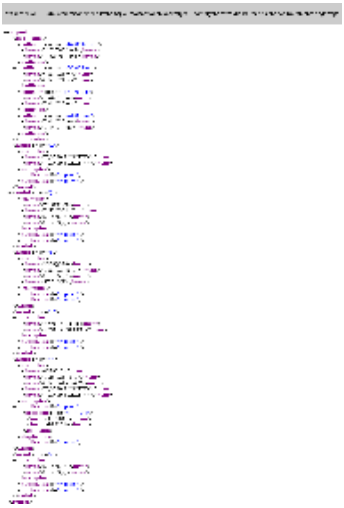
Report zur Inventarnummerverwendung

Die Inventarnummer-Analyse-Pipeline erzeugt einen Report im XML-Format, mit dem sich manche wahrscheinlichen Duplikate in den Inventarnummern, sowie einige weitere Artefakte in der Inventarnummerverwendung aufspüren lassen. Für eine brauchbare Ansicht sollte die Datei in einem Browser geöffnet werden, hier z.B. in Firefox (sieht in Chrome sehr ähnlich aus):



Die Fehlermeldung "*Mit dieser XML-Datei sind anscheinend keine Style-Informationen verknüpft. Nachfolgend wird die Baum-Ansicht des Dokuments angezeigt.*" ist erwartet: Die Baumansicht ist bereits recht übersichtlich, und eine formatierte Ansicht mit gleicher Funktionalität nicht ohne weiteres realisierbar.

Ausführung

Die Inventarnummer-Analyse-Pipeline muss mit den korrekten Werten für das jeweilige Museum ausgeführt werden, entsprechend den jeweiligen Regeln zu Bildung der reduzierten Inventarnummern der Dateinamenskonvention für die Medienbereitstellung.

Zeichenkategorie	Konfigurationsparameter	Beschreibung	im Beispiel verwendet
Trennzeichen	expoSync.invnr.replace	Zeichen, die in der Inventarnummer durch Bindestrich ersetzt werden	• Leerzeichen • Bindestrich • Schrägstrich / • Punkt
entfernte Zeichen	expoSync.invnr.remove	Zeichen, die in der Inventarnummer restlos entfernt werden	• Apostroph '
verbotene Zeichen	expoSync.invnr.ignore	Zeichen, die nicht gemeldet werden, weil deren Präsenz in einer Inventarnummer diese explizit als nicht für Zuordnung geeignet markiert. Alle anderen Zeichen werden im Report als unerwartetes Sonderzeichen gemeldet.	• Raute #

Dies geschieht entweder in der Job-Konfiguration für regelmäßige Jobs oder nach Auswahl der Pipeline, wenn die Pipeline manuell ausgeführt wird. In letzteren Fall bitte die Email-Adressen leer lassen und den Report direkt herunterladen.

Meldungen

Duplikate

Hier werden alle Inventarnummern angezeigt, die nach den angegebenen Regeln auf dieselbe reduzierte Inventarnummer gemappt werden. Bilder für diese Inventarnummern können nicht automatisch verknüpft werden, weil nicht klar ist, auf welche der beiden Inventarnummern die reduzierte Inventarnummer zu mappen ist. Hier mit den Trennzeichen: Leerzeichen, Schrägstrich /, Leerzeichen, sowie als entferntes Zeichen.


```
−<duplicates>
  −<collision code="h-62-001-10-1">
      <invnr>H 62/001/10.1</invnr>
      <invnr>H 62/001.10.1</invnr>
    </collision>
  −<collision code="g-53-021-w">
      <invnr>G 53/021.w</invnr>
      <invnr>G 53/021 w</invnr>
    </collision>
  −<collision code="d-44-019-o">
      <invnr>D 44/019 o</invnr>
      <invnr>D 44/019 o'</invnr>
    </collision>
  +<collision code="d-44-019-m"></collision>
  </duplicates>
```


Dabei gibt `code="xyz"` die reduzierte Inventarnummer an; die `<invnr>`-Tags die Inventarnummern, die auf diese reduzierte Inventarnummer reduziert werden.

Dieser Bereich findet aber auch manche doppelte Inventarnummern. Hier sind H 62/001/10.1 und H 62/001.10.1, sowie G 53/021.w und G 53/021 w, wahrscheinlich zwei Einträge für das gleiche Objekt, d.h., diese Datensätze sollten genauer überprüft werden. Wenn es sich um zwei verschiedene Objekte handelt oder wenn die Inventarnummern so strukturiert sind, dass es sich um semantisch verschiedene Nummern handelt, müssen die Regeln zu Bildung der reduzierten Inventarnummern entsprechend angepasst werden.

