

Dokumentation intranda Regelsatzeditor

Technische Dokumentation

Revision	Datum	Version	Änderungen
1	22.04.2014	1.0	Initialer Entwurf
2	15.05.2014	1.0	Integration aktualisierter Screenshots
3	19.05.2014	1.0	Kleine Korrekturen

Inhaltsverzeichnis

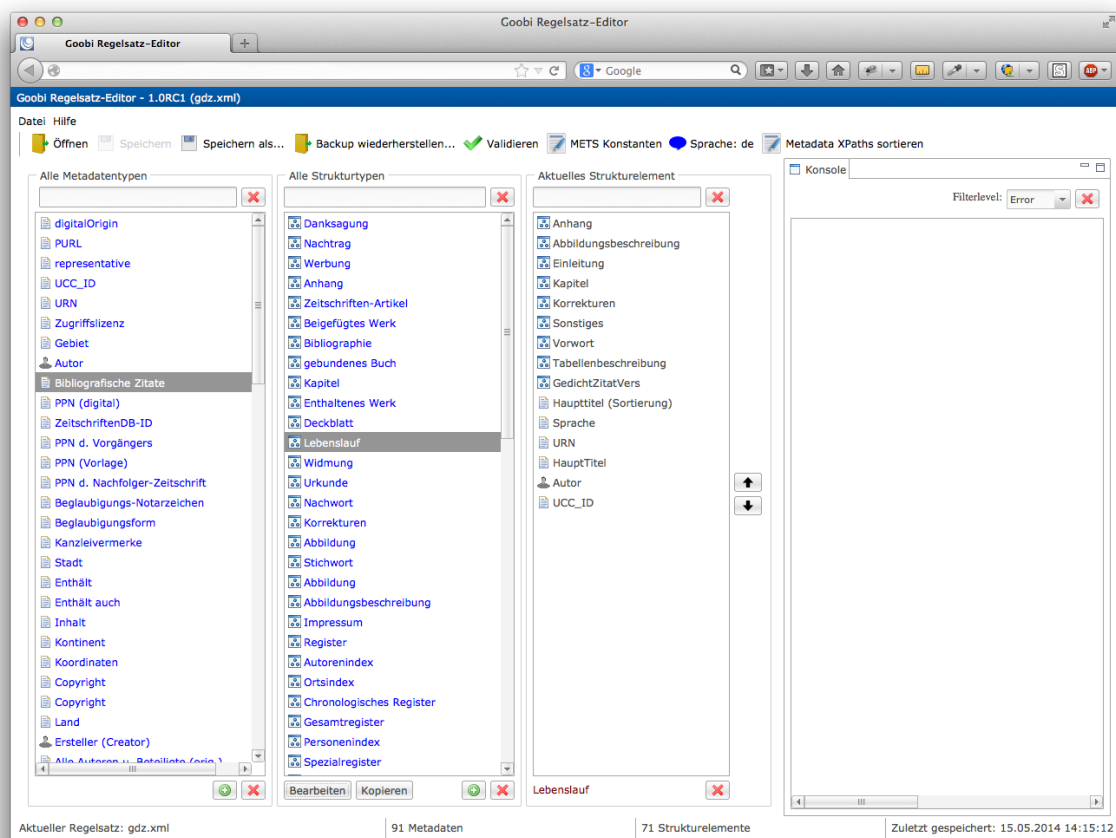
1	Übersicht	3
2	Regelsätze speichern und laden	4
2.1	Kommentarblöcke in Regelsätzen	4
2.2	Synchronisation mit Server	5
3	Validierung.....	5
3.1	Validierung beim Laden	5
3.2	Der Validierungsdialog	6
4	Sprache	8
5	Allgemeine METS/MODS-Konfiguration	10
6	Bearbeiten des Regelsatzes	11
6.1	Bearbeiten der Struktur	11
6.2	Erstellen und Bearbeiten von Strukturelementtypen.....	13
6.3	Erstellen und Bearbeiten von Metadatentypen.....	14
6.4	Austausch von Elementen zwischen verschiedenen Regelsätzen.....	17
7	Konfiguration.....	18
8	Zusätzliche Details	18

1 Übersicht

Der Regelsatzeditor ist eine Web-Applikation zum Erstellen, Bearbeiten und Validieren von Goobi-Regelsätzen. Durch Hoch- und Runterladen von Regelsatzdateien von und zum Goobi-Server erlaubt der Regelsatzeditor außerdem den Austausch von Regelsätzen, und das schnelle Übernehmen von Strukturelementen und Metadaten von einem Regelsatz in einen anderen.

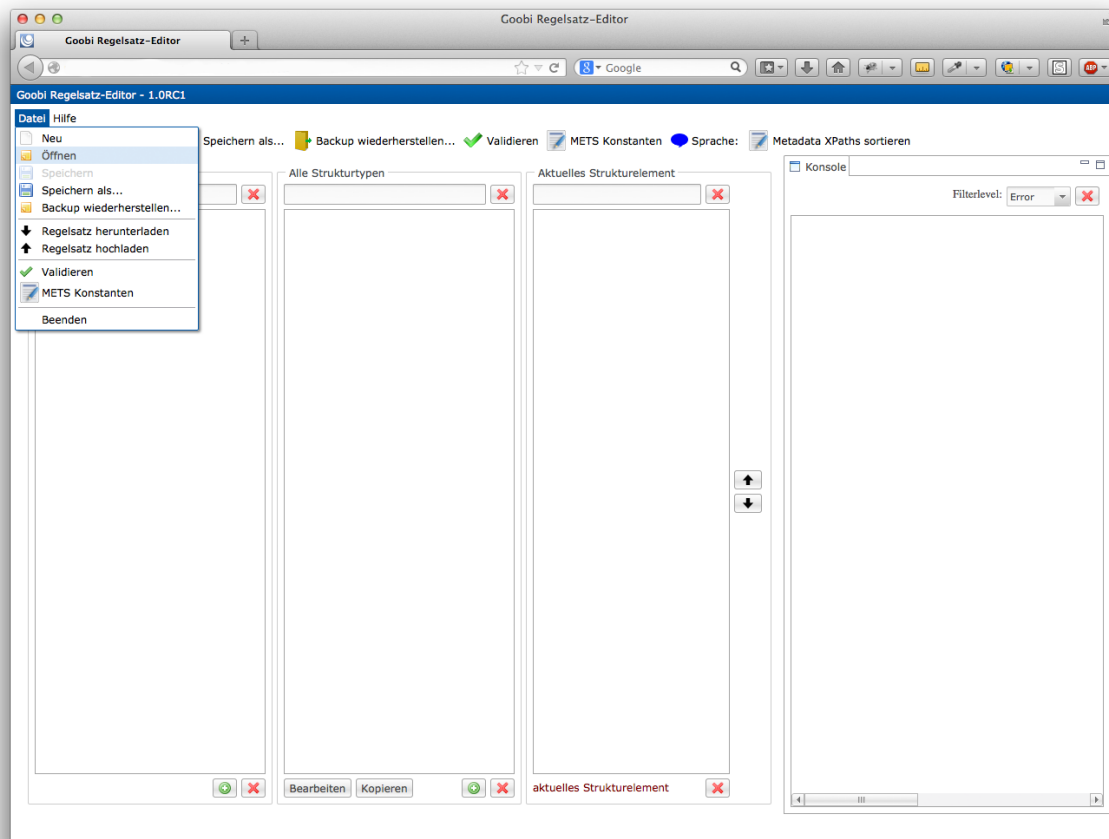
Die Hauptansicht des Editors zeigt eine Liste aller Metadattentypen und eine Liste aller Strukturelementtypen des Regelsatzes. Eine dritte Liste führt alle bestehenden Kindelemente auf, also diejenigen Strukturelementtypen und Metadattentypen, die dem aktuell ausgewählten Strukturelementtyp zugewiesen wurden. Ein Konsolenfenster dient darüber hinaus zur Anzeige von wichtigen Meldungen und Validierungsergebnissen.

