

# Goobi.Production 1.11

## Beauftragte Weiterentwicklungen

ZED-Server Goobi Production 1.11 ZED-Server Zeutschel GmbH Deutsch • English • français

> Startseite  
- Allgemeines -  
Technischer Hintergrund  
Aktive Benutzer

Startseite

**Goobi**  
DIGITALLIBRARYMODULES  
1.10

**Datenbankstatistik**

Benutzer:	325
Benutzergruppen:	15
Vorgänge:	66000
Aufgaben:	600000
Physische Vorlagen:	67500
Werkstücke:	67500

**Startseite**

Die Software Goobi dient zur Organisation von Digitalisierungsworkflows. Das Softwarepaket für diese kooperative und kollaborative Arbeitsumgebung ist der zentrale Baustein und erfüllt folgende Funktionalitäten: Plattform-Unabhängigkeit (Web-Applikation), da Partner weltweit auf das Tool zugreifen müssen; Zentrale Metadaten-Verwaltung, d.h. Katalogisieren und Vervollständigung der Metadaten von verschiedenen Standorten aus (zum Beispiel Erstellung der russischen Metadaten in Moskau, Transliterationen in Hannover); Zentrale Verwaltung der Digitalisate (Images); Import- und Exportschnittstellen für Metadaten und für Fremd-Digitalisate (zum Beispiel aus Russland); Controlling-Mechanismen: welcher Partner ist wie weit mit seinen Arbeiten, welche Zeitschrift ist in welchem Bearbeitungsstadium etc.; Fehlermeldungen; Abschließen eines Arbeitsschrittes und Weiterleiten zur nächsten Arbeitstufe (eventuell hier auch Wechsel/Übergabe von einem Partner zum anderen Partner). Komponenten für das verteilte Workflowmanagement sind integriert worden, um die Verwaltung einer verteilten Produktion und Kommunikation zwischen den unterschiedlichen Partnern zu gewährleisten. Für jeden einzelnen Workflowschritt kann eingestellt werden, ob er parallel zum vorherigen durchgeführt werden kann oder erst nach dessen erfolgreichem Abschluss. Ferner werden Korrekturschleifen abgebildet, um in späteren Schritten festgestellte fehlerhafte Daten früherer Schritte zu vermerken. Die Applikation ist transparent konzipiert, somit ist es für jeden Projektpartner jederzeit möglich, den Status der Teilschritte des Workflows inklusive Personenzuordnung für einzelne Werke detailliert einzusehen.

Login   
Passwort:   
Einloggen

<http://www.goobi.org/> | Goobi Community | Impressum

**ZEUSCHEL**

Vor Beginn aller Arbeiten Anleitung lesen!

Zeuschel GmbH  
Heerweg 2  
72070 Tübingen

Telefon: 07071 / 97060  
Telefax: 07071 / 970644  
E-Mail: [info@zeuschel.de](mailto:info@zeuschel.de)  
Internet: <http://www.zeuschel.de/>

Zeut-16060-DE

© 2014 Zeuschel GmbH

**Informationen zu dieser Bedienungsanleitung**

Diese Anleitung ermöglicht einen sichereren und effizienteren Umgang mit der Software. Die Anleitung ist Bestandteil der Weiterentwicklung und muss in unmittelbarer Nähe der Software für das Personal jederzeit zugänglich aufbewahrt werden.

Das Personal muss diese Anleitung vor Beginn aller Arbeiten sorgfältig durchgelesen und verstanden haben. Grundvoraussetzung für sicheres Arbeiten ist die Einhaltung aller angegebenen Sicherheitshinweise und Handlungsanweisungen in dieser Anleitung.

Abbildungen in dieser Anleitung dienen dem grundsätzlichen Verständnis und können von der tatsächlichen Ausführung der Software abweichen.

Neben dieser Anleitung gelten die benannten technischen Standards.

**Urheberschutz**

Dieses Werk ist lizenziert unter einer Creative Commons Namensnennung – Nicht kommerziell – Keine Bearbeitungen 4.0 International Lizenz.<sup>1</sup>

Zuwendungen verpflichten zu Schadensersatz. Weitere Ansprüche bleiben vorbehalten.

**Garantiebestimmungen**

Die Garantiebestimmungen sind in den Allgemeinen Geschäftsbedingungen des Herstellers enthalten.

**Kundendienst**

Telefonnummer: 07071 / 97060

E-Mail: [info@zeutschel.de](mailto:info@zeutschel.de)

---

<sup>1</sup> <https://creativecommons.org/licenses/by-nc-nd/4.0/>

## Inhaltsverzeichnis

<b>1</b>	<b>Sicherheit .....</b>	<b>5</b>
1.1	Symbole in dieser Anleitung .....	5
1.2	Bestimmungsgemäße Verwendung .....	6
1.3	Besondere Gefahren .....	7
1.4	Hinweise für den Betreiber .....	9
1.5	Personalanforderungen .....	10
<b>2</b>	<b>Änderungen und Erweiterungen .....</b>	<b>11</b>
2.1	DMS-Export als lang laufende Aufgabe .....	11
2.2	Import aller Subfelder beim PICA-Katalogimport .....	12
2.3	Möglichkeit zum Zusammenmappen mehrerer Metadatenfelder .....	13
2.4	Verwaltung gruppierter Metadaten .....	14
2.5	Metadaten beim Import bedingt kopieren .....	17
2.6	Auswahl eines Vorschaubildes im Metadateneditor .....	18
2.7	Prozess-ID exportieren .....	20
2.8	Export fortlaufender Sammelwerke mit Querbezügen ...	21

# 1 Sicherheit

## 1.1 Symbole in dieser Anleitung

### Sicherheitshinweise

Sicherheitshinweise sind in dieser Anleitung durch Symbole gekennzeichnet. Die Sicherheitshinweise werden durch Signalworte eingeleitet, die das Ausmaß der Gefährdung zum Ausdruck bringen.



#### **HÖCHSTE LEBENSGEFAHR!**

Diese Kombination aus Symbol und Signalwort weist auf eine unmittelbar gefährliche Situation hin, die zum Tod oder zu schweren Verletzungen führt, wenn sie nicht gemieden wird.



#### **LEBENSGEFAHR!**

Diese Kombination aus Symbol und Signalwort weist auf eine möglicherweise gefährliche Situation hin, die zum Tod oder zu schweren Verletzungen führen kann, wenn sie nicht gemieden wird.



#### **VERLETZUNGSGEFAHR!**

Diese Kombination aus Symbol und Signalwort weist auf eine möglicherweise gefährliche Situation hin, die zu geringfügigen oder leichten Verletzungen führen kann, wenn sie nicht gemieden wird.