D 44/019 o und D 44/019 o' illustrieren einen anderen Fall: Apostroph ' wurde als entferntes Zeichen angegeben, weil einige doppelte Inventarnummern nach dem mathematischen Schema a, a' benannt wurden. Damit hier d-44-019-o nicht D 44/019 o zugeordnet wird, obwohl eventuell D 44/019 o' richtig wäre, werden beide Inventarnummern absichtlich auf dieselbe reduzierte Inventarnummer reduziert. Dadurch wird die automatische Zuordnung verhindert und es muss manuell das korrekte Objekt ausgewählt werden.

Solche Duplikate können nicht versteckt werden. Der Eintrag im Report lässt sich zusammenklappen (d-44-019-m im Beispiel), aber er lässt sich nicht ganz entfernen. Deshalb wird empfohlen, Objekte mit doppelter Inventarnummer nach dem Schema "A 123 #1", "A 123 #2", "A 123 #3" umzubenennen (siehe Dateinamenskonvention für die Medienbereitstellung). Das Beispiel hatte doppelte Inventarnummern nach Schema "A 123 #1", aber da Raute # als verbotenes Zeichen angegeben wurde, werden diese nicht gemeldet. Sie können stattdessen sehr einfach in imdas pro gefunden werden.

Wenn zwei Inventarnummern exakt gleich aussehen, ist eine zusätzliche Recherche notwendig. Sie können nicht exakt gleich sein, das würde imdas pro nicht zulassen (wenn die automatische Inventarnummernkontrolle aktiviert ist). Einige wenige Zeichen werden im Browser nicht oder nicht richtig angezeigt, so dass es sinnvoll sein kann, die Datei in einem Texteditor (z.B. Notepad++, nicht der Windows-Editor Notepad) zu öffnen und nach der Inventarnummer zu suchen. Leerzeichen am Ende einer Inventarnummer (wie in einem Museum praktiziert, um die automatische Inventarnummernkontrolle zu überlisten) sind z.B. auch sehr schwer zu erkennen und in einem Texteditor mit fester Zeichenbreite leichter zu sehen.

 in der reduzierten Inventarnummer ist ein nicht erlaubtes Zeichen und sollte unter Sonderzeichen nochmals aufgelistet werden. Die Ausnahme sind die expliziten Verbotenen Zeichen; diese tauchen in der reduzierten Inventarnummer ebenfalls als auf.

Sonderzeichen

Sonderzeichen können aus den verschiedensten Gründen in der Inventarnummern vorkommen und können harmlos sein oder unerwünscht. Aus Sicht der Medienbereitstellung verhindern sie aber die Zuordnung eines Medienobjekts zu einem Museumsobjekt mit dieser Inventarnummer.

Typisches Beispiel für eine Meldung in diesem Bereich:

```
- <weird char="+">
  - <examples>
      <invnr>NM01-710a+b</invnr>
      <invnr>NM01-K134+135</invnr>
    </examples>
    <duplicates if="replace"/>
    <duplicates if="remove"/>
  </weird>
```

`char="+"` gibt hier das Sonderzeichen an, wobei "exotische" Zeichen u.U. nicht richtig angezeigt werden. Dann werden alle Inventarnummern aufgezählt, die dieses Sonderzeichen enthalten. Wenn dies sehr viele sind, lohnt es sich den `<examples>`-Tag zuzuklappen.

Hier wird das Plus-Zeichen offensichtlich für "NM01-710a und NM01-710b" sowie für "NM01-K134 und NM01-K135" benutzt. Im ersten Fall wäre (in diesem Museum) "NM01-710 a-b" korrekt, und der Datensatz sollte korrigiert werden. Für den zweiten Fall muss entschieden werden, wie solche Kombinations-Objekte benannt werden. Z.B. könnten sie als "NM01-K134; NM01-K135" erstellt (und Semicolon ; als Trennzeichen hinzugefügt), oder so wie vorhanden repräsentiert (und Plus + als Trennzeichen hinzugefügt) werden.

Für jedes Sonderzeichen wird außerdem gemeldet, ob es zu Kollisionen kommen würde, wenn das Zeichen zu den Trennzeichen (`if="replace"`) oder den Entfernten Zeichen (`if="remove"`) hinzugefügt würde:

```
- <weird char="_">
  - <examples>
      <invnr>00492_2</invnr>
      <invnr>D 38/021.16_2</invnr>
      <invnr>G 55/041 c.24_2</invnr>
      <invnr>III/3589_LÖSCHEN</invn
      <invnr>III/3590_LÖSCHEN</invn
    </examples>
  - <duplicates if="replace">
    - <collision code="i-1810-b">
        <invnr>I_1810 b</invnr>
        <invnr>I/1810 b</invnr>
      </collision>
    </duplicates>
    <duplicates if="remove"/>
  </weird>
```

Hier würde z.B. eine Kollision erzeugt, wenn der Unterstrich _ zu den Trennzeichen hinzugefügt würde. Als Entferntes Zeichen würde er keine Kollision erzeugen. Nachdem "I_1810 b" und "I/1810 b" aber vermutlich dasselbe Objekt sind, wäre es hier sinnvoller, beide als Objekte mit doppelter Inventarnummer zu kennzeichnen (oder direkt eines davon zu entfernen, wenn offensichtlich).

Der Suffix "_2" könnte ebenfalls auf doppelte Inventarnummern hindeuten, oder einfach Teil des Inventarnummerschemas sein. Da es keine "_1" Objekte gibt, handelt es sich vermutlich um Duplikate, aber der Report versucht nicht, solche semantischen Entscheidungen zu treffen.

Klammern

Klammern werden meist in Paaren gemeldet, da von Menschen zumeist in Paaren benutzt. Ein typischer Fall ist "(?)" bei unsicheren oder unlesbaren Inventarnummern:

```
−<weird char="(">
  −<examples>
      <invnr>2008()029</invnr>
      <invnr>D 53/030 k (2)</invnr>
      <invnr>ST 675 (?)</invnr>
      <invnr>ST 713(?)</invnr>
    </examples>
    <duplicates if="replace"/>
    <duplicates if="remove"/>
  </weird>
−<weird char=")">
  −<examples>
      <invnr>2008()029</invnr>
      <invnr>D 53/030 k (2)</invnr>
      <invnr>ST 675 (?)</invnr>
      <invnr>ST 713(?)</invnr>
    </examples>
    <duplicates if="replace"/>
    <duplicates if="remove"/>
  </weird>
−<weird char="?">
  −<examples>
      <invnr>ST 675 (?)</invnr>
      <invnr>ST 713(?)</invnr>
    </examples>
    <duplicates if="replace"/>
    <duplicates if="remove"/>
  </weird>
```


Hier lohnt sich eventuell eine Überprüfung der entsprechenden Objekte, sofern praktikabel. Vor allem, weil diese "unsicheren" Inventarnummern in der Regel eindeutig sind, d.h. es gibt nur "ST 675 (?)", aber nicht "ST 675".

Bei Klammern werden in der Regel keine "Was-wäre-wenn"-Duplikate gefunden, weil der Report nur *einzelne* Zeichen betrachtet. Egal welches einzelne Zeichen von "(", "?" und ")" ersetzt wird, "ST 675 (?)" und "ST 675" wären aber immer noch verschieden ("ST 765 ?)", "ST 765 ()" sowie "ST 765 (?"). Hier lohnt es sich also, den Report nochmal zu erzeugen, diesmal mit allen drei Zeichen als Entfernte Zeichen. Eventuelle Kollisionen werden dann ganz oben unter `<duplicates>` angezeigt.

Die Entstehung von "2008()029" ist kryptisch. Das Beispiel ist hier zur Illustration, dass nicht alle Anomalien in den Inventarnummern rational erscheinen müssen.