Zuordnungen von Datentypen zu externen Formaten – sogenannte Mappings – werden größtenteils in den Metadattentypen und Strukturelementen selbst festgelegt. Globale Einstellungen für METS/MODS sind hingegen über die Werkzeugleiste erreichbar.



2 Regelsätze speichern und laden

Ein Regelsatz kann über die entsprechenden Elemente in der Menü- oder Werkzeugleiste erstellt, gespeichert, geladen und unter anderem Namen - und damit in einer anderen Datei - gespeichert werden.



Existierende Regelsatzdateien werden erst beim Klick auf `Speichern` überschrieben, wobei die bisherige Version immer als Backup behalten wird. Diese Backups können über `Backup wiederherstellen` zu einem späteren Zeitpunkt auch wieder geladen werden. Die Backups eines Regelsatzes sind dazu nach dem Zeitpunkt ihres Erstellens sortiert.

Wurden nach dem letzten Speichern eines Regelsatzes noch Daten verändert, werden diese ebenfalls vorgehalten, und das Programm bietet an, diese Daten wieder herzustellen, wenn der Regelsatz das nächste Mal geladen wird.

2.1 Kommentarblöcke in Regelsätzen

Kommentare, die außerhalb der xml-Struktur oder direkt in den Elementen `Preferences`, `Formats` und den direkten Kindelementen von `Formats` stehen, bleiben beim Laden und Speichern der Regelsätze erhalten. Andere Kommentare - z.B. innerhalb eines Metadatentyps - bleiben hingegen nicht erhalten.

2.2 Synchronisation mit Server

Die im Regelsatzeditor verfügbaren Regelsätze werden serverseitig verwaltet. Dies wird im Abschnitt Konfiguration detailliert erläutert. Während durchaus mehrere Client-Sitzungen gleichzeitig laufen können, kann ein Regelsatz nur jeweils von einem Benutzer gleichzeitig geöffnet sein.

Es ist möglich, den aktuell geöffneten Regelsatz über den Browser lokal herunterzuladen. Das geschieht durch einen einfachen Klick auf `Regelsatz herunterladen` im Reiter `Datei` der Menüleiste. Der Download findet anschließend gemäß den Browsereinstellungen statt. Umgekehrt kann man durch einen Klick auf `Regelsatz hochladen` im selben Menü eine oder mehrere Dateien auf den Server hochladen. Dies öffnet einen Dialog, in den die Pfade zu den hochzuladenden Regelsätzen eingetragen werden können. Mit `Browse` kann ebenso im Dateisystem gesucht werden. Anschließend wird versucht, die zuletzt hochgeladene Datei als Regelsatz geladen.

3 Validierung

Der Regelsatzeditor verfügt über zwei Validierungsmechanismen. Es gibt eine Validierung beim Laden eines Regelsatzes und es gibt einen Validierungsdialog, der durch Klick auf die Schaltfläche `Validieren` in der Menüleiste oder Werkzeugleiste geöffnet werden kann.

3.1 Validierung beim Laden

Die Validierung beim Laden eines Regelsatzes überprüft zunächst die interne Stimmigkeit der Regelsatzstruktur. Regelsätze, deren xml-Struktur fehlerhaft ist, oder denen das `Formats`-Element fehlt, können nicht geladen werden.

Weitere Fehler im Regelsatz, die der inneren Logik des Regelsatzes widersprechen, werden beim Laden gemeldet und es werden dem Benutzer Lösungsvorschläge angeboten, um diese Widersprüche aufzulösen. Dies betrifft vorrangig die folgenden Fehlertypen:

- Strukturelementtypen oder Metadatentypen, die als Kindelemente von Strukturelementtypen auftauchen, aber im Regelsatz nicht deklariert sind.
- Zuordnungen (`mappings`) zu Fremdformaten wie METS/MODS, die keinem deklarierten Metadatentypen oder Strukturelementtypen zugeordnet sind.
- Deklarierte Metadatentypen oder Strukturelementtypen ohne Namen.
- Mehrfach deklarierte Metadatentypen und Strukturelementtypen (mehrere Elemente mit gleichem Namen)
- Fehlende essentielle Metadatentypen und Strukturelementtypen für die physische Struktur.

In den ersten beiden Fällen wird angeboten, ein passendes Element (Metadatentyp oder Strukturelementtyp) aus der Liste der deklarierten Elemente auszuwählen. Wird dies jedoch abgelehnt, kann ein neues passendes Element erstellt, der Eintrag ignoriert, oder der Ladevorgang ganz abgebrochen werden.

Existieren Elemente ohne Namen, wird das gemeldet, und das Element bekommt einen Namen vom Programm zugewiesen. Der Benutzer kann dann das Element später umbenennen oder ganz löschen.

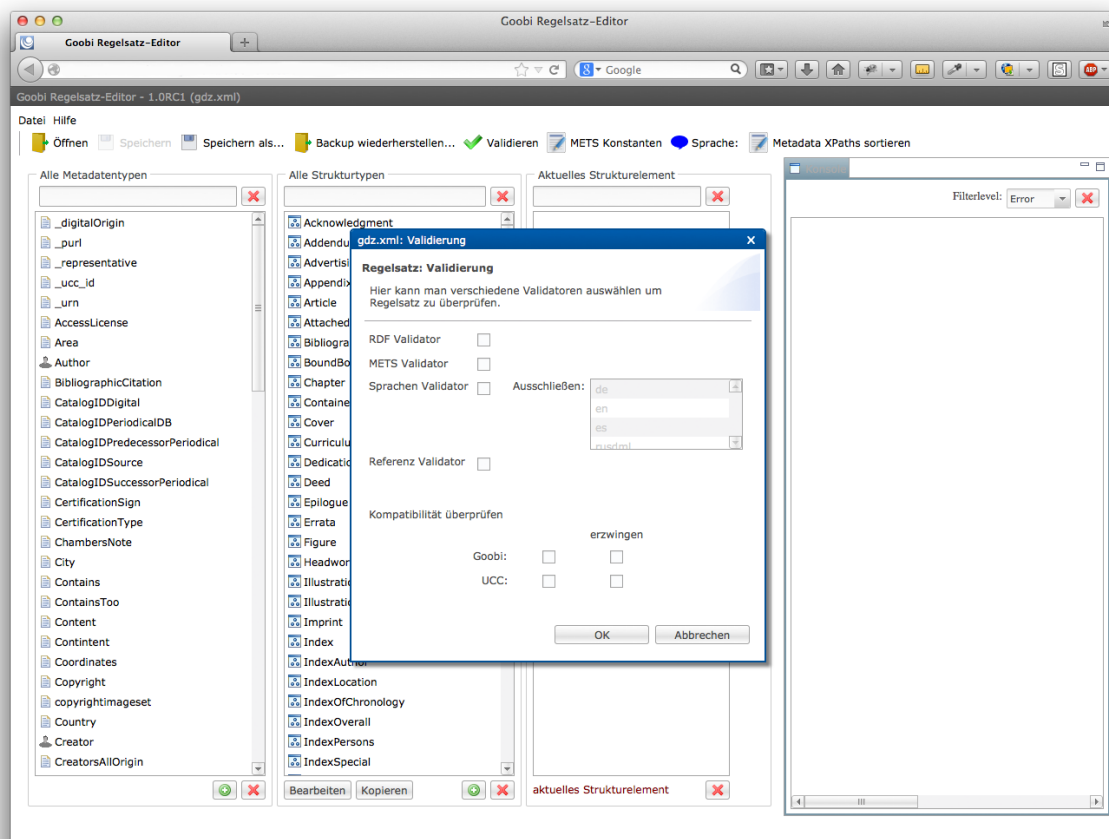
Existieren Metadaten- oder Strukturelementtypen mit gleichem Namen, wird das in der Konsole gemeldet, jedoch zunächst nicht verändert. Der Regelsatz kann aber erst gespeichert werden, wenn der Konflikt behoben wurde, indem die redundanten Elemente umbenannt oder gelöscht werden.

