in Produkte von Drittanbietern wie z.B. Excel kopieren.

 © 2019-2025 Altova GmbH

- Diese intelligenten Bearbeitungsfunktionen von XMLSpy stehen auch in der Tabellenansicht zur Verfügung.

Anzeigen eines Elements mit mehreren Instanzen als Tabelle

Um den Elementtyp `Person` in Tabellenform anzuzeigen, gehen Sie folgendermaßen vor:

1. Klicken Sie, wie oben beschrieben, auf das **Tabellensicht**-Symbol der ersten Instanz des `Person`-Elements.
2. Wählen Sie die Menüoption **Ansicht | Optimale Breite** oder das Symbol **Optimale Breite** der Grid-Ansichts-Symbolleiste.

Anmerkung: Die Tabellensicht kann für alle Elemente mit mehreren Instanzen aktiviert oder deaktiviert werden. Child-Elemente, die als Tabellen angezeigt wurden, werden jedoch weiterhin als Tabellen angezeigt.

Eingabe von Inhalt in der Tabellenansicht

Um Inhalt für das zweite `Person`-Element einzugeben, doppelklicken Sie in jede der Tabellenzellen in der zweiten Zeile und geben Sie Daten ein. Die intelligenten Bearbeitungsfunktionen sind auch innerhalb der Zellen einer Tabelle aktiv, sodass Sie, wo verfügbar, Optionen aus Dropdown-Listen auswählen können (z.B. Booleschen Inhalt und die Enumerationen des Attributs `Degree`).

	Manager	Degree	Programmer	First	Last	PhoneExt	Email
1	true	BA	false	Fred	Smith	22	Smith@work.com
2	false	MA	true	Alfred	Aldrich	33	Aldrich@work.com

Dynamische Valdierung

Beachten Sie jedoch, dass `PhoneExt`, wie im Schema definiert, eine Ganzzahl zwischen `0` und `99` sein muss, damit die Datei gültig ist. Sie können die XMLSpy-Funktion zur Validierung während der Bearbeitung aktivieren. Wenn diese Funktion aktiv ist, wird die Datei, jedes Mal, wenn der Fokus auf einen neuen Node verschoben wird, validiert. Testen Sie die dynamische Valdierung folgendermaßen:

1. Aktivieren Sie den Menübefehl **XML | Bei Bearbeitung validieren**.
2. Geben Sie, wie in der Abbildung unten gezeigt, einen ungültigen `PhoneExt`-Wert ein (einen beliebigen Wert, der größer als `99` ist).
3. Drücken Sie die **Tabulatortaste**. Daraufhin erscheinen in der Zelle `PhoneExt` ein Fehlersymbol und ein Smart Fix-Symbol (*siehe Abbildung unten*).
4. Wenn Sie die Maus über das Fehlersymbol platzieren, sehen Sie die Validierungsfehlermeldung (*Abbildung unten*).

Last	PhoneExt	Email
Aldrich	33	Aldrich@work
Coletti	444	Coletti@work.com
Smith	22	Smith@work.com

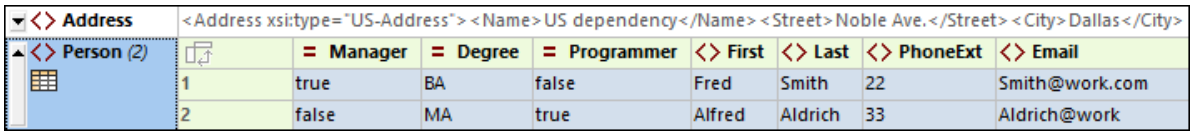
Der Wert '444' ist für das Element <PhoneExt> nicht zulässig.

5. Klicken Sie auf das Smart Fix-Symbol und anschließend auf die angezeigte Smart Fix-Option. Daraufhin wird der ungültige Werte durch einen gültigen Wert ersetzt und das Fehlersymbol verschwindet.

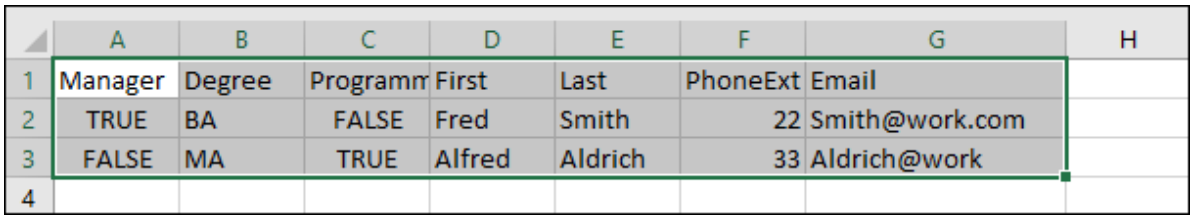
Kopieren von XML Daten aus und in Tabellenkalkulationsprogramme

In der Tabellenansicht können Sie Daten als tabulatorgetrennten Text kopieren, sodass er mit Tabellenkalkulationsprogrammen wie z.B. MS Excel ausgetauscht werden kann. Um aus Ihrer XML-Datei zu kopieren, gehen Sie folgendermaßen vor:

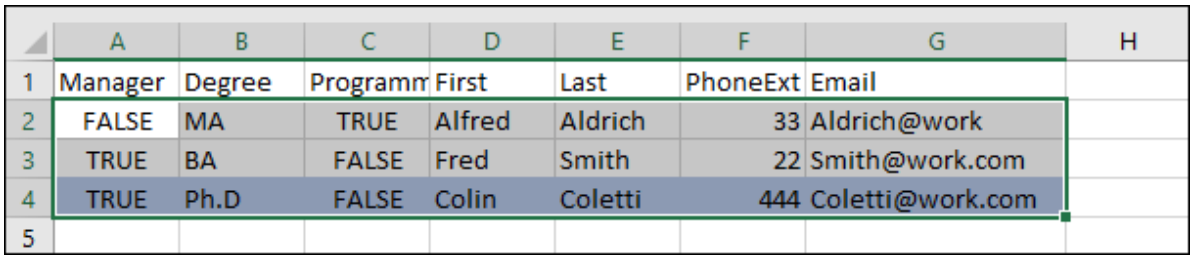
1. Klicken Sie auf das Element `Person` (*siehe Abbildung unten*). Dadurch werden die Spaltenüberschriften sowie beide Zeilen der Tabelle ausgewählt.



2. Klicken Sie mit der rechten Maustaste auf die Auswahl und wählen Sie im daraufhin angezeigten Kontextmenü den Befehl **Kopieren | Als tabulatorgetrennten Text kopieren**. Drücken Sie alternativ dazu **Strg+ C**.
3. Wechseln Sie zu einem Excel-Arbeitsblatt, wählen Sie die Zelle A1 aus und fügen Sie die XML-Daten ein (**Strg+V**). Die Daten werden gemäß der Tabellenstruktur in der Tabellenansicht in Form von Zeilen eingegeben (*siehe Abbildung unten*).



4. Geben Sie, wie in der Abbildung unten gezeigt, eine neue Datenzeile in Excel ein. Stellen Sie sicher, dass Sie für das Element `PhoneExt` eine dreistellige Zahl (z.B. `444`) eingeben.
5. Markieren Sie die Tabellendaten in Excel (ohne die Spaltenüberschriften) (*grüne Umrandung in der Abbildung unten*) und kopieren Sie sie mit **Strg+C**.



6. Stellen Sie in XMLSpy sicher, dass der Befehl **XML | bei Bearbeitung validieren** aktiviert ist.
7. Wählen Sie in XMLSpy in der Tabellenansicht Ihres XML-Dokuments die Zelle `Manager` der ersten Zeile aus und fügen Sie den Inhalt der Zwischenablage mit **Strg+V** ein. Ihre neue Tabelle sieht in etwa wie in der Abbildung unten aus.