Doppelte Leerzeichen

Diese spezielle Regel sucht nach Inventarnummern, die 2 Leerzeichen nacheinander enthalten, oder mit Leerzeichen beginnen oder enden. Diese sind normalerweise kein Problem für die Medienbereitstellung: Durch die üblichen Regeln werden doppelte Leerzeichen zu einem einfachen "-"; Leerzeichen am Anfang oder Ende werden ganz entfernt. Leerzeichen an unerwarteten Stellen erzeugen aber visuell erhebliche Verwirrung und werden daher gemeldet.


```
−<double-space>
    <invnr search="065034__doppelt">065034 doppelt</invnr>
    <invnr search="OA__26529_a-g">OA 26529 a-g</invnr>
    <invnr search="SA__02065_L">SA 02065 L</invnr>
    <invnr search="SA__02473_L">SA 02473 L</invnr>
  </double-space>
−<space-start>
    <invnr search="_A_41626"> A 41626</invnr>
  </space-start>
−<space-end>
    <invnr search="SA_02229_L_">SA 02229 L </invnr>
    <invnr search="SA_02438_L_">SA 02438 L </invnr>
    <invnr search="MO-Kopie_">MO-Kopie </invnr>
  </space-end>
```

Diese Inventarnummern sollten fast immer korrigiert werden. Doppelte Leerzeichen sind im Report leider komplett unsichtbar (sie sind in der Datei drin, aber Browser zeigen sie nicht an) und man kann sie auch nur als einzelnes Leerzeichen rauskopieren. Deshalb wird zusätzlich eine "Suchhilfe" ausgegeben, in der die Leerzeichen durch "_" (Unterstrich; Platzhalter für ein beliebiges einzelnes Zeichen in der imdas-Suche) ersetzt sind. Diese illustrieren, wo die Leerzeichen sind, und lassen sich direkt kopieren zur Suche nach den "Übeltätern". Allerdings werden u.U. auch andere, unproblematische Datensätze gefunden, z.B. findet das obige "SA_02229_L_" natürlich auch "SA 02229-LC".

False Positives

Typisches Beispiel für harmlose Meldungen sind das "ö" in "löschen", oder andere Sonderzeichen in Objekten, die offensichtlich gelöscht werden sollen:

```
- <weird char="Ö">
  - <examples>
      <invnr>III/3589_LÖSCHEN</invnr>
      <invnr>III/3590_LÖSCHEN</invnr>
    </examples>
    <duplicates if="replace"/>
    <duplicates if="remove"/>
  </weird>
```

False Negatives (Dinge, die der Report nicht findet)

Unsinnige Verwendung von Trennzeichen

Wenn Trennzeichen in einer für Menschen unsinnigen Form benutzt werden, findet der Report das nur, wenn das zu Duplikaten bei der reduzierten Inventarnummer führt.

"NM01--710a" (zwei Bindestriche hintereinander) wird z.B. nur gefunden, wenn es auch "NM01-710a" (nur ein Bindestrich) gibt. Ansonsten wird "NM01--710a" zu "nm01-710a" und ist verknüpfbar. Entsprechend sind "A 123 /- 5", "A 123 // 2", "A 123 (1(" oder "A 123 -" für Menschen offensichtlich merkwürdig, aber solange die reduzierten Inventarnummern "a-123-5", "a-123-2", "a-123-1" sowie "a-123" eindeutig sind, findet der Report nichts.

Duplikate, die mit erlaubten Trennzeichen aufgelöst wurden

Das System hat keine Chance zu erkennen, ob "D 38 / 030 - 2" der zweite Teil eines zweiteiligen Objekts, ein hier semantisch korrektes Zeichen zur Strukturierung der Inventarnummer, oder ein Duplikat zu "D 38 / 030" ist. Darunter fällt auch "D 38 / 030 doppelt": diese Nummer wird auf die eindeutige reduzierte Inventarnummer "d-38-030-doppelt" gemappt und vom Report nicht gefunden. (Dafür allerdings mit einer Suche nach "doppelt" in imdas pro.)

Es ist deshalb empfehlenswert, Duplikate so zu markieren, dass sie offensichtlich erkennbar sind. Die Verwendung von #1, #2, #3 in der Dateinamenskonvention für die Medienbereitstellung, zusammen mit # als Verbotenes Zeichen, macht den Report nützlicher und Duplikate leichter zu finden.

Unsinnige aber eindeutige Inventarnummern

Der Report kennt nicht die lokalen Inventarnummer-Konventionen im Museum und überprüft sie auch nicht.

Er findet also in der Regel keine unsinnigen Inventarnummern, bzw. die Verwendung des Inventarnummern-Feldes für Werte, die keine Inventarnummern sind. Z.B. würde "Schrank 4" (vermutlich der Standort, nicht die Inventarnummer) einem Menschen sofort auffallen, wenn alle anderen Inventarnummern der Form "X 1234 / 42" folgen. Da "schrank-4" aber eindeutig ist, wird nichts gemeldet.