

# **Altova EBA Add-in für Excel, Version 2024r2 Enterprise Edition**



**Benutzer- und Referenzhandbuch**

# **Altova EBA Add-in für Excel, Version 2024r2**

## **Enterprise Edition**

### **Benutzer- und Referenzhandbuch**

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: 2024

© 2018-2024 Altova GmbH

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>6</b>
1.1	Systemanforderungen.....	8
1.2	Installation und Lizenzierung.....	9
<b>2</b>	<b>Erstellen eines neuen Berichts</b>	<b>10</b>
2.1	Neuer Bericht.....	11
2.2	Filing-Fenster.....	15
2.3	Dateneingabe.....	17
2.4	Dateneingabe in 3-D-Tabellen.....	21
2.5	Festlegen der Genauigkeit von Zellen.....	23
2.6	Validieren von Daten.....	25
2.7	Exportieren von Daten in XBRL.....	27
2.8	Importieren von Daten aus XBRL.....	28
2.9	Batch-Konvertierung von XBRL in Excel.....	29
<b>3</b>	<b>Taxonomie-Manager</b>	<b>31</b>
3.1	Ausführen des Taxonomie-Managers.....	34
3.2	Statuskategorien.....	37
3.3	Anwenden eines Patch oder Installation einer Taxonomie.....	39
3.4	Deinstallieren einer Taxonomie, Zurücksetzen.....	41
3.5	Befehlszeilenschnittstelle (CLI).....	42
3.5.1	help .....	42
3.5.2	info .....	43
3.5.3	initialize.....	43
3.5.4	install .....	44
3.5.5	list .....	44
3.5.6	reset .....	45
3.5.7	uninstall.....	46

---

3.5.8	update .....	47
3.5.9	upgrade.....	47
<b>4</b>	<b>Menübefehle</b>	<b>49</b>
<b>5</b>	<b>Einstellungen</b>	<b>52</b>
<b>6</b>	<b>COM API</b>	<b>55</b>
6.1	Access API.....	56
6.2	C# Example.....	58
6.3	VBA Example.....	63
6.4	API Reference.....	64
6.4.1	IAutomationAPI.....	64
6.4.2	IEntryPointTree.....	68
6.4.3	IEntryPointGroup.....	68
6.4.4	IEntryPoint.....	70
6.4.5	ITableTree.....	72
6.4.6	IGroupNode.....	73
6.4.7	ITableNode.....	75
6.4.8	IForm .....	79
6.4.9	IReportProperties.....	82
6.4.10	IFormProperties.....	86
6.4.11	ICellProperties.....	89
6.4.12	IDimension.....	91
6.4.13	IValidationReport.....	92
6.4.14	IValidationReportMessage.....	93
<b>7</b>	<b>Lizenzinformationen</b>	<b>95</b>
7.1	Electronic Software Distribution.....	96
7.2	Software-Aktivierung und Lizenzüberwachung.....	97
7.3	Altova Endbenutzer-Lizenzvereinbarung.....	99

---

**Index**

**100**

# 1 Einführung

Mit Hilfe des **Altova® European Banking Authority (EBA) XBRL Add-in für Excel** kann Ihr Unternehmen XBRL-Berichte erstellen, die der XBRL-Taxonomie der EBA (European Banking Authority/Europäische Bankenaufsichtsbehörde) oder länderspezifischen XBRL-Taxonomien entsprechen.



Mit dem EBA XBRL Add-in können Sie Folgendes tun:

- Eingabe von XBRL-Daten in Microsoft Excel über ein vordefiniertes Vorlagen-Arbeitsblatt, in dem die Daten auf die XBRL-Taxonomie gemappt werden
- Validierung der Berichtsdaten direkt in Excel, um die Konformität der Daten mit der XBRL-Taxonomie zu gewährleisten.
- Export der Berichtsdaten aus Excel in das XBRL-Format
- Import der Daten aus vorhandenen XBRL-Berichten in Excel
- Batch-Konvertierung von XBRL-Dateien in das Excel-Format (**.xlsx**) (*Enterprise Edition*).

## XBRL-Taxonomien

Die Liste der unterstützten XBRL-Taxonomien wird regelmäßig aktualisiert, damit neuere Versionen unabhängig von Altova Add-in-Releases inkludiert werden. Länderspezifische XBRL-Taxonomien und ältere Versionen der EBA XBRL-Taxonomie werden bei der Installation des Add-in standardmäßig nicht unterstützt. Sie können die Taxonomien bei Bedarf mit dem [XBRL-Taxonomie-Manager](#)<sup>31</sup> anzeigen, installieren, upgraden und deinstallieren.

### Allgemeine Taxonomien

Die folgenden allgemeinen, nicht länderspezifischen Taxonomien werden unterstützt:

- EBA (European Banking Authority) XBRL-Taxonomie (ab Version 2.0 bis zur neuesten Version)
- SFRDP (Supervisory Financial Reporting Data Points) (wird von der EZB bereitgestellt)
- SRB RES REP: Single Resolution Board Resolution Reporting für die Meldung von Verbindlichkeitsdaten, wichtigen Funktionen und Marktinfrastruktur
- SRB SRF EAC: Single Resolution Board Ex-Ante Contributions to the Single Resolution Fund

### Länderspezifische Taxonomien

Die folgenden länderspezifischen Taxonomien werden unterstützt:

- ACPR (Banque de France Autorité de Contrôle Prudentiel et de Résolution)
  - LCB-FT: Zur Bekämpfung von Geldwäsche und Terrorismusfinanzierung
  - RUBA: Unified Reporting Banks and Assimilates
  - COREP: Gemeinsames Rahmenwerk für die Berichterstattung mit den Modulen LCR\_CON, LCR\_DA, NSFR\_CON and ALM\_CON
  - CREDITIMMO: Rahmenwerk für die Berichterstattung über private Immobilienkredite und ausstehende Kreditforderungen in Frankreich.
- BBK (Deutsche Bundesbank)
  - Durch die deutsche Basistaxonomie erweiterte EBA-ITS-Berichterstattung

- BDP (Banco de Portugal)
  - Rahmenwerk für die Finanzberichterstattung und das Meldewesen
- BOE (Bank of England)
  - BANKING: Regulatory Reporting für den Bankensektor
  - STATISTICS: Statistische Berichtspflichten
- CBI (Central Bank of Ireland)
  - FSP: Fund Service Provider, Investmentfirmen und andere Zahlungsinstitute
- DNB (De Nederlandsche Bank)
  - BISCBS: BIS Consolidated Banking Statistics
  - BSI: Balance Sheet Items Taxonomy
  - DGS: Deposit Guarantee Scheme Reporting Framework
  - FBO: Taxonomie für die Finanzberichterstattung für Investmentfonds und Fondsmanager
  - FEH: Foreign Equity Holdings
  - MESREP: Macroeconomic Statistics Reporting
  - MIR: MFI Interest Rates
  - MSR: Taxonomie für monatliche Wertpapierberichte
  - OTC: Derivat-Taxonomie
  - PAY: Payment Statistics Reporting Framework
  - PLA: Profit and Loss Account-Taxonomie
  - PPSP: Taxonomie für Payment Processing Service Providers
  - SPV: Special Purpose Vehicles-Taxonomie
- NBB (National Bank of Belgium)
  - BANKING: Domain MBS-XBRL Berichterstattung: FINREP, TREP

## Diese Dokumentation

Diese Dokumentation sollte in Verbindung mit den in die EBA XBRL-Taxonomie inkludierten Dokumenten zu dieser Taxonomie gelesen werden:

- *Description of DPM formal model*
- *EBA Architecture for XBRL representation of DPM*
- *EBA XBRL Filing Rules*

*Letzte Aktualisierung: 04.04.2024*

## 1.1 Systemanforderungen

Um das Add-in installieren und ausführen zu können, gelten die folgenden Systemvoraussetzungen:

- Windows 10, Windows 11, Windows Server 2016 oder höher
- Microsoft Excel ab Version 2010
- [Visual Studio 2010-Tools für Office-Laufzeit](#)
- Microsoft .NET Framework ab Version 4.8

### Achtung

- Das Add-in steht für Microsoft Excel 32-Bit und 64-Bit zur Verfügung. Microsoft Excel 64-Bit wird empfohlen, wenn große Taxonomien wie COREP CON, COREP IND, FINREP geladen werden müssen. Andernfalls kann es zu Fehlern im Zusammenhang mit zu wenig Speicherplatz kommen, wenn Sie versuchen, solche Taxonomien mit Microsoft Excel 32-Bit zu laden.
- Damit XBRL-Berichte erstellt, validiert und exportiert werden können, benötigt das Add-in vollständigen Zugriff auf das Excel-Dokument. Falls Ihr Unternehmen Information Rights Management (IRM) mittels Azure Information Protection oder einer ähnlichen Technologie verlangt, kann der Zugriff auf das Excel-Dokument eingeschränkt sein. Informationen dazu, wie Sie Code im Hintergrund von Dokumenten mit eingeschränkten Berechtigungen ausführen lassen können, finden Sie in der [Microsoft-Dokumentation](#).

## 1.2 Installation und Lizenzierung

Um das EBA XBRL Add-in zu installieren, laden Sie die ausführbare Datei vom [Altova Download Center](#) herunter. Starten Sie die ausführbare Datei und befolgen Sie die Anweisungen des Installationsassistenten, um die Installation fertig zu stellen. Sie müssen der Lizenzvereinbarung und den Datenschutzbestimmungen zustimmen, um mit der Installation fortfahren zu können. Stellen Sie sicher, dass Sie die entsprechende ausführbare Datei für Ihr Betriebssystem und Ihre Excel-Plattform (32-Bit oder 64-Bit) herunterladen. Die ausführbare 32-Bit-Datei kann sowohl auf 32-Bit- als auch auf 64-Bit-Windows-Systemen installiert werden. Allerdings unterstützt die ausführbare 32-Bit-Datei nur Excel 32-Bit. Beachten Sie, dass, wenn Sie die 64-Bit-Version des Add-in für eine Excel-32-Version installieren, weiterhin die 32-Bit-Version ausgeführt wird.

Nach der Installation steht im Excel-Menüband ein neues Register namens **EBA** zur Verfügung. Um die aktuelle Version des Add-in zu sehen, öffnen Sie im Excel-Menüband das Add-in-Register und klicken Sie auf **About EBA Add-In**.

### *Mögliche Probleme bei der Installation*

Wenn Sie Excel nach der Installation des Add-in zum ersten Mal starten, werden die DLL-Dateien des Add-in in `C:\Benutzer\\AppData\Local\assembly\d13` kopiert. Eventuell werden die Ausführung der DLL-Dateien von diesem Ordner aus und die Installation des Add-in durch die [Microsoft ASR-Regeln](#) blockiert. Falls es zu Problemen mit der Installation des Add-in kommt, überprüfen Sie die für Ihr System konfigurierten Gruppenrichtlinien.

### Lizenzierung

Um das EBA XBRL Add-in verwenden zu können, benötigen Sie einen gültigen Lizenz-Keycode. Um einen neuen Keycode zu erwerben oder eine kostenlosen Testlizenz von der Altova Website anzufordern, öffnen Sie das Add-in-Menüband und klicken Sie auf **Add-In Activation**. Daraufhin sehen Sie eine Anleitung, wie Sie eine neue Lizenz von Altova erhalten oder eine vorhandene Lizenz verwalten.

Gehen Sie nach Erwerb der Lizenz von Altova vor, wie oben beschrieben, um das Aktivierungsfeld zu öffnen und die Lizenzdatei hochzuladen. Alternativ dazu können Sie die erworbenen Lizenzen über den in Ihrem Firmennetzwerk laufenden [Altova LicenseServer](#) hochladen. Altova LicenseServer ist ein kostenloses Produkt, mit dem Organisationen alle Altova-Lizenzen zentral verwalten können. Siehe auch [Lizenzinformationen](#)<sup>95</sup>.

## 2 Erstellen eines neuen Berichts

Dieser Abschnitt bietet eine Einstiegshilfe bei der Erstellung eines EBA-Berichts. Dieser Abschnitt ist in die folgenden Kapitel gegliedert:

- [Neuer Bericht](#) <sup>11</sup>
- [Dateneingabe](#) <sup>17</sup>
- [Dateneingabe in 3-D-Tabellen](#) <sup>21</sup>
- [Festlegen der Genauigkeit von Zellen](#) <sup>23</sup>
- [Validieren der Daten](#) <sup>25</sup>
- [Exportieren von Daten in XBRL](#) <sup>27</sup>
- [Importieren von Daten aus XBRL](#) <sup>28</sup>
- [Batch-Konvertierung von XBRL in Excel](#) <sup>29</sup>

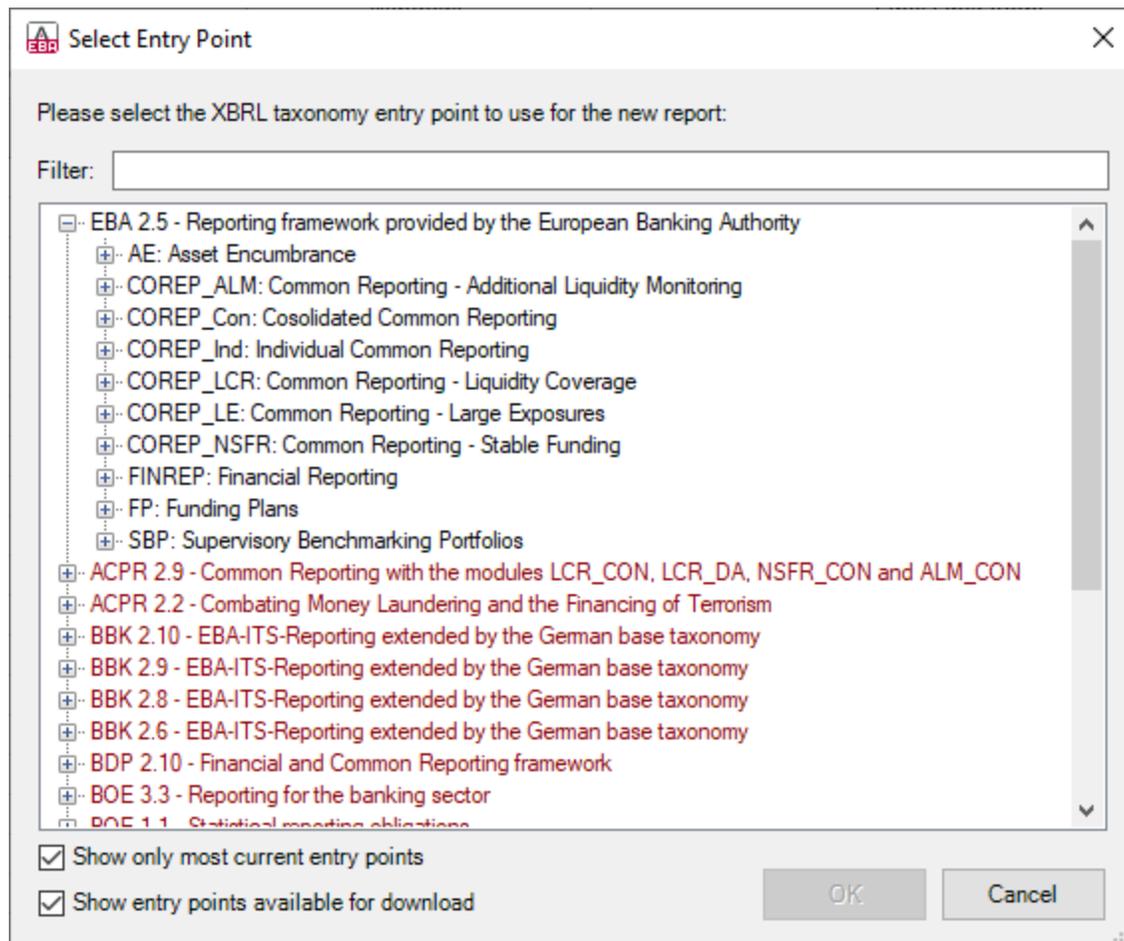
## 2.1 Neuer Bericht

In der nachstehenden Anleitung wird beschrieben, wie Sie auf Basis der im EBA XBRL Add-in verfügbaren Standard-EBA-Taxonomie einen neuen XBRL-Bericht erstellen. Diese XBRL-Taxonomie wird bei der Installation des Add-in standardmäßig auf Ihrem Rechner installiert. Zusätzliche Taxonomien können separat installiert werden. Nähere Informationen dazu finden Sie unter [XBRL-Taxonomie-Manager](#)<sup>31</sup>.

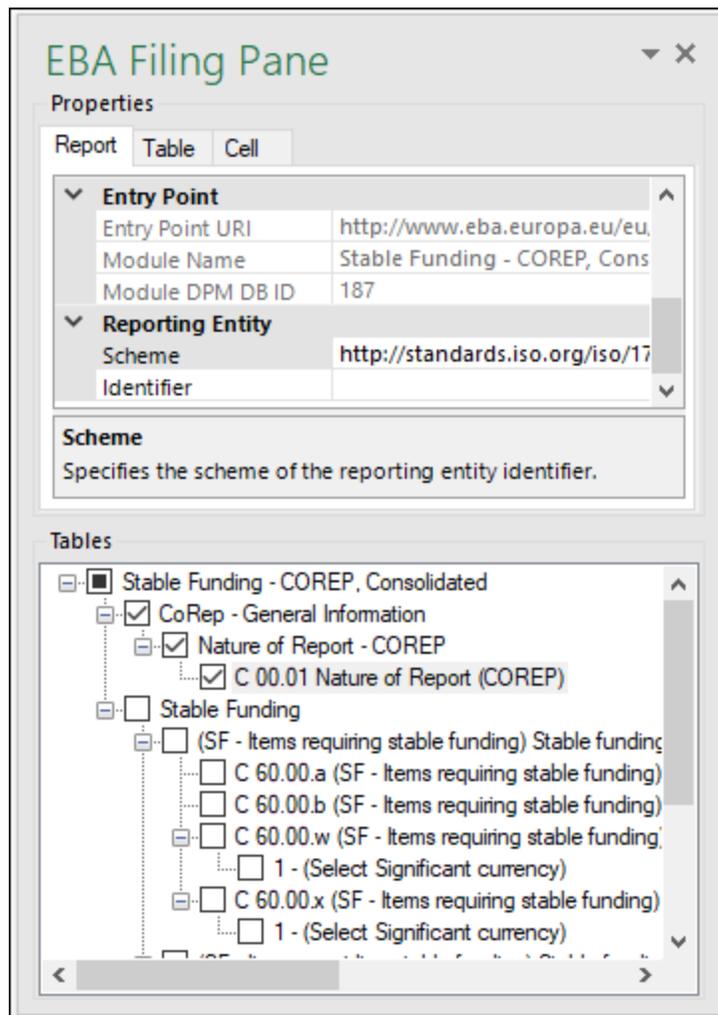
### Neuer Bericht

Um einen neuen Bericht zu erstellen, gehen Sie folgendermaßen vor:

1. Klicken Sie im Excel-Menüband auf das Add-in-Register.
2. Klicken Sie auf **Insert New Report**. Daraufhin wird das Dialogfeld **Select Entry Point** geöffnet.
3. Wählen Sie den gewünschten Taxonomie-Eintrittspunkt für Ihren Bericht aus (*siehe Abbildung unten*). Über den Filter am oberen Rand des Dialogfelds können Sie Eintrittspunkte nach Schlüsselwörtern filtern. Standardmäßig werden nur die aktuellsten Eintrittspunkte für die aktuelle Version des Add-in angezeigt. Um alle XBRL-Taxonomien, die zum Download zur Verfügung stehen, zu sehen, aktivieren Sie das Kontrollkästchen *Show entry points available for download*. Um alle Versionen zu sehen, deaktivieren Sie dieses Kontrollkästchen. Nicht installierte Eintrittspunkte werden rot angezeigt. Um die entsprechenden XBRL-Taxonomien zu installieren, wählen Sie den Eintrittspunkt aus und klicken Sie auf **OK**. Daraufhin wird der XBRL-Taxonomie-Manager geöffnet, über den Sie die Installation fertig stellen können. Nähere Informationen dazu finden Sie unter [XBRL-Taxonomie-Manager](#)<sup>31</sup>. Aufgrund von Arbeitsspeicheranforderungen können einige Eintrittspunkte in der 32-Bit-Version des Add-in nicht geladen werden. In diesem Fall erscheinen Sie im Dialogfeld ausgegraut. Um diese Eintrittspunkte laden zu können, verwenden Sie Excel 64-Bit und installieren Sie die 64-Bit Version des Add-in.



4. Im nächsten Schritt werden die Berichtstabellen in Excel geladen. Während dieses Vorgangs werden Sie in einem Dialogfeld über den Fortschritt informiert. Sobald die Berichtstabellen fertig geladen wurden, beachten Sie den Abschnitt *Tables* im Fenster EBA Filing (*siehe unten*).



5. Aktivieren Sie die Kontrollkästchen neben den Tabellen, die in den Bericht inkludiert werden sollen. Jede inkludierte Tabelle wird in der Excel-Arbeitsmappe auf einem neuen Arbeitsblatt angezeigt. Beachten Sie bitte, dass durch die ausgewählten Tabellen ein "true" Filing Indicator und durch nicht ausgewählte Tabellen ein "false" Filing Indicator generiert wird. Sie können Filing Indicators durch Aktivieren/Deaktivieren der entsprechenden Tabellen im Abschnitt *Tables* des Fensters definieren. Jede aktivierte Tabelle wird zum Bericht hinzugefügt und als eingereicht (filed) definiert; alle anderen werden nicht zum Bericht hinzugefügt.

Sie können nun mit der Eingabe der Daten in die Tabellen beginnen, die Daten validieren und ins XBRL-Format exportieren. Nähere Informationen dazu finden Sie in den folgenden Kapiteln:

- [Dateneingabe](#) <sup>17</sup>
- [Überprüfen der Genauigkeit von Zellen](#) <sup>23</sup>
- [Validieren der Daten](#) <sup>25</sup>
- [Exportieren der Daten in XBRL](#) <sup>27</sup>

### Aktualisieren des Eintrittspunkts (Enterprise Edition)

Wenn Sie bereits einen Bericht erstellt haben und zu einer anderen Version derselben Taxonomie (einer

neueren oder gegebenenfalls älteren) wechseln möchten, können Sie dies mit dem Menübandbefehl **Update Entrypoint** tun. Bei Auswahl dieses Befehls wird das Dialogfeld **Select Entry Point** aufgerufen (*siehe Unterabschnitt weiter oben*), über das Sie eine andere Version derselben Taxonomie auswählen und installieren können. Die aktuell verwendete Taxonomieversion ist im Dialogfeld **Select Entry Point** ausgegraut.