 © 2019-2025 Altova GmbH

Address	<Address xsi:type="US-Address"><Name>US dependency</Name><Street>Noble Ave.</Street><City>Dallas</City>							
Person (3)		Manager	Degree	Programmer	First	Last	PhoneExt	Email
	1	FALSE	MA	TRUE	Alfred	Aldrich	33	Aldrich@work
	2	TRUE	BA	FALSE	Fred	Smith	22	Smith@work.com
	3	TRUE	Ph.D	FALSE	Colin	Coletti	444	Coletti@work.com

8. Der Grund für die Validierungsfehler für die Booleschen Werte ist der Unterschied in der Groß- und Kleinschreibung zwischen XML und Excel. Um diese Fehler zu beheben, wenden Sie die Smart Fix-Korrekturen auf die entsprechenden Tabellenzellen an.

Sortieren der Tabellen nach Spalteninhalt

Eine Tabelle kann in der Tabellenansicht nach jeder ihrer Spalten in aufsteigender oder absteigender Reihenfolge sortiert werden. Wir möchten die Tabelle **`Person`** nach Nachnamen sortieren. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie auf die Kopfzeile der Spalte **`Last`**.

Person (3)		Manager	Degree	Programmer	First	Last	PhoneExt	Email
	1	true	BA	false	Fred	Smith	22	Smith@work.com
	2	false	MA	true	Alfred	Aldrich	33	Aldrich@work
	3	true	Ph.D	false	Colin	Coletti	444	Coletti@work.com

2. Wählen Sie die Menüoption **XML | Aufsteigend sortieren** oder klicken Sie das Symbol **Aufsteigend sortieren** der Grid-Ansichts-Symbolleiste. Die Spalte wird unter Einbeziehung der gesamten Tabelle alphabetisch sortiert. Die Spalte bleibt hierbei markiert.

Person (3)		Manager	Degree	Programmer	First	Last	PhoneExt	Email
	1	false	MA	true	Alfred	Aldrich	33	Aldrich@work
	2	true	Ph.D	false	Colin	Coletti	444	Coletti@work.com
	3	true	BA	false	Fred	Smith	22	Smith@work.com

Da die Telefondurchwahl **`444`** korrekt, aber ungültig ist, müssen wir das XML-Schema ändern, damit diese Nummer gültig ist. Dies wird im nächsten Abschnitt beschrieben.

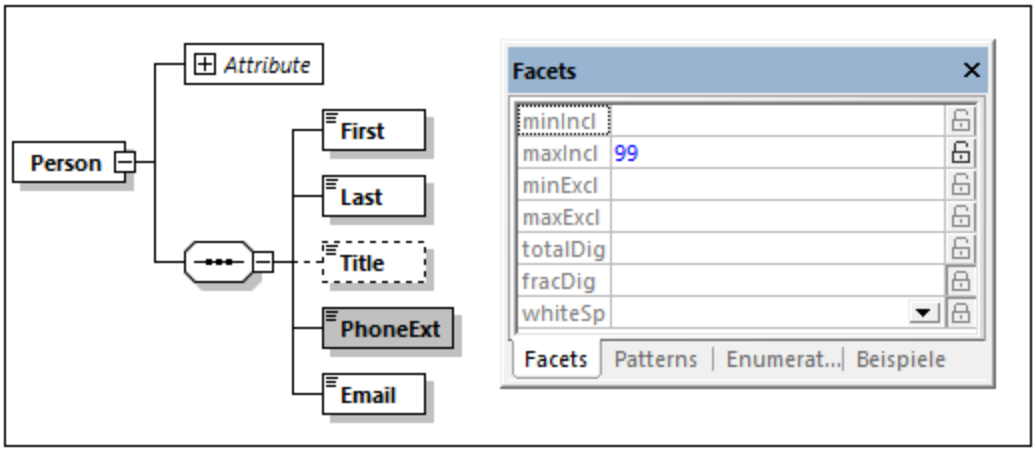
5.8 Ändern des Schemas

Da wir eine Durchwahl haben, die sich außerhalb des im XML-Schema definierten Bereichs (**`0`** bis **`99`**) befindet, wollen wir den Bereich bis **`999`** erweitern. Gehen Sie dazu vor, wie unten beschrieben.