Im Falle von fehlenden benötigten Elementen für die physische Struktur werden diese automatisch eingefügt und dies als `Info` in der Konsole gemeldet. Das heißt, die Meldung ist nur dann für den Benutzer sichtbar, wenn der Filterlevel der Konsole auf `Info` steht.

Beim Laden wird zusätzlich eine METS/MODS-Validierung ausgeführt, deren Ergebnisse als `Warnung` in der Konsole angezeigt werden. Dies bedeutet, dass die Ergebnisse der METS/MODS-Validierung nur dann für den Nutzer sichtbar sind, wenn das Filterlevel der Konsole auf `Info` oder `Warn` steht.

3.2 Der Validierungsdialog

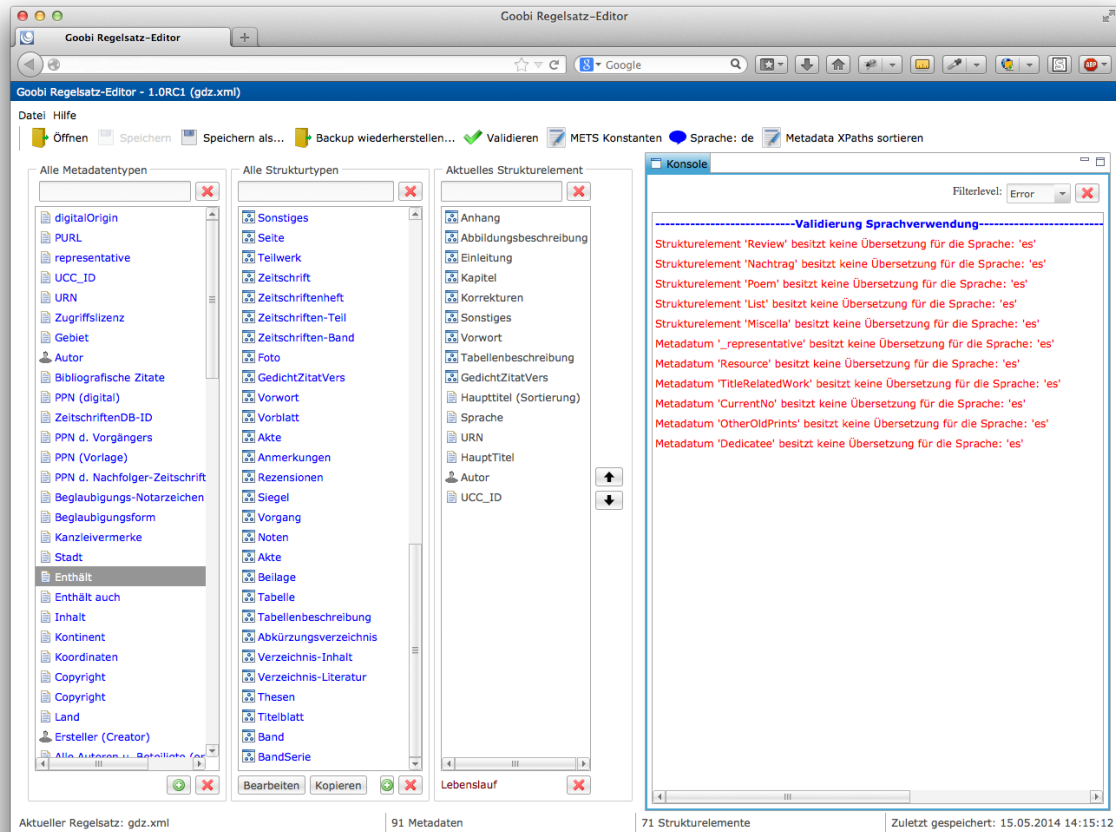
Der Validierungsdialog kann nur dann gestartet werden, wenn keine Metadaten- oder Strukturelementtypen mehrfach vorkommen. Andernfalls meldet das Programm die problematischen Elemente ohne den Validierungsdialog zu öffnen.



Der Dialog selbst besitzt eine Reihe von Optionen, die aktiviert werden können, um sie in die nachfolgende Validierung aufzunehmen. Die einzelnen Optionen sind folgende:

- **RDF-Validator:** Überprüft die Validität der RDF-Mappings, insbesondere ob Metadatentypen oder Strukturelementtypen ohne RDF-Mapping existieren.
- **METS-Validator:** Überprüft die Validität der METS/MODS-Mappings, insbesondere ob Metadatentypen oder Strukturelementtypen ohne METS/MODS-Mapping existieren.
- **Sprachen-Validator:** Überprüft das Vorhandensein von Übersetzungen für alle Metadaten- und Strukturelementtypen im Regelsatz. Im Feld `Ausschließen` können hierbei diejenigen Sprachen markiert werden, die in der Validierung nicht berücksichtigt werden sollen. Die Tasten `Shift` oder `Ctrl` muss hierbei gedrückt gehalten werden, um mehrere Sprachen gleichzeitig zu markieren.
- **Referenz-Validator:** Überprüft, ob Strukturelementtypen oder Metadatentypen existieren, die im Regelsatz nicht verwendet werden, also weder als `Anchor` noch als `TopStruct` deklariert sind, und von keinen anderen Strukturelementtypen verwendet werden.
- **Kompatibilitäts-Validator:** Überprüft die Eignung des Regelsatzes für Goobi oder wahlweise für die Scan-Software UCC. Für Goobi betrifft dies die Existenz und korrekte Verwendungshäufigkeit der Metadatentypen `CatalogIDDigital`, `TitleDocMain` und `singleDigCollection`. Für den UCC betrifft es die Existenz und Verwendung von `_ucc_id`. Die Aktivierung der Option `erzwingen` korrigiert die Regelsätze automatisch so, dass sie zu Goobi bzw. zum UCC kompatibel sind.

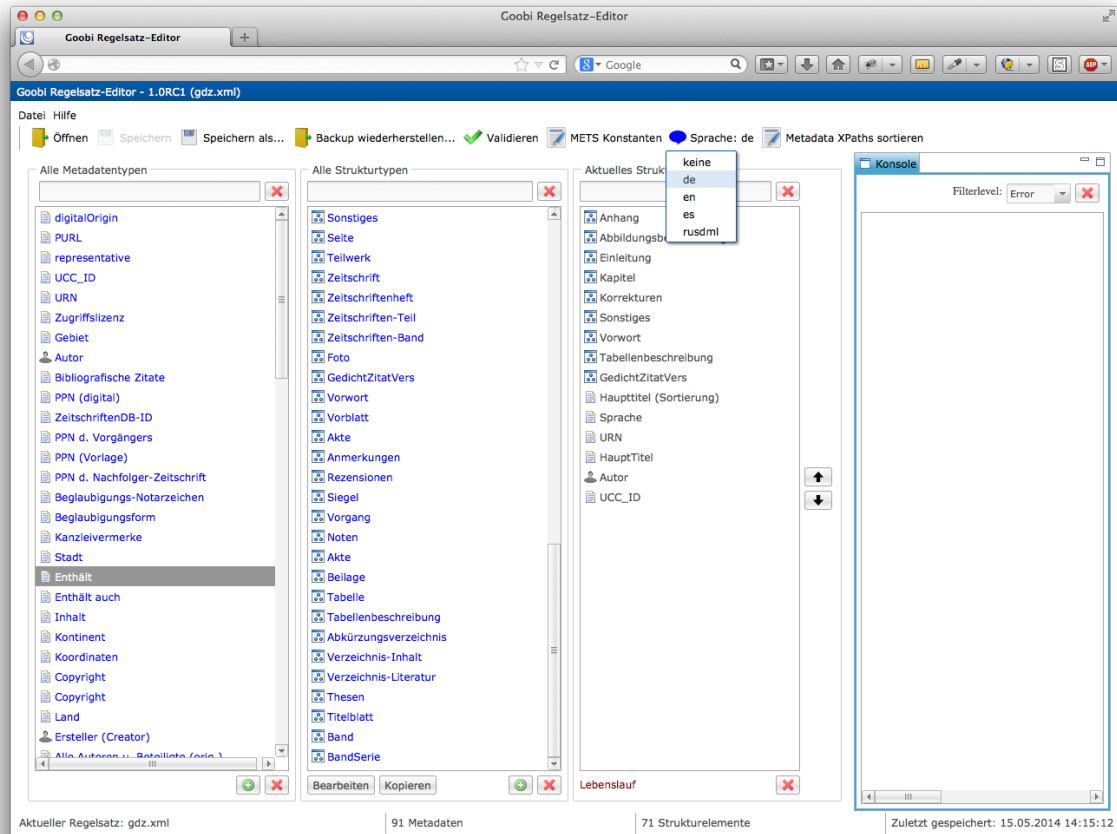
Die Validierung startet mit Klick auf die Schaltfläche `OK` des Dialogs. Anschließend werden die Ergebnisse innerhalb der Konsole angezeigt, je nach Schwere des Problems auf Filterstufe `Info`, `Warn` oder `Error`.



Die angezeigte Filterstufe lässt sich über die Auswahl von `Filterlevel` ändern. Per default werden nur Fehlermeldungen der Stufe `Error` angezeigt.

4 Sprache

In der Werkzeugleiste gibt es eine Schaltfläche `Sprache`, über die aus den im Regelsatz vorhandenen Sprachen die gewünschte Sprache für die Darstellung der Regelsatzdaten ausgewählt werden kann. Beispielsweise führt die Auswahl der Sprache `de` dazu, dass für alle Metadatenarten und Strukturelementtypen die vorhandenen deutschen Namen angezeigt werden, sofern dies für alle bestehenden Elemente möglich ist. Die Auswahl der Option `keine` zeigt den internen Namen des Elements an.

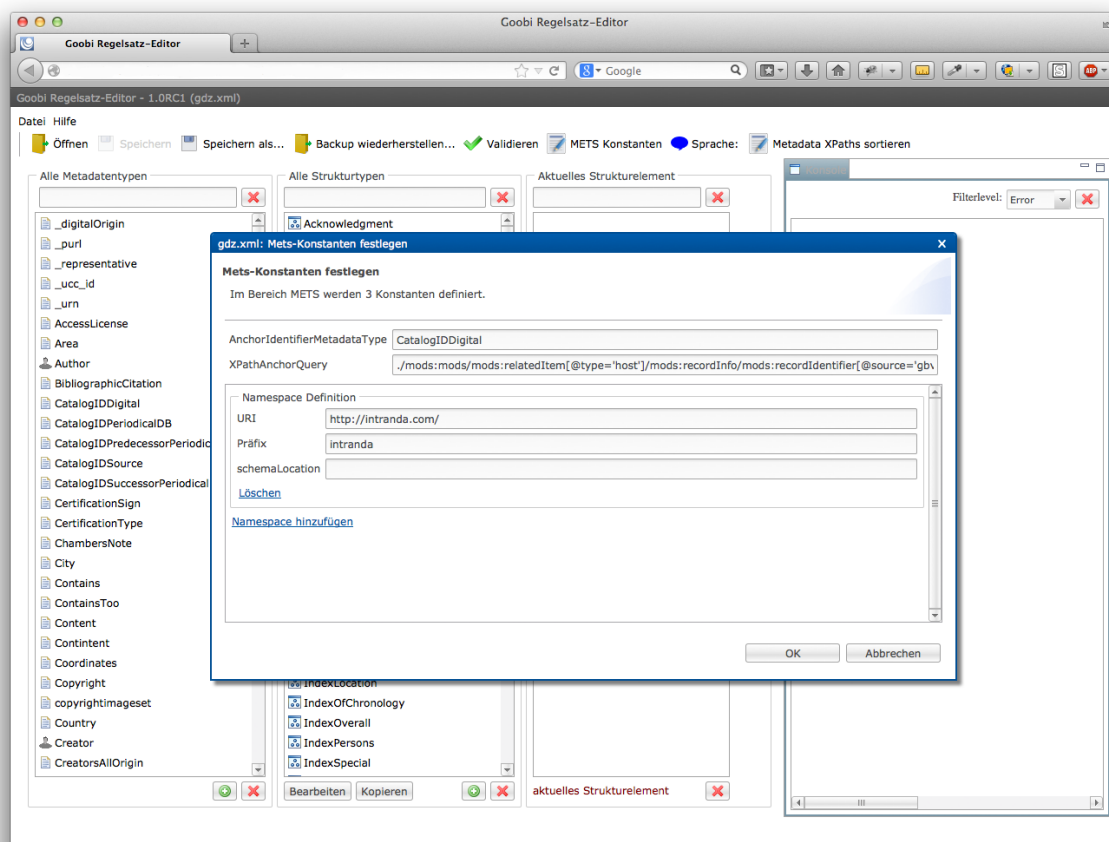


Bei jeder Einstellung außer der Auswahl `keine`, werden alle Metadaten- und Strukturelementtypen blau dargestellt, für die eine passende Übersetzung existiert. Alle anderen Elemente werden hingegen mit ihrem internen Namen in schwarzer Schrift dargestellt. Auf diese Weise lässt sich leicht überprüfen, welche Übersetzungen fehlen, indem einfach nach schwarz verbleibenden Einträgen gesucht wird.