Wenn Sie eine andere Taxonomieversion für die Installation auswählen, wird diese Taxonomieversion in einer neuen Arbeitsmappe geöffnet. Beachten Sie, dass die Facts aus der ursprünglichen Arbeitsmappe, die den Concepts im neuen Eintrittspunkt vollständig entsprechen, in die neue Arbeitsmappe kopiert werden. Wenn die Facts aus irgendeinem Grund im neuen Eintrittspunkt nicht zur Verfügung stehen (z.B. weil die Datentypen in der ursprünglichen Arbeitsmappe nicht mit denen des neu installierten Eintrittspunkts übereinstimmen), können Sie diese Facts manuell aus der ursprünglichen Arbeitsmappe kopieren. Dadurch dass die ursprüngliche Arbeitsmappe erhalten bleibt, vermeiden Sie einen Verlust von Daten.

Alle Facts, die nicht in den neu installierten Eintrittspunkt eingefügt werden können, werden im Validierungsbericht angezeigt. Diese Informationen werden gleichzeitig im Validierungsbericht der ursprünglichen Version und in dem des neu installierten Eintrittspunkts angezeigt. Der Validierungsbericht enthält Links zu den Facts, die nicht in den neuen Bericht importiert werden können. Wenn Sie auf diese Links klicken, werden die entsprechenden Zellen im ursprünglichen Bericht markiert.

## 2.2 Filing-Fenster

Im Fenster EBA Filing können Sie Tabellen in den Bericht inkludieren und davon ausnehmen, Informationen zu jeder Zelle anzeigen und verschiedene Berichtseigenschaften definieren. Standardmäßig ist dieses Fenster sichtbar. Durch Auswahl des Befehls **Toggle EBA Filing Pane** des Menübands können Sie das Fenster ein- bzw. ausblenden. Das Fenster EBA Filing besteht aus zwei Hauptabschnitten: *Properties* und *Tables* (siehe *Unterabschnitte weiter unten*).

### Properties (Eigenschaften)

Die im Fenster EBA Filing angezeigten Eigenschaften wirken sich direkt auf den Inhalt der beim Export der XBRL-Instanz erstellten XBRL-Instanzdatei aus. Um zu sehen, was die einzelnen Eigenschaften bewirken, klicken Sie darauf und lesen Sie die im grauen Feld unterhalb des Rasters angezeigte Beschreibung. Ausgegraute Eigenschaften sind schreibgeschützt. Andernfalls kann eine Eigenschaft durch Eingabe von Text bzw. Auswahl eines Werts bearbeitet werden.

Die Eigenschaften *Scheme* und *Identifizier* unter *Reporting Entity* werden normalerweise von den jeweiligen zuständigen Stellen bereitgestellt.

Zwar beginnen einige Eigenschaftswerte mit `http` (z.B. *XSD-Eintrittspunkt-URI*), verweisen aber nicht notwendigerweise auf Live Webressourcen und sollten daher nicht als tote Links betrachtet werden. Zum Auflösen von Eintrittspunkt-URIs werden im Add-in mit Hilfe von Katalog-Mechanismen URIs auf Dateien im lokalen System gemappt. Der Grund dafür ist hauptsächlich die Größe der Taxonomien und die Tatsache, dass diese Tausende von Dateien enthalten. Ein Aufruf der Taxonomiedateien über das Internet würde das System extrem verlangsamen, selbst wenn die Herausgeber der Taxonomien diese auf diese Art zur Verfügung stellen würden.

Eigenschaften werden in die folgenden drei Register gruppiert:

- Auf dem Register "Report" werden Eigenschaften den gesamten Bericht betreffend angezeigt: Ein einziger Bericht entspricht einer einzigen Excel-Arbeitsmappe.
- Auf dem Register "Table" werden die Eigenschaften der aktuell ausgewählten Tabelle angezeigt. Eine Tabelle entspricht normalerweise einem einzelnen Excel-Arbeitsblatt. Immer wenn Sie daher in ein neues Excel-Blatt klicken, werden die Eigenschaften aktualisiert, um das neue Arbeitsblatt anzuzeigen.
- Auf dem Register "Cell" werden die Eigenschaften der aktuell ausgewählten Zelle angezeigt. Immer wenn Sie in eine neue Zelle klicken, werden die Eigenschaften der Zelle entsprechend aktualisiert.

Sie können die Eigenschaften im Zusammenhang mit der Genauigkeit (accuracy) auf Berichts-, Tabellen- oder Zellenebene definieren. Nähere Informationen dazu finden Sie unter [Festlegen der Genauigkeit von Zellen](#)<sup>23</sup>.

### Tables (Tabellen)

Um eine Tabelle in den Bericht zu inkludieren, aktivieren Sie das dazugehörige Kontrollkästchen im Fenster. Jede inkludierte Tabelle wird in der Excel-Arbeitsmappe auf einem neuen Arbeitsblatt angezeigt. Um ein bestimmtes Blatt aufzurufen, navigieren Sie, wie bei Excel üblich, dorthin oder klicken Sie im Fenster EBA Filing auf die entsprechende Tabelle. Um eine bestimmte Tabelle aus dem Bericht zu entfernen, deaktivieren Sie das Kontrollkästchen neben der Tabelle. Nicht ausgewählte Tabellen werden nicht in den Bericht inkludiert.

Einige Tabellen unterstützen eine Z-Achse (eine dritte Dimension). Informationen dazu, wie Sie eine Z-Achse zu einer Tabelle hinzufügen, finden Sie unter [Dateneingabe in 3-D-Tabellen](#)<sup>21</sup>.

Jede im Fenster EBA Filing angezeigte Berichtstabelle ist XBRL-gebunden, d.h. die Daten, die Sie direkt in die

Tabellenzellen eingeben, scheinen in der XBRL-Instanzdatei auf, sobald der Bericht fertig ist. Nähere Informationen dazu finden Sie unter [Exportieren der Daten in XBRL](#)<sup>27</sup>. Solange Sie noch an Ihren Berichtsdaten arbeiten, können Sie die Excel-Arbeitsmappe speichern und später wieder öffnen.

Arbeitsblätter, die Tabellen enthalten, sind an die XBRL-Taxonomie gebunden. Diese Tabellen dürfen daher nicht gelöscht werden. Es wird empfohlen, solche Blätter umzubenennen. Falls nötig, können Sie neue Blätter zur Arbeitsmappe hinzufügen. Diese Blätter sind jedoch nicht an die XBRL-Taxonomie gebunden. Infolgedessen werden diese Blätter ignoriert, wenn Sie die XBRL-Instanzdatei generieren.

## 2.3 Dateneingabe

Sie können einen Bericht mit Daten befüllen, indem Sie die Daten manuell in Zellen eingeben oder Werte einfügen. Bei einigen Zellen können Sie einen Wert aus einer vordefinierten Liste wie z.B. einer Länder- oder Währungsliste auswählen. In einigen Berichtstabellen müssen Sie eventuell neue Zeilen oder Spalten hinzufügen. Im Folgenden finden Sie Informationen darüber, wie Sie Daten eingeben können.

### Editierbare im Gegensatz zu nicht editierbaren Zellen

Grau angezeigte Zellen dürfen im Allgemeinen nicht bearbeitet werden. Nur Zellen, die in den XBRL-gebundenen Bereich (der durch Tabellenumrandungen begrenzt ist) inkludiert sind, sind zu bearbeiten. Nähere Informationen über verschiedene Zellen und die darin einzugebenden Daten finden Sie in den Zelleninformationen (Eigenschaften) auf dem Register "Cell" des Fensters EBA Filing,

### Einfügen von Daten

Wenn Sie Daten aus mehreren Spalten einfügen, sollte die Anzahl der eingefügten Spalten der Anzahl der Spalten im vordefinierten Blatt entsprechen. Wenn Sie irrtümlich mehr Spalten des Typs "Text" außerhalb der Standardtabelle einfügen, scheinen nicht gewünschte Spalten unter Umständen außerhalb des XBRL-gebundenen Bereichs auf. Um die nicht gewünschten Spalten zu löschen, klicken Sie mit der rechten Maustaste in die Zelle und wählen Sie den Befehl **Delete | Table Columns**. Damit Excel nicht automatisch neue Spalten und Zeilen hinzufügt, gehen Sie zu **Datei | Optionen | Dokumentprüfung | AutoKorrektur-Optionen | AutoFormat während der Eingabe | Während der Eingabe anwenden** und deaktivieren Sie das Kontrollkästchen *Neue Zeilen und Spalten in Tabelle inkludieren*.

Wenn Sie Daten einfügen, wird empfohlen, nur die Werte, nicht aber die Formatierung beizubehalten. Wählen Sie dazu beim Einfügen von Zellen oder Zeilen die Option **Werte einfügen** aus.

### Tatsächlicher Zellenwert im Gegensatz zum angezeigten Zellenwert

Das Add-in ignoriert beim Generieren der XBRL-Instanzdatei alle Zellenformatierungsinformationen und exportiert den *tatsächlichen* Wert der Excel-Zelle. Beachten Sie allerdings, dass der tatsächliche Wert aufgrund der Zellenformatierungsinformationen nicht unbedingt dem in der Zelle angezeigten Wert entsprechen muss. Sie sehen den tatsächlichen Wert, der in die XBRL-Instanz geschrieben wird, in der Formelleiste von Excel (*siehe Beispiel unten*).

	A	B	C	D	E	F	G
1	C 66.01.a Maturity ladder. Total. Overnight and higher maturity						
2							
3							Overnight
4							0020
5					Liabilities resulting from	0010	12.345,00

Im obigen Beispiel ist der Wert, der in die XBRL-Instanz geschrieben wird, 12345. Beachten Sie, dass die Genauigkeit der in der XBRL-Instanzdatei ausgegebenen Zahl auch von dem in den *Accuracy*-Eigenschaften ausgewählten Wert abhängt. Nähere Informationen über die Genauigkeit von Zellen finden Sie unter [Festlegen der Genauigkeit von Zellen](#) <sup>23</sup>.

## Enumerationswerte

In einigen Zellen wird ein festgelegter vordefinierter Wert erwartet (z.B. in Zellen, die Währungen oder Länder repräsentieren). In diesem Fall wird im Add-in ein kleiner Tooltip angezeigt, wenn Sie in die Zelle klicken. Sie können den erforderlichen Wert aus der Dropdown-Liste auswählen:

	A	B	C	D	E	F	G	H
1						F 34.00.c (AE-CONT) Asset encumbrance: Contingent encumbr		
2								
3						Sheet per Significant currency		
4								
5						Contingent Encumbrance B. Net ef		

Um die vollständige Liste aller möglichen Werte zu sehen, klicken Sie in die Zelle und werfen Sie einen Blick in die Zelleigenschaften des Registers "Cell" des Fensters EBA Filing.

## Bedingte Zellen

In einigen Tabellen muss zuerst ein Zellwert ausgefüllt werden, damit andere Zellen der Tabelle editierbar werden. So muss z.B. die Zelle F3 ausgefüllt werden, bevor alle anderen Zellen in derselben Spalte bearbeitet werden können (siehe Abbildung unten).

	A	B	C	D	E	F	G	H
1	F 34.00.c (AE-CONT) Asset encumbrance: Contingent encumbr							
2								
3	Sheet per Significant currency							
4								
5						Contingent Encumbr		
6						B. Net ef 10% depreciation of significant currencies		
7						Additional amount of encumbered assets		
8						026		
9	Carrying amount of selected				010			
10	Derivatives				020			
11	of which: Over-The-				030			
12	Deposits				040			
13	Repurchase agreements				050			
14	of which: central banks				060			
15	Collateralised deposits				070			
16	of which: central banks				080			
17	Debt securities issued				090			

Select Significant currency  
Please select a value from the list.

F 33.00.b | F 34.00.a | F 34.00.b | **F 34.00.c (1)** (+)

### Zellen mit mehreren Werten

Je nach XBRL-Taxonomie haben einige Berichte eventuell Facts, die eine Liste mehrerer kommasetrennter Werte repräsentieren. Daher müssen auch mehrere Werte in dieselbe Zelle eingegeben werden. Um Daten für solche Zellen einzugeben, erweitern Sie die Dropdown-Liste und klicken Sie auf alle relevanten Einträge. Alternativ dazu können Sie alle numerischen Werte, getrennt durch ein Komma, eingeben (*siehe Abbildung unten*). Wie bereits erwähnt, können alle möglichen Werte einer Zelle auf dem Register "Cell" des Fensters EBA Filing angezeigt werden.

	A	B	C
1	S.25.01.21.03 Basic Solvency Capital Requirement (U)		
2			
3			USP
4			C0090
5	Life underwriting risk	R0030	
6	Health underwriting risk	R0040	2,3
7	Non-life underwriting risk	R0050	

Nachdem Sie die Zelle mit mehreren Werten verlassen haben, wird diese Zelle automatisch neu gezeichnet, damit alle ausgewählten Werte in einer lesbaren Form angezeigt werden (auch wenn Sie möglicherweise nur Zahlen eingegeben haben).

## Neue Zeilen

In einigen Tabellen müssen Sie eventuell neue Zeilen erstellen. Dies ist z.B. bei *Tabelle C 10.02* aus dem Eintrittspunkt *EBA 2.6 COREP CON* der Fall. Sie können neue Zeilen mit den Excel-Standardbefehlen oder Tastenkürzeln hinzufügen. Alternativ dazu können Sie in der Menüleiste auf die Schaltfläche **Add Row** klicken. Um z.B. eine neue Zeile zur *Tabelle C 10.02* des oben erwähnten Eintrittspunkts hinzuzufügen, wählen Sie eine der folgenden Methoden:

- Öffnen Sie im Excel-Menüband das Add-in-Register und klicken Sie auf **Add Row | Insert Row Below**. Beachten Sie, dass die Befehle zum Einfügen oder Löschen von Zeilen nur dann aktiv sind, wenn das Hinzufügen neuer Zeilen für diese Tabelle unterstützt wird.
- Klicken Sie auf die am weitesten rechts gelegene Zelle der letzten Reihe in der Tabelle und drücken Sie die **Tabulatortaste**.
- Klicken Sie in eine Zelle in der leeren Zeile und wählen Sie im Kontextmenü den Befehl **Insert | Table Row Below**.

**Anmerkung:** Neu hinzugefügte Zeilen müssen sich innerhalb des klar schwarz umrandeten XBRL-gebundenen Bereichs der Tabelle befinden.

## Neue Spalten

Für einige Tabellen müssen Sie eventuell zusätzliche Spalten hinzufügen, d.h. diese Tabellen können in horizontaler Richtung wachsen. Sie können neue Spalten auf die folgenden Arten zu solchen Tabellen hinzufügen:

- Öffnen Sie im Excel-Menüband das Add-in-Register und klicken Sie auf **Add Column**. Beachten Sie, dass dieser Befehl nur dann aktiv ist, wenn das Hinzufügen neuer Spalten für diese Tabelle unterstützt wird.
- Klicken Sie auf die Schaltfläche **Add** neben der am weitesten rechts gelegenen Spalte der Tabelle.
- Klicken Sie mit der rechten Maustaste in eine Tabellenzelle und wählen Sie im Kontextmenü den Befehl **Insert | Table Columns to the Right**.

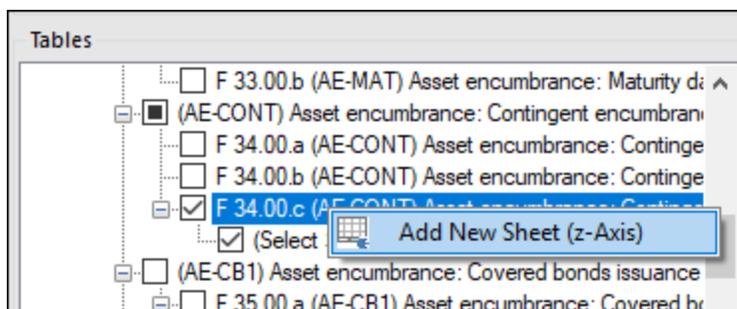
## 2.4 Dateneingabe in 3-D-Tabellen

Die meisten der Berichtstabellen haben nur zwei Dimensionen: die X-Achse (Spalten) und die Y-Achse (Zeilen). Bei einigen Tabellen müssen Sie aber unter Umständen Daten in eine dritte Dimension (die Z-Achse) eingeben, z.B. *Tabelle F 34.00.c (AE-CON) aus dem Eintrittspunkt AE-CON: Asset Encumbrance, Consolidated*. Für diese Tabelle benötigen Sie eventuell ein zusätzliches Blatt für die einzelnen Währungen. Wie unten gezeigt, handelt es sich bei der Zelle F3 um eine Dropdown-Liste, aus der Sie eine Währung auswählen können.

	A	B	C	D	E	F	G	H	
1	F 34.00.c (AE-CONT) Asset encumbrance: Contingent encumbr								
2									
3	Sheet per Significant currency								
4									
5						Continge Encumbr B. Net ef			

In Fällen wie dem obigen können Sie ein neues Blatt folgendermaßen entlang der Z-Achse (dritte Dimension) der Tabelle hinzufügen:

1. Klicken Sie im Excel-Menüband auf das Add-in-Register.
2. Klicken Sie auf die Schaltfläche **Add Sheet (z-Axis)**. Beachten Sie, dass die Befehle zum Einfügen oder Löschen neuer Z-Achsen-Blätter nur dann aktiv sind, wenn das Hinzufügen der Z-Achse für diese Tabelle von der Taxonomie unterstützt wird. Klicken Sie alternativ dazu mit der rechten Maustaste in das Fenster EBA Filing und wählen Sie im Kontextmenü den Befehl **Add New Sheet (z-Axis)** (siehe *Abbildung unten*). Dadurch wird ein neues Blatt erstellt, auf dem die dritte Dimension der Tabelle (Z-Achse) angezeigt wird. Das Blatt für die Z-Achse hat immer einen bezeichnenden Namen, der dem der Originaltabelle ähnelt.



Daten aus der dritten Dimension (Z-Achse) einer Tabelle werden als neue Blätter in Excel dargestellt. Dreidimensionale Tabellen umfassen daher mehr als in Arbeitsblatt. Dies stellt eine Ausnahme für die Regel dar, dass ein einziges Excel-Blatt einer einzigen Tabelle in einem XBRL-Bericht entspricht. In der XBRL-Instanz werden Daten, die zur Z-Achse gehören, korrekt als Teil derselben Tabelle ausgegeben.

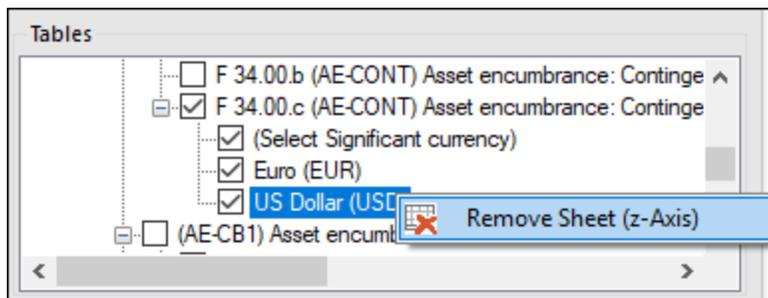
Wenn Sie in eine Zelle klicken, die die Z-Achse repräsentiert, werden alle möglichen Werte der Dropdown-Liste auf dem Register "Cell" des Fensters EBA Filing (siehe *Abbildung unten*) angezeigt.

Dimension	
Name	CUS
Label	Currency with significant liability
Verbose Label	Defines the currencies of the si
DPM DB ID	775
Type	enumeration
Possible Values	
Afghani (AFN)	eba_CU:AFN
Algerian Dinar (DZD)	eba_CU:DZD
Argentine Peso (ARS)	eba_CU:ARS
Armenian Dram (AMD)	eba_CU:AMD
Aruban Florin (AWG)	eba_CU:AWG
Australian Dollar (AUD)	eba_CU:AUD

## Löschen von Z-Achsen-Blättern

Sie können Blätter, die Daten aus der dritten Dimension (Z-Achse) enthalten, folgendermaßen löschen:

1. Wählen Sie das entsprechende Arbeitsblatt aus oder klicken Sie im Abschnitt *Tables* des Fensters EBA Filing auf den entsprechenden Eintrag.
2. Klicken Sie auf dem Register EBA auf **Delete Sheet (z-Axis)**. Klicken Sie alternativ dazu mit der rechten Maustaste in das Fenster EBA Filing und wählen Sie im Kontextmenü den Befehl **Delete Sheet (z-Axis)** (siehe Abbildung unten).



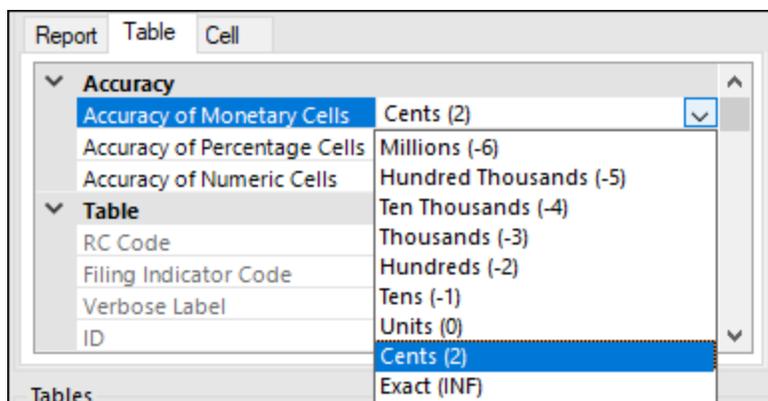
## 2.5 Festlegen der Genauigkeit von Zellen

Sie können die Genauigkeit monetärer und anderer numerischer Werte im XBRL-Bericht durch Definieren der Genauigkeit von monetären Zellen, numerischen Zellen und Prozentwertzellen festlegen. Diese Eigenschaften stehen im Fenster EBA Filing im Abschnitt *Properties* zur Verfügung.

Der Parameter "accuracy" gibt den Genauigkeitsgrad von in Berichten ausgegebenen Facts an und definiert die Genauigkeit des ausgegebenen Fact in Form der Anzahl der Dezimalstellen an. Der Genauigkeitsgrad wird mit Hilfe des Attributs `decimals` kommuniziert. Das `decimals`-Attribut kann eine Ganzzahl oder ein INF-Wert sein. Der INF-Wert repräsentiert den genauen Wert eines ausgegebenen Fact. In der Abbildung unten entspricht der Genauigkeitswert in Klammern dem Wert des Attributs `decimals` in der XBRL-Instanzdatei.