1. Wählen Sie in der Grid-Ansicht eine beliebige der **`PhoneExt`**-Zellen aus (*siehe Abbildung unten*).

Person (3)		Manager	Degree	Programmer	First	Last	PhoneExt	Email
	1	false	MA	true	Alfred	Aldrich	33	Aldrich@work
	2	true	Ph.D	false	Colin	Coletti	444	Coletti@work.com
	3	true	BA	false	Fred	Smith	22	Smith@work.com

2. Wählen Sie die Menüoption **DTD/Schema | Gehe zu Definition** oder klicken Sie das Symbol **Gehe zu Definition** der Grid-Ansichts-Symbolleiste. Dadurch wird das damit verknüpfte Schema, in diesem Fall **`AddressLast.xsd`** geöffnet und die Definition **`PhoneExt`** wird markiert (*siehe Abbildung unten*).



3. Das Facet `maxIncl` des Elements ist `99` (*siehe Abbildung*). Ändern Sie diesen Wert in `999` und speichern Sie das Schema anschließend.
4. Gehen Sie zurück zum XML-Dokument und validieren Sie es. Es ist nun gültig.
5. Speichern Sie Ihre Datei unter `CompanyLast.xml`.

Anmerkung: Der Ordner "Tutorial" von XMLSpy enthält eine Datei namens `CompanyLast.xml`, die dieselben Daten enthält wie die Datei, die Sie nach Beendigung dieses Tutorials gespeichert haben.

 © 2019-2025 Altova GmbH

6 XSLT-Transformationen

Ziel

Erstellen einer HTML-Datei anhand 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 drei in XMLSpy integrierten Altova XSLT-Prozessoren durch (*siehe Hinweis unten*).

In diesem Abschnitt verwendete Befehle

Die folgenden XMLSpy-Befehle werden in diesem Abschnitt verwendet:

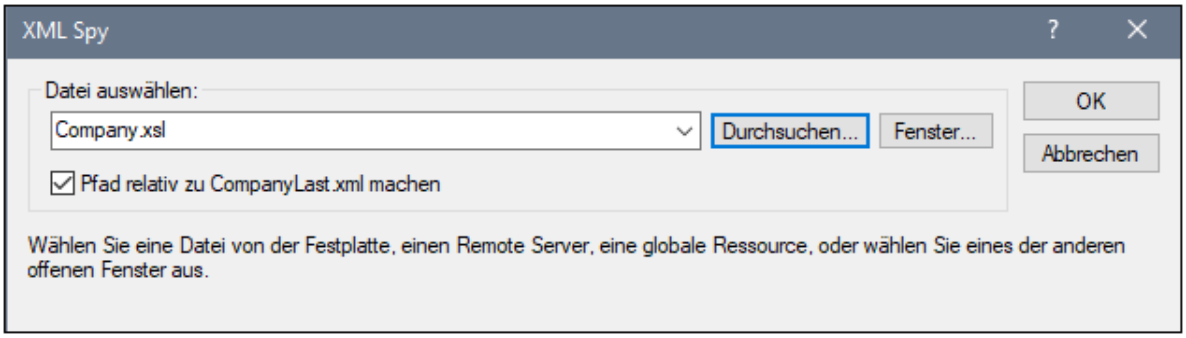
	XSL/XQuery | XSL zuweisen weist dem aktiven XML-Dokument eine XSL-Datei zu.
	XSL/XQuery | Gehe zu XSL öffnet die vom aktiven XML-Dokument referenzierte XSL-Datei.
	XSL/XQuery | XSL-Transformation (F10) oder die Symbolleistenschaltfläche transformiert das aktive XML-Dokument mit Hilfe des diesem zugewiesenen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

Anmerkung: XMLSpy verfügt über integrierte XSLT-Prozessoren für XSLT 1.0, 2.0 und 3.0. 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 ein XSLT 3.0 Stylesheet. Bei Aufruf des Befehls **XSL-Transformation** wird automatisch der XSLT 3.0-Prozessor verwendet.

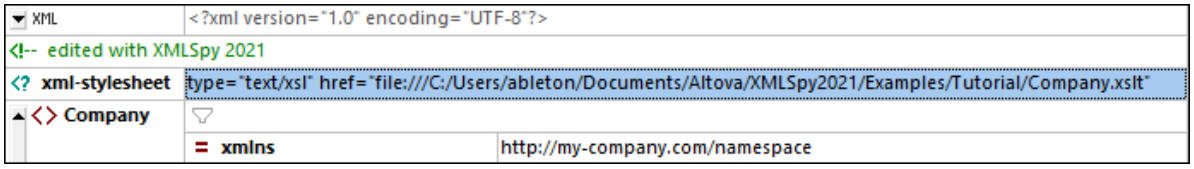
6.1 Zuweisen einer XSLT-Datei

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.mxl machen*" aktiviert haben, ist der Pfad relativ; andernfalls ist er absolut (wie in der Abbildung).



6.2 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 in der Symbolleiste auf die entsprechende Befehlsschaltfläche. Daraufhin wird die Transformation anhand des im XML-Dokument referenzierten XSLT-Stylesheet gestartet. Das Ausgabedokument wird in der Browser-Ansicht angezeigt; es hat den Namen `XSL Output.html`. Die `Company/Address`-Daten werden in einem Block links oben angezeigt, während die `Company/Person`-Daten unterhalb davon in Tabellenform dargestellt werden.

 © 2019-2025 Altova GmbH

Your Company

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

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

Anmerkung: Da die Datei `Company.xslt` ein XSLT 3.0-Dokument ist, wird für die Transformation automatisch der integrierte Altova XSLT 3.0-Prozessor ausgewählt. (Wenn die HTML-Ausgabedatei nicht generiert wird, stellen Sie sicher, dass die Standarddateierweiterung im Dialogfeld "Optionen" (**Extras | Optionen**) im Abschnitt "*XSL*" auf `.html` gesetzt wurde.) Damit wird sichergestellt, dass der Browser das Ausgabedokument korrekt als HTML-Datei liest.

6.3 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 `#ccccff` in `#99cc99` geändert werden. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie auf das Register `CompanyLast.xml` um das Dokument zu aktiven zu machen.
2. Wählen Sie die Menüoption **XSL/XQuery | Gehe zu XSL**. Daraufhin wird die im XML-Dokument referenzierte Datei `Company.xslt` geöffnet.
3. Suchen Sie nach dem Start-Tag des Elements `table` und anschließend dem Attribut `bgcolor` (*in der Abbildung unten markiert*). Ändern Sie den Wert des Attributs von `#ccccff` in `#99cc99`.

```
6   <xsl:template match="/">
7      <html>
8         <head><title>Your company</title></head>
9         <body>
10           <h1><center>Your Company</center></h1>
11           <xsl:apply-templates select="//my:Address"/>
12              <table border="1" bgcolor="#ccccff">
13                 <thead align="center">
14                    <td><strong>First</strong></td>
15                    <td><strong>Last</strong></td>
16                    <td><strong>Ext.</strong></td>
17                    <td><strong>E-Mail</strong></td>
18                    <td><strong>Manager</strong></td>
19                    <td><strong>Degree</strong></td>
20                    <td><strong>Programmer</strong></td>
21                 </thead>
22                 <xsl:apply-templates select="//my:Person"/>
23              </table>
24        </body>
25     </html>
26  </xsl:template>
27
```

4. Wählen Sie die Menüoption **Datei | Speichern**, um die Änderung zu speichern.
5. Klicken Sie auf das Register `CompanyLast.xml`, um die XML-Datei zur aktiven zu machen.
6. Wählen Sie den Befehl **XSL/XQuery | XSL-Transformation** oder drücken Sie alternativ dazu **F10**. In der Browser-Ansicht wird daraufhin eine neue `XSLT Output.html`-Datei mit einer Tabelle mit einer neuen Hintergrundfarbe angezeigt (*siehe Abbildung unten*).

Your Company

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

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

7. Wählen Sie die Menüoption **Datei | Speichern** und speichern Sie das Ausgabedokument unter dem Namen `Company.html`.

© 2019-2025 Altova GmbH

7 Projektverwaltung

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

7.1 Vorteile von Projekten

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

- Dateien können nach einem gemeinsamen Kriterium in Form von Ordnern gegliedert werden. So könnten Sie z.B. XML- und XSD-Dateien in separate Ordner einteilen. Sie können ganz nach Bedarf jede beliebige Hierarchie erstellen.
- Jeder Ordner hat bestimmte Eigenschaften, die Sie definieren können. So können Sie etwa einem Ordner von XML-Dateien ein Schema für die Validierung zuweisen. Daraufhin können alle Dateien in diesem Projektordner auf einmal anhand der Schema-Datei des Ordners validiert werden. Wenn Sie die Schemazuweisung des Projektordners ändern, können Sie schnell eine neue Batch-Validierung durchführen. Sie können eine Reihe weiterer nützlicher Ordnereigenschaften definieren, wie z.B. eine XSLT-Datei für Batch-Transformationen anhand einer einzigen XSLT-Datei.
- Die Batch-Verarbeitung kann auf bestimmte Ordner oder das ganze Projekt angewendet 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 die XML-Dateien in diesem Ordner anhand der zugewiesenen XSLT-Datei zu transformieren.
- Es können Zielordner für die XSL-Transformationsdateien definiert werden.

Alle oben aufgelisteten Einstellungen können über die Menüoption **Projekt | Eigenschaften**definiert werden. Projektbefehle stehen außerdem in den Kontextmenüs des Projekts und einzelner Projektordner zur Verfügung. 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**). (Nähere Informationen dazu finden Sie im Abschnitt Versionskontrolle).
- Sie können zu Projekten externe Ordner in Ihrem Netzwerk sowie Web-Ordner hinzufügen. Dadurch können alle Funktionalitäten von Projektordnern wie z.B. Validierung und Transformationen auf Ordner in Ihrem Netzwerk oder im Internet angewendet werden.

7.2 Erstellen eines Projekts

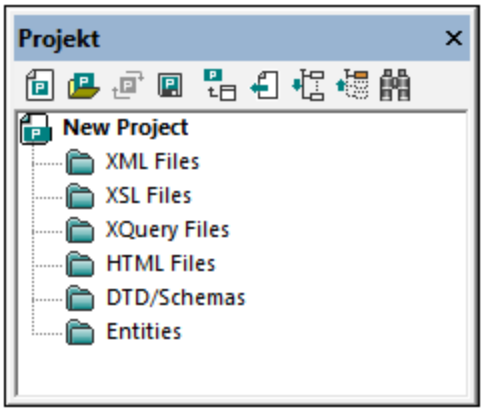
Im Hauptfenster ist nun eine Reihe von Tutorial-Dateien geöffnet. Sie können diese Dateien in einem Tutorial-Projekt gliedern. Zuerst müssen Sie ein neues Projekt erstellen und anschließend die Tutorial-Dateien zu den jeweiligen Unterordnern des Projekts hinzufügen.

 © 2019-2025 Altova GmbH

Erstellen eines neuen Projekts

Gehen Sie dazu folgendermaßen vor:

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



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`. Die XML-Datei wird zum Unterordner *XML* hinzugefügt, da sie die aktive Datei ist. Die Schema-Datei wird zum Ordner *DTD/Schemas* hinzugefügt, da die XML-Datei eine Referenz auf diese Datei enthält und sie zu einer verwandten Datei macht. Beachten Sie, dass durch Processing Instructions 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`.

Anmerkung: Ordner (nicht aber das Projekt) haben jeweils eine Eigenschaft namens *Dateierweiterungen*. Es handelt sich dabei um eine Liste von durch Kommas getrennten Dateierweiterungen (z.B. `xml;svg;wml`). Anhand dieser Liste wird ermittelt, welche Dateien zu welchen Ordnern hinzugefügt werden, wenn Dateien zu einem Projekt hinzugefügt werden. Wenn z.B. wie oben aktive und verwandte Dateien zu einem Projekt hinzugefügt werden, wird anhand der *Dateierweiterungseigenschaften* des Ordners ermittelt, zu welchem Ordner die Dateien hinzugefügt werden sollen.

Projekt- und Ordnereigenschaften

Eigenschaften (wie z.B. das Schema für die Validierung und die XSLT-Datei für die Transformation) können nicht nur für das gesamte Projekt, sondern auch für einzelne Ordner definiert werden. Anschließend können Sie am gesamten Projekt oder einzelnen Ordnern Aktionen wie Validierung und Transformation durchführen. Um eine Aktion durchzuführen, klicken Sie mit der rechten Maustaste auf das Projekt oder den Ordner und wählen Sie im Kontextmenü die gewünschte Aktion aus.

Beachten Sie die folgenden Punkte:

- Eine für einen Ordner definierte Eigenschaft setzt dieselbe Eigenschaft des Projekts außer Kraft.
- Wenn eine Eigenschaft für ein Projekt definiert wurde, wird sie auf alle Ordner, für die diese Eigenschaft noch nicht definiert wurde, angewendet.

- Wenn eine Aktion an einem Projekt durchgeführt wird, wird sie auf alle anwendbaren Dateitypen in allen Ordnern des Projekts angewendet. Wenn z.B eine Validierung an einem Projekt durchgeführt wird, wird sie an allen XML-Dateien in allen Ordnern des Projekts durchgeführt. In diesem Fall wird das für das Projekt definierte Schema für alle Validierungen mit Ausnahme der Dateien, die sich in Ordnern befinden, für deren Schemavalidierungseigenschaft ein anderes Schema definiert wurde, verwendet.

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 (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 zu Projekt hinzufügen**. Die XML-Datei wird anhand ihres Dateityps (`.xml`) zum Ordner *XML Files* hinzugefügt.
3. Fügen Sie auf dieselbe Art eine HTML-Datei und eine XSDL-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 | 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**).

Anmerkung: 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 zu Projekt hinzufügen** 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.

© 2019-2025 Altova GmbH

8 Fertig!

Wenn Sie bis hierher gekommen sind, herzlichen Glückwunsch und vielen Dank!

Wir hoffen, dass Ihnen diese Anleitung geholfen hat, um die Grundzüge von XMLSpy kennen zu lernen und dass Sie nun in der Lage sind, Ihre XML-Dateien mit XMLSpy als Editor zu bearbeiten. Wenn Sie nähere Informationen zu bestimmten Funktionen benötigen, konsultieren Sie bitte den Index oder die Suchfunktion dieses Handbuchs. Sie können die PDF-Version dieses Tutorials auch ausdrucken. Sie steht unter dem Namen **`Tutorial.pdf`** in Ihrem XMLSpy Applikationsordner zur Verfügung.

Benutzeroberfläche und Umgebung

Dieser Abschnitt enthält eine Beschreibung :

- der Benutzeroberfläche der Applikation und
- der Applikationsumgebung.

Der Abschnitt "Benutzeroberfläche" beginnt mit einer Übersicht über die Benutzeroberfläche. Anschließend werden die einzelnen Fenster der Benutzeroberfläche näher beschrieben. Außerdem erfahren Sie, wie Sie die Größe der Fenster anpassen und die Fenster verschieben können und mit den Fenstern und der Benutzeroberfläche arbeiten.

Im Abschnitt "Applikationsumgebung" werden die verschiedenen Einstellungen erläutert, mit denen festgelegt wird, wie die Dateien angezeigt und bearbeitet werden können. Außerdem wird darin erklärt, wie und wo Sie Ihre Applikation anpassen können. In diesem Abschnitt erfahren Sie, wo wichtige Beispiel- und Tutorial-Dateien auf Ihrem Rechner installiert wurden. Weiter hinten in diesem Abschnitt finden Sie einen Link zur Altova Website, wo Sie eine Feature Matrix Ihrer Applikation finden, Ihr Benutzerhandbuch in verschiedenen Formaten abrufen können und Informationen über die verschiedenen Support-Optionen für Sie finden. Des Weiteren finden Sie dort Informationen über andere Produkte von Altova.

© 2019-2025 Altova GmbH

Index

A

Attribute, 41
in der Content Model-Ansicht ein- und ausblenden, 41
in Schemadefinitionen, 41

B

Beispieldateien,
Tutorial, 5

C

ComplexType,
Definition erweitern, 31
ComplexTypes,
in Schema-Definitionen, 31
Content Model,
Attribute ein- und ausblenden, 41
einfaches Content Model erstellen, 19
Content Model-Ansicht, 16

D

Datenbank/Tabellen-Ansicht,
Verwendung, 68
Details-Eingabehilfe, 19
Dokumentieren,
Schema, 46
Drag & Drop,
Elemente hinzufügen, 24

E

Eingabehilfe,
Details, 19
Eingabehilfen,
in Grid-Ansicht, 66
Elemente, 27
Inhalt einschränken, 27
optional machen, 27
Elementtyp,
in XML-Dokument festlegen, 53
Enhanced Grid-Ansicht,
siehe Grid-Ansicht, 56
Enumerationen,
für Attribute definieren, 41

G

Globale Elemente,
in XML-Schemas verwenden, 39
Grid-Ansicht, 66
Dateneingaben, 56
Eingabehilfen verwenden, 66
Elemente und Attribute anhängen, 66
und Tabellenansicht, 68

I

Identity Constraint,
in der Content Model-Ansicht ein- und ausblenden, 41
Instanzen,
Anzahl, 19

K

Komponenten-Definitionen,
wiederverwenden, 31
Kompositor,
für Sequences, 19

N

Namespaces,
in Schemas, 18

© 2019-2025 Altova GmbH

Navigation,
Kürzel im Schema-Design, 44
Neues XML Dokument,
erstellen, 52

O

Optionale Elemente,
erstellen, 27

P

Projekte in XMLSpy,
erstellen, 78
Vorteile, 78
Projektverwaltung in XMLSpy, 78

S

Schema,
Dokumentation, 46
siehe XML-Schema, 16
Schema-Ansicht,
konfigurieren, 25
Schema-Übersicht, 16
Sequence-Kompositor,
verwenden, 19
SimpleTypes,
in Schema-Definitionen, 31

T

Tabellenansicht,
Verwendung, 68
Textansicht,
bearbeiten in, 57
Tutorial, 5
Beispieldateien, 5
Ziele, 5
Typ,
Erweiterung im XML-Dokument, 53

V

Validieren,
XML-Dokumente, 62
Vorlagen,
Ordner, 5

W

Wohlgeformtheitsprüfung,
bei XML-Dokument, 62

X

XML-Dokument,
in Textansicht bearbeiten, 57
neues erstellen, 52
XML-Dokumente,
Überprüfung der Gültigkeit (Validierung), 62
XML-Dokumenterstellung,
Tutorial, 51
XML-Schema, 16
Ansicht konfigurieren, 25
einfaches Schema erstellen, 16
Komponenten hinzufügen, 19
Namespaces definieren, 18
Navigation in der Design-Ansicht, 44
neue Datei erstellen, 16
Tutorial, 16
während der Bearbeitung des XML-Dokuments ändern, 71
XML-Schema-Definitionen,
komplexe Definitionen, 31
xsi:type,
Verwendung, 53
XSLT,
in XMLSpy ändern, 75
XSL-Transformation,
siehe XSLT, 73
XSLT-Transformation,
in XMLSpy, 74
Tutorial, 73
XSLT-Datei zuweisen, 73

© 2019-2025 Altova GmbH