5 Allgemeine METS/MODS-Konfiguration

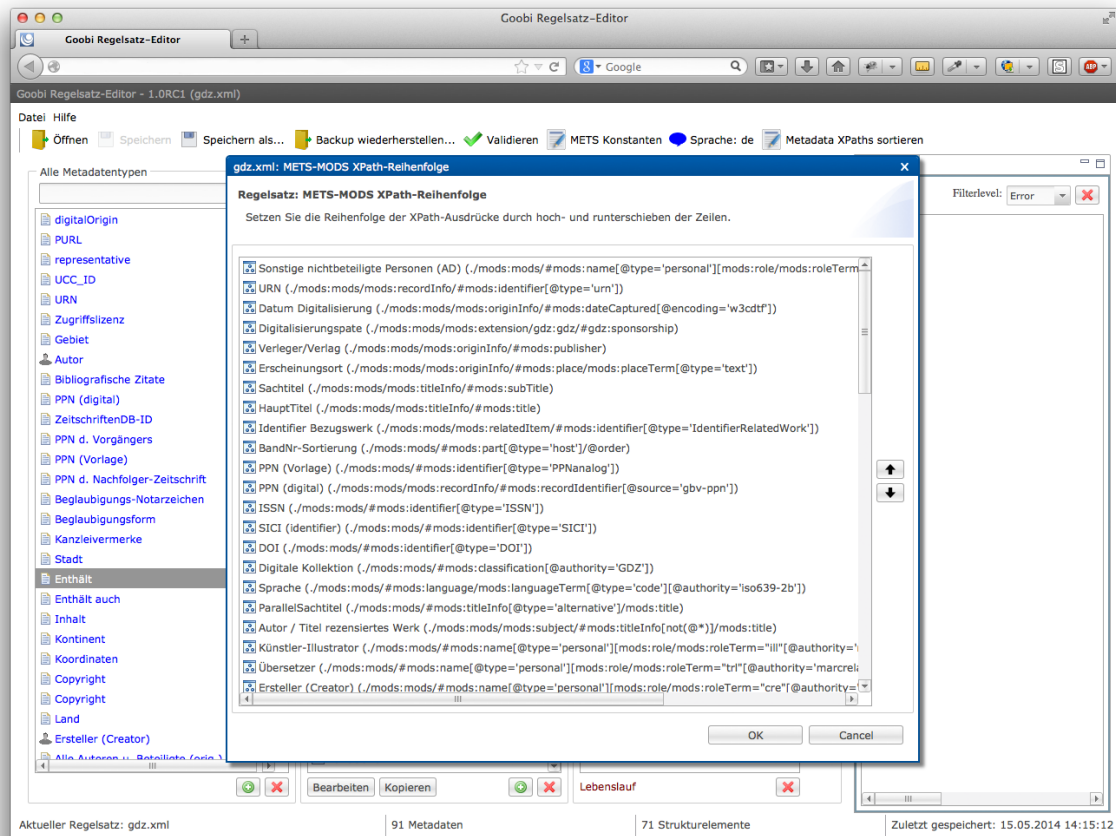
Es gibt zwei allgemeine Konfigurationen für das Mapping nach METS/MODS: Globale Mets-Konstanten und die Sortierung der Mappings.

Die Einstellung der globalen Konstanten ist über die Schaltfläche `METS Konstanten` in der Werkzeugleiste zu erreichen. Hier werden die Einstellungen zur Auflösung des Anchors eines Werkes in METS/MODS vorgenommen. `AnchorIdentifizierMetadataType` bezeichnet dabei das Metadatum aus den Metadaten der `meta_anchor.xml`, das als Identifier des Anchors verwendet wird. `XpathAnchorQuery` bezeichnet den MODS-Pfad als XPath, in den die Daten des Anchors geschrieben werden.



Darüber hinaus können hier zusätzlich im METS/MODS-Dokument zu deklarierende Namespaces definiert werden. Diese werden als scrollbare Liste dargestellt, an deren Ende weitere Namespaces nach Bedarf hinzugefügt werden können.

Die Schaltfläche `Metadata XPaths sortieren` innerhalb der Werkzeugleiste führt zu einer Liste von Metadatatypen mit zugehörigen XPath-Ausdrücken.

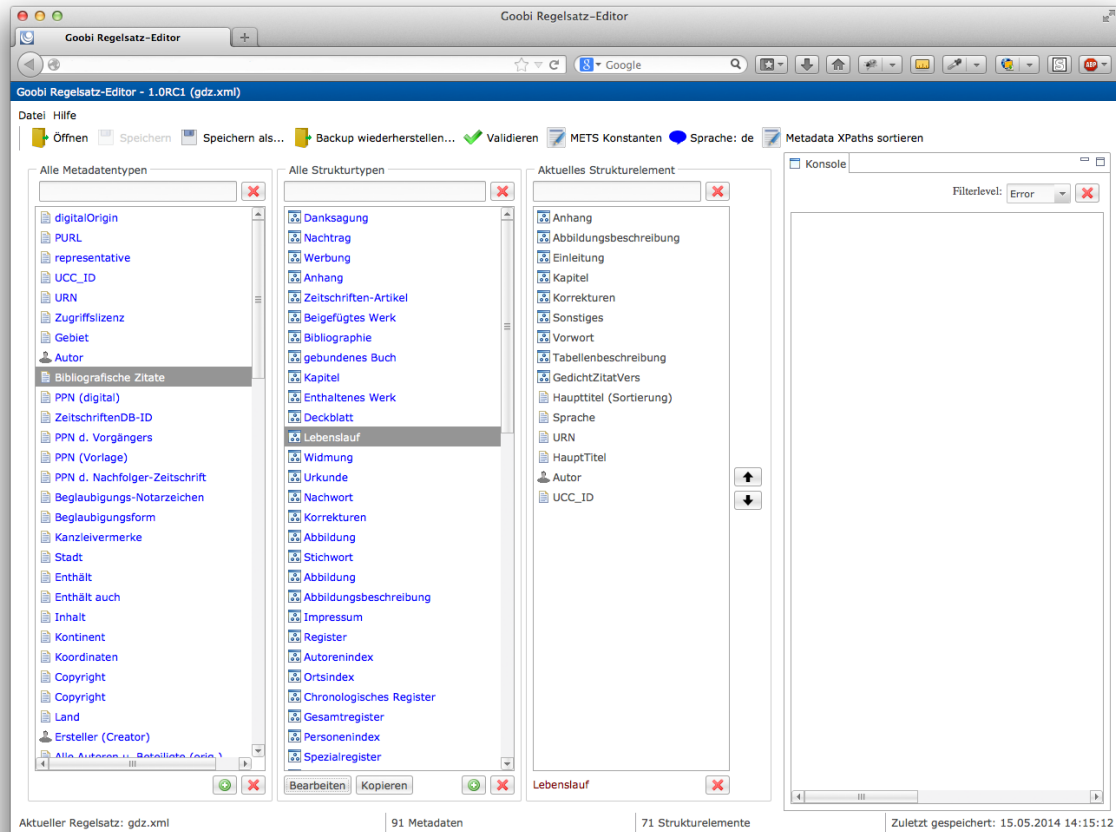


Die Reihenfolge der Listenelemente ist hierbei diejenige Reihenfolge, in der die XPath-Mappings nach METS/MODS in die Regelsatzdatei geschrieben werden sollen. Sie bestimmt damit die Reihenfolge, in der die Metadaten im MODS-Format aufgeführt werden. Die Reihenfolge kann über die Pfeiltasten rechts neben der Liste verändert werden, die ein oder mehrere markierte Listenelemente nach oben oder unten verschieben. Hier können mehrere Elemente mit Hilfe der Tasten **Shift** und **Ctrl** markiert werden.

6 Bearbeiten des Regelsatzes

6.1 Bearbeiten der Struktur

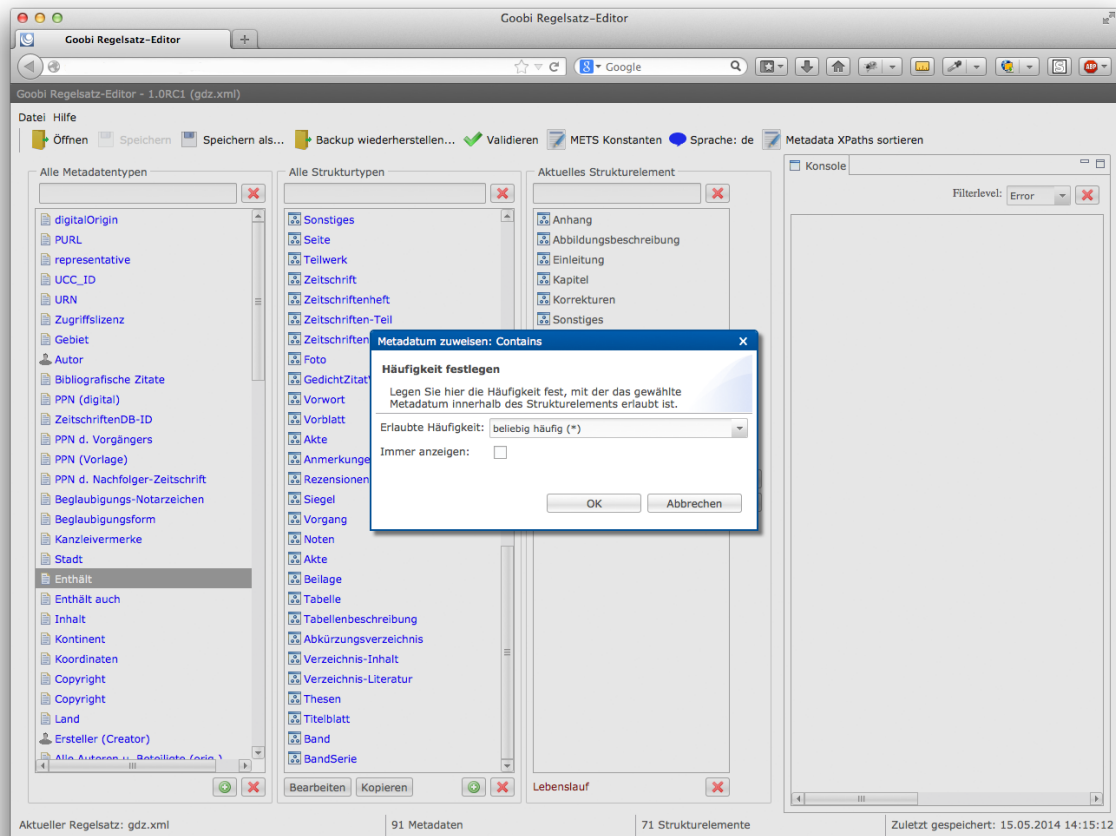
Die Bearbeitung der Struktur eines Regelsatzes bezieht sich auf das Anlegen und Entfernen von Metadatentypen und Strukturelementtypen sowie auf die Zuweisung von Metadaten- und Strukturelementtypen als mögliche Kindelemente von anderen Strukturelementtypen. In der Hauptansicht sind hierfür Listen aller vorhandenen Metdaten- und Strukturelementtypen aufgeführt, zusätzlich zu einer Liste mit den Kindelementen des zuletzt mit Rechtsklick ausgewählten Strukturelementtyps.



Mit den Schaltflächen unterhalb der Listen lassen sich Einträge hinzufügen oder entfernen. Beim Entfernen von Elementen, die von anderen Strukturelementtypen referenziert werden, erscheint zuvor ein Bestätigungsdialog mit einer Liste aller Strukturelementtypen, die dieses Element als Kindelement besitzen. Bei den Strukturelementtypen gibt es zusätzlich einen Knopf **Kopieren**, um eine vollständige Kopie des aktuell markierten Strukturelements zu erstellen.

Metadatentypen und Strukturelementtypen lassen sich per Drag & Drop in die Liste **Aktuelles Strukturelement** schieben, um sie dem aktiven - mit Rechtsklick ausgewählten - Strukturelement als Kindelemente hinzuzufügen. Die Elemente werden an der Stelle der Mausposition eingefügt. Zusätzlich lassen sich die Kindelemente durch die beiden Pfeile rechts neben der Liste nach oben oder unten verschieben, um die Reihenfolge der Kindelemente zu verändern.

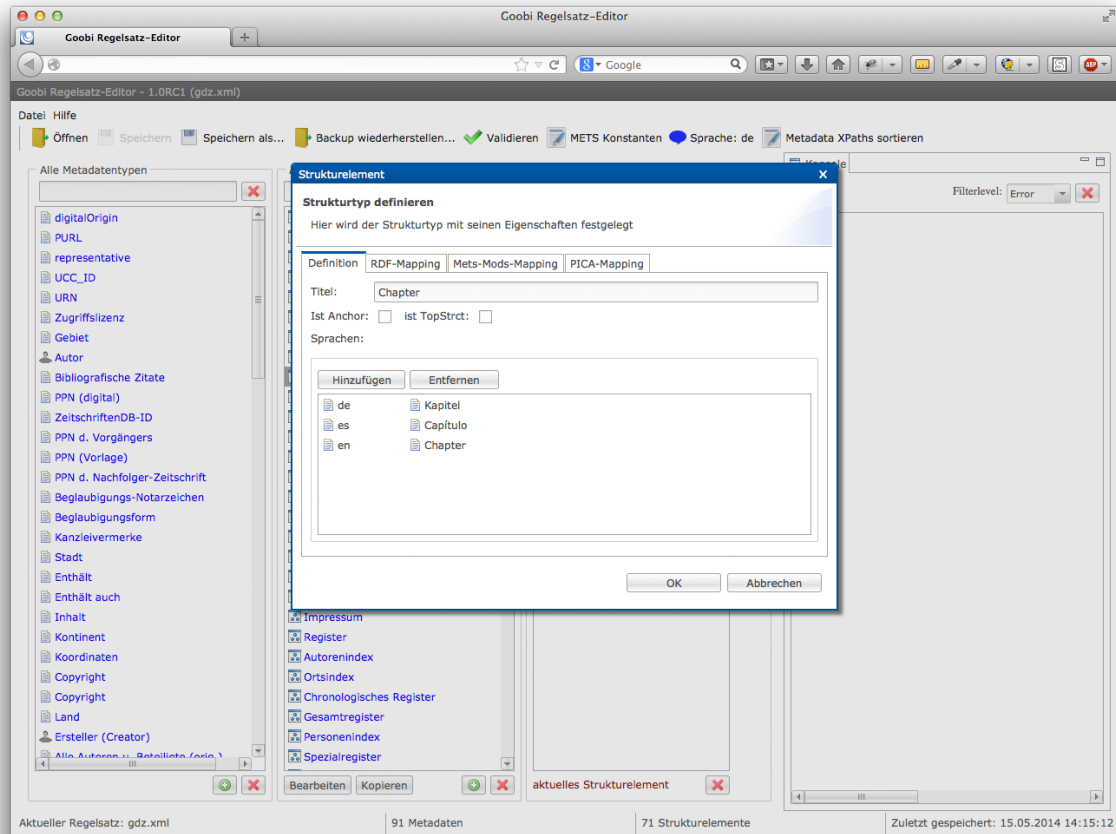
Ein Doppelklick auf ein Element der Listen öffnet einen Bearbeitungsdialog für dieses Element. Für Metadaten- und Strukturelementtypen wird dieser Dialog in den nachfolgenden Abschnitten detailliert beschrieben. Für Metadaten des aktiven Strukturelementtyps lässt sich beim hinzufügen hier die Erlaubte Häufigkeit (*, +, 1o oder 1+) sowie das Attribut **DefaultDisplay** (immer anzeigen) festlegen.



Über jeder Liste existiert jeweils ein Suchschlitz, in dem nach den Namen von Metadaten- und Strukturelementtypen gesucht werden kann. Für die Suche werden dabei immer die internen Namen und gegebenenfalls die Übersetzungen der aktuell ausgewählten Sprache berücksichtigt. Die Schaltfläche neben dem Suchschlitz löscht den aktuellen Inhalt des Suchschlitzes und setzt somit die Suche zurück.

6.2 Erstellen und Bearbeiten von Strukturelementtypen

Beim Doppelklick auf einen Strukturelementtyp, oder dem Anlegen eines neuen Strukturelementtyps - zum Beispiel auch mittels der Schaltfläche **Kopieren** - erscheint ein Dialog, um Details zum Strukturelementtyp festzulegen.



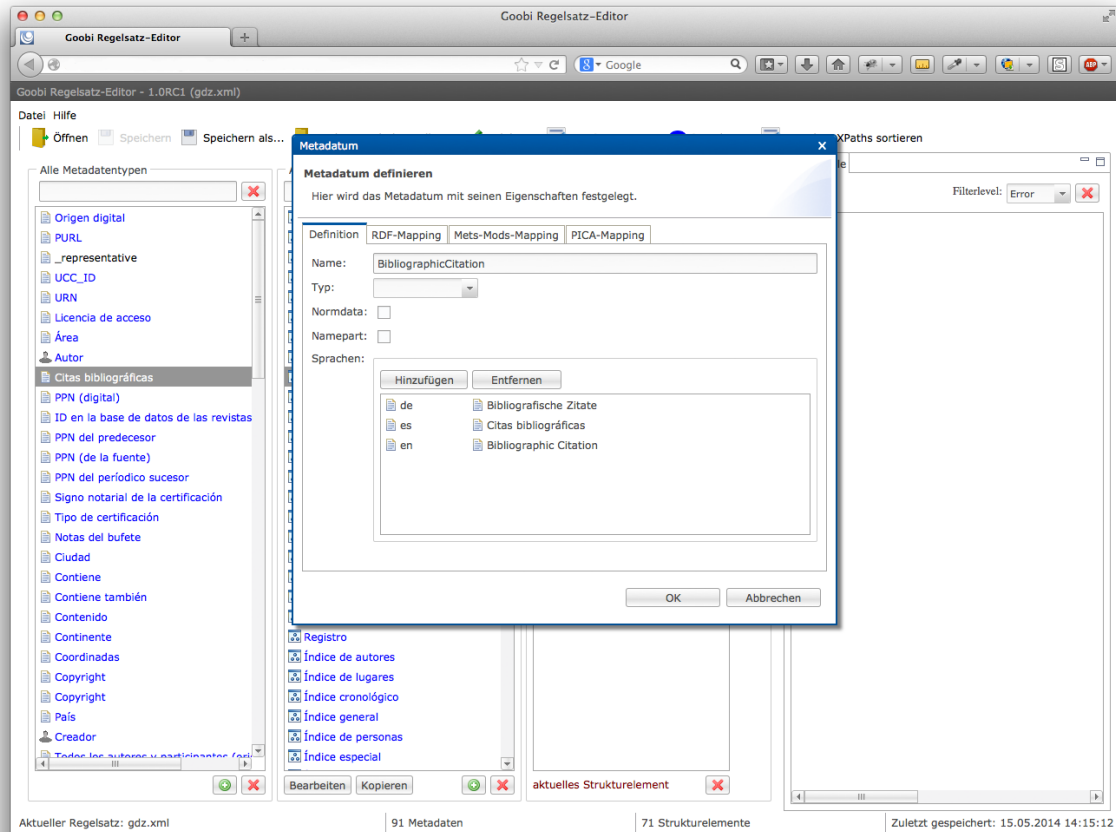
Im Reiter `Definition` werden der (interne) Name des Strukturelementtyps festgelegt, sowie eine Liste von vorhandenen Übersetzungen in verschiedene Sprachen. Übersetzungen können mit Hilfe der entsprechenden Knöpfe hinzugefügt oder entfernt werden. Ein Doppelklick auf eine Übersetzung öffnet sie zur Bearbeitung.

Zusätzlich kann hier der Status des Strukturelementtyps als `Anchor` - also dem physischen Werk übergeordnete logische Einheit - oder als `TopStrct` – d.h. als oberstes logisches Element eines physischen Werkes - festgelegt werden. Die Kennzeichnung als `TopStrct` ist nicht zwingend erforderlich, wird jedoch für die Validierung und im Hinblick auf mögliche zukünftige Entwicklungen empfohlen.

Die anderen Reiter dieses Dialogs beziehen sich auf die Zuordnungen (`Mappings`) des Strukturelementtyps auf andere Metadaten-Formate und sind weitgehend selbsterklärend.

6.3 Erstellen und Bearbeiten von Metadatatypen

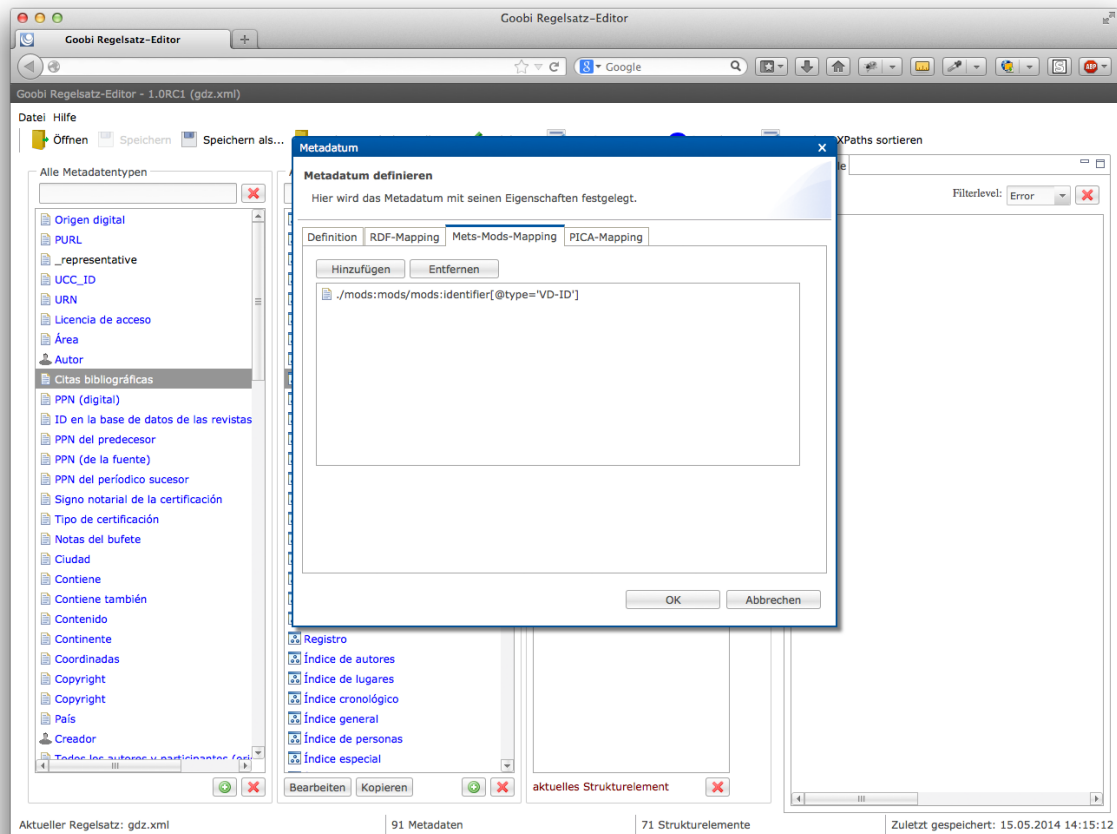
Beim Doppelklick auf einen Metadatatyp oder dem Anlegen eines neuen Metadatatyps erscheint ein Dialog um Details des Metadatatyps festzulegen.



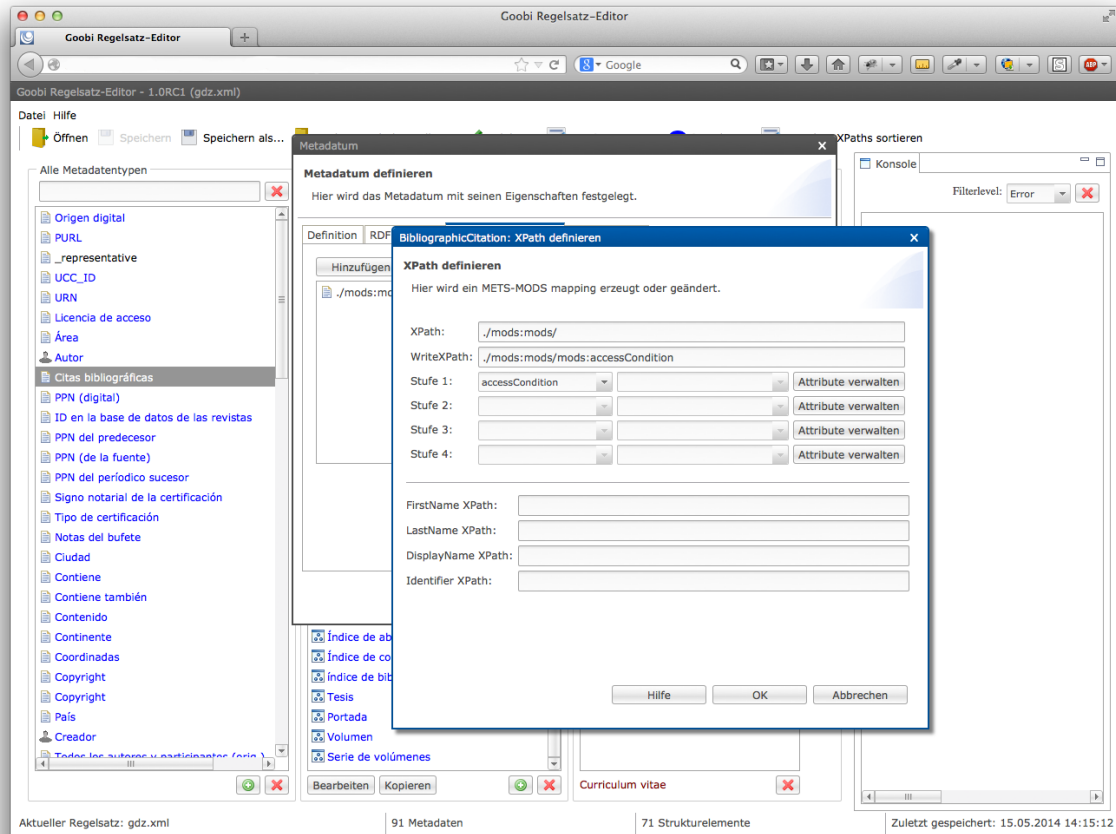
Im Reiter `Definition` werden der interne Name des Metadatentyps, sowie eine Liste von vorhandenen Übersetzungen in verschiedene Sprachen festgelegt. Übersetzungen können mit Hilfe der entsprechenden Knöpfe hinzugefügt oder entfernt werden. Ein Doppelklick auf eine Übersetzung öffnet diese zur Bearbeitung.

Außerdem kann im Auswahlfeld `Typ` der Metadatentyp festgelegt werden: Entweder als `Person` mit Vor- und Zunamen und veränderten MODS-Mappings, oder als `Identifizier` zur Identifikation eines Datensatzes in Goobi. Jedes Strukturelement darf jeweils nur ein Metadatum vom Typ `Identifizier` besitzen, was auch in der Kompatibilitätsvalidierung mit Goobi überprüft wird.

Die anderen Reiter behandeln wie bei den Strukturelementtypen das Mapping in andere Metadatenformate. Dabei bedarf der Reiter `Mets-Mods-Mapping` einiger zusätzlicher Erläuterungen.



Hier können ein oder mehrere Mappings ins METS/MODS-Format definiert werden. Das Hinzufügen eines Mappings oder Editieren mittels Doppelklick, öffnet stets einen weiteren Dialog zur Zusammenstellung der XPath-Ausdrücke.



Jedes Mapping schreibt zwei Elemente in den Regelsatz; XPath zur Überführung aus METS/MODS zu Goobi-Metadatendateien (Goobi-METS), und WriteXPath zum Export nach METS/MODS aus Goobi heraus. Verwendung findet hauptsächlich der WriteXPath, weshalb dieser mit Hilfe von Auswahlfeldern aus erlaubten Ausdrücken zusammengestellt werden kann. Details dazu kann man über die Schaltfläche `Hilfe` abfragen.

Für Personen-Metadaten gibt es hier außerdem die Felder `FirstName`, `LastName`, `DisplayName` und `Identifier`, die mit den relativen Pfaden dieser Daten zum WriteXPath gefüllt werden können. Dies geschieht automatisch für `FirstName` und `LastName`.

6.4 Austausch von Elementen zwischen verschiedenen Regelsätzen

Es ist möglich, Metadatatypen und Strukturelementtypen von einem Regelsatz in einen anderen zu übertragen. Bei Strukturelementtypen ist dabei zu beachten, dass mit dieser Aktion auch alle im Ziel-Regelsatz nicht vorhandenen Kindelemente mit übertragen werden. In diesem Fall wird ein Dialog mit einer Liste aller betroffenen Metadaten- und Strukturelementtypen geöffnet, in dem der Vorgang bestätigt oder abgebrochen werden kann.

Mittels der Tastenkombination `ALT-C` wird ein Metadatatyp oder ein Strukturelementtyp in die Zwischenablage kopiert. Anschließend kann ein anderer Regelsatz geöffnet werden, und dort das Element durch Verwendung der Tastenkombination `ALT-V` eingefügt werden. Alternativ kann das

kopierte Element auch in einem anderen Browserfenster eines parallel geöffneten Regelsatzes eingefügt werden.

Bitte beachten Sie hierbei, dass die Tastenkombinationen `ALT-C` und `ALT-V` absichtlich so gewählt wurden, um Überschneidungen mit der Standardzwischenablage auf dem jeweiligen Betriebssystem zu vermeiden.

7 Konfiguration

Die globale Konfigurationsdatei des Regelsatzeditors liegt direkt im Arbeitsverzeichnis des Applikationsservers. Diese ist bei Apache Tomcat beispielsweise das folgende Verzeichnis.

```
/var/lib/tomcat7/
```

Die Konfigurationsdatei, die in diesem Verzeichnis vorliegt lautet `ruleseteditor_config.properties`. Existiert diese Datei nicht, werden vom Regelsatzeditor automatisch die default-Werte für die Konfigurationen verwendet.

Folgende Werte können konfiguriert werden:

- `localConfig`: Existiert die Konfigurationsdatei mit diesem Pfad, so werden die Konfigurationsparameter aus dieser Datei gelesen.
- `rulesetDirectory`: Der Pfad zum Verzeichnis, in dem die Regelsatz-Dateien liegen. Default ist `/opt/digiverso/goobi/rulesets/`
- `backupDirectory`: Der Pfad zum Verzeichnis, in das Backup-Dateien geschrieben werden. Das Verzeichnis wird vom Programm gegebenenfalls automatisch angelegt. Default ist `/opt/digiverso/goobi/rulesets/backup/`

8 Zusätzliche Details

Für weitere Details stehen die Entwickler des Regelsatzeditors jederzeit zur Verfügung. Bitte wenden Sie sich hierzu an folgende Ansprechpartner:

Ansprechpartner:

Florian Alpers
Steffen Hankiewicz
Robert Sehr

Kontakt:

intranda GmbH
Bertha-von-Suttner Str. 9
D – 37085 Göttingen

<http://www.intranda.com>
goobi@intranda.com