Bei monetären und numerischen Zellen kann der `decimals`-Wert positiv oder negativ sein. Ein positiver Wert  $N$  definiert die Genauigkeit von bis zu  $N$  Stellen rechts vom Dezimaltrennzeichen. So gibt der Wert 2 etwa an, dass monetäre Zellen mit einer Genauigkeit von Cent angegeben werden müssen. Ein negativer Wert  $N$  definiert die Genauigkeit von bis zu  $N$  Stellen links vom Dezimaltrennzeichen. So gibt der Wert -3 etwa an, dass die Genauigkeit bis zu Tausend sein muss, während der Wert -6 angibt, dass die Genauigkeit bis zu Millionen sein muss. Nähere Informationen zur Genauigkeit (Accuracy) finden Sie in der [XBRL 2.1 Recommendation](#).

Sie können die Eigenschaften im Zusammenhang mit der Genauigkeit (accuracy) auf Berichts-, Tabellen- oder Zellenebene definieren (siehe Abbildung unten). Wenn Sie die Genauigkeit auf mehreren Ebenen definieren, setzt die spezifischere Eigenschaft immer die allgemeinere außer Kraft. So hat etwa die Genauigkeit auf Zellenebene Vorrang vor der Genauigkeit auf Tabellenebene.



### Accuracy of monetary cells (Genauigkeit von monetären Zellen)

Die Eigenschaft *Accuracy of Monetary Cells* wird auf numerische Zellen, die einen monetären Wert darstellen, angewendet. Die von Ihnen ausgewählte Option definiert die Genauigkeit der Zahl relativ zum Dezimalkomma (bzw. Dezimalpunkt in englischen Systemen). Standardmäßig ist diese Eigenschaft auf *Cents (2)* gesetzt, d.h. die Zahl ist auf 2 Stellen rechts vom Dezimalkomma genau.

### Accuracy of percentage cells (Genauigkeit von Prozentwertzellen)

Die Eigenschaft *Accuracy of Percentage Cells* wird auf Werte, die einen Prozentwert darstellen, angewendet. Die von Ihnen ausgewählte Option definiert die Genauigkeit der Prozentzahl relativ zum Dezimalkomma (bzw. Dezimalpunkt in englischen Systemen). Standardmäßig ist diese Eigenschaft auf *Basis Points (4)* gesetzt, d.h. der Prozentwert ist auf 4 Stellen rechts vom Dezimalkomma genau.

### Accuracy of numeric cells (Genauigkeit von numerischen Zellen)

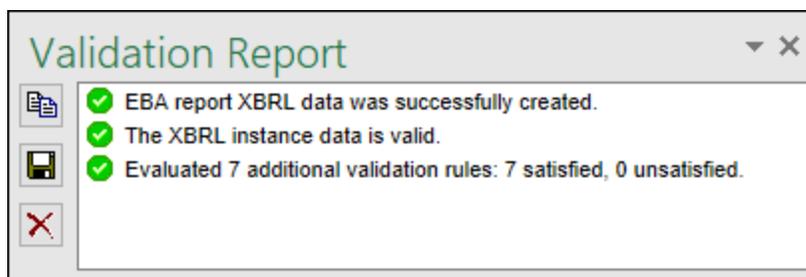
Die Eigenschaft *Accuracy of Numeric Cells* wird auf numerische Werte, bei denen es sich weder um einen

monetären Wert noch um einen Prozentwert handelt, angewendet. Die von Ihnen ausgewählte Option definiert die Genauigkeit der Zahl relativ zum Dezimalkomma (bzw. Dezimalpunkt in englischen Systemen). Standardmäßig ist diese Option auf *Exact (INF)* gesetzt.

## 2.6 Validieren von Daten

Mit der Validierung stellen Sie sicher, dass die eingereichten XBRL-Daten der XBRL-Spezifikation entsprechen. Ihre Berichtsdaten müssen vor dem Export unbedingt validiert werden. Um Daten zu validieren, klicken Sie auf dem Add-in-Register des Excel-Menübands auf die Schaltfläche **Validate**. Nachdem die Instanz fertig validiert wurde, wird ein Validierungsbericht ähnlich dem unten gezeigten angezeigt.

Wenn die Validierung fehlschlägt, werden im Validierungsfenster Links zu der Zelle, in der der Fehler aufgetreten ist, angezeigt. Um die Zelle mit dem Fehler schnell zu finden, klicken Sie auf den unterstrichenen Text, woraufhin der Cursor automatisch in die entsprechende Zelle springt. Beachten Sie, dass es Fälle gibt, in denen mehrere Zellen von einer einzelnen Validierungsüberprüfung betroffen sind. In solchen Fällen wird bei Klick auf den Fehler-Link eine der betroffenen Zellen ausgewählt.



### Validierungsergebnisse

Das Validierungsergebnis kann eine der folgenden Validierungsmeldungen enthalten:

Meldungstyp	Bedeutung
	Die Instanzdaten sind gültig.
	Die Instanzdaten sind gültig, enthalten aber Inkonsistenzen oder Warnungen.
	Die Instanzdaten sind nicht gültig.

#### Informationsmeldungen, Warnungen und Fehler

Zusätzlich dazu werden im Dialogfeld **Validation Report** eventuell die folgenden Arten von Meldungen angezeigt: Informationen, Warnungen und Fehler.

Meldungstyp	Bedeutung
	Dies ist eine Informationsmeldung. Durch Informationsmeldungen wird die XBRL-Instanz nicht ungültig.
	Dies ist eine Warnmeldung oder eine Meldung über eine Inkonsistenz. Durch Warnungen und Inkonsistenzen wird die XBRL-Instanz nicht ungültig.
	Kennzeichnet einen Fehler. Wenn Validierungsfehler vorhanden sind, ist die XBRL-Instanz nicht gültig. In diesem Fall müssen Sie die Berichtsdaten bearbeiten, um die einzelnen

Meldungstyp	Bedeutung
	<p>Fehler zu korrigieren, bevor Sie die XBRL-Daten exportieren.</p> <p>Bei der Validierung überprüft das Add-in XBRL Formula Assertions und gibt diese als Fehler aus. Wenn Sie <a href="#">Altova RaptorXML+XBRL Server</a> für die Validierung verwenden, können XBRL Formula Assertions optional so konfiguriert werden, dass sie nicht als Fehler ausgegeben werden.</p>

**Anmerkung:** Standardmäßig behandelt das Add-in ungültige Zellenwerte als Fehler. Gegebenenfalls können Sie das Add-in so konfigurieren, dass ungültige Zellenwerte stattdessen als Warnungen ausgegeben werden. Nähere Informationen dazu finden Sie unter [Einstellungen](#)<sup>52</sup>.

#### Kopieren, Speichern, Löschen

Um den Inhalt des Validierungsberichts in die Zwischenablage zu kopieren, klicken Sie auf  und fügen Sie den Inhalt in eine Zieldatei (z.B. eine E-Mail) ein. Klicken Sie alternativ dazu mit der rechten Maustaste in das Fenster "Validation Report" und wählen Sie im Kontextmenü den Befehl **Copy All Messages**.

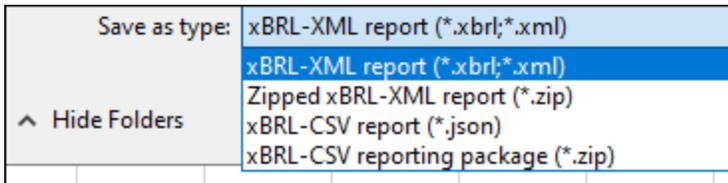
Um den Validierungsbericht als Text oder HTML zu speichern, klicken Sie auf . Klicken Sie alternativ dazu mit der rechten Maustaste in das Fenster "Validation Report" und wählen Sie im Kontextmenü den Befehl **Save Validation Report**.

Um den Validierungsbericht zu löschen, klicken Sie auf . Klicken Sie alternativ dazu mit der rechten Maustaste in das Fenster "Validation Report" und wählen Sie im Kontextmenü den Befehl **Clear**.

## 2.7 Exportieren von Daten in XBRL

Sobald Ihr Bericht fertig erstellt und gültig ist, können Sie eine XBRL-Instanzdatei generieren. Klicken Sie dazu im Add-in-Menüband auf **Export to XBRL**. Instanzdateien werden standardmäßig mit der Erweiterung **.xbrl** gespeichert. Wenn die exportierte Datei eine andere Erweiterung (z.B. **.xml**) erhalten soll, geben Sie die Dateierweiterung in das Dialogfeld **Export** ein.

Sie können Ihre Datei in die in der Abbildung unten gezeigten Formate exportieren.



Die Daten werden beim Export automatisch validiert. Fehler, Inkonsistenzen und Warnungen werden nach Beendigung des Exports auf dem Bildschirm angezeigt. Nähere Informationen dazu finden Sie unter [Validieren von Daten](#)<sup>25</sup>.

**Anmerkung:** Ungültige Zellenwerte, d.h. Zellen, die dem Datentyp des zugrunde liegenden XBRL Concept nicht entsprechen, verhindern, dass der Bericht exportiert werden kann.

Tipps dazu, wie Sie Datenformatierungsfehler vermeiden, finden Sie unter [Dateneingabe](#)<sup>17</sup>. Beachten Sie jedoch, dass nicht alle XBRL-Validierungsfehler aufgrund inkorrekt formatierter Daten entstehen. Einige Fehler treten auf, weil die Daten nicht den für den einzureichenden Bericht geltenden XBRL-Validierungsregeln entsprechen.

## 2.8 Importieren von Daten aus XBRL

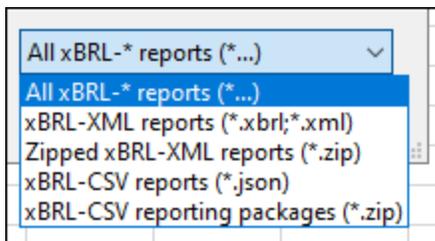
Sie können Daten aus vorhandenen XBRL-Berichtsinstanzen in Excel importieren (normalerweise Dateien mit der Erweiterung `.xbrl`). Damit die Daten erfolgreich importiert werden können, muss es sich dabei um gültige XBRL-Berichte handeln. Dies können zuvor mit **EBA XBRL Add-in** generierte Berichte sein oder Berichte, die Sie von anderen erhalten haben.

Um eine vorhandene XBRL-Instanzdatei in Excel zu importieren, gehen Sie folgendermaßen vor:

1. Klicken Sie im Excel-Menüband auf das Add-in-Register.
2. Klicken Sie auf **Import from XBRL** und navigieren Sie zur XBRL-Instanzdatei.

**Anmerkung:** Wenn in Excel bereits ein Bericht offen ist, ist die Schaltfläche **Import from XBRL** deaktiviert. Um den Befehl zu aktivieren, speichern und schließen Sie den aktuellen Bericht (Arbeitsmappe) und erstellen Sie eine neue Arbeitsmappe.

In der Abbildung unten sehen Sie, welche Formate importiert werden können.



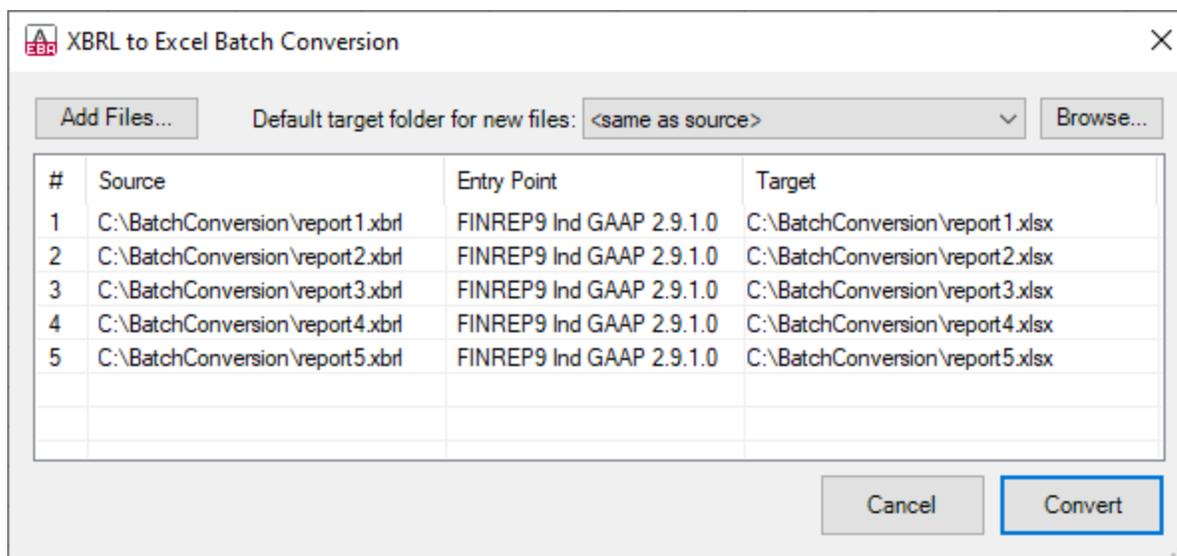
Beim Import werden Sie im Dialogfeld **Importing XBRL report** über den Fortschritt des Imports informiert. Während die Berichtsdaten in Excel geladen werden, werden sie automatisch validiert und im Dialogfeld über potenzielle Warnungen, Inkonsistenzen und/oder Fehler informiert. Nähere Informationen dazu finden Sie unter [Validieren von Daten](#) <sup>25</sup>.

**Anmerkung:** Während des Imports werden XBRL Formula Assertions validiert. Der Bericht wird jedoch auch bei Assertions, die nicht erfüllt werden, importiert.

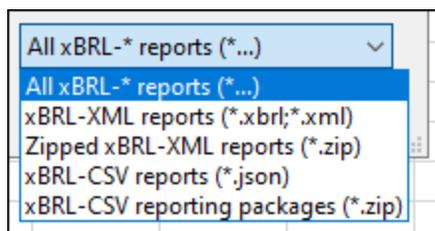
## 2.9 Batch-Konvertierung von XBRL in Excel

Mit Hilfe des Befehls **Batch Conversion** im Excel-Menüband können Sie mehrere XBRL-Instanzdateien in das Excel-Format konvertieren. Das Ergebnis ist dasselbe wie beim [Import](#)<sup>28</sup> mehrerer XBRL-Instanzdateien und dem anschließenden Speichern im Excel-Format. Der Hauptvorteil ist, dass die Konvertierung im Stapel erfolgt. Um eine Batch-Konvertierung durchzuführen, fügen Sie alle benötigten Dateien folgendermaßen zum Stapel hinzu:

1. Klicken Sie im Excel-Menüband auf das Add-in-Register.
2. Klicken Sie auf die Schaltfläche **Batch Conversion**.
3. Klicken Sie auf **Add Files**. Klicken Sie alternativ dazu mit der rechten Maustaste in das Raster und wählen Sie im Kontextmenü den Befehl **Add Files**.



In der Abbildung unten sehen Sie, welche Formate für die Batch-Konvertierung zur Verfügung stehen:



### Zusätzliche Optionen im Dialogfeld "Batch Conversion"

Im Konvertierungsdialogfeld können Sie die folgenden zusätzlichen Aufgaben durchführen:

1. Klicken Sie auf **Add Files**, um Dateien aus zusätzlichen Quellordnern zum selben Stapel hinzuzufügen.
2. Der Standard-Zielordner ist immer derselbe wie der Quellordner der jeweiligen Datei. Wenn Sie allen neuen Dateien standardmäßig einen bestimmten Zielordner zuweisen möchten, wählen Sie den

gewünschten Ordner aus der Liste namens *Default target folder for new files* aus. Um neue Einträge zur Liste hinzuzufügen, klicken Sie auf **Browse** und wählen Sie einen Ordner aus. Standardmäßig wirkt sich die Option *Default target folder for new files* auf neue Dateien, die Sie zum Stapel hinzufügen, aus. Wenn Sie diese Option jedoch ändern und die Dateien im Stapel (im Raster) bereits vorhanden sind, werden Sie informiert, dass die Änderung sich auf alle neuen zum Stapel hinzugefügten Dateien auswirkt. Klicken Sie auf **Yes**, wenn auch der Zielordner bereits vorhandener Dateien geändert werden soll.

3. Sie können auswählen, ob alle konvertierten Dateien im selben Zielordner gespeichert werden sollen oder Sie können für jede einzelne Datei einen anderen Zielordner auswählen. Um den Zielordner bestimmter Dateien zu ändern, wählen Sie die entsprechenden Dateien im Raster aus, klicken Sie mit der rechten Maustaste ins Raster und wählen Sie im Kontextmenü den Befehl **Set target folder**.
4. Sie können die Zieldateien umbenennen. Wählen Sie dazu zuerst die Dateien im Raster aus. Klicken Sie anschließend mit der rechten Maustaste in das Raster und wählen Sie im Kontextmenü den Befehl **Rename** (oder drücken Sie **F2**). Sie können sowohl den Dateinamen als auch den Dateipfad ändern. Wenn Sie den Pfad jedoch ändern, stellen Sie sicher, dass der Ordner vorhanden ist.
5. Um Dateien aus dem Stapel zu entfernen, wählen Sie sie aus, klicken Sie mit der rechten Maustaste ins Raster und wählen Sie im Kontextmenü den Befehl **Remove** (oder drücken Sie die **Entf**-Taste).

## Auswahl mehrerer Dateien

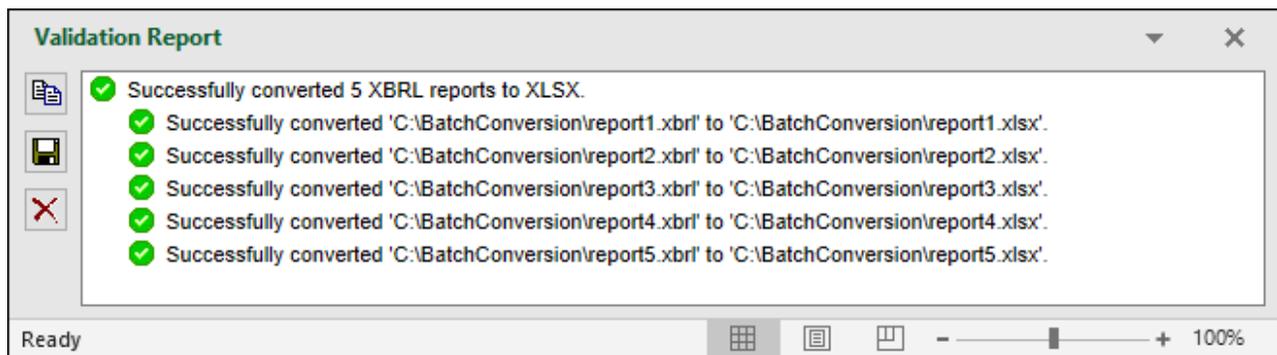
Um mehrere Dateien im Raster auszuwählen, verwenden Sie die Windows-Standard-Tastaturkürzel, z.B.:

- Klicken Sie bei gedrückter **Strg**-Taste auf die gewünschten Dateien, um sie auszuwählen.
- Klicken Sie im Dialogfeld in einen leeren Bereich und ziehen Sie den Cursor über die gewünschten Dateien (Rechteckauswahl).
- Drücken Sie **Strg+A**, um alle Dateien im Raster auszuwählen.

Sobald der Stapel fertig ist, klicken Sie auf **Convert**, um alle Dateien darin zu verarbeiten.

## Validierungsbericht

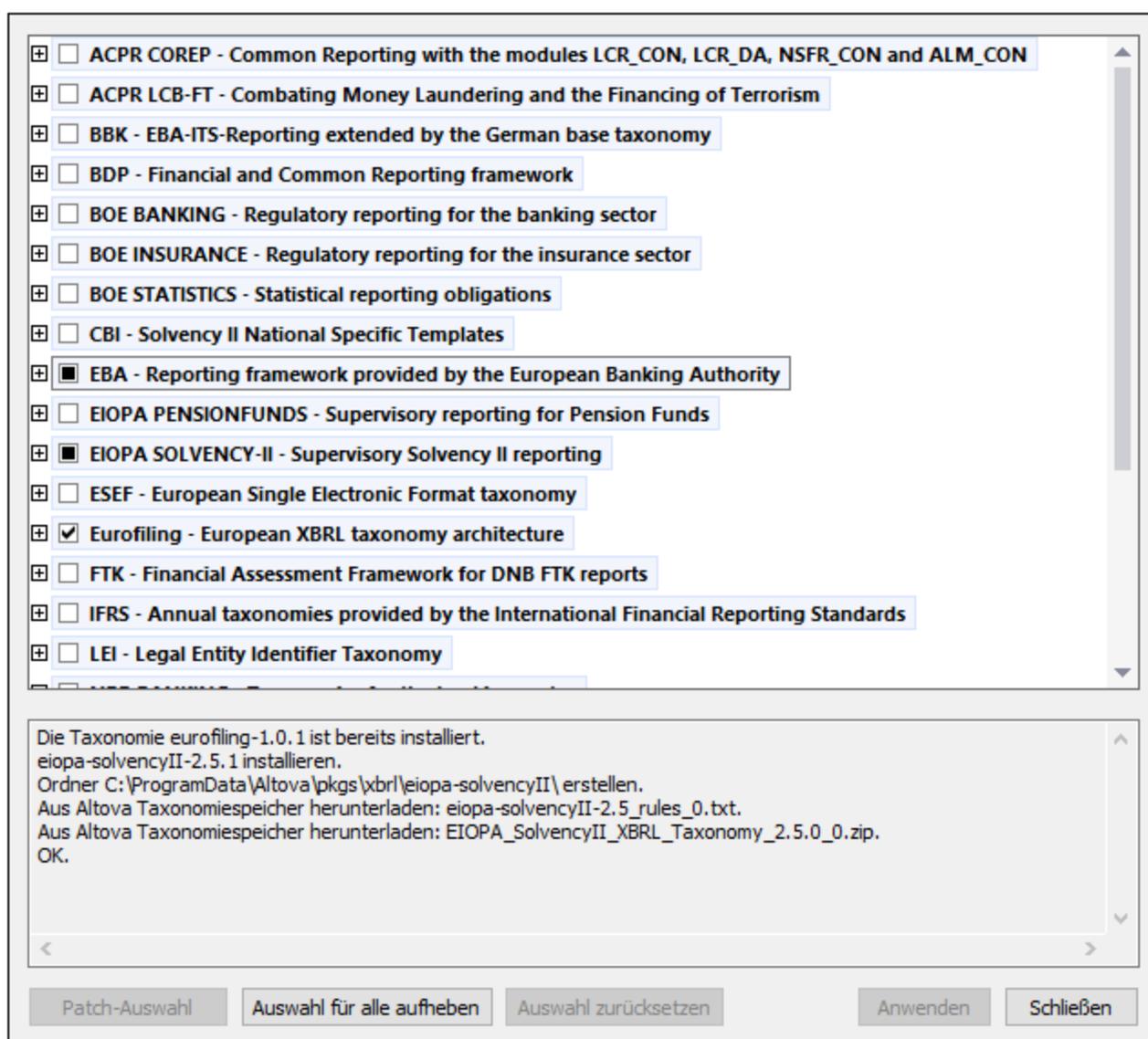
Das Ergebnis der Konvertierung wird im Fenster "Validation Report" ausgegeben (*siehe Abbildung unten*). Nähere Informationen zu Validierungsmeldungen finden Sie unter [Validieren von Daten](#) <sup>25</sup>.



### 3 Taxonomie-Manager

Der XBRL-Taxonomie-Manager ist ein Altova-Tool, mit dem Sie XBRL-Taxonomien zentral installieren und verwalten können, um diese in allen XBRL-fähigen Applikationen von Altova einschließlich EBA XBRL Add-in verwenden zu können.

- Unter Windows hat der Taxonomie-Manager eine grafische Benutzeroberfläche (siehe Abbildung unten) und steht auch über die Befehlszeile zur Verfügung. (Die Desktop-Applikationen von Altova stehen nur unter Windows zur Verfügung; siehe Liste unten).
- Unter Linux und MacOS steht der Taxonomie-Manager nur über die Befehlszeile zur Verfügung. (Die Server-Applikationen von Altova stehen unter Windows, Linux und macOS zur Verfügung; siehe Liste unten).



XBRL-fähige Applikationen von Altova

Desktop-Applikationen (nur Windows)	Server-Applikationen (Windows, Linux, macOS)
Altova XBRL Add-ins für Excel (EBA, ESEF, Solvency II, WIP)	MapForce Server (Standard und Advanced Edition)
MapForce Enterprise Edition	RaptorXML+XBRL Server
StyleVision Enterprise Edition	StyleVision Server
XMLSpy Enterprise Edition	

## Installation und Deinstallation des Taxonomie-Managers

Der Taxonomie-Manager wird bei der ersten Installation einer neuen Version der Altova Mission Kit Enterprise Edition oder einer der XBRL-fähigen Applikationen von Altova (*siehe Tabelle oben*) automatisch installiert.

Ebenso wird er auch automatisch entfernt, wenn Sie die letzte XBRL-fähige Applikation von Altova auf Ihrem Rechner deinstallieren.

## Taxonomie-Manager-Funktionalitäten

Im Taxonomie-Manager stehen die folgenden Funktionalitäten zur Verfügung:

- Anzeigen der auf Ihrem Rechner installierten XBRL-Taxonomien und Überprüfung, ob neue Versionen zum Download zur Verfügung stehen.
- Download neuer Versionen von XBRL-Taxonomien unabhängig vom Altova Produkt-Release-Zyklus. (Die Taxonomien werden von Altova online bereitgestellt und können über den Taxonomie-Manager heruntergeladen werden).
- Installation oder Deinstallation jeder beliebigen (oder ggf. aller) der zahlreichen Versionen einer bestimmten Taxonomie.
- Eine XBRL-Taxonomie kann Abhängigkeiten von anderen Taxonomien aufweisen. Bei der Installation oder Deinstallation einer bestimmten Taxonomie informiert Sie der Taxonomie-Manager über davon abhängige Taxonomien und installiert bzw. entfernt diese ebenfalls automatisch.
- Der Taxonomie-Manager ordnet Schema-Referenzen mit Hilfe des [XML-Katalogs](#) lokalen Dateien zu. Dadurch lassen sich große XBRL-Taxonomien schneller verarbeiten, als wenn sie sich unter einem entfernten Pfad befinden.
- Alle wichtigen Taxonomien werden über den Taxonomie-Manager bereitgestellt und regelmäßig auf die jeweils neuesten Version aktualisiert. Dadurch können alle Ihre Taxonomien zentral verwaltet werden und stehen allen XBRL-fähigen Applikationen von Altova jederzeit zur Verfügung.
- Im Taxonomie-Manager vorgenommene Änderungen werden für alle auf dem Rechner installierten Altova-Produkte wirksam.

## Funktionsweise

Alle in Altova-Produkten verwendeten XBRL-Taxonomien werden von Altova online bereitgestellt. Dieser Speicher wird bei Veröffentlichung neuer Versionen der Taxonomien aktualisiert. Im Taxonomie-Manager werden sowohl bei Aufruf über die Benutzeroberfläche als auch über das CLI Informationen über die neuesten verfügbare Taxonomien angezeigt. Sie können die gewünschten Taxonomien dann über den Taxonomie-Manager installieren, aktualisieren oder deinstallieren.

Taxonomien können vom Taxonomie-Manager auch auf eine weitere Art installiert werden. Sie können eine Taxonomie und die davon abhängigen Taxonomien auf der [Altova Website](https://www.altova.com/de/taxonomy-manager) (<https://www.altova.com/de/taxonomy-manager>) auswählen. Daraufhin wird auf der Website eine Datei des Typs `.altova_taxonomies` mit Informationen über Ihre ausgewählte Taxonomie zum Download vorbereitet. Bei Doppelklick auf diese Datei oder bei Übergabe an den Taxonomie-Manager über das CLI als Argument des Befehls `install`<sup>44</sup> installiert der Taxonomie-Manager die ausgewählten Taxonomien.

#### Lokaler Cache: Überprüfung Ihrer Taxonomien

Alle Informationen über installierte Taxonomien werden in einem zentralen Cache-Verzeichnis auf Ihrem Rechner aufgezeichnet. Das Verzeichnis befindet sich hier:

<i>Windows</i>	C:\ProgramData\Altova\pkgs\.cache oder <BENUTZER-HOME>\Dokumente\Altova\pkgs\.cache (der Pfad hängt von dem für das Feld <i>Taxonomies Folder</i> im Dialogfeld <b>Einstellungen</b> ausgewählten Verzeichnis ab)
<i>Linux</i>	/var/opt/Altova/pkgs/.cache
<i>macOS</i>	/var/Altova/pkgs

Dieses Cache-Verzeichnis wird regelmäßig mit dem neuesten Status der Taxonomien im Online-Speicher von Altova aktualisiert. Diese Aktualisierungen finden unter den folgenden Bedingungen statt:

- bei jedem Start von Taxonomie-Manager.
- Wenn Sie EBA XBRL Add-in zum ersten Mal an einem bestimmten Kalendertag starten.
- Wenn EBA XBRL Add-in länger als 24 Stunden geöffnet ist, findet alle 24 Stunden eine Aktualisierung des Cache statt.
- Sie können den Cache auch durch Ausführung des `update`<sup>47</sup>-Befehls über die Befehlszeilenschnittstelle aktualisieren.

Der Taxonomie-Manager kann somit Ihre installierten Taxonomien über den Cache ständig anhand der online verfügbaren Taxonomien auf der Altova Website überprüfen.

#### **Nehmen Sie keine manuellen Änderungen am Cache vor!**

Das lokale Cache-Verzeichnis wird automatisch auf Basis der installierten oder deinstallierten Taxonomien verwaltet; es sollte nicht manuell geändert oder gelöscht werden. Falls Sie den Taxonomie-Manager je in seinen Originalzustand zurücksetzen möchten, (i) führen Sie den CLI-Befehl `reset`<sup>45</sup> der Befehlszeilenschnittstelle und (ii) anschließend den Befehl `initialize`<sup>43</sup> aus. (Führen Sie alternativ dazu den Befehl `reset` mit der Option `-i` aus).

## HTTP-Proxy

Sie können für Taxonomie-Manager-Verbindungen einen HTTP-Proxy verwenden. Es werden die Proxy-Einstellungen des Windows-Systems verwendet.

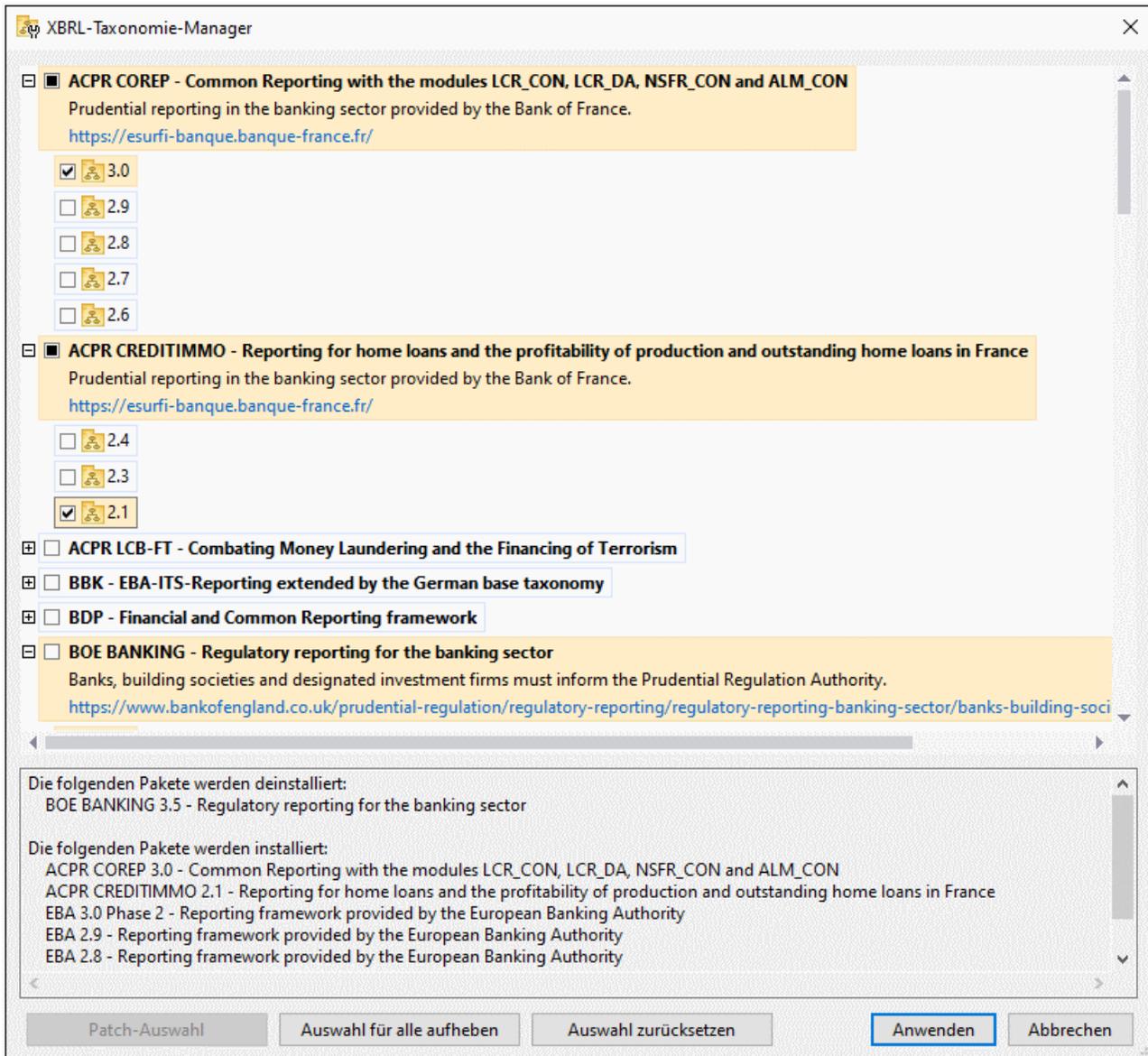
## 3.1 Ausführen des Taxonomie-Managers

### Grafische Benutzeroberfläche

Sie können die Benutzeroberfläche des Taxonomie-Managers auf eine der folgenden Arten aufrufen:

- *Bei der Installationen von EBA XBRL Add-in:* Aktivieren Sie gegen Ende der Installation das Kontrollkästchen *Altova-Taxonomie-Manager aufrufen*, wodurch Sie die Benutzeroberfläche des XBRL-Taxonomie-Managers direkt aufrufen können. Auf diese Art können Sie Taxonomien während der Installation Ihrer Altova-Applikation installieren.
- *Nach der Installation von EBA XBRL Add-in:* Nachdem die Applikation installiert wurde, können Sie die Benutzeroberfläche des Taxonomie-Managers jederzeit durch Aufruf des Befehls **Manage Taxonomies** über das Add-in-Menüband aufrufen.
- über die vom [XBRL-Taxonomie-Download-Center von Altova](#) heruntergeladene `.altova_taxonomies-` Datei: Doppelklicken Sie auf die heruntergeladene Datei, um den Taxonomie-Manager zu starten, der daraufhin die (auf der Website) ausgewählten Taxonomien installiert. (Um das Online-XBRL-Taxonomie-Download Center von Altova in Ihrem Browser zu öffnen und die gewünschten Taxonomien auszuwählen, klicken Sie im Add-in-Menüband auf **Altova on the Web | Taxonomy Download**).

Nachdem die Benutzeroberfläche des Taxonomie-Managers geöffnet wurde (*Abbildung unten*), werden bereits installierte Taxonomien markiert angezeigt. Wenn eine zusätzliche Taxonomie installiert werden soll, aktivieren Sie diese. Wenn eine bereits installierte Taxonomie deinstalliert werden soll, deaktivieren Sie diese. Nachdem Sie Ihre Auswahl getroffen haben, können Ihre Änderungen angewendet werden. Die Taxonomien, die installiert bzw. deinstalliert werden, werden markiert und im Fenster "Meldungen" am unteren Rand des Taxonomie-Manager-Fensters (*siehe Abbildung*) erscheint eine Meldung über die bevorstehenden Änderungen.



## Befehlszeilenschnittstelle

Sie können den Taxonomie-Manager über eine Befehlszeilenschnittstelle starten, indem Sie Befehle an die ausführbare Datei `taxonomymanager.exe` senden.

Die Datei `taxonomymanager.exe` steht im folgenden Ordner zur Verfügung:

- *unter Windows:* `C:\ProgramData\Altova\SharedBetweenVersions`
- *Unter Linux oder macOS (nur Server-Applikationen):* `%INSTALLDIR%/bin`, wobei `%INSTALLDIR%` das Installationsverzeichnis des Programms ist.

Anschließend können Sie jeden der im Abschnitt zur CLI-Befehlsreferenz aufgelisteten Befehle verwenden.

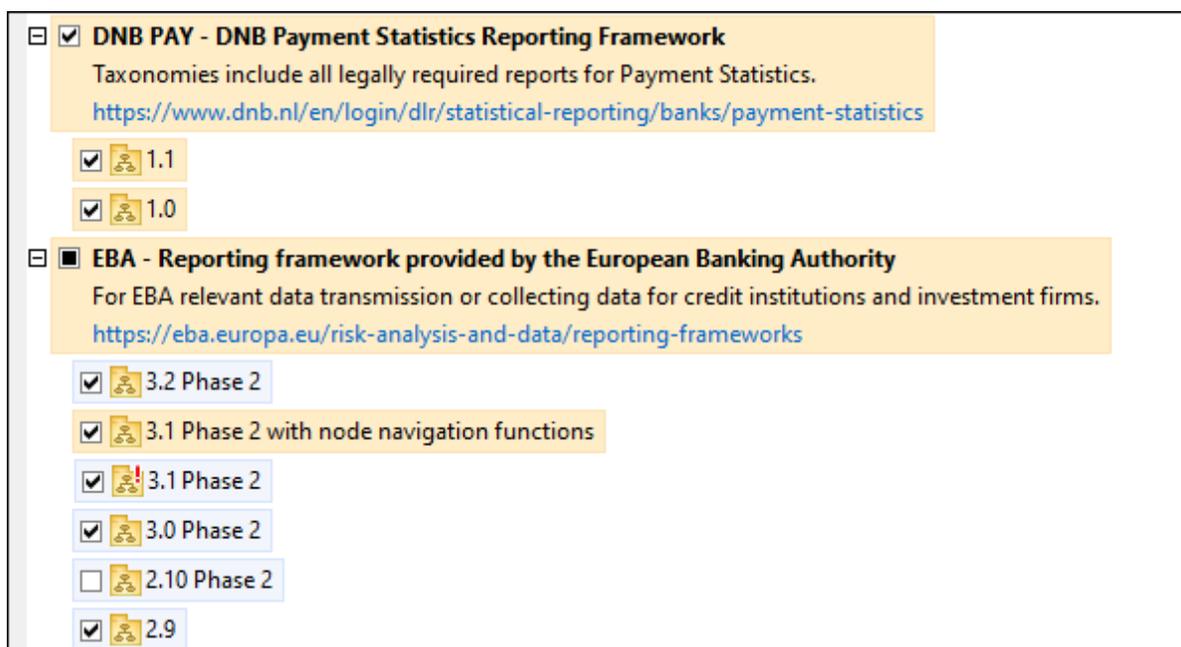
Um die Hilfe zu den Befehlen anzuzeigen, führen Sie den folgenden Befehl aus:

- *unter Windows:* `taxonomymanager.exe --help`
- *Unter Linux oder macOS (nur Server-Applikationen):* `sudo ./taxonomymanager --help`

## 3.2 Statuskategorien

Der Taxonomie-Manager unterscheidet folgendermaßen zwischen den von ihm verwalteten Taxonomien:

- *Installierte Taxonomien* Diese werden auf der Benutzeroberfläche mit einem Häkchen angezeigt (*in der Abbildung unten sind die mit einem Häkchen versehenen Versionen der DNB- und EBA-Taxonomie installiert*). Wenn alle Versionen einer Taxonomie ausgewählt sind, wird ein Häkchen angezeigt. Wenn zumindest eine Version nicht ausgewählt ist, wird ein gefülltes Quadrat angezeigt. Sie können die Auswahl für eine installierte Taxonomie aufheben, um sie zu **deinstallieren**.
- *Nicht installierte verfügbare Taxonomien*. Diese werden auf der Benutzeroberfläche mit einem deaktivierten Kontrollkästchen angezeigt. Sie können die Taxonomien, die **installiert** werden sollen, auswählen.



- *Taxonomien, für die ein Upgrade zur Verfügung steht* sind diejenigen, die seit ihrer Installation vom Herausgeber überarbeitet wurden. Sie werden auf der Benutzeroberfläche durch ein  Symbol gekennzeichnet (*siehe Abbildung oben*). Sie können für eine installierte Taxonomie ein **Patch** der verfügbaren überarbeiteten Version installieren.

### Wichtige Punkte

- In der Abbildung oben sind beide DNB-Taxonomien und einige der EBA-Taxonomien ausgewählt. Diejenigen mit einem blauen Hintergrund sind bereits installiert. Diejenige mit gelbem Hintergrund sind nicht installiert und wurden für die Installation ausgewählt. Beachten Sie, dass (i) die EBA 2.10 Phase 2-Taxonomie nicht installiert wurde und nicht für die Installation ausgewählt wurde, (ii) die EBA 3.1 Phase 2-Taxonomie installiert wurde, dafür aber seit ihrer Installation ein Patch vom Herausgeber zur Verfügung gestellt wurde, das noch nicht installiert wurde.
- Bei Ausführung des Taxonomie-Managers über die Befehlszeile wird der Befehl `list`<sup>44</sup> mit verschiedenen Optionen verwendet, um verschiedene Taxonomiekategorien aufzulisten:

<code>taxonomymanager.exe list</code>	Listet alle installierten und verfügbaren Taxonomien auf; auch verfügbare Upgrades werden angezeigt.
<code>taxonomymanager.exe list -i</code>	Listet nur installierte Taxonomien auf; auch verfügbare Upgrades werden angezeigt.
<code>taxonomymanager.exe list -u</code>	Listet für Taxonomien auf, für die Upgrades zur Verfügung stehen

**Anmerkung:** Verwenden Sie unter Linux und macOS, `sudo ./taxonomymanager list`

## 3.3 Anwenden eines Patch oder Installation einer Taxonomie

### Anwenden eines Patch auf eine installierte Taxonomie

Von Zeit zu Zeit werden von den Herausgebern der XBRL-Taxonomien Patches (Upgrades oder Überarbeitungen) veröffentlicht. Wenn der Taxonomie-Manager erkennt, dass Patches zur Verfügung stehen, werden diese in der Taxonomieliste des Taxonomie-Managers angezeigt und Sie können diese Patches schnell installieren.

#### Über die Benutzeroberfläche

Patches werden mit dem Symbol  gekennzeichnet. (Siehe auch vorhergehendes Kapitel über [Statuskategorien](#)<sup>37</sup>). Falls Patches zur Verfügung stehen, ist die Schaltfläche **Patch-Auswahl** aktiv. Klicken Sie darauf, um alle Patches für die Installation auszuwählen und vorzubereiten. Auf der Benutzeroberfläche ändert sich das Symbol von Taxonomien, für die ein Patch installiert wird von  in , und im Fenster "Meldungen" am unteren Rand des Dialogfelds werden die Patches, die angewendet werden, aufgelistet. Sobald Sie mit der Auswahl fertig sind, klicken Sie auf **Anwenden**. Alle Patches werden gemeinsam angewendet. Beachten Sie, dass eine für die Installation eines Patch markierte Taxonomie deinstalliert wird, wenn Sie die Auswahl für diese Taxonomie aufheben.

#### Über das CLI

So wenden Sie einen Patch über die Befehlszeilenschnittstelle an:

1. Führen Sie den Befehl `list -u`<sup>44</sup> aus. Daraufhin werden alle Taxonomien, für die Patch Upgrades zur Verfügung stehen, aufgelistet.
2. Führen Sie den Befehl `upgrade`<sup>47</sup> aus, um alle Patches zu installieren.

### Installieren einer verfügbaren Taxonomie

Sie können Taxonomien entweder über die Benutzeroberfläche des Taxonomie-Managers oder durch Senden der Taxonomie-Manager-Installationsbefehle über die Befehlszeile installieren.

**Anmerkung:** Wenn die aktuelle Taxonomie andere Taxonomien referenziert, werden auch die referenzierten Taxonomien installiert.

#### Über die Benutzeroberfläche

Um Taxonomien über die Benutzeroberfläche des Taxonomie-Managers zu installieren, wählen Sie die gewünschten Taxonomien aus und klicken Sie auf **Anwenden**.

Sie können die gewünschten Taxonomien auch auf der [Altova Website](#) auswählen und eine herunterladbare `.altova_taxonomies`-Datei generieren. Bei Doppelklick auf diese Datei wird der Taxonomie-Manager aufgerufen, in dem die gewünschten Taxonomien bereits vorausgewählt sind. Sie müssen nur mehr auf **Anwenden** klicken.

#### Über das CLI

Um Taxonomien über die Befehlszeile zu installieren, rufen Sie den Befehl `install`<sup>44</sup> auf:

```
taxonomymanager.exe install [options] Taxonomy+
```

wobei es sich bei `taxonomy` um die gewünschte(n) Taxonomie(n) bzw. eine `.altova_taxonomies`-Datei handelt. Eine Taxonomie wird von einem Identifier im Format `<name>-<version>` referenziert. (Die Identifier von Taxonomien werden angezeigt, wenn Sie den Befehl `list`<sup>44</sup> ausführen.) Sie können beliebig viele Taxonomien eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls `install`<sup>44</sup>.

**Anmerkung:** Verwenden Sie unter Linux oder macOS den Befehl `sudo ./taxonomymanager`.

#### Installation einer benötigten Taxonomie

Wenn Sie einen XBRL-Befehl in EBA XBRL Add-in ausführen und EBA XBRL Add-in erkennt, dass eine zur Ausführung des Befehls erforderliche Taxonomie nicht vorhanden oder unvollständig ist, wird der Taxonomie-Manager mit Informationen über die fehlende Taxonomie aufgerufen. Sie können die gewünschten Taxonomien dann über den Taxonomie-Manager direkt installieren.

Alle bereits installierten Taxonomien können jederzeit durch Aufruf des Taxonomie-Managers über **Extras | Taxonomie-Manager** über die Benutzeroberfläche des Taxonomie-Managers angezeigt werden.

## 3.4 Deinstallieren einer Taxonomie, Zurücksetzen

### Deinstallieren einer Taxonomie

Sie können Taxonomien entweder über die Benutzeroberfläche des Taxonomie-Managers oder durch Senden der Taxonomie-Manager-Deinstallationsanweisungen über die Befehlszeile deinstallieren.

**Anmerkung:** Wenn die gewünschte Taxonomie andere Taxonomien referenziert, so werden auch die referenzierten Taxonomien deinstalliert.

#### Über die Benutzeroberfläche

Um Taxonomien über die Benutzeroberfläche des Taxonomie-Managers zu deinstallieren, deaktivieren Sie die Kontrollkästchen der entsprechenden Taxonomien und klicken Sie auf **Anwenden**. Daraufhin werden die ausgewählten Taxonomien und die davon referenzierten Taxonomien deinstalliert.

Um alle Taxonomien zu deinstallieren, klicken Sie auf **Auswahl für alle aufheben** und anschließend auf **Anwenden**.

#### Über das CLI

Um Taxonomien über die Befehlszeile zu deinstallieren, rufen Sie den Befehl `uninstall` auf:

```
taxonomymanager.exe uninstall [options] Taxonomy+
```

wobei es sich beim Argument `Taxonomy` eine zu deinstallierende Taxonomie oder eine `.altova_taxonomies`-Datei handelt. Eine Taxonomie wird von einem Identifier im Format `<name>-<version>` definiert. (Die Identifier von Taxonomien werden angezeigt, wenn Sie den Befehl `list`<sup>44</sup> ausführen.) Sie können beliebig viele Taxonomien eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls `uninstall`<sup>46</sup>.

**Anmerkung:** Verwenden Sie unter Linux oder macOS den Befehl `sudo ./taxonomymanager`.

### Zurücksetzen des Taxonomie-Managers

Sie können den Taxonomie-Manager zurücksetzen.

- Klicken Sie auf der Benutzeroberfläche auf **Auswahl zurücksetzen**. Damit wird die Benutzeroberfläche zurückgesetzt, und Sie sehen, welche Taxonomien derzeit installiert sind. Jegliche Auswahl bzw. Aufhebung einer Auswahl, die der Benutzer in der aktuellen Sitzung vorgenommen hat, wird aufgehoben.
- Führen Sie über die Benutzeroberfläche den Befehl `reset`<sup>45</sup> aus. Damit werden alle installierten Taxonomien und das Cache-Verzeichnis entfernt.

Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize`<sup>43</sup> ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl `reset`<sup>45</sup> mit der Option `-i` aus.

Beachten Sie, dass mit `reset-i`<sup>45</sup> die Originalinstallation des Produkts wiederhergestellt wird, daher wird empfohlen, nach dem Zurücksetzen auch den Befehl `update`<sup>47</sup> auszuführen. Führen Sie alternativ dazu den Befehl `reset`<sup>45</sup> mit den Optionen `-i` und `-u` aus.

## 3.5 Befehlszeilenschnittstelle (CLI)

Um den Taxonomie-Manager über die Befehlszeile aufzurufen, müssen Sie den Pfad zur ausführbaren Datei kennen. Standardmäßig befindet sich die ausführbare Taxonomie-Manager-Datei hier:

```
C:\ProgramData\Altova\SharedBetweenVersions\TaxonomyManager.exe
```

**Anmerkung:** Nachdem Sie auf Linux- und macOS-Systemen das Verzeichnis in dasjenige, das die ausführbare Datei enthält, geändert haben, können Sie die ausführbare Datei mit `sudo ./taxonomymanager` aufrufen. Das Präfix `./` gibt an, dass sich die ausführbare Datei im aktuellen Verzeichnis befindet. Das Präfix `sudo` gibt an, dass der Befehl mit Root-Rechten ausgeführt werden muss.

### Befehlszeilensyntax

Die allgemeine Syntax zur Verwendung der Befehlszeile lautet folgendermaßen:

```
<exec> -h | --help | --version | <command> [options] [arguments]
```

Der senkrechte Balken `|` im Codefragment oben trennt eine Gruppe einander gegenseitig ausschließender Elemente. Optionale Elemente stehen innerhalb von eckigen Klammern `[ ]`. Im Prinzip können Sie den Pfad zur ausführbaren Datei, gefolgt von entweder `--h`, `--help` oder `--version`-Optionen oder gefolgt von einem Befehl eingeben. Jeder Befehl kann Optionen und Argumente haben. Die Liste der Befehle wird in den folgenden Abschnitten beschrieben.

### 3.5.1 help

Mit diesem Befehl erhalten Sie Hilfe zu Befehlen zur ausführbaren Taxonomie-Manager-Datei.

#### Syntax

```
<exec> help [Befehl]
```

[Befehl] ist hierbei ein optionales Argument zur Angabe jedes beliebigen gültigen Befehlsnamens.

Beachten Sie dazu Folgendes:

- Sie können die Hilfe zu einem Befehl auch durch Eingabe des Befehls, gefolgt von `-h` oder `--help` aufrufen, z.B: `<exec> list -h`
- Wenn Sie `-h` oder `--help` direkt nach dem Namen der ausführbaren Datei und vor einem Befehl eingeben, wird die allgemeine Hilfe (und nicht die Hilfe zu einem bestimmten Befehl) angezeigt, z.B: `<exec> -h list`

#### Beispiel

Mit dem folgenden Befehl wird Hilfe zum Befehl `list` angezeigt:

```
taxonomymanager help list
```

## 3.5.2 info

Mit diesem Befehl werden ausführliche Informationen über die einzelnen als `Taxonomy`-Argument angegebenen Taxonomien angezeigt. Darin enthalten sind Titel, Version, Beschreibung, Herausgeber der jeweils angegebenen Taxonomie und davon abhängige Taxonomien sowie die Information, ob die Taxonomie installiert ist oder nicht.

### Syntax

```
<exec> info [options] Taxonomy+
```

- Das Argument `Taxonomy` ist der Name einer Taxonomie oder Teil eines Taxonomienamens. (Die Paket-ID einer Taxonomie und detaillierte Informationen über ihren Installationsstatus erhalten Sie mit dem Befehl [list](#) <sup>44</sup>.)
- Mit `<exec> info -h` können Sie die Hilfe zum Befehl anzeigen.

### Beispiel

Mit dem folgenden Befehl werden Informationen über die Taxonomien `eba-2.1.0` und `us-gaap-2020.0` angezeigt:

```
taxonomymanager info eba-2.1.0 us-gaap-2020.0
```

## 3.5.3 initialize

Mit diesem Befehl wird die Taxonomie-Manager-Umgebung initialisiert. Sie erstellen damit ein Cache-Verzeichnis, in dem Informationen über alle Taxonomien lokal gespeichert werden. Die Initialisierung erfolgt automatisch bei der ersten Installation einer XBRL-fähigen Altova-Applikation. Normalerweise muss dieser Befehl nicht ausgeführt werden. Nach Ausführung des `reset`-Befehls ist dies allerdings erforderlich.

### Syntax

```
<exec> initialize | init [options]
```

#### Optionen

Für den Befehl `initialize` stehen die folgenden Optionen zur Verfügung:

<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <code>false</code> .
<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

### Beispiel

Mit dem folgenden Befehl wird der Taxonomie-Manager initialisiert:

```
taxonomymanager initialize
```

### 3.5.4 install

Mit diesem Befehl installieren Sie eine oder mehrere Taxonomien.

#### Syntax

```
<exec> install [options] Taxonomy+
```

Um mehrere Taxonomien zu installieren, fügen Sie das Argument **Taxonomy** mehrmals hinzu.

Als **Taxonomy**-Argument kann eines der folgenden verwendet werden:

- Ein Taxonomie-Identifizier (im Format **<name>-<version>**, z.B: **eba-2.10**). Um die Taxonomie-Identifizier der gewünschten Taxonomien zu eruieren, führen Sie den Befehl [list](#)<sup>44</sup> aus. Sie können auch einen abgekürzten Identifizier verwenden, sofern dieser eindeutig ist, z.B. **eba**. Falls Sie einen abgekürzten Identifizier verwenden, wird die neueste Version dieser Taxonomie installiert.
- Der Pfad zu einer von der Altova-Website heruntergeladenen **.altova\_taxonomies**-Datei. Informationen zu diesen Dateien finden Sie in [der Einführung zum Taxonomie-Manager.: Funktionsweise](#)<sup>31</sup>.

#### Optionen

Für den Befehl **install** stehen die folgenden Optionen zur Verfügung:

<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <b>false</b> .
<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <b>false</b> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

#### Beispiel

Mit dem folgenden Befehl werden die neuesten **eba** (European Banking Authority) und **us-gaap** (US Generally Accepted Accounting Principles) Taxonomien installiert:

```
taxonomymanager install eba us-gaap
```

### 3.5.5 list

Mit diesem Befehl werden vom Taxonomie-Manager verwaltete Taxonomien aufgelistet. In der Liste wird eine der folgenden Informationen angezeigt:

- Alle verfügbaren Taxonomien
- Taxonomien, die im Namen den im Argument **Taxonomy** angegebenen String enthalten
- Nur installierte Taxonomien
- Nur Taxonomien, für die ein Upgrade installiert werden kann

## Syntax

```
<exec> list | ls [options] Taxonomy?
```

Wenn kein `taxonomy`-Argument angegeben wird, werden alle verfügbaren Taxonomien aufgelistet. Andernfalls werden die durch die angegebenen Optionen definierten Taxonomien aufgelistet (*siehe Beispiel unten*). Beachten Sie, dass Sie das Argument `taxonomy` mehrfach angeben können.

### Optionen

Für den Befehl `list` stehen die folgenden Optionen zur Verfügung:

<code>--installed, --i</code>	Auflisten nur der installierten Taxonomien. Der Standardwert ist <code>false</code> .
<code>--upgradeable, --u</code>	Auflisten nur derjenigen Taxonomien, für die Upgrades (Patches) zur Verfügung stehen. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

## Beispiele

- Um alle verfügbaren Taxonomien aufzulisten, führen Sie den folgenden Befehl aus: `taxonomymanager list`
- Um installierte Taxonomien aufzulisten, führen Sie `taxonomymanager list -i` aus.
- Um Taxonomien, die in ihrem Namen entweder "eba" oder "us-gaap" enthalten, aufzulisten, führen Sie `taxonomymanager list eba us-gaap` aus.

## 3.5.6 reset

Mit diesem Befehl werden alle installierten Taxonomien und das Cache-Verzeichnis entfernt. Ihre Taxonomieumgebung wird vollständig zurückgesetzt. Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize`<sup>43</sup> ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl `reset` mit der Option `-i` aus. Da mit `reset-i` die Originalinstallation des Produkts wiederhergestellt wird, wird empfohlen, nach dem Zurücksetzen und Initialisieren auch den Befehl `update`<sup>47</sup> auszuführen. Führen Sie alternativ dazu den Befehl `reset` mit den Optionen `-i` und `-u` aus.

## Syntax

```
<exec> reset [Optionen]
```

### Optionen

Für den Befehl `reset` stehen die folgenden Optionen zur Verfügung:

<code>--init, --i</code>	Initialisierung des Taxonomie-Managers nach dem Zurücksetzen. Der Standardwert ist <code>false</code> .
<code>--update, --u</code>	Aktualisiert die Liste der verfügbaren Taxonomien im Cache. Der Standardwert ist <code>false</code> .

<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <code>false</code> .
<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

## Beispiele

- Um den Taxonomie-Manager zurückzusetzen, führen Sie den folgenden Befehl aus: `taxonomymanager reset`
- Um den Taxonomie-Manager zurückzusetzen und ihn zu initialisieren, führen Sie `taxonomymanager reset -i` aus.
- Um den Taxonomie-Manager zurückzusetzen, ihn zu initialisieren und seine Taxonomieliste zu aktualisieren, führen Sie `taxonomymanager reset -i -u` aus.

## 3.5.7 uninstall

Mit diesem Befehl deinstallieren Sie eine oder mehrere Taxonomien. Standardmäßig werden auch alle Taxonomien, die von der aktuellen Taxonomie referenziert werden, deinstalliert. Um nur die aktuelle Taxonomie zu deinstallieren und die referenzierten Taxonomien beizubehalten, setzen Sie die Option `--k`.

### Syntax

```
<exec> uninstall [options] Taxonomy+
```

Um mehrere Taxonomien zu deinstallieren, fügen Sie das Argument `taxonomy` mehrmals hinzu.

Als `taxonomy`-Argument kann eines der folgenden verwendet werden:

- Ein Taxonomie-Identifizier (im Format `<name>-<version>`, z.B. `eba-2.10`). Um die Taxonomie-Identifizier der installierten Taxonomien zu eruieren, führen Sie den Befehl `list -i`<sup>44</sup> aus. Sie können auch einen abgekürzten Taxonomienamen verwenden, sofern dieser eindeutig ist, z.B. `eba`. Falls Sie einen abgekürzten Namen verwenden, werden alle Taxonomien, die die Abkürzung in ihrem Namen enthalten, deinstalliert.
- Der Pfad zu einer von der Altova-Website heruntergeladenen `.altova_taxonomies`-Datei. Informationen zu diesen Dateien finden Sie [der Einführung zum Taxonomie-Manager](#): [Funktionsweise](#)<sup>31</sup>.

### Optionen

Für den Befehl `uninstall` stehen die folgenden Optionen zur Verfügung:

<code>--keep-references, --k</code>	Definieren Sie diese Option, um referenzierte Taxonomien beizubehalten. Der Standardwert ist <code>false</code> .
<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <code>false</code> .

<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

## Beispiel

Mit dem folgenden Befehl werden die Taxonomien `eba-2.10` und `us-gAAP-2020.0` und deren Abhängigkeiten deinstalliert:

```
taxonomymanager uninstall eba-2.10 us-gAAP-2020.0
```

Mit dem folgenden Befehl wird die `eba-2.10`-Taxonomie, nicht aber die davon referenzierten Taxonomien deinstalliert:

```
taxonomymanager uninstall --k eba-2.10
```

## 3.5.8 update

Mit diesem Befehl wird die Liste der über den Online-Speicher verfügbaren Taxonomien abgefragt und das lokale Cache-Verzeichnis wird aktualisiert. Normalerweise muss dieser Befehl nur ausgeführt werden, wenn Sie [reset](#)<sup>45</sup> und [initialize](#)<sup>43</sup> ausgeführt haben.

### Syntax

```
<exec> update [options]
```

#### Optionen

Für den Befehl `update` stehen die folgenden Optionen zur Verfügung:

<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <code>false</code> .
<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

## Beispiel

Mit dem folgenden Befehl wird der lokale Cache mit der Liste der neuesten Taxonomien aktualisiert:

```
taxonomymanager update
```

## 3.5.9 upgrade

Mit diesem Befehl werden alle installierten Taxonomien, für die ein Upgrade installiert werden kann, auf die neueste verfügbare *Patch*-Version aktualisiert. Um herauszufinden, welche Taxonomien aktualisiert werden können, starten Sie den Befehl [list -u](#)<sup>44</sup>.

**Anmerkung:** Der Befehl `upgrade` würde eine veraltete Taxonomie entfernen, falls keine neuere Version zur Verfügung steht.

## Syntax

`<exec> upgrade [Optionen]`

### Optionen

Für den Befehl `upgrade` stehen die folgenden Optionen zur Verfügung:

<code>--silent, --s</code>	Nur Fehlermeldungen anzeigen. Der Standardwert ist <code>false</code> .
<code>--verbose, --v</code>	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist <code>false</code> .
<code>--help, --h</code>	Anzeige der Hilfe zum Befehl.

## 4 Menübefehle

In diesem Abschnitt werden die Befehle aus dem EBA-Menüband beschrieben (siehe Abbildung unten). Diese Befehle sind in vier Gruppen unterteilt: *Bericht*, *Fenster*, *Tabellenoperationen* und *Add-In*. Nähere Informationen dazu finden Sie in den Unterabschnitten weiter unten.



### Bericht

#### Insert New Report

Erstellt einen neuen Bericht. Diese Befehl ist deaktiviert, wenn das Berichtsblatt bereits in die Arbeitsmappe eingefügt wurde. Nähere Informationen dazu finden Sie unter [Erstellen eines neuen Berichts](#) <sup>11</sup>.

#### Import from XBRL

Importiert eine XBRL-Instanzdatei in das aktuelle Excel-Arbeitsblatt. Diese Befehl ist deaktiviert, wenn der Bericht bereits in die Arbeitsmappe eingefügt wurde. Um den Befehl zu aktivieren, speichern und schließen Sie den aktuellen Bericht (Arbeitsmappe) und erstellen Sie eine neue Arbeitsmappe. Nähere Informationen dazu finden Sie unter [Importieren von Daten aus XBRL](#) <sup>28</sup>.

#### Update Entrypoint

Mit Hilfe des Befehls **Update Entrypoint** können Sie zu einer anderen Version derselben Taxonomie wechseln. Bei Aufruf dieses Befehls wird das Dialogfeld **Select Entry Point** geöffnet, in dem Sie eine andere Taxonomieversion auswählen können, um diese herunterzuladen.

#### Export to XBRL

Exportiert Daten aus allen aktuell aktiven Arbeitsblättern in eine XBRL-Instanzdatei. Nähere Informationen dazu finden Sie unter [Exportieren von Daten in XBRL](#) <sup>27</sup>.

#### Validate

Validiert Berichtsdaten anhand der zugrunde liegenden XBRL-Taxonomie und zeigt die Validierungsergebnisse im Dialogfeld **Validation Report** an. Nähere Informationen dazu finden Sie unter [Validieren von Daten](#) <sup>25</sup>.

#### Batch Conversion (Enterprise Edition)

Konvertiert mehrere XBRL-Instanzdateien in Excel. Nähere Informationen dazu finden Sie unter [Batch-Konvertierung von XBRL in Excel](#) <sup>29</sup>.

### Fenster

#### Toggle Filing Pane

Blendet das Fenster "Filing" ein oder aus. Standardmäßig ist dieses Fenster sichtbar.

☒ Toggle Validation Report

Blendet das Fenster "Validation Report" ein oder aus. Siehe [Validieren von Daten](#)<sup>25</sup>.

## Tabellenoperationen

☒ Add Sheet (z-Axis)

Fügt ein neues Arbeitsblatt hinzu, wodurch Sie Daten in einer dritten Dimension eingeben können. Diese Schaltfläche ist nur dann aktiv, wenn für die aktuelle Tabelle laut XBRL-Taxonomie eine dritte Dimension zulässig ist. Nähere Informationen dazu finden Sie unter [Dateneingabe in 3-D-Tabellen](#)<sup>21</sup>.

☒ Delete Sheet (z-Axis)

Entfernt das zuvor hinzugefügte Arbeitsblatt für die Z-Achse.

☒ Add Row

Fügt oberhalb oder unterhalb der aktuell ausgewählten Zeile eine neue Zeile hinzu. Diese Schaltfläche ist nur dann aktiv, wenn eine vertikale Erweiterung der aktuellen Tabelle von der Taxonomie unterstützt wird.

☒ Delete Row

Löscht die aktuell ausgewählte Zeile. Diese Schaltfläche ist nur dann aktiv, wenn eine vertikale Erweiterung der aktuellen Tabelle von der Taxonomie unterstützt wird.

☒ Add Column

Fügt links oder rechts von der aktuell ausgewählten Spalte eine neue Spalte hinzu. Diese Schaltfläche ist nur dann aktiv, wenn eine horizontale Erweiterung der aktuellen Tabelle von der Taxonomie unterstützt wird.

☒ Delete Column

Entfernt die aktuell ausgewählte Spalte. Diese Schaltfläche ist nur dann aktiv, wenn eine horizontale Erweiterung der aktuellen Tabelle von der Taxonomie unterstützt wird.

## Add-In

☒ Settings

Ruft ein Dialogfeld auf, in dem Sie die Add-in-Einstellungen anzeigen und ändern können.

☒ Manage Taxonomies

Mit diesem Befehl wird der XBRL-Taxonomie-Manager geöffnet, ein Tool, mit dem Sie XBRL-Taxonomien anzeigen, installieren und deinstallieren können. Siehe [XBRL-Taxonomie-Manager](#)<sup>31</sup>.

☒ Check for Updates

Überprüft, ob am Altova Server eine neuere Version Ihres Produkts vorhanden ist und zeigt eine

entsprechende Meldung an.

☒ Add-In Activation

Zeigt den Aktivierungsstatus des Add-in an und enthält Optionen, um einen Lizenz-Keycode eingeben oder erwerben zu können.

☒ About EBA XBRL Add-in

Zeigt Informationen über die Add-in-Version an.

☒ Altova on the Web

Enthält Links zur Altova Website, darunter zum Online Support Center, zum XBRL-Taxonomie-Download-Center, zu Training und Tutorials.

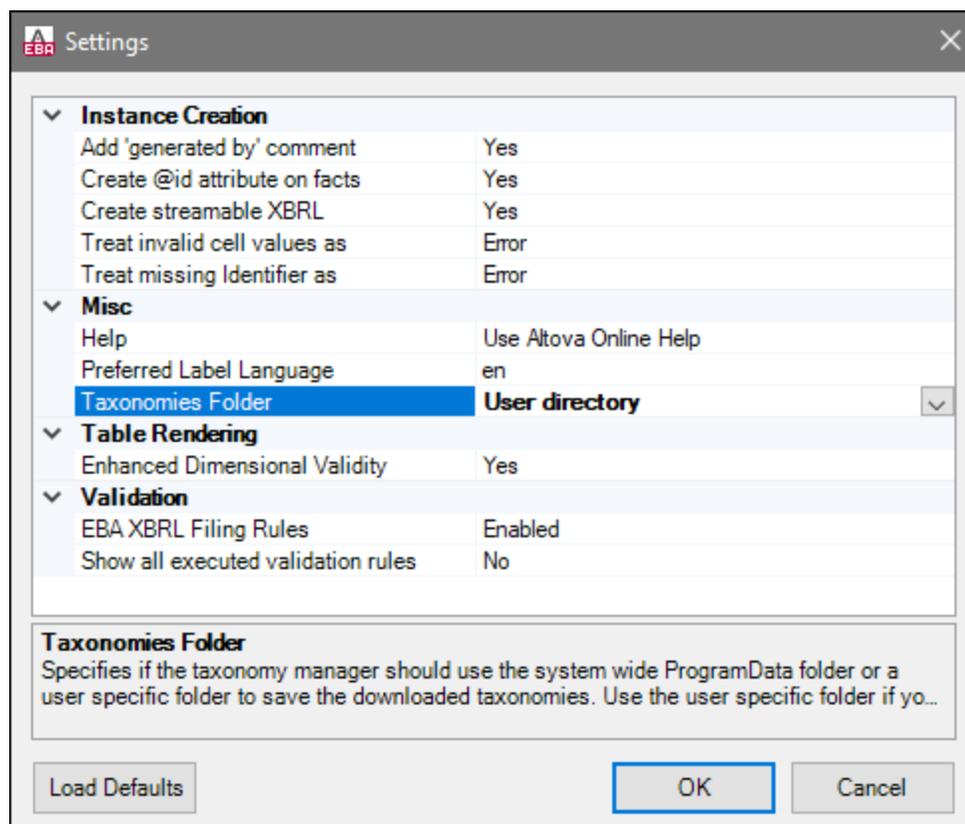
☒ Help

Mit dem Befehl **Help** wird die Hilfe-Dokumentation (das Benutzerhandbuch) des Add-in geöffnet. Standardmäßig wird die Online-Hilfe im HTML-Format auf der Altova Website aufgerufen.

Falls Sie keinen Internet-Zugriff haben oder die Online-Hilfe aus einem anderen Grund nicht aufrufen möchten, können Sie die lokal gespeicherte Version des Benutzerhandbuchs verwenden. Dabei handelt es sich um eine PDF-Datei namens **EBA XBRL Add-in.pdf**, die sich im Applikationsordner (im Ordner "Programme") befindet. Über den Abschnitt *Misc* des Dialogfelds **Settings** können Sie ein anderes Standardformat für das Benutzerhandbuch auswählen.

## 5 Einstellungen

Im Dialogfeld **Settings** (siehe Abbildung unten) können Sie Parameter für die Instanzerstellung, Hilfe und Sprache, Tabellendarstellung konfigurieren und Validierungseinstellungen definieren. Um die Add-in-Einstellungen anzuzeigen oder zu ändern, klicken Sie im Add-in-Menüband auf **Settings**.



Unterhalb sehen Sie eine Liste der konfigurierbaren Einstellungen:

### Instance Creation

Im Abschnitt *Instance Creation* können Sie allgemeine Einstellungen im Zusammenhang mit der Instanzerstellung definieren (z.B. wie ungültige Zellenwerte behandelt werden sollen).

#### Add 'generated by' comment

Gibt an, ob die exportierten XBRL-Instanzen einen *Generated by*-Kommentar enthalten sollen. Setzen Sie diese Option auf **NO**, wenn die zuständige Behörde Kommentare in den erstellten Berichten nicht erlaubt. Der Standardwert ist **YES**.

#### Create @id attribute on facts

Gemäß den [EBA XBRL Filing Rules \(Section 3.7\)](#) sollten nicht verwendete **@id**-Attribute in Facts nicht in den Bericht inkludiert werden, es sei denn, sie werden im Bericht benötigt. Im EBA Add-in werden zu allen Facts **@id**-Attribute hinzugefügt, da Sie dadurch zu Zellen mit Validierungsfehlern oder Warnungen

navigieren können. Standardmäßig ist die Option *Create @id attribute on facts* auf **Yes** gesetzt. Wenn Sie **No** auswählen, werden nicht verwendete **@id**-Attribute in Facts nicht in den XBRL-Bericht inkludiert.

Create streamable XBRL

Gibt an, ob die exportierten XBRL-Instanzen streambar sein und die Processing Instruction `xbrl-streamable-instance` enthalten sollen. Setzen Sie diese Option auf **No**, wenn Ihre Behörde Processing Instructions in Berichten nicht erlaubt. Der Standardwert ist **Yes**. Nähere Informationen zu streambaren XBR-Instanzen finden Sie unter [XBRL Streaming Extensions Module 1.0](#).

Treat invalid cell values as

Gibt an, ob ungültige Zellenwerte als Validierungsfehler oder Warnungen behandelt werden sollen. Der Standardwert ist **Error**.

Treat missing Identifier as

Die Eigenschaft *Identifier* steht im Fenster "Filing" zur Verfügung. Standardmäßig ist dieser Parameter leer. Die Option *Treat missing Identifier as* gibt an, ob eine leere *Identifier*-Eigenschaft einen Validierungsfehler oder eine Warnung auslösen soll. Der Standardwert ist **Error**.

## Misc

Im Abschnitt *Misc* können Sie das Format des Add-in-Handbuchs angeben, die in Überschriften verwendete Sprache und den Pfad von heruntergeladenen Taxonomien.

Help

Gibt an, ob die Altova Online-Hilfe im HTML-Format (Standardoption) oder die lokal installierte PDF-Datei verwendet werden soll.

Preferred Label Language

Definiert, welche Sprache in den Überschriften von erstellten Arbeitsblättern vorzugsweise verwendet werden soll. Beachten Sie, dass die entsprechende Label-Ressource in der Taxonomie definiert sein muss, damit diese Einstellung wirksam werden kann. Der Standardwert ist **en**.

Taxonomies Folder

Über diese Einstellung können Sie festlegen, ob heruntergeladene Taxonomien im global-package-Ordner **ProgramData** (`C:\ProgramData\Altova\pkgs\.cache`) oder im Benutzerordner (`<USER-HOME>\Dokumente\Altova\pkgs\.cache`) gespeichert werden sollen. Wählen Sie den Benutzerordner aus, wenn Sie keine Schreibrechte für den Ordner **ProgramData** haben.

Beachten Sie, dass nichts im global-package-Ordner und dem Benutzerordner gemeinsam verwendet wird. Wenn Sie daher beschließen, von einem Ordner zum anderen zu wechseln (z.B. vom global-package-Ordner zum Benutzerordner) und Sie dieselben Taxonomien im zweiten Ordner benötigen, müssen Sie diese Taxonomien erneut installieren.

## Table Rendering

Der Abschnitt *Table Rendering* enthält die Eigenschaft *Dimension Validity*.

### Enhanced Dimensional Validity

Mit dieser Option werden nur diejenigen Zellen aktiviert, die in Ihrer Dimension in mindestens einer Basisgruppe mit einer Linkrole, die den entsprechenden RCCode der Tabelle enthält, gültig sind. Die empfohlene Standardeinstellung für die Option *Enhanced Dimensional Validity* ist `Yes`. Wenn bei der Einreichung Ihrer Daten bei der zuständigen Stelle Probleme auftreten, setzen Sie diese Option auf `No`. Nähere Informationen zur Gültigkeit von Dimensionen finden Sie unter [Technical Considerations for the Use of XBRL Dimensions 1.0](#).

## Validation

Im Abschnitt *Validation* können Sie angeben, ob zusätzliche XBRL Filing Rules verwendet werden sollen und ob alle ausgeführten Validierungsregeln in den Bericht inkludiert werden sollen.

### XBRL Filing Rules

Definiert, ob die zusätzlichen (in den [EBA XBRL Filing Rules](#) definierten) EBA XBRL Filing Rules überprüft werden sollen. Der Standardwert ist `Enabled`.

### Show all executed validation rules

Zeigt alle ausgeführten Validierungsregeln einschließlich der fehlgeschlagenen im Bericht an. Die Standardoption ist `No`.

## 6 COM API

The add-in provides a COM API that can be used from programming languages that support interacting with Excel and accessing COM objects programmatically, such as VBA or .NET languages. Specifically, the API enables you to create, import and export XBRL reports, read and write form data.

Platform requirements:

- .NET Framework 4.5 or later
- Visual Studio Tools for Office runtime, Version 4.0 or later

If you intend to distribute the API to other clients, note the following:

1. Excel and EBA XBRL Add-in must be installed on each client machine.
2. Each API client that uses your custom code or application must hold a valid EBA XBRL Add-in license.

## 6.1 Access API

You can access COM API of the add-in programmatically in one of the following ways:

- From Excel by using [Visual Basic for Applications](#)
- From your custom program by using the Office Interop API from any .NET language

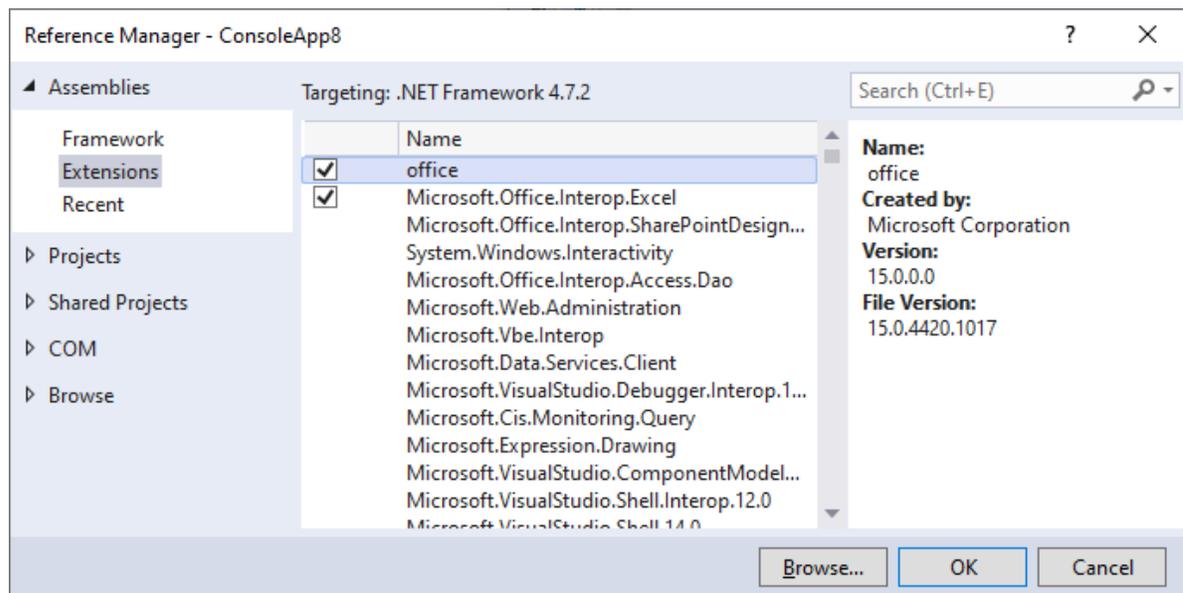
The main interface is the `IAutomationAPI` interface. The code listing below shows how to create a new instance of the automation object in VBA:

```
Dim automationObject As Object
Set automationObject = Application.COMAddIns.Item("Altova.EBAAddIn").Object
```

### Access COM API from a .NET project

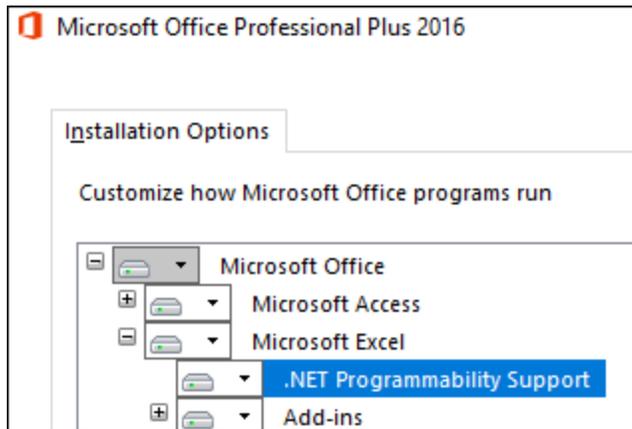
To access the COM API from a Visual Studio .NET project, add a reference to the assemblies **Microsoft Office Object Library (office.dll)** and **Microsoft.Office.Interop.Excel**, as shown below:

1. Right-click your project's name in **Solution Explorer** and then click **Add Reference**. The **Add Reference** dialog box appears.
2. On the **Assemblies** page, select **office** and **Microsoft.Office.Interop.Excel** from the component list and click **OK**.



If you do not see the assemblies above, take the following steps:

1. Make sure that you have installed Microsoft Office and that you have selected the **.NET Programmability Support** feature for Excel (see *screenshot below*).



2. Run the Visual Studio setup and make sure that you choose the **Office/SharePoint development** workload or **Microsoft Office Developer Tools**, if applicable.

For more information about accessing Office interop assemblies from .NET projects, see the [Microsoft documentation](#). After adding the assembly references, you can create a new add-in instance as shown below.

C#

```
// Make sure that your project references the following two assemblies:  
// * Microsoft.Office.Interop.Excel  
// * Microsoft Office Object Library (office.dll)  
var app = new Microsoft.Office.Interop.Excel.Application();  
dynamic automationObject = app.COMAddIns.Item("Altova.EBAAddIn").Object;
```

## 6.2 C# Example

Each of the C# code listings illustrated below represents the `Program.cs` file in a standard .NET Framework console application. Before attempting to run the program code, make sure that you have added the required assembly references to the Visual Studio project, as described in [Access API](#)<sup>56</sup>.

### Export XBRL from Excel

The code listing below shows how to create a new Excel workbook using a specific XBRL entry point, populate a few properties and data cells, and save data to an XBRL file on the disk. The code will help you save both the Excel workbook and the XBRL file to the `C:\XBRL_Examples` directory.

```
using System;
using Excel = Microsoft.Office.Interop.Excel;

namespace EbaAddinClient
{
    class Program
    {
        static void Main(string[] args)
        {
            var app = new Excel.Application();

            try
            {
                // Suppress Excel alerts and create a new workbook
                Console.WriteLine("Creating a new workbook...");
                app.DisplayAlerts = false;
                var wb = (Excel._Workbook)(app.Workbooks.Add());

                // Get the Automation API object
                Console.WriteLine("Getting the COM automation object...");
                dynamic addIn = app.COMAddIns.Item("Altova.EBAAddIn");
                dynamic automationObject = addIn.Object;

                // Create a new report using taxonomy entry point
                Console.WriteLine("Creating the new report...");
                automationObject.InsertNewReport("http://www.eba.europa.eu/eu/fr/xbrl/1/crr/fws/fp/gl-2019-05/2020-02-29/mod/fp_con.xsd");

                // Set the report properties
                Console.WriteLine("Setting the report properties...");
                var rp = automationObject.GetReportProperties(wb);
                rp.ReportingEntityScheme = "http://standards.iso.org/iso/17442";
                rp.ReportingEntityIdentifier = "123456";

                // Find table by its code and ensure it is included in this report
                var tab = automationObject.GetTableTree(wb);
                var tableNode = tab.FindTableByRCCode("P 00.01");
                tableNode.IncludeInFiling = true;

                // Populate cells
                Console.WriteLine("Populating cells...");
                tableNode.Forms.Item(0).DataRange.Item(1).Value = "National GAAP";
            }
        }
    }
}
```

```

        tableNode.Forms.Item(0).DataRange.Item(2).Value = "Consolidated";

        // Export data to XBRL
        Console.WriteLine("Exporting the XBRL instance...");
        automationObject.ExportXBRL(wb, @"C:\XBRL_Examples\Example.xbrl");

        // Save and close the .xlsx workbook
        Console.WriteLine("Saving the .xlsx file...");
        wb.SaveAs(@"C:\XBRL_Examples\Example.xlsx");
        wb.Close();

        Console.WriteLine("Task completed.");
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
    finally
    {
        app.DisplayAlerts = true;
        app.Quit();
    }
}
}
}
}
}

```

## Import XBRL to Excel

The code listing below shows how to convert an XBRL file to an Excel file. To run this example successfully, an XBRL instance file must exist at `C:\XBRL_Examples\Example.xbrl`. Otherwise, change the path accordingly. Use the **Export** command to create an XBRL file by running the previous code listing or manually from Excel. For more information, see [Export Data to XBRL](#) <sup>27</sup>.

```

using System;
using Excel = Microsoft.Office.Interop.Excel;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            var app = new Excel.Application();
            try
            {
                // Suppress Excel alerts and create a new workbook
                Console.WriteLine("Creating a new workbook...");
                app.DisplayAlerts = false;
                var wb = (Excel._Workbook)(app.Workbooks.Add());

                // Get the Automation API object
                Console.WriteLine("Getting the COM automation object...");
                dynamic addIn = app.COMAddIns.Item("Altova.EBAAddIn");
                dynamic automationObject = addIn.Object;

                // Import EBA report eba_example.xbrl

```

```

        Console.WriteLine("Importing XBRL...");
        automationObject.ImportXBRL(@"C:\XBRL_Examples\Example.xbrl");

        // Save as xlsx
        Console.WriteLine("Saving the .xlsx file...");
        wb.SaveAs(@"C:\XBRL_Examples\Example.xlsx");
        wb.Close();

        Console.WriteLine("Task complete.");
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
    finally
    {
        app.DisplayAlerts = true;
        app.Quit();
    }
}
}
}

```

## Map XBRL table to Excel report table

The code listing below shows how to map an XBRL table to a report table in Excel.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Excel = Microsoft.Office.Interop.Excel;

namespace EBAAddInAPITest
{
    class Program
    {
        internal struct Data
        {
            internal Data(string rc, string cc, string value)
            {
                RowCode = rc;
                ColumnCode = cc;
                Value = value;
            }

            internal string RowCode;
            internal string ColumnCode;
            internal string Value;
        }

        static Data[] testData = new Data[]
        {
            new Data("0010", "0060", "163.59000" ),
            new Data("0010", "0070", "2032.33000" ),
        }
    }
}

```

```
new Data("0020", "0010", "1016.16000"),
new Data("0022", "0010", "1000.00000"),
new Data("0040", "0010", "16.16000"),
new Data("0050", "0010", "1016.16000"),
new Data("0020", "0030", "16.16000"),
new Data("0020", "0050", "1016.16000"),
new Data("0020", "0060", "81.29000"),
new Data("0040", "0030", "16.16000"),
new Data("0050", "0030", "1016.16000"),
new Data("0050", "0050", "1016.16000"),
new Data("1050", "0060", "81.29000")
};

static void Main(string[] args)
{
    var app = new Excel.Application();

    try
    {
        // Suppress Excel alerts and create a new workbook
        Console.WriteLine("Creating a new workbook...");
        app.DisplayAlerts = false;
        app.Visible = false;
        var wb = (Excel._Workbook)(app.Workbooks.Add());

        // Get the Automation API object
        Console.WriteLine("Getting the COM automation object...");
        // dynamic addIn = app.COMAddIns.Item("Altova.EBAAddIn");
        dynamic addIn = app.COMAddIns.Item("EBAAddInForExcel");
        dynamic automationObject = addIn.Object;

        // Create a new report using taxonomy entry point
        Console.WriteLine("Creating the new report...");
        automationObject.InsertNewReport("http://www.eba.europa.eu/eu/fr/xbml/crr/f
ws/if/its-002-2021/2021-05-08/mod/if_class2_con.xsd");

        // Set the report properties
        Console.WriteLine("Setting the report properties...");
        var rp = automationObject.GetReportProperties(wb);
        rp.ReportingEntityScheme = "http://standards.iso.org/iso/17442";
        rp.ReportingEntityIdentifier = "123456";

        // Find table by its code and ensure it is included in this report
        var tab = automationObject.GetTableTree(wb);
        var tableNode = tab.FindTableByRCCode("C 21.00");
        tableNode.IncludeInFiling = true;

        // Read column/row headers
        var columnIndices = new Dictionary<string, int>();
        var rowIndices = new Dictionary<string, int>();
        var form = tableNode.Forms.Item(0);
        var dataRange = form.DataRange;
        for (int i = 1; i <= dataRange.Columns.Count; ++i)
            columnIndices.Add(dataRange.Item(0, i).Value.ToString(), i);
        for (int i = 1; i <= dataRange.Rows.Count; ++i)
            rowIndices.Add(dataRange.Item(i, 0).Value.ToString(), i);
    }
}
```

```

        // Populate cells
        Console.WriteLine("Populating cells...");
        foreach (var data in testData)
        {
            int row, column;
            if (!rowIndices.TryGetValue(data.RowCode, out row))
            {
                Console.WriteLine(String.Format("Warning: RowCode {0} not found in
form {1}!", data.RowCode, form.Text));
                continue;
            }
            if (!columnIndices.TryGetValue(data.ColumnCode, out column))
            {
                Console.WriteLine(String.Format("Warning: ColumnCode {0} not found
in form {1}!", data.ColumnCode, form.Text));
                continue;
            }

            dataRange.Item(row, column).Value = data.Value;
        }

        // Export data to XBRL
        Console.WriteLine("Exporting the XBRL instance...");
        automationObject.ExportXBRL(wb, @"C:\XBRL_Examples\Example.xbml");

        var valReport = automationObject.GetValidationReport(wb);
        if (valReport != null)
            Console.WriteLine(valReport.CreateTextReport());

        // Save and close the .xlsx workbook
        Console.WriteLine("Saving the .xlsx file...");
        wb.SaveAs(@"C:\XBRL_Examples\Example.xlsx");
        wb.Close();

        Console.WriteLine("Task completed.");
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
    finally
    {
        app.DisplayAlerts = true;
        app.Quit();
    }
}
}
}

```

## 6.3 VBA Example

The following code listing shows how to insert an XBRL report into an Excel file and populate the first cell using VBA:

```
' VBA Example 1:
' Creates a new EBA COREP ALM report with form 'C 68.00.a' and sets the value of the
first cell
Sub Example1()
    Dim addIn As COMAddIn
    Dim automationObject As Object
    Dim Workbook As Object
    Dim tableTree As Object
    Dim tableNode As Object

    ' Get the Automation API object
    Set addIn = Application.COMAddIns.Item("Altova.EBAAddIn")
    Set automationObject = addIn.Object

    ' Insert a new EBA 2.9.1 COREP ALM report
    Set Workbook =
automationObject.InsertNewReport("http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/
cir-680-2014/2019-11-15/mod/corep_alm_con.xsd")
    Set tableTree = automationObject.GetTableTree(Workbook)

    ' Find table tree node for form 'C 68.00.a'
    Set tableNode = tableTree.FindTableByRCCode("C 68.00.a")

    ' Include this table in the filing (this will also create the respective Excel
worksheet)
    tableNode.IncludeInFiling = True

    ' Get the Data range of this form and set the value of the first cell to 42
    tableNode.Forms.Item(0).DataRange.Item(1).Value = "42"
End Sub
```

## 6.4 API Reference

This section provides information about the interfaces of the EBA Add-in COM API.

### 6.4.1 IAutomationAPI

The `IAutomationAPI` interface is the main automation interface of the EBA Add-in. This interface is the starting point for doing any further operations with the add-in. This interface allows you to create, import and export reports, read and write data. For information about creating an instance of this interface, see [Accessing the API](#)<sup>56</sup>.

#### Methods

The `IAutomationAPI` interface allows you to use the following methods:

##### InsertNewReport

Use this method to insert a new report of the respective taxonomy entry point.

##### Signature

```
InsertNewReport(in entryPointUrl:String) -> Microsoft.Office.Interop.Excel.Workbook
```

##### Parameters

Name	Type	Description
<b>entryPointUrl</b>	<code>String</code>	The URI of the taxonomy a report should be created for. Use the <code>GetEntryPointTree</code> method to get the available entry points.

##### ImportXBRL

Imports an XBRL report. Returns the Excel workbook that contains the imported XBRL report.

##### Signature

```
ImportXBRL(in inputFile:String) -> Microsoft.Office.Interop.Excel.Workbook
```

##### Parameters

Name	Type	Description
<b>inputFile</b>	<code>String</code>	The path to the XBRL report which should be imported.

##### ExportXBRL

Exports the report from the respective Excel workbook to XBRL and also validates it. To get the validation

results, call `GetValidationReport` after calling this method. You can get the file name by using the `GetReportBaseName` method.

#### Signature

```
ExportXBRL(workbook as Microsoft.Office.Interop.Excel.Workbook, outputFile as String)
-> Void
```

#### Parameters

Name	Type	Description
workbook	Microsoft.Office.Interop.Excel.Workbook	The Excel workbook to be exported.
outputFile	String	The path to the output XBRL file.

#### ExportXBRL\_CSV

Exports the report from the respective Excel workbook to the XBRL-CSV report package and validates it. To get the validation results, call `GetValidationReport` after calling this method.

#### Signature

```
ExportXBRL_CSV(in workbook:Microsoft.Office.Interop.Excel.Workbook, in
outputFile:String) -> Void
```

#### Parameters

Name	Type	Description
<b>workbook</b>	Microsoft.Office.Interop.Excel.Workbook	The Excel workbook to be exported.
<b>outputFile</b>	String	The path to the output XBRL-CSV report package file.

#### Validate

Validates the current report. To get the validation results, call `GetValidationReport` after calling this method.

#### Signature

```
Validate(in workbook:Microsoft.Office.Interop.Excel.Workbook) -> Void
```

#### Parameters

Name	Type	Description
<b>workbook</b>	Microsoft.Office.Interop.Excel.Workbook	The Excel workbook to be validated.

#### GetValidationReport

Returns an `IVValidationReport` object representing the validation report currently shown in the validation report pane.

#### Signature

```
GetValidationReport(in workbook:Microsoft.Office.Interop.Excel.Workbook) ->
IVValidationReport
```

#### Parameters

Name	Type	Description
<b>workbook</b>	<code>Microsoft.Office.Interop.Excel.Workbook</code>	The Excel workbook from which to get the validation report.

#### [-] GetEntryPointTree

Returns an `IEntryPointTree` object representing a tree of the available taxonomy entry points.

#### Signature

```
GetEntryPointTree() -> IEntryPointTree
```

#### [-] GetTableTree

Returns an `ITableTree` object representing the tree of the available tables in the report opened in the specified Excel workbook.

#### Signature

```
GetTableTree(in workbook:Microsoft.Office.Interop.Excel.Workbook) -> ITableTree
```

#### Parameters

Name	Type	Description
<b>workbook</b>	<code>Microsoft.Office.Interop.Excel.Workbook</code>	The Excel workbook containing the XBRL report.

#### [-] GetReportProperties

Returns an `IReportProperties` object providing the properties of the XBRL report.

#### Signature

```
GetReportProperties(in workbook:Microsoft.Office.Interop.Excel.Workbook) ->
IReportProperties
```

#### Parameters

Name	Type	Description
<b>workbook</b>	<code>Microsoft.Office.Interop.Excel.Workbook</code>	The workbook containing the XBRL report.

### GetFormProperties

Returns an `IFormProperties` object providing the properties of the XBRL form in the specified Excel worksheet.

#### Signature

```
GetFormProperties(in worksheet:Microsoft.Office.Interop.Excel.Worksheet) ->
IFormProperties
```

#### Parameters

Name	Type	Description
<b>worksheet</b>	<code>Microsoft.Office.Interop.Excel.Worksheet</code>	An Excel worksheet containing an XBRL form.

### GetCellProperties

Returns an `ICellProperties` object providing the properties of the fact in the specified Excel range.

#### Signature

```
GetCellProperties(in cell:Microsoft.Office.Interop.Excel.Range) -> ICellProperties
```

#### Parameters

Name	Type	Description
<b>cell</b>	<code>Microsoft.Office.Interop.Excel.Range</code>	An Excel cell containing an XBRL fact (must be within the data range of a form).

### GetReportBaseName

Returns the file name that Excel provides automatically when you export an XBRL file (via the `ExportXBRL` method) from an Excel workbook.

#### Signature

```
GetReportBaseName(workbook as Excel.Workbook) -> String
```

#### Parameters

Name	Type	Description
workbook	<code>Excel.Workbook</code>	The name of the XBRL file exported via the <code>ExportXBRL</code> method.

## 6.4.2 IEntryPointTree

The `IEntryPointTree` interface provides information about the available taxonomy entry points in a structured way.

### Properties

Name	Description
<a href="#">Groups</a> <sup>68</sup>	Read-only. Returns a collection of <code>IEntryPointGroup</code> representing the available taxonomy groups, for example, "2.9" or "Bank of England Banking Taxonomy 3.1.1".

### 6.4.2.1 Properties

#### 6.4.2.1.1 Groups

Returns a collection of `IEntryPointGroup` representing the available taxonomy groups, for example, "2.9" or "Bank of England Banking Taxonomy 3.1.1".

### Signature

Groups : [Collection](#)

## 6.4.3 IEntryPointGroup

The `IEntryPointGroup` interface provides information about a group of taxonomy entry points.

### Properties

Name	Description
<a href="#">Name</a> <sup>69</sup>	Read-only. The Name of the group. For example, "EBA 2.9".
<a href="#">Country</a> <sup>69</sup>	Read-only. The country code for which this entry point group is relevant. This is set only for country-specific taxonomies such as "Bank of England Banking Taxonomy".
<a href="#">Version</a> <sup>69</sup>	Read-only. The version of the taxonomy. May be empty for sub-groups.

Name	Description
<a href="#">IsCurrentVersion</a> <sup>70</sup>	Read-only. <b>True</b> if this group contains the current version of the taxonomy, <b>false</b> for older versions.
<a href="#">Groups</a> <sup>70</sup>	Read-only. A collection of <code>IEntryPointGroup</code> representing the sub-groups of this entry point group.
<a href="#">EntryPoints</a> <sup>70</sup>	Read-only. A collection of <code>IEntryPoint</code> representing the specific taxonomy entry points of this group.

## 6.4.3.1 Properties

### 6.4.3.1.1 Name

The Name of the group. For example, "EBA 2.9".

#### Signature

Name : `String`

### 6.4.3.1.2 Country

The country code for which this entry point group is relevant. This is set only for country-specific taxonomies such as "Bank of England Banking Taxonomy".

#### Signature

Country : `String`

### 6.4.3.1.3 Version

The version of the taxonomy. May be empty for sub-groups.

#### Signature

Version : `String`

#### 6.4.3.1.4 IsCurrentVersion

**True** if this group contains the current version of the taxonomy; **false** for older versions.

#### Signature

```
IsCurrentVersion : Boolean
```

#### 6.4.3.1.5 Groups

A collection of `IEntryPointGroup` representing the sub-groups of this entry point group.

#### Signature

```
Groups : Collection
```

#### 6.4.3.1.6 EntryPoints

A collection of `IEntryPoint` representing the specific taxonomy entry points of this group.

#### Signature

```
EntryPoints : Collection
```

### 6.4.4 IEntryPoint

The `IEntryPoint` interface provides information about a specific taxonomy entry point.

#### Properties

Name	Description
<a href="#">Name</a> <sup>71</sup>	Read-only. The name of the entry point. For example, "Additional Liquidity Monitoring - COREP, Consolidated".
<a href="#">ShortName</a> <sup>71</sup>	Read-only. The abbreviated form of the entry points name. For example, "COREP ALM Con".
<a href="#">Version</a> <sup>71</sup>	Read-only.

Name	Description
	The version of the entry point. This may be empty (use the parents group version in this case).
<a href="#">URI</a> <sup>72</sup>	Read-only. The URI of the taxonomy entry point, for example, "http://www.eba.europa.eu/eu/fr/xbml/crr/fws/corep/cir-680-2014/2019-04-30/mod/corep_alm_con.xsd". Pass this to the <code>IAutomationAPI.InsertNewReport</code> method.
<a href="#">Needs64Bit</a> <sup>72</sup>	Read-only. <b>True</b> if the entry point requires the 64-bit version of Excel; <b>false</b> otherwise.

## 6.4.4.1 Properties

### 6.4.4.1.1 Name

The name of the entry point. For example, "Additional Liquidity Monitoring - COREP, Consolidated".

#### Signature

Name : `String`

### 6.4.4.1.2 ShortName

The abbreviated form of the entry points name. For example, "COREP ALM Con".

#### Signature

ShortName : `String`

### 6.4.4.1.3 Version

The version of the entry point. This may be empty (use the parents group version in this case).

#### Signature

Version : `String`

#### 6.4.4.1.4 URI

The URI of the taxonomy entry point, for example, "http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/cir-680-2014/2019-04-30/mod/corep\_alm\_con.xsd". Pass this to the `IAutomationAPI.InsertNewReport` method.

#### Signature

```
URI : String
```

#### 6.4.4.1.5 Needs64Bit

**True** if the entry point requires the 64-bit version of Excel; **false** otherwise.

#### Signature

```
Needs64Bit : Boolean
```

### 6.4.5 ITableTree

The `ITableTree` interface provides structured information about the available tables and forms in an XBRL report.

#### Properties

Name	Description
<a href="#">Nodes</a> <sup>73</sup>	Read-only. Returns a collection of <code>IGroupNode</code> and <code>ITableNode</code> objects, which represent groups of tables and tables respectively.

#### Methods

Name	Description
<a href="#">FindTableByRCCode</a> <sup>73</sup>	Returns the table node with the specified RC Code.

## 6.4.5.1 Properties

### 6.4.5.1.1 Nodes

Returns a collection of `IGroupNode` and `ITableNode` objects, which represent groups of tables and tables respectively.

#### Signature

```
Nodes : Collection
```

## 6.4.5.2 Methods

### 6.4.5.2.1 FindTableByRCCode

Returns the table node with the specified RC Code.

#### Signature

```
FindTableByRCCode(in rcCode:string) -> ITableNode
```

#### Parameters

Name	Type	Description
<b>rcCode</b>	<code>string</code>	The RC Code of the desired table node.

## 6.4.6 IGroupNode

The `IGroupNode` interface provides information about a group of tables.

#### Properties

Name	Description
<a href="#">Text</a> <sup>74</sup>	Read-only. The name of the group of tables as displayed in the EBA Filing Pane.
<a href="#">IsGroup</a> <sup>74</sup>	Read-only.

Name	Description
	This is always <b>true</b> for group nodes. Use this to distinguish between <code>IGroupNode</code> and <code>ITableNode</code> members of the <code>Nodes</code> collection.
<a href="#">IsTable</a> <sup>74</sup>	Read-only. This is always <b>false</b> for group nodes. Use this to distinguish between <code>IGroupNode</code> and <code>ITableNode</code> members of the <code>Nodes</code> collection.
<a href="#">Nodes</a> <sup>75</sup>	Read-only. Returns a collection of <code>IGroupNode</code> and <code>ITableNode</code> objects, which represent groups of tables and tables respectively.

## 6.4.6.1 Properties

### 6.4.6.1.1 Text

The name of the group of tables as displayed in the EBA Filing Pane.

#### Signature

Text : **String**

### 6.4.6.1.2 IsGroup

This is always **true** for group nodes. Use this to distinguish between `IGroupNode` and `ITableNode` members of the `Nodes` collection.

#### Signature

IsGroup : **Boolean**

### 6.4.6.1.3 IsTable

This is always **false** for group nodes. Use this to distinguish between `IGroupNode` and `ITableNode` members of the `Nodes` collection.

## Signature

IsTable : **Boolean**

### 6.4.6.1.4 Nodes

Returns a collection of `IGroupNode` and `ITableNode` objects, which represent groups of tables and tables respectively.

## Signature

Nodes : **Collection**

## 6.4.7 ITableNode

The `ITableNode` interface provides information about a table.

### Properties

Name	Description
<a href="#">Text</a> <sup>76</sup>	Read-only. The name of the table as displayed in the EBA Filing Pane.
<a href="#">IsGroup</a> <sup>76</sup>	Read-only. This is always <b>false</b> for table nodes. Use this to distinguish between <code>IGroupNode</code> and <code>ITableNode</code> members of the <code>Nodes</code> collection.
<a href="#">IsTable</a> <sup>77</sup>	Read-only. This is always <b>true</b> for table nodes. Use this to distinguish between <code>IGroupNode</code> and <code>ITableNode</code> members of the <code>Nodes</code> collection.
<a href="#">Forms</a> <sup>77</sup>	Read-only. Returns a collection of <code>IForm</code> objects that represent concrete forms that can be displayed as Excel worksheet.
<a href="#">CanAddSubForm</a> <sup>77</sup>	Read-only. This is <b>true</b> if additional sub forms can be added to the table, for example, if the table has open aspects on the z-axis. In most cases, this is a form for each country or currency.
<a href="#">FilingIndicator</a> <sup>77</sup>	Read-only. The filing indicator code of the table.

Name	Description
<a href="#">RCCode</a> <sup>78</sup>	Read-only. The RC code of the table.
<a href="#">IncludeInFiling</a> <sup>78</sup>	<b>True</b> if the table should be part of the report; <b>false</b> otherwise. If you set this property to <b>true</b> for the first time, a new worksheet for this table will be created.
<a href="#">HasOpenXAxis</a> <sup>78</sup>	<b>True</b> for tables with an open X-axis; <b>false</b> otherwise.
<a href="#">HasOpenYAxis</a> <sup>78</sup>	<b>True</b> for tables with an open Y-axis; <b>false</b> otherwise.

## Methods

Name	Description
<a href="#">AddSubForm</a> <sup>79</sup>	Creates a new sub-form of this table and returns the respective <code>IForm</code> object. This method returns null if no sub-form can be added.

## 6.4.7.1 Properties

### 6.4.7.1.1 Text

The name of the table as displayed in the EBA Filing Pane.

#### Signature

```
Text : String
```

### 6.4.7.1.2 IsGroup

This is always **false** for table nodes. Use this to distinguish between `IGroupNode` and `ITableNode` members of the `Nodes` collection.

#### Signature

```
IsGroup : Boolean
```

### 6.4.7.1.3 IsTable

This is always **true** for table nodes. Use this to distinguish between `IGroupNode` and `ITableNode` members of the `Nodes` collection.

#### Signature

```
IsTable : Boolean
```

### 6.4.7.1.4 Forms

Returns a collection of `IForm` objects, which represent concrete forms that can be displayed as an Excel worksheet.

#### Signature

```
Forms : Collection
```

### 6.4.7.1.5 CanAddSubForm

This is **true** if additional sub forms can be added to the table, for example, if the table has open aspects on the z-axis. In most cases, this is a form for each country or currency.

#### Signature

```
CanAddSubForm : Boolean
```

### 6.4.7.1.6 FilingIndicator

The filing indicator code of the table.

#### Signature

```
FilingIndicator : String
```

### 6.4.7.1.7 RCode

The RC code of the table.

#### Signature

```
RCode : String
```

### 6.4.7.1.8 IncludeInFiling

**True** if the table should be part of the report; **false** otherwise. If you set this property to **true** for the first time, a new worksheet for this table will be created.

#### Signature

```
IncludeInFiling : Boolean
```

### 6.4.7.1.9 HasOpenXAxis

**True** for tables with an open X-axis; **false** otherwise.

#### Signature

```
HasOpenXAxis : Boolean
```

### 6.4.7.1.10 HasOpenYAxis

**True** for tables with an open Y-axis; **false** otherwise.

#### Signature

```
HasOpenYAxis : Boolean
```

## 6.4.7.2 Methods

### 6.4.7.2.1 AddSubForm

Creates a new sub-form of this table and returns the respective `IForm` object. This method returns null if no sub-form can be added.

#### Signature

```
AddSubForm() -> IForm
```

## 6.4.8 IForm

The `IForm` interface provides information about a form.

### Properties

Name	Description
<a href="#">Text</a> <sup>80</sup>	Read-only. The name of the form as displayed in the EBA Filing Pane.
<a href="#">DataRange</a> <sup>80</sup>	Read-only. The Excel range containing the data of this form.
<a href="#">FormSelectorRange</a> <sup>80</sup>	Read-only. The Excel range containing the form selector of this form. Namely, the cells that contain the data that distinguishes this form from the other forms of the same table. This returns null if the table does not consist of multiple forms.
<a href="#">Worksheet</a> <sup>81</sup>	Read-only. The Excel worksheet containing this form. This may be null if <code>IncludeInFiling</code> is false.
<a href="#">IncludeInFiling</a> <sup>81</sup>	<b>True</b> if this form should be part of the report; <b>false</b> otherwise. This shows/hides the respective Excel worksheet.

### Methods

Name	Description
<a href="#">Remove</a> <sup>81</sup>	Removes this form and deletes the respective Excel worksheet.
<a href="#">AddRow</a> <sup>81</sup>	Adds a row to the data range of tables with open y-axis. Returns the excel range of the new row.

Name	Description
<a href="#">AddColumn</a> <sup>82</sup>	Adds a column to the data range of tables with open x-axis. Returns the excel range of the new column.

## 6.4.8.1 Properties

### 6.4.8.1.1 Text

The name of the form as displayed in the EBA Filing Pane.

#### Signature

Text : `String`

### 6.4.8.1.2 DataRange

The Excel range containing the data of this form.

#### Signature

DataRange : `Microsoft.Office.Interop.Excel.Range`

### 6.4.8.1.3 FormSelectorRange

The Excel range containing the form selector of this form. Namely, the cells that contain the data that distinguishes this form from the other forms of the same table. This returns null if the table does not consist of multiple forms.

#### Signature

FormSelectorRange : `Microsoft.Office.Interop.Excel.Range`

#### 6.4.8.1.4 Worksheet

The Excel worksheet containing this form. This may be null if `IncludeInFiling` is `false`.

##### Signature

```
Worksheet : Microsoft.Office.Interop.Excel.Worksheet
```

#### 6.4.8.1.5 IncludeInFiling

**True** if this form should be part of the report; **false** otherwise. This shows/hides the respective Excel worksheet.

##### Signature

```
IncludeInFiling : Boolean
```

### 6.4.8.2 Methods

#### 6.4.8.2.1 Remove

Removes this form and deletes the respective Excel worksheet.

##### Signature

```
Remove() -> Void
```

#### 6.4.8.2.2 AddRow

Adds a row to the data range of tables with open y-axis. Returns the excel range of the new row.

##### Signature

```
AddRow() -> Microsoft.Office.Interop.Excel.Range
```

### 6.4.8.2.3 AddColumn

Adds a column to the data range of tables with open x-axis. Returns the excel range of the new column.

#### Signature

```
AddColumn() -> Microsoft.Office.Interop.Excel.Range
```

## 6.4.9 IReportProperties

The `IReportProperties` interface provides properties of the whole XBRL report.

#### Properties

Name	Description
<a href="#">EntryPointURI</a> <sup>83</sup>	Read-only. The URI of the taxonomy entry point. For example, "Additional Liquidity Monitoring - COREP, Consolidated".
<a href="#">EntryPointModuleName</a> <sup>83</sup>	Read-only. The module name of the taxonomy entry point. For example, "http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/cir-680-2014/2019-04-30/mod/corep_alm_con.xsd".
<a href="#">EntryPointModuleDPMID</a> <sup>83</sup>	Read-only. The Data Point Model Database ID of the module. For example, "218".
<a href="#">ReportingEntityScheme</a> <sup>84</sup>	The scheme of the reporting entity. For example, "http://standards.iso.org/iso/17442".
<a href="#">ReportingEntityIdentifier</a> <sup>84</sup>	The identifier of the reporting entity.
<a href="#">ReferenceDate</a> <sup>84</sup>	The reference date of the report.
<a href="#">MonetaryCellsAccuracy</a> <sup>84</sup>	The accuracy of monetary facts in this report. Applies to each monetary fact for which no separate accuracy was specified (at cell or table level).
<a href="#">PercentageCellsAccuracy</a> <sup>84</sup>	The accuracy of percentage facts in this report. Applies to each percentage fact for which no separate accuracy was specified (at cell or table level).
<a href="#">PureCellsAccuracy</a> <sup>85</sup>	The accuracy of pure facts in this report. Applies to each pure fact for which no separate accuracy was specified (at cell or table level).

Name	Description
<a href="#">ReportingCurrency</a> <sup>85</sup>	The reporting currency used in the XBRL report as an ISO 4217 currency code.
<a href="#">ReportingLanguage</a> <sup>85</sup>	The language of footnotes in the XBRL report as BCP-47 language tag. For example, "en-US".

## 6.4.9.1 Properties

### 6.4.9.1.1 EntryPointURI

The URI of the taxonomy entry point. For example, "Additional Liquidity Monitoring - COREP, Consolidated".

#### Signature

```
EntryPointURI : String
```

### 6.4.9.1.2 EntryPointModuleName

The module name of the taxonomy entry point. For example, "http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/cir-680-2014/2019-04-30/mod/corep\_alm\_con.xsd".

#### Signature

```
EntryPointModuleName : String
```

### 6.4.9.1.3 EntryPointModuleDPMID

The Data Point Model Database ID of the module. For example, "218".

#### Signature

```
EntryPointModuleDPMID : String
```

#### 6.4.9.1.4 ReportingEntityScheme

The scheme of the reporting entity. For example, "http://standards.iso.org/iso/17442".

##### Signature

```
ReportingEntityScheme : String
```

#### 6.4.9.1.5 ReportingEntityIdentifier

The identifier of the reporting entity.

##### Signature

```
ReportingEntityIdentifier : String
```

#### 6.4.9.1.6 ReferenceDate

The reference date of the report.

##### Signature

```
ReferenceDate : DateTime
```

#### 6.4.9.1.7 MonetaryCellsAccuracy

The accuracy of monetary facts in this report. Applies to each monetary fact for which no separate accuracy was specified (at cell or table level).

##### Signature

```
MonetaryCellsAccuracy : String
```

#### 6.4.9.1.8 PercentageCellsAccuracy

The accuracy of percentage facts in this report. Applies to each percentage fact for which no separate accuracy was specified (at cell or table level).

## Signature

```
PercentageCellsAccuracy : String
```

### 6.4.9.1.9 PureCellsAccuracy

The accuracy of pure facts in this report. Applies to each pure fact for which no separate accuracy was specified (at cell or table level).

## Signature

```
PureCellsAccuracy : String
```

### 6.4.9.1.10 ReportingCurrency

The reporting currency used in the XBRL report as an ISO 4217 currency code.

## Signature

```
ReportingCurrency : String
```

### 6.4.9.1.11 ReportingLanguage

The language of footnotes in the XBRL report as BCP-47 language tag. For example, "en-US".

## Signature

```
ReportingLanguage : String
```

## 6.4.10 IFormProperties

The `IFormProperties` interface provides properties of one form of the report.

### Properties

Name	Description
<a href="#">MonetaryCellsAccuracy</a> <sup>86</sup>	The accuracy of monetary facts in this form. Applies to each monetary fact for which no separate accuracy was specified.
<a href="#">PercentageCellsAccuracy</a> <sup>87</sup>	The accuracy of percentage facts in this form. Applies to each percentage fact for which no separate accuracy was specified.
<a href="#">PureCellsAccuracy</a> <sup>87</sup>	The accuracy of pure facts in this form. Applies to each pure fact for which no separate accuracy was specified.
<a href="#">TableRCCode</a> <sup>87</sup>	Read-only. The RC Code of the table. For example, "C 66.01.a".
<a href="#">FilingIndicatorCode</a> <sup>87</sup>	Read-only. The filing indicator code of the table. For example, "C_66.01".
<a href="#">Label</a> <sup>88</sup>	Read-only. The label of the table. For example, "C 66.01.a".
<a href="#">VerboseLabel</a> <sup>88</sup>	Read-only. The verbose label of the table. For example, "C 66.01.a Maturity ladder. Total. Overnight and higher maturity".
<a href="#">TableID</a> <sup>88</sup>	Read-only. The id of the table resource. For example, "eba_tC_66.01.a".
<a href="#">TableDPMID</a> <sup>88</sup>	Read-only. The Data Point Model Database ID of the table. For example, "429.1431".
<a href="#">ValidationRules</a> <sup>89</sup>	Read-only. The collection of validation rules that apply to this table.

### 6.4.10.1 Properties

#### 6.4.10.1.1 MonetaryCellsAccuracy

The accuracy of monetary facts in this form. Applies to each monetary fact for which no separate accuracy was specified.

## Signature

```
MonetaryCellsAccuracy : String
```

### 6.4.10.1.2 PercentageCellsAccuracy

The accuracy of percentage facts in this form. Applies to each percentage fact for which no separate accuracy was specified.

## Signature

```
PercentageCellsAccuracy : String
```

### 6.4.10.1.3 PureCellsAccuracy

The accuracy of pure facts in this form. Applies to each pure fact for which no separate accuracy was specified.

## Signature

```
PureCellsAccuracy : String
```

### 6.4.10.1.4 TableRCCode

The RC Code of the table. For example, "C 66.01.a".

## Signature

```
TableRCCode : String
```

### 6.4.10.1.5 FilingIndicatorCode

The filing indicator code of the table. For example, "C\_66.01".

## Signature

```
FilingIndicatorCode : String
```

#### 6.4.10.1.6 Label

The label of the table. For example, "C 66.01.a".

##### Signature

```
Label : String
```

#### 6.4.10.1.7 VerboseLabel

The verbose label of the table. For example, "C 66.01.a Maturity ladder. Total. Overnight and higher maturity".

##### Signature

```
VerboseLabel : String
```

#### 6.4.10.1.8 TableID

The id of the table resource. For example, "eba\_tC\_66.01.a".

##### Signature

```
TableID : String
```

#### 6.4.10.1.9 TableDPMID

The Data Point Model Database ID of the table. For example, "429.1431".

##### Signature

```
TableDPMID : String
```

### 6.4.10.1.10 ValidationRules

The collection of validation rules that apply to this table.

#### Signature

ValidationRules : [Collection](#)

## 6.4.11 ICellProperties

The `ICellProperties` interface shows properties of one fact of the report.

#### Properties

Name	Description
<a href="#">Accuracy</a> <sup>90</sup>	The accuracy of the numeric item as string. This can be "INF" or a number representing the number of decimal places for which this fact is accurate.  <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;">This is null for non-numeric items.</div>
<a href="#">Footnote</a> <sup>90</sup>	The footnote of the fact.
<a href="#">Name</a> <sup>90</sup>	Read-only. The concept name of the fact. For example, "eba_met:mi256".
<a href="#">Type</a> <sup>90</sup>	Read-only. The type of the fact. For example, "xbri:monetaryItemType".
<a href="#">Label</a> <sup>91</sup>	Read-only. The label of the fact. For example, "Cash value".
<a href="#">DPMID</a> <sup>91</sup>	Read-only. The Data Point Model Database ID of this fact. For example, "6062".
<a href="#">Dimensions</a> <sup>91</sup>	Read-only. The collection of <code>IDimension</code> objects representing the dimensions for which this fact is reported.

## 6.4.11.1 Properties

### 6.4.11.1.1 Accuracy

The accuracy of the numeric item as string. This may be "INF" or a number representing the number of decimal places for which this fact is accurate.

This is null for non-numeric items.

#### Signature

Accuracy : **String**

### 6.4.11.1.2 Footnote

The footnote of the fact.

#### Signature

Footnote : **String**

### 6.4.11.1.3 Name

The concept name of the fact. For example, "eba\_met:mi256".

#### Signature

Name : **String**

### 6.4.11.1.4 Type

The type of the fact. For example, "xbrli:monetaryItemType".

#### Signature

Type : **String**

### 6.4.11.1.5 Label

The label of the fact. For example, "Cash value".

#### Signature

```
Label : String
```

### 6.4.11.1.6 DPMID

The Data Point Model Database ID of this fact. For example, "6062".

#### Signature

```
DPMID : String
```

### 6.4.11.1.7 Dimensions

The collection of `IDimension` objects representing the dimensions for which this fact is reported.

#### Signature

```
Dimensions : Collection
```

## 6.4.12 IDimension

The `IDimension` interface provides basic information of a `Dimension` and its value.

#### Properties

Name	Description
<a href="#">Name</a> <sup>92</sup>	Read-only. The name of the dimension.
<a href="#">Value</a> <sup>92</sup>	Read-only. The value of the dimension as <code>String</code> .

## 6.4.12.1 Properties

### 6.4.12.1.1 Name

The name of the dimension.

#### Signature

Name : `String`

### 6.4.12.1.2 Value

The value of the dimension as `String`.

#### Signature

Value : `String`

## 6.4.13 IValidationReport

The `IValidationReport` interface provides access to the validation report.

### Properties

Name	Description
<a href="#">Messages</a> <sup>93</sup>	Read-only. Returns a collection of <code>IValidationReportMessage</code> representing the main lines of the validation report.

### Methods

Name	Description
<a href="#">CreateTextReport</a> <sup>93</sup>	Returns a textual representation of the validation report.
<a href="#">CreateHTMLReport</a> <sup>93</sup>	Returns an HTML representation of the validation report.

## 6.4.13.1 Properties

### 6.4.13.1.1 Messages

Returns a collection of `IVValidationReportMessage` representing the main lines of the validation report.

#### Signature

```
Messages : Collection
```

## 6.4.13.2 Methods

### 6.4.13.2.1 CreateTextReport

Returns a textual representation of the validation report.

#### Signature

```
CreateTextReport() -> String
```

### 6.4.13.2.2 CreateHTMLReport

Returns an HTML representation of the validation report.

#### Signature

```
CreateHTMLReport() -> String
```

## 6.4.14 IValidationReportMessage

The `IVValidationReportMessage` interface represents a main line in the validation report.

### Properties

Name	Description
<a href="#">Severity</a> <sup>94</sup>	Read-only.

Name	Description
	Returns the severity of this report message as <i>String</i> . Possible values are: "success", "info", "warning" and "error".
<a href="#">Text</a> <sup>94</sup>	Read-only. Returns the value of this report message.
<a href="#">Details</a> <sup>94</sup>	Read-only. Returns the details of this report message (if any) as <i>String</i> .

## 6.4.14.1 Properties

### 6.4.14.1.1 Severity

Returns the severity of this report message as *String*. Possible values are: "success", "info", "warning" and "error".

#### Signature

```
Severity : String
```

### 6.4.14.1.2 Text

Returns the value of this report message.

#### Signature

```
Text : String
```

### 6.4.14.1.3 Details

Returns the details of this report message (if any) as *String*.

#### Signature

```
Details : String
```

## 7 Lizenzinformationen

Dieser Anhang enthält die folgenden Informationen:

- Informationen über den Vertrieb dieses Software-Produkts
- Informationen zur Software-Aktivierung und Lizenzüberwachung
- die Lizenzvereinbarung zu diesem Software-Produkt

Lesen Sie die Informationen bitte sorgfältig - sie sind rechtlich bindend, da Sie sich bei der Installation dieses Software-Produkts damit einverstanden erklärt haben.

Den Inhalt aller Altova-Lizenzvereinbarungen finden Sie auf der [Altova Website](#) unter [Rechtliches](#).

## 7.1 Electronic Software Distribution

Dieses Produkt ist über EDS (Electronic Software Distribution), also auf elektronischem Weg erhältlich, eine Methode, die die folgenden einzigartigen Vorteile bietet:

- Sie können die Software kostenlos 30 Tage lang testen, bevor Sie sich zu einem Kauf entscheiden. *(Anmerkung: Die Lizenz für MobileTogether Designer ist kostenlos.)*
  - Wenn Sie sich entschieden haben, die Software zu kaufen, können Sie Ihre Bestellung online auf der [Altova Website](#) tätigen. Sie erhalten dann innerhalb weniger Minuten ein vollständig lizenziertes Produkt.
  - Sie erhalten immer die neueste Version unserer Software
  - Die Software enthält ein umfassendes Hilfesystem, das Sie von der Benutzeroberfläche der Applikation aus aufrufen können. Die neueste Version des Benutzerhandbuchs steht auf unserer Website [www.altova.com](http://www.altova.com) (i) im HTML-Format zum Aufrufen online und (ii) im PDF-Format zum Download und Ausdrucken zur Verfügung.
- 

### 30-Tage-Evaluierungszeitraum

Nachdem Sie dieses Software-Produkt heruntergeladen haben, können Sie es 30 Tage lang kostenlos testen. Während dieses Zeitraums werden Sie nach etwa 20 Tagen in regelmäßigen Abständen daran erinnert, dass die Software noch nicht lizenziert wurde. Diese Erinnerungsmeldung wird allerdings nur einmal, nämlich bei jedem Start des Programms, angezeigt. Wenn Sie das Programm nach Ablauf des 30-tägigen Evaluierungszeitraums weiterhin verwenden möchten, müssen Sie eine Produktlizenz erwerben, die Sie in Form einer Lizenzdatei mit einem Keycode erhalten. Laden Sie die Lizenzdatei über das Dialogfeld "Software-Aktivierung" Ihres Produkts hoch, um das Produkt freizuschalten.

Sie können Ihre Produktlizenz über <https://shop.altova.com/> erwerben.

---

### Weitergabe der Software an andere Mitarbeiter in Ihrem Unternehmen zu Testzwecken

Wenn Sie die Evaluierungsversion der Software auch anderen Personen in Ihrem Unternehmen über das Netzwerk zur Verfügung stellen möchten oder wenn Sie sie auf einem PC installieren möchten, der nicht mit dem Internet verbunden ist, dürfen Sie nur das Installationsprogramm weitergeben, vorausgesetzt es wurde nicht modifiziert. Jeder, der das von Ihnen zur Verfügung gestellte Installationsprogramm aufruft, muss einen eigenen Evaluierungs-Keycode für 30 Tage anfordern. Nach Ablauf des Testzeitraums, muss eine Lizenz erworben werden, damit das Produkt weiter verwendet werden kann.

## 7.2 Software-Aktivierung und Lizenzüberwachung

Im Rahmen der Aktivierung der Software durch Altova, verwendet die Software unter Umständen Ihr internes Netzwerk und Ihre Internetverbindung, um die Lizenzdaten während der Installation, Registrierung, der Verwendung oder der Aktualisierung an einen von Altova betriebenen Lizenzserver zu übertragen und die Authentizität der Lizenzdaten zu überprüfen, damit Altova-Software nicht ohne Lizenz oder auf unzulässige Art und Weise verwendet werden kann und um den Kundenservice gleichzeitig zu verbessern. Bei der Aktivierung werden zwischen Ihrem Computer und dem Altova-Lizenzserver für die Lizenzierung erforderliche Daten wie Informationen über Betriebssystem, IP-Adresse, Datum/Uhrzeit, Software-Version und Computernamen sowie andere Informationen ausgetauscht.

Ihr Altova-Produkt verfügt über ein integriertes Lizenzüberwachungsmodul, das ebenfalls dazu beiträgt, unbeabsichtigte Verletzungen der Lizenzvereinbarung zu vermeiden. Ihr Produkt kann entweder mit einer Einzelplatzlizenz oder einer Mehrfachlizenz erworben werden. Je nach Lizenz stellt das Lizenzüberwachungsmodul sicher, dass nicht mehr als die lizenzierte Anzahl an Benutzern die Applikation gleichzeitig verwendet.

Bei dieser Lizenzüberwachungsmethode wird Ihr LAN-Netzwerk verwendet, um die Kommunikation zwischen Instanzen der Applikation, die auf verschiedenen Computern laufen, zu überwachen.

### Einzelplatzlizenz

Beim Start der Applikation wird im Rahmen der Lizenzüberprüfung ein kurzes Broadcast-Datagramm abgesendet, um andere Instanzen des Produkts, die auf anderen Computern im selben Netzwerk laufen, zu finden. Wenn keine Antwort einlangt, wird ein Port geöffnet, der Informationen von anderen Instanzen der Applikation empfangen kann.

### Mehrplatzlizenz

Wenn Sie im selben LAN mehrere Instanzen der Applikation verwenden, kommunizieren diese beim Start kurz miteinander, um Keycode-Informationen auszutauschen, damit Sie sicher sein können, dass nicht mehr als die lizenzierte Anzahl an Lizenzen gleichzeitig in Verwendung ist. Dieselbe Lizenzüberwachungstechnologie wird auch bei Unix und vielen anderen Datenbankentwicklungstools verwendet. Sie gestattet Benutzern den Erwerb von Parallellizenzen für mehrere Benutzer zu vernünftigen Preisen.

Wir sind außerdem bestrebt, nur wenige, kleine Netzwerkpakete zu versenden, um Ihr Netzwerk nicht zu überlasten. Die von Ihrem Altova Produkt verwendeten TCP/IP Ports (2799) sind offiziell bei IANA registriert, (*nähere Informationen siehe [IANA Service Name Registry](#)*) und unser Lizenzüberwachungsmodul basiert auf einer bewährten und erprobten Technologie.

Wenn Sie eine Firewall verwenden, werden Sie unter Umständen feststellen, dass die Computer, auf denen Altova-Produkte laufen, über Port 2799 miteinander kommunizieren. Sie können diesen Netzwerkverkehr zwischen verschiedenen Gruppen in Ihrem Unternehmen natürlich blockieren, solange Sie mit anderen Mitteln sicherstellen können, dass Ihre Lizenzvereinbarung eingehalten wird.

### Anmerkung zu Zertifikaten

Ihre Altova Applikation kontaktiert den Altova Lizenzierungsserver über HTTPS ([link.altova.com](https://link.altova.com)). Für diese Kommunikation verwendet Altova ein registriertes SSL-Zertifikat. Wenn dieses Zertifikat ersetzt wird (z.B. von Ihrer IT-Abteilung oder einer externen Agentur), werden Sie von Ihrer Altova Applikation gewarnt, dass die Verbindung nicht sicher ist. Sie könnten Ihre Altova Applikation mit dem Ersetzungszertifikat starten. Dies

würde jedoch auf Ihr eigenes Risiko geschehen. Wenn Sie eine Warnung sehen, dass die *Verbindung nicht sicher* ist, überprüfen Sie den Ursprung des Zertifikats und wenden Sie sich an Ihr IT-Team (die in der Lage sein sollten, zu entscheiden, ob das Abfangen und die Ersetzung des Altova-Zertifikats fortgesetzt werden soll).

Wenn Ihr Unternehmen sein eigenes Zertifikat verwenden muss (z.B. um die Kommunikation zu und von Client-Rechnern zu überwachen), empfehlen wir Ihnen, [Altova LicenseServer](#), die kostenlose Lizenzverwaltungssoftware von Altova in Ihrem Netzwerk zu installieren. Client-Rechner verwenden mit dieser Konfiguration weiterhin die Zertifikate Ihres Unternehmens, während der Altova LicenseServer für die Kommunikation mit Altova das Altova-Zertifikat verwenden kann.

## 7.3 Altova Endbenutzer-Lizenzvereinbarung

- Die Altova-Endbenutzer-Lizenzvereinbarung kann unter <https://www.altova.com/de/legal/eula> eingesehen werden.
- Die Altova-Datenschutzbestimmungen finden Sie unter <https://www.altova.com/de/privacy>.

# Index

## 6

- 64-Bit Excel,**  
Add-in verwenden in, 8

## A

- API Reference,**  
IAutomationAPI, 64  
interfaces, 64
- Azure Information Protection,**  
und eingeschränkter Zugriff, 8

## B

- Batch-Konvertierung,**  
ausführen, 29
- Bericht, 10**  
Batch-Konvertierung, 29  
bedingte Zellen, 17  
Daten aus XBRL importieren, 28  
Daten einfügen, 17  
Daten in 3-D-Zellen eingeben, 21  
Daten in XBRL exportieren, 27  
Daten validieren, 25  
Dateneingabe, 17  
EBA Filing-Fenster, 11  
editierbare Zellen, 17  
Genauigkeit von Zellen, 23  
neue Spalten, 17  
neue Zeilen, 17  
neuen Bericht erstellen, 11  
nicht editierbare Zellen, 17  
tatsächliche Zellenwerte, 17  
Validierung, 25  
Z-Achsen-Blätter, 21  
Zellen mit mehreren Werten, 17

## C

- C#,**  
API, 55, 58
- COM API, 55**  
accessing the, 56  
examples, 58, 63
- Copyright-Informationen, 95**

## E

- EBA XBRL Add-in,**  
allgemeine Informationen, 6  
Befehlsreferenz, 49  
Einführung, 6  
installation, 9  
Lizenzierung, 9  
Menübandbefehle, 49  
Menübefehle, 49  
Systemanforderungen, 8  
unterstützte Taxonomien, 6
- Einstellungen,**  
@id-Attribut in Facts erstellen, 52  
alle ausgeführten Validierungsregeln anzeigen, 52  
ändern, 52  
bevorzugte Label-Sprache, 52  
Dimensional Validity, 52  
fehlende Identifier behandeln als, 52  
'generiert von'-Kommentar hinzufügen, 52  
Hilfe, 52  
Instanzerstellung, 52  
Referenz, 52  
streambare XBRL-Daten erstellen, 52  
Tabellendarstellung, 52  
ungültige Zellen behandeln als, 52  
XBRL Filing Rules, 52
- Endbenutzer-Lizenzvereinbarung, 95, 99**
- Evaluierungszeitraum,**  
für Altova-Software-Produkte, 96  
von Altova Software-Produkten, 95

**I****IAutomationAPI,**

methods, 64

**Information Rights Management,**

und eingeschränkter Zugriff, 8

**Installation,**

DLL-Dateien, 9

Probleme, 9

**L****Lizenz, 99**

Informationen, 95

**Lizenzierung, 9****Lizenzüberwachung,**

in Altova-Produkten, 97

**M****Menübefehle,**

Add-in-Aktivierung, 49

Altova im Web, 49

auf Updates überprüfen, 49

aus XBRL importieren, 49

Batch-konvertieren, 49

Batch-Konvertierung, 49

Blatt hinzufügen (Z-Achse), 49

Blatt löschen (Z-Achse), 49

Einstellungen, 49

Filing-Fenster ein/ausschalten, 49

in XBRL exportieren, 49

neuen Bericht einfügen, 49

Spalte hinzufügen, 49

Spalte löschen, 49

Taxonomien verwalten, 49

über EBA XBRL Add-in, 49

validieren, 49

Validierungsbericht ein/aus, 49

Zeile hinzufügen, 49

Zeile löschen, 49

**R****Rechtliches, 95****S****Software-Produktlizenz, 99****T****Taxonomie-Manager,**

CLI-Befehl Help, 42

CLI-Befehl Info, 43

CLI-Befehl Initialize, 43

CLI-Befehl Install, 44

CLI-Befehl List, 44

CLI-Befehl Reset, 45

CLI-Befehl Uninstall, 46

CLI-Befehl Update, 47

CLI-Befehl Upgrade, 47

CLI-Übersicht, 42

Installieren einer Taxonomie, 39

Patch auf Taxonomie anwenden, 39

starten, 34

Status von Taxonomien, 37

Taxonomie deinstallieren, 41

Taxonomien nach Status auflisten, 37

Übersicht, 31

Upgrade für eine Taxonomie installieren, 39

zurücksetzen, 41

**V****VBA,**

API, 55, 63

**Vertrieb,**

von Altova Software-Produkten, 95

von Altova-Software-Produkten, 96