#### **WARNUNG!**

Diese Kombination aus Symbol und Signalwort weist auf eine möglicherweise gefährliche Situation hin, die zu Sach- und Umweltschäden führen kann, wenn sie nicht gemieden wird.

### Sicherheitshinweise in Handlungsanweisungen

Sicherheitshinweise können sich auf bestimmte, einzelne Handlungsanweisungen beziehen. Solche Sicherheitshinweise werden in die Handlungsanweisung eingebettet, damit sie den Lesefluss beim Ausführen der Handlung nicht unterbrechen. Es werden die oben beschriebenen Signalworte verwendet.



*Dieses Symbol hebt nützliche Tipps und Empfehlungen sowie Informationen für einen effizienten und störungsfreien Betrieb hervor.*

## Tipps und Empfehlungen

## Weitere Kennzeichnungen

Zur Hervorhebung von Handlungsanweisungen, Ergebnissen, Auflistungen, Verweisen und anderen Elementen werden in dieser Anleitung folgende Kennzeichnungen verwendet:

Kennzeichnung	Erläuterung
1., 2., 3. ...	Schritt-für-Schritt-Handlungsanweisungen
⇒	Ergebnisse von Handlungsschritten
↪	Verweise auf Abschnitte dieser Anleitung und auf mitgeltende Unterlagen
•	Auflistungen ohne festgelegte Reihenfolge
[Taste]	Bedienelemente (z.B. Tasten, Schalter), Anzeigeelemente (z.B. Signalleuchten)
“Anzeige”	Bildschirmelemente (z.B. Schaltflächen, Belegung von Funktionstasten)

## 1.2 Bestimmungsgemäße Verwendung

Die Software ist ausschließlich für die hier beschriebene bestimmungsgemäße Verwendung konzipiert und konstruiert.

Die Software Goobi.Production dient ausschließlich zur Organisation von Digitalisierungsworkflows und zum Erstellen von Metadatendateien im Format METS/MODS.

Zur bestimmungsgemäßen Verwendung gehört auch die Einhaltung aller Angaben in dieser Anleitung.

Jede über die bestimmungsgemäße Verwendung hinausgehende oder andersartige Benutzung gilt als Fehlgebrauch.

Ansprüche jeglicher Art wegen Schäden aufgrund nicht bestimmungsgemäßer Verwendung sind ausgeschlossen.

### 1.3 Besondere Gefahren

#### Gesundheitsbeschwerden



#### **VERLETZUNGSGEFAHR!**

##### **Gesundheitsbeschwerden durch monotone Belastungen!**

Sich häufig wiederholende, monotone Bewegungen an Computerarbeitsplätzen können zu Beschwerden wie Nacken-, Schulter-, Arm- und/oder Handbeschwerden führen.

Deshalb:

- Nur an ergonomischen Computerarbeitsplätzen arbeiten.
- Auf eine gute Sitzhaltung achten.
- Die Arbeit regelmäßig unterbrechen.
- Auf viel Bewegung Wert legen.

#### Datenverlust



#### **WARNUNG!**

##### **Gefahr von Sachschäden durch Datenverlust!**

Menschliches und technisches Versagen kann zu Datenverlust führen.

Deshalb:

- Täglich eine vollständige Datensicherung durchführen.
- Neue Softwareversionen vor der produktiven Nutzung auf einem Testsystem vollständig testen.
- Vor dem Herunterfahren auch einzelner Komponenten sicherstellen, dass diese gerade nicht von Benutzern verwendet werden.
- Datenbanksicherungen bei heruntergefahrenem Servlet-Container durchführen.
- Für alle Zugriffe Benutzerkonten mit möglichst eingeschränkten Berechtigungen verwenden.

## Urheberrechtsverletzungen



### **WARNUNG!**

#### **Gefahr von Urheberrechtsverletzungen!**

Mangelnde Rechtskenntnis und/oder mangelndes Unrechtsbewusstsein kann zu Urheberrechtsverletzungen führen.

Deshalb:

- Vor der Durchführung von Digitalisierungsprojekten die urheberrechtliche Situation für die zu digitalisierenden Vorlagen klären.
- Im Zweifelsfall Maßnahmen zur Wahrung des Urheberschutzes treffen.
- Alle Mitarbeiter für das Thema Urheberrecht sensibilisieren.

## Ausspähen von Daten



### **WARNUNG!**

#### **Gefahr des Ausspähens von Daten!**

Auf dem Computer, auf dem Goobi.Production ausgeführt wird, befindliche Daten können in falsche Hände gelangen.

Deshalb:

- Soweit vermeidbar auf dem Computer keine schützenswerten Daten speichern.
- Vorkonfigurierte Standardbenutzerkonten löschen oder, wo nicht möglich, sperren (z.B. Betriebssystem, Tomcat, Goobi.Production, MySQL).
- Den Computer gegen unbefugten Zugang sichern (z.B. Rechnerraum abschließen, nicht benötigte Serverdienste abschalten, Passwortübertragungen verschlüsseln, unsichere Passwörter verbieten, zulässige IP-Bereiche für Anmeldevorgänge beschränken).
- Sicherheitsupdates für alle verwendeten Softwarekomponenten immer so zeitnah wie möglich einspielen.
- Für die Benutzerkonten in Goobi.Production eigene Passwörter verwenden.
- Für den Zugriff auf zentrale Server auf diesen Benutzerkonten mit möglichst reduzierten Berechtigungen.



**WARNUNG!****Gefahr der illegitimen Ressourcennutzung!**

Der Computer, auf dem Goobi.Production ausgeführt wird, kann zu illegitimen Zwecken missbraucht werden.

Deshalb:

- Für notwendige Serverdienste Benutzerkonten mit möglichst eingeschränkten Berechtigungen verwenden.
- Vorkonfigurierte Standardbenutzerkonten löschen oder, wo nicht möglich, sperren (z.B. Betriebssystem, Tomcat, Goobi.Production, MySQL).
- Datenverkehr auf System Ports soweit möglich sperren.
- Den Computer gegen unbefugten Zugang sichern (z.B. Rechnerraum abschließen, nicht benötigte Serverdienste abschalten, Passwortübertragungen verschlüsseln, unsichere Passwörter verbieten, zulässige IP-Bereiche für Anmeldevorgänge beschränken).
- Sicherheitsupdates für alle verwendeten Softwarekomponenten immer so zeitnah wie möglich einspielen.
- Das WebInterface in der web.xml auskommentieren, wenn es nicht genutzt werden soll.
- Die Berechtigungen für die Plug-In-Verzeichnisstruktur so restriktiv wie möglich halten.
- Die Ressourcennutzung des Computers überwachen.

**Illegitime Ressourcennutzung****1.4 Hinweise für den Betreiber**

Der Betreiber der Software unterliegt allen seinen gesetzlichen Pflichten. Neben den Sicherheitshinweisen in dieser Anleitung müssen die für den Einsatzbereich der Software gültigen Vorschriften eingehalten werden.

Dabei gilt insbesondere Folgendes:

- Der Betreiber muss die Zuständigkeiten für Installation, Administration und Rechnerbetrieb eindeutig regeln und festlegen.
- Weiterhin ist der Betreiber dafür verantwortlich, dass das Softwaresystem stets in technisch einwandfreiem Zustand ist.

## 1.5 Personalanforderungen



### **LEBENSGEFAHR!**

#### **Verletzungsgefahr bei unzureichender Qualifikation!**

Unsachgemäßer Umgang kann zu erheblichen Personen- und Sachschäden führen.

- Alle Tätigkeiten nur durch dafür qualifiziertes Personal durchführen lassen.

In dieser Anleitung werden die im Folgenden aufgeführten Qualifikationen des Personals für die verschiedenen Tätigkeitsbereiche benannt:

### **System-/Netzwerkadministrator**

Der System-/Netzwerkadministrator ist aufgrund seiner fachlichen Ausbildung, Kenntnisse und Erfahrung sowie Kenntnis der einschlägigen Bestimmungen in der Lage, Arbeiten an elektronischen Datenverarbeitungsanlagen vorzunehmen und mögliche Gefahren selbstständig zu erkennen und zu vermeiden. Der System-/Netzwerkadministrator ist für den speziellen Einsatzort, in dem er tätig ist, ausgebildet und kennt die relevanten Normen und Bestimmungen.

### **Bediener**

Der Bediener nutzt und bedient die Software im Rahmen der bestimmungsgemäßen Verwendung.

Als Bediener und Personal sind nur Personen zugelassen, von denen zu erwarten ist, dass sie ihre Arbeit zuverlässig ausführen. Personen, deren Reaktionsfähigkeit beeinflusst ist, z. B. durch Drogen, Alkohol oder Medikamente, sind nicht zugelassen.

## 2 Änderungen und Erweiterungen

### 2.1 DMS-Export als lang laufende Aufgabe

#### Ziel

Das Ausführen eines DMS- Exports kann langsam sein, wenn viele Bilder kopiert werden müssen. Eine globale Option, um den DMS-Export alternativ als lang laufende Aufgabe im Hintergrund durchführen zu können, sollte hinzugefügt werden.

#### Design-Verbesserungen

Der LongRunningTaskManager wurde durch einen neu geschriebenen Taskmanager ersetzt. Ein TaskSitter pflegt die Aufgabenliste und sorgt für die verzögerte Ausführung von Aufgaben und das Herunterfahren von Threads und Vollstrecker, wenn die Web-App gestoppt wird. Eine neue Klasse EmptyTask wurde als Basisklasse für neue Aufgaben implementiert.

Verbesserungen:

- Der Timer wurde durch einen Executor ersetzt, um ein Herunterfahren des Tomcat zu ermöglichen.
- Der Thread-Zustand wird von Thread-Status, Unterbrechungsstatus und Auftreten einer Ausnahme abgeleitet, nicht mehr von irgendeiner Integer-Variable mit Magischen Zahlen, die in JSP aufgelöst werden.
- Die Anzahl der gleichzeitig startenden Threads ist nun begrenzt und verzögertes Starten ist vorgesehen.
- Ausnahmen werden an einer zentralen Stelle gefangen und können in einem Pop-up-Fenster im Frontend betrachtet werden.
- Das Löschen einer Aufgabe aus der Aufgabenliste impliziert immer auch den Abbruch des Threads.
- Um den Neustart eines vom Benutzer unterbrochenen Tasks zu ermöglichen wird die Klasse geklont. So haben Task-Implementationen nur genau die Variablenfelder, die sie wirklich brauchen.
- Der binäre Dateikopier-Code mit dem ineffizient kleinen Puffer wurde durch einen Aufruf der Apache Commons-Bibliothek, ersetzt, die bereits im Lieferumfang von Produktion enthalten ist.
- Die Ausgaben im Front-End zeigen nun mehr sinnvolle Details. Der Task-Status wird übersetzt.
- Ausführliches Javadoc in den neuen Klassen des Task Packages
- Der Task-Titel wird als Threadname benutzt, welcher im Debugger sichtbar ist.

**Neue Konfigurationseinträge**

Folgende Konfigurationseinträge wurden ergänzt:

`asynchronousAutomaticExport`

ob beim Anklicken des DMS-Export-Buttons ein Task im Taskmanager erstellt wird

`taskManager.autoRunLimit`

maximale Anzahl gleichzeitig zu startender Tasks

`taskManager.inspectionIntervalMillis`

Zeitintervall zwischen zwei Überprüfungen der Taskliste

`taskManager.keepThreads.failed.count`

maximale Anzahl abgestürzter Threads, die noch im Speicher gehalten werden

`taskManager.keepThreads.failed.minutes`

maximale Zeit, die abgestürzte Threads noch im Speicher gehalten werden

`taskManager.keepThreads.successful.count`

maximale Anzahl erfolgreich abgeschlossener Threads, die noch im Speicher gehalten werden

`taskManager.keepThreads.successful.minutes`

maximale Zeit, die erfolgreich abgeschlossene Threads im Speicher gehalten werden

`taskManager.showInSidebar`

ob der Taskmanager in der Navigationsleiste erscheint (ersetzt die alte Option `show_taskmanager` und ist jetzt standardmäßig eingeschaltet)

`taskManager.showSampleTask`

Ob die Option zum „Hinzufügen einer Beispielaufgabe“ angezeigt werden soll (ist jetzt standardmäßig ausgeschaltet)

**Weitere Änderungen**

Weitere Änderungen:

- Beseitigung des doppelt registrierten Listeners für `org.goobi.production.imageIOInitializer` aus der `web.xml` der nichts weiter tat, als beim Hochfahren eine Warnung auszugeben
- Verschiedenen auskommentierten Code gelöscht.
- Beseitigung des `TiffWriterTask`, der nie etwas produktives getan hat.

**2.2 Import aller Subfelder beim PICA-Katalogimport****Ziel**

Der PICA-Import sollte erweitert werden, sodass *alle* Vorkommen eines Unterfelds, das mehrfach erscheinen darf, als wiederholte Vorkommen dieses Metadatenelementes importiert werden. Dies funktionierte beispielsweise bei mehreren Sprachen nicht.

**Lösung**

Wie sich herausstellte handelte es sich um einen Bug, da alle Vorkommen zunächst importiert wurden, bei Sprachen – wo das Fehlen weiterer Subfelder aufgefallen war – aber danach für die Verbalisierung über `goobi_opacLanguages.txt` wieder gelöscht und hier nur

eine Sprache erneut eingetragen wurde. Das PICA-Plug-In wurde korrigiert, dass nun alle Sprachfelder korrekt bearbeitet werden.

## 2.3 Möglichkeit zum Zusammenmappen mehrerer Metadatenfelder

<b>Ziel</b>	Es sollten Möglichkeiten hinzugefügt werden, um mehrere Metadatenfelder in eins zusammenzumappen.
<b>Bisherige Situation</b>	<p>OPAC-Beautifler können in <code>goobi_opac.xml</code> konfiguriert werden. Bislang gab es bereits folgende Verwendungsmöglichkeit:</p> <pre>&lt;setvalue tag="010@" subtag="a" value="ger"&gt;   &lt;condition tag="010@" subtag="a" value="deu?"/&gt; &lt;/setvalue&gt;</pre> <p>Wenn ein Datensatz ein Feld, wie es mit den <code>setvalue</code> <code>tag</code> und <code>subtag</code> Attributen bezeichnet ist, enthält und wenn alle als <code>condition</code> angegebenen Bedingungskonstrukte erfüllt sind, wobei <code>value</code> ein Regulärer Ausdruck gemäß Java-Spezifikation ist, der auf den gesamten Feldinhalt des in dem <code>condition</code>-Konstrukt benannten Subfeldes matchen muss (das heißt, das obige Beispiel entspricht <code>/^deu?\$/</code> in Perl-Notation), wird das in <code>setvalue</code> benannte Feld auf den dort angegebenen Wert gesetzt.</p>
<b>Neuer Parameter <code>mode</code></b>	<p>Um mehrere Metadatenfelder in eins zusammenzumappen wurden die OPAC-Beautifler-Definitionen um einen neuen, optionalen Parameter <code>mode</code> erweitert. Ohne Angabe von <code>mode</code> bleibt das bisherige Verhalten bestehen. Das Attribute <code>mode</code> wie nachfolgend dargestellt zu setzen erzeugt ebenfalls das bisherige Verhalten:</p> <pre>&lt;setvalue tag="010@" subtag="a" value="ger" mode="replace"&gt;   &lt;condition tag="010@" subtag="a" value="deu?" mode="matches"/&gt; &lt;/setvalue&gt;</pre>
<b>Erweiterte Funktionalität</b>	<p>Erweiterungen:</p> <ul style="list-style-type: none"> <li>• Es ist jetzt alternativ möglich, im <code>condition</code> Konstrukt <code>mode="find"</code> zu verwenden, mit dem Ziel, lediglich eine Teilzeichenkette im angegebenen Feld zu finden, anstelle gegen den gesamten Feldinhalt zu matchen. (Das obige Beispiel würde dann <code>/deu?/</code> in Perl-Notation entsprechen.)</li> <li>• Es ist jetzt möglich, das Ergebnis der (letzten, wenn es mehrere sind) überprüften Bedingung im <code>value</code>-Attribut des <code>setvalue</code>-Konstrukts zu verwenden. Hierzu gibt es zwei Möglichkeiten: <ul style="list-style-type: none"> <li>○ <code>{@}</code> kann als Platzhalter für alle aneinandergehängten Ergebnisse des wiederholten Aufrufens von <code>find</code> verwendet werden</li> <li>○ <code>{1}</code>, <code>{2}</code>, <code>{3}</code>, ... kann verwendet werden, um gematchte Gruppen einzufügen.</li> </ul> </li> <li>• Es ist jetzt möglich, für <code>setvalue</code> die Modi <code>prepend</code> oder <code>append</code> zu verwenden, um den Wert aus <code>value</code> vorne oder hinten anzufügen.</li> <li>• Da die in Goobi zum Zugriff auf <code>goobi_opac.xml</code> verwendete API Leerzeichen vorne und hinten von Attributen entfernt</li> </ul>

wurde die Möglichkeit ergänzt, eine offene Schachtel („`<u>`“, U+2423) als Platzhalter für ein Leerzeichen zu verwenden.

## Beispiele

Beispiele:

```
<!-- Löschen aller @-Zeichen aus 027A a -->
<setvalue tag="027A" subtag="a" value="{@}" mode="replace">
  <condition tag="027A" subtag="a" value="^[@]" mode="find"/>
</setvalue>

<!-- Den Wert von 064D b getrennt durch ein Leerzeichen
      vor 046D a schreiben, wenn seine Länge > 0 ist -->
<setvalue tag="046D" subtag="a" value="{@}_ " mode="prepend">
  <condition tag="046D" subtag="b" value="." mode="matches"/>
</setvalue>
```

## 2.4 Verwaltung gruppierter Metadaten

### Ziel

Es sollte ermöglicht werden, gruppierte Metadaten zu importieren, zu bearbeiten und anzuzeigen, sowie zu exportieren.

### Bisherige Situation

Eine Unterstützung gruppierter Metadaten ist in der Goobi zugrunde liegenden Programmbibliothek UGH bereits enthalten, die Verwendung dieser Funktionen war jedoch nicht offensichtlich dokumentiert und gruppierte Metadaten konnten weder importiert, noch in der Goobi-Oberfläche betrachtet oder verändert werden.

### Erweiterungen

Der Metadateneditor wurde erweitert, sodass Metadatengruppen angelegt werden können, angezeigt werden und geändert und gelöscht werden können. Die Subfelder sprechen auf die Konfiguration mit Metadaten Display Rules an.

Das PICA-Plug-In wurde erweitert, sodass Metadaten in Gruppen importiert werden können.

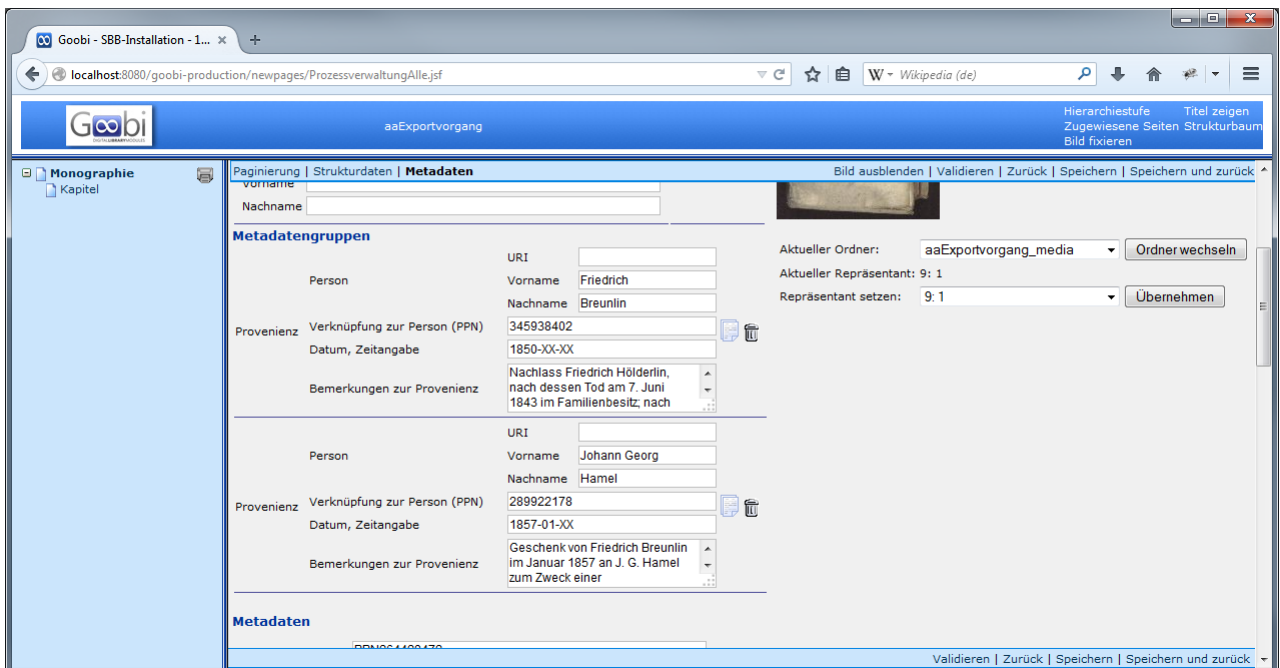


Bild 1: Metadaten im Metadateneditor

## Konfiguration

Um Metadaten Gruppen in Goobi nutzen zu können, müssen im Regelsatz zuerst die Metadaten Typen definiert werden, die in der Gruppe vorkommen.

Die soll am Beispiel von Provenienzdaten (PICA+-Feld 220B) erläutert werden:

```
<MetadataType type="person">
  <Name>ProvPerson</Name>
  <language name="de">Person</language>
  <language name="en">Person</language>
</MetadataType>
<MetadataType>
  <Name>ProvPersonPPN</Name>
  <language name="de">Verknüpfung zur Person (PPN)</language>
  <language name="en">Link to the person (PPN record)</language>
</MetadataType>
<MetadataType>
  <Name>ProvDateTime</Name>
  <language name="de">Datum, Zeitangabe</language>
  <language name="en">Date and time</language>
</MetadataType>
<MetadataType>
  <Name>ProvRemarks</Name>
  <language name="de">Bemerkungen zur Provenienz</language>
  <language name="en">Remarks to the provenience</language>
</MetadataType>
```

Dann muss im Regelsatz die Metadaten Gruppe definiert werden:

```
<Group>
  <Name>Proveniece</Name>
  <language name="de">Provenienz</language>
  <language name="en">Proveniece</language>
  <metadata>ProvPerson</metadata>
  <metadata>ProvPersonPPN</metadata>
  <metadata>ProvDateTime</metadata>
  <metadata>ProvRemarks</metadata>
</Group>
```

Desweiteren muss die Metadaten Gruppe zu denjenigen DocStrctType-Definitionen hinzugefügt werden, bei denen sie erlaubt sein soll:

```
<DocStrctType>
  <Name>Monograph</Name>
  <language name="en">Monograph</language>
  <language name="de">Monographie</language>
  <!-- ... -->
  <group num="*">Proveniece</group>
</DocStrctType>
```

Optional kann die gewünschte Art der Eingabeelemente in goobi\_metadataDisplayRules.xml konfiguriert werden:

```
<ruleSet>
  <context projectName="My project name">
    <bind>create</bind>
    <!-- ... -->
    <input tns:ref="ProvPersonPPN" />
    <input tns:ref="ProvDateTime" />
  </context>
</ruleSet>
```

```

        <textarea tns:ref="ProvRemarks" />
    </context>
    <context projectName="My project name">
        <bind>edit</bind>
        <!-- ... -->
        <input tns:ref="ProvPersonPPN" />
        <input tns:ref="ProvDateTime" />
        <textarea tns:ref="ProvRemarks" />
    </context>

```

Wenn die Metadaten­gruppe aus PicaPlus importiert werden soll, muss ein entsprechendes Import-Mapping angegeben werden:

```

<PicaPlus>
  <!-- ... -->
  <MetadataGroup>
    <Name>Provenience</Name>
    <picaMainTag>220B</picaMainTag>
  </MetadataGroup>
  <Person>
    <Name>ProvPerson</Name>
    <picaMainTag>220B</picaMainTag>
    <picaSubTag type="expansion">8</picaSubTag>
  </Person>
  <Metadata>
    <Name>ProvPersonPPN</Name>
    <picaMainTag>220B</picaMainTag>
    <picaSubTag>9</picaSubTag>
  </Metadata>
  <Metadata>
    <Name>ProvDateTime</Name>
    <picaMainTag>220B</picaMainTag>
    <picaSubTag>d</picaSubTag>
  </Metadata>
  <Metadata>
    <Name>ProvRemarks</Name>
    <picaMainTag>220B</picaMainTag>
    <picaSubTag>k</picaSubTag>
  </Metadata>
</PicaPlus>

```

Wenn die Metadaten­gruppe nach METS/MODS exportiert werden soll, muss ein Exportmapping angegeben werden:

```

<METS>
  <!-- ... -->
  <Group>
    <InternalName>Provenience</InternalName>
    <WriteXPath>./mods:mods/#mods:note[@type='custodialHistory']</WriteXPath>
    <Person>
      <InternalName>ProvPerson</InternalName>
      <WriteXPath>./mods:name[@type='ownership']</WriteXPath>
      <FirstnameXPath>./mods:namePart[@type='given']</FirstnameXPath>
      <LastnameXPath>./mods:namePart[@type='family']</LastnameXPath>
      <DisplayNameXPath>./mods:displayForm</DisplayNameXPath>
    </Person>
  </Group>

```



```

<Metadata>
  <InternalName>ProvPersonPPN</InternalName>
  <WriteXPath>./mods:name[@type='ownership'][@authority='swb-ppn']/@ppn</WriteXPath>
</Metadata>
<Metadata>
  <InternalName>ProvDateTime</InternalName>
  <WriteXPath>./mods:date[@encoding='YYYY-MM-DD']</WriteXPath>
</Metadata>
<Metadata>
  <InternalName>ProvRemarks</InternalName>
  <WriteXPath>./mods:note[@type='description']</WriteXPath>
</Metadata>
</Group>
</METS>

```

Listing 1: Exportmapping für Metadatengruppen

## 2.5 Metadaten beim Import bedingt kopieren

### Ziel

Es sollte eine Möglichkeit geschaffen werden, Metadatenfeldern mit den Werten anderer Felder zu belegen, wenn diese nach einem Katalogimport leer geblieben sind. Beispielsweise sollte es damit möglich sein,

- Die Sortierzählung eines Bandes in die übergeordnete Einheit zu übernehmen oder
- Den Titel eines Bandes mit dem Titel der übergeordneten Einheit zu befüllen.

### Lösung

Es wurde ein neuer Konfigurationseintrag `copyData.onCatalogue Query` vorgesehen. Mit diesem können Kopierregeln definiert werden, die nach einem Datenimport angewendet werden und es u.a. ermöglichen, den Inhalt eines Metadatenfeldes einem anderen Metadatenfeld zuzuweisen, wenn ersteres nicht bereits vorhanden ist.

Eine Kopierregel hat das Format *Ziel – Zuweisungsoperator – Argument*, wobei diese Elemente durch Leerzeichen („ “, U+0020) voneinander getrennt werden. Mehrere Kopierregeln können durch Semikola („;“, U+003B) getrennt werden. Eine Regel wird immer nur dann angewendet, wenn das Argument aufgelöst werden kann und – in diesem Fall – nur dann, wenn das Ziel noch nicht existiert.

Das **Ziel** wird durch einen Metadatenauswahlausdruck beschrieben, der einen Pfad durch die logische Dokumentstruktur beschreibt. Die Pfad-Elemente werden durch Schrägstriche („/“, U+002F) voneinander abgegrenzt. Ein einzelner Schrägstich markiert den Wurzelknoten des logischen Dokumentstrukturbaumes. Es kann entweder nach einem bestimmten Strukturelement selektiert, oder mit einem Stern („\*“, U+002A) ein beliebiges Strukturelement angesprochen werden. Mit eckigen Klammern kann ein Element indiziert werden. Die Zählung ist null-basiert. Eine schließende Spitzklammer („>“, U+003E) referenziert das jeweils letzte Element. Das auszuwählende Metadatum wird durch ein vorangestelltes At-Zeichen („@“, U+0040) bezeichnet.

Beispiele:

Ausdruck	Erläuterung
<code>/@TitleDocMain</code>	Titel des obersten Strukturelementes
<code>/*[0]@CurrentNoSorting</code>	Sortier-Nr. des ersten Kindelementes

Für den Zweck, Metadatenfelder nur dann anzulegen, wenn es das entsprechende Metadatenfeld nicht bereits gibt wurde der bedingte **Zuweisungsoperator** `""=` implementiert, der als Argument einen Verweis auf die zuzuweisenden Daten erwartet. Auch hierfür kann wiederum ein Metadatenauswahlausdruck verwendet werden, sodass sich für den beschriebenen Verwendungszweck die folgenden beiden Zuweisungen ergeben:

Metadatenkopierregel	Erläuterung
<code>/@CurrentNoSorting ""= /*[0]@CurrentNoSorting</code>	Setze die Sortierzählung des obersten Strukturelementes auf den Wert des ersten Kindelementes, wenn das oberste Strukturelement noch kein Metadatum Sortierzählung trägt.
<code>/*[0]@TitleDocMain ""= /@TitleDocMain</code>	Setze den Titel des ersten Kindelementes auf den Wert des obersten Strukturelementes, wenn das erste Kindelement existiert, aber noch kein Metadatum Titel trägt.

Beim Erstellen des Konfigurationseintrags mit diesen beiden Regeln soll dem Gleichheitszeichen („=“, U+003D) ein Rückstrich („\“, U+005C) vorangestellt werden:

```
copyData.onCatalogueQuery= /@CurrentNoSorting ""\= /*[0]@
CurrentNoSorting;/*[0]@TitleDocMain ""\= /@TitleDocMain
```

## 2.6 Auswahl eines Vorschaubildes im Metadateneditor

Es sollte ermöglicht werden, ein Vorschaubild im Metadateneditor auszuwählen und beim METS-Export in Form einer speziellen Dateigruppe, die nur dieses Vorschaubild enthält, zu exportieren.

Goobi wurde so erweitert, dass in den Projekteinstellungen eine Dateigruppe konfiguriert werden kann, in welche nur das ausgewählte Vorschaubild exportiert wird.

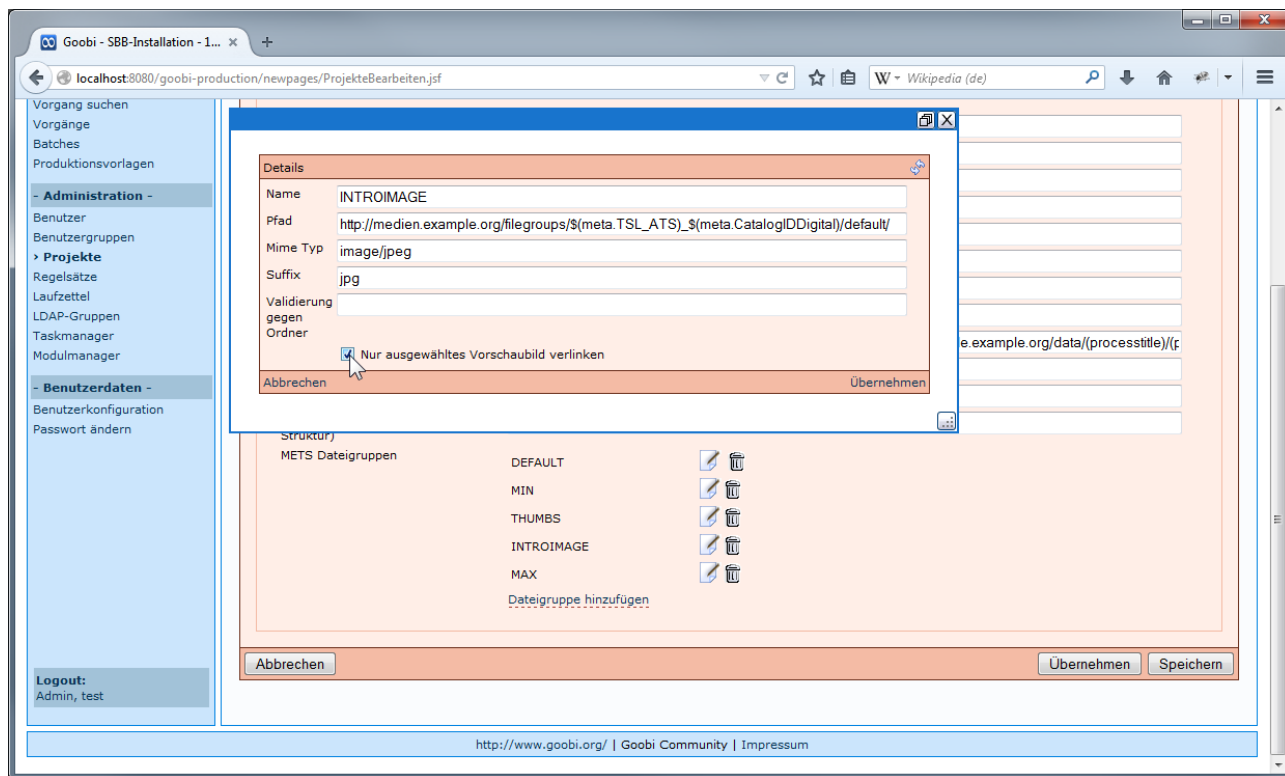


Bild 2: In den Einstellungen für die METS Dateigruppen kann eine Dateigruppe definiert werden, in die nur das ausgewählte Vorschaubild exportiert wird.

Im Regelsatz muss der Metadattentyp `_representative` konfiguriert werden:

```
<MetadataType>
  <Name>_representative</Name>
  <language name="de">Vorschaubild</language>
  <language name="en">Preview image</language>
</MetadataType>
```

Dieser muss dem DocStrctType `page` zugeordnet werden:

```
<DocStrctType>
  <Name>page</Name>
  <!-- ... -->
  <metadata num="10">_representative</metadata>
</DocStrctType>
```

Im Metadateneditor lässt sich unterhalb des Vorschaubildes der Repräsentant auswählen. Der Editor wurde erweitert, sodass das aktuell eingestellte Vorschaubild über dem Auswahlfenster angezeigt wird.

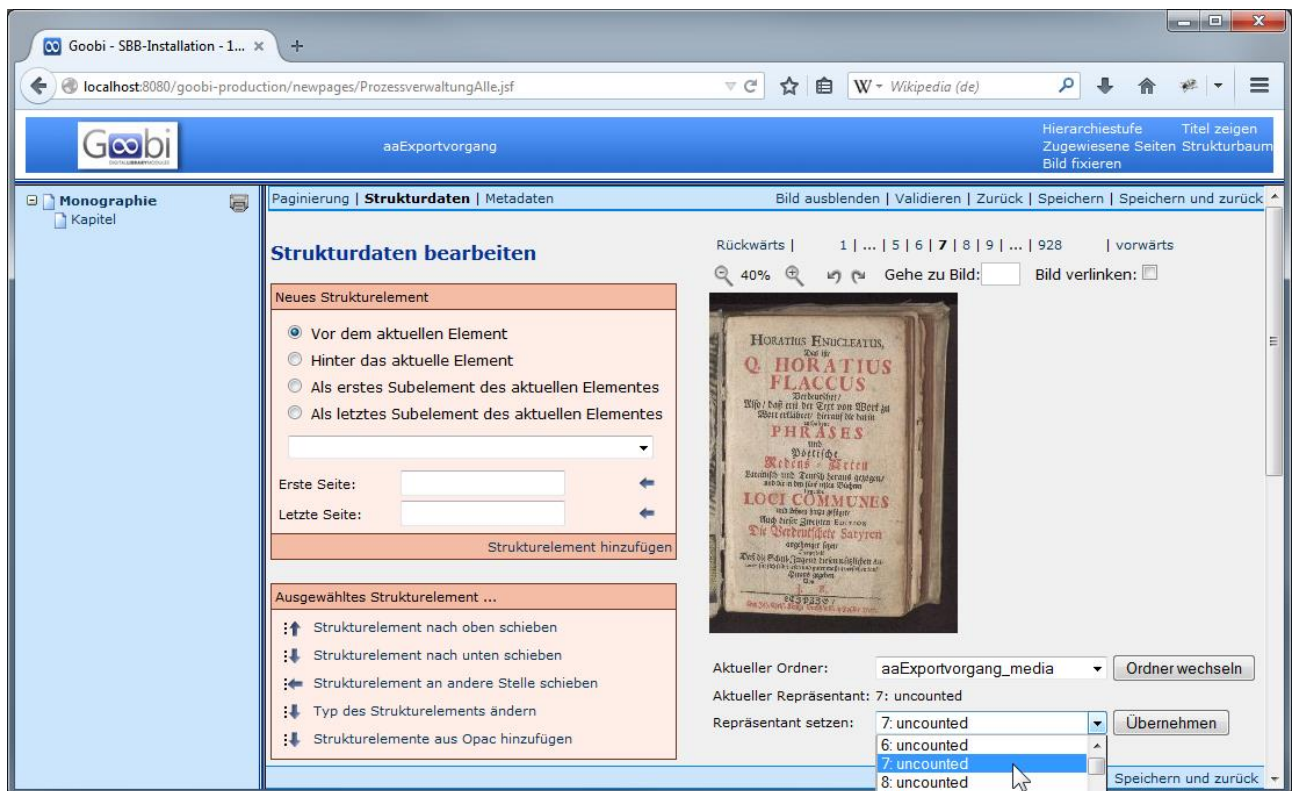


Bild 3: Im Metadateneditor lässt sich der Repräsentant setzen.

## 2.7 Prozess-ID exportieren

### Ziel

Es sollte eine Möglichkeit geschaffen werden, die interne Prozess-ID während des METS-Exports in ein Metadatenfeld zu exportieren.

### Lösung

Es wurde ein neuer Konfigurationseintrag `copyData.onExport` vorgesehen. Mit diesem können Kopierregeln definiert werden, die während des Datenexports angewendet werden und es u.a. ermöglichen, den Inhalt einer internen Variable einem Metadatenfeld zuzuweisen.

Eine Kopierregel ist wie in Abschnitt 2.5 definiert. Das **Ziel** wird wieder durch einen Metadatenauswahlausdruck, wie in Abschnitt 2.5 definiert, beschrieben.

Für den Zweck, ein Metadatenfeld anzulegen wurde der **Zuweisungsoperator** = implementiert, der als Argument einen Verweis auf die zuzuweisenden Daten erwartet.

Zum Zugriff auf die internen Variablen wurde ein **Variablenselektor** implementiert. Ein Variablenselektor besteht aus einem Dollarzeichen („\$“, U+0024) gefolgt vom Namen der auszuwählenden Variablen, ausgehend von der Klasse `de.sub.goobi.metadaten.copier.CopierData`. Mehrere Variablenselektoren können mit einem Punkt („.“, U+002E) verknüpft werden. Als Ergebnis wird `String.valueOf()` der letzten Variable der Folge zurückgegeben.

Beispiele:

Variablenselektor	Erläuterung
<code>\$process.id</code>	Goobi Prozess-ID
<code>\$process.docket.file</code>	Dateiname des Laufzettels
<code>\$process.projekt.titel</code>	Name des Digitalisierungsprojektes

Beim Erstellen des Konfigurationseintrags mit diesen beiden Regeln soll dem Gleichheitszeichen („=“, U+003D) ein Rückstrich („\“, U+005C) vorangestellt werden:

```
copyData.onExport= /@GoobiProcessID \= $process.id
```

## 2.8 Export fortlaufender Sammelwerke mit Querbezügen

### Ziel

Es sollte eine Funktionalität entwickelt werden, sodass beim Export fortlaufender Sammelwerke in allen Ankerdateien die Querbezüge zu allen Werken mit ausgegeben werden.

### Konfiguration

Im Regelsatz muss ein Metadattentyp `MetsPointerURL` konfiguriert und dem `DocStrctType` `Volume` zugeordnet werden:

```
<MetadataType>
  <Name>MetsPointerURL</Name>
  <language name="de">METS-Verweis-URL</language>
  <language name="en">METS pointer URL</language>
</MetadataType>
<!-- ... -->
<DocStrctType>
  <Name>Volume</Name>
  <!-- ... -->
  <metadata num="*">MetsPointerURL</metadata>
</DocStrctType>
```

### Bedienung

Alle Prozesse, die zu derselben übergeordneten Gesamtheit gehören müssen im Bildschirmdialog *Batches* einem Batch zugeordnet und für diesen der Typ „fortlaufendes Sammelwerk“ gesetzt werden. Dann können die Vorgänge über den Schalter *[Gesamten Batch ins DMS exportieren]* gemeinsam exportiert werden, wobei Querbezüge zu allen Bänden in den Ankerdateien aller Vorgängen generiert werden.

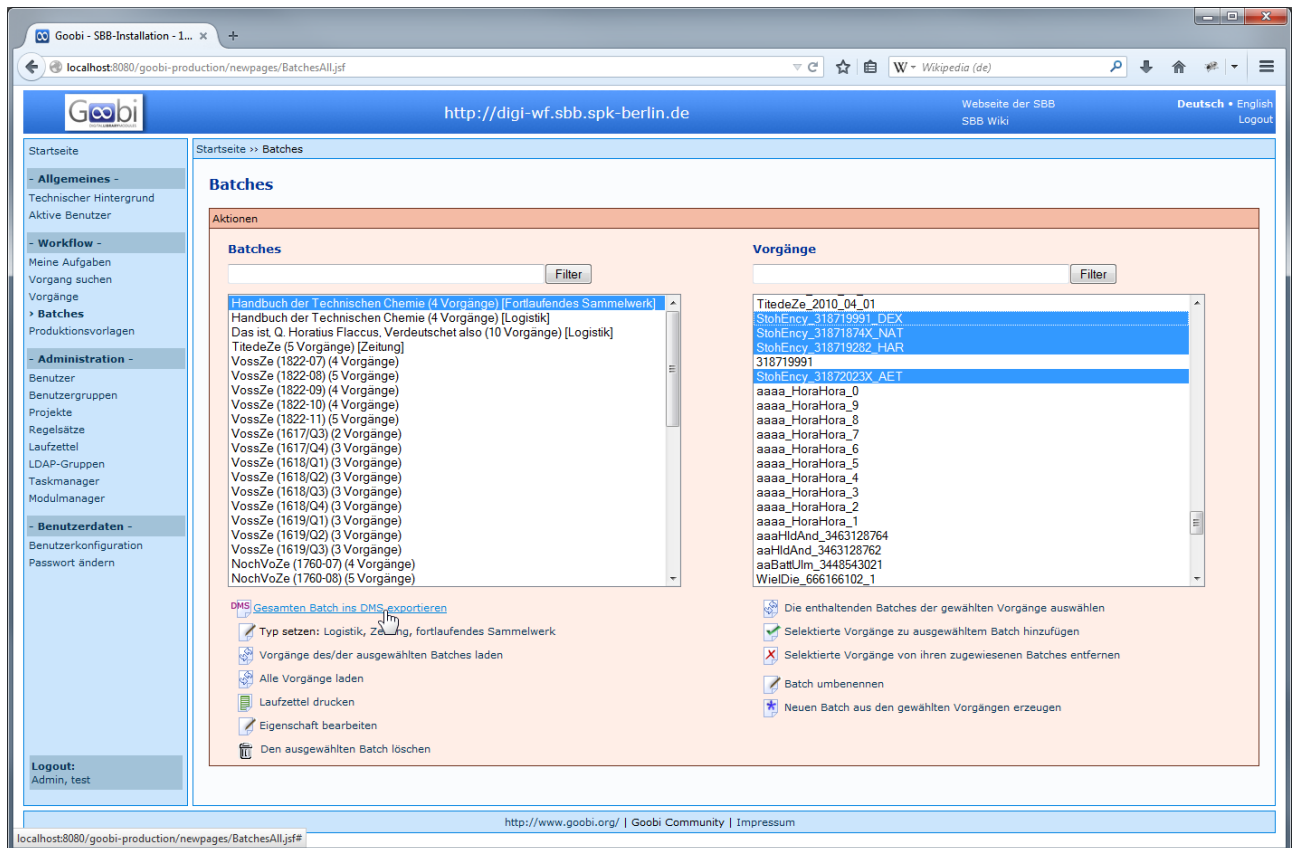


Bild 4: Export fortlaufender Sammelwerke aus dem Dialog „Batches“.