

# Dokumentation: «Dummie Letters»

## Problemstellung/Voraussetzungen

Die BEBB umfasst insgesamt 5'423 Briefe, für die eine Systemnummer existiert. Von denen 1'684 im «unproduktiven Wiki» und 1'529 im «produktiven Wiki» bereits als XML-Dateien existieren. Damit können sie in BEOL importiert werden, was sie auffindbar und referenzierbar macht.

Die verbleibenden 2'210 Briefe haben zwar eine Systemnummer und sind im HAN-Katalog verzeichnet, da für sie aber keine XML-Dateien mit Metainformationen existieren, können sie nicht in BEOL importiert werden.

Meine Aufgabe war es darum, ein Python-Programm zu schreiben, das für die besagten Briefe leere XML-Dateien erstellt. Diese sollten strukturell den XML-Dateien für bestehende Briefe entsprechen, um eine serielle Bearbeitung aller Briefe zu ermöglichen und sie mit der bestehenden Infrastruktur in BEOL importieren zu können.

Die Dateien sollen leere Tags enthalten, wo eigentlich die Transkription des Briefes steht (welche es aber noch nicht gibt). Alle Metadaten, die sich im HAN-Katalog zu den Briefen finden, müssen aber in den Dateien enthalten sein.

## Umsetzung

Die Problemstellung erfordert ein Programm, das folgende Arbeitsschritte durchführt:

- 1) Einlesen einer Liste aller Systemnummern, die für die BEBB relevant sind.
  - ⇒ Diese Liste existierte bereits, sie musste lediglich in ein computerlesbares Format gebracht werden.
- 2) Erstellen einer Liste aller Systemnummern, für welche bereits eine XML-Datei besteht.
  - ⇒ Hierfür hat Fabienne Schwizer mir alle bestehenden XML-Dateien des Projekts zur Verfügung gestellt.
  - ⇒ Das Programm iteriert über alle XML-Dateien und liest deren Systemnummern. Anschliessend werden alle Systemnummern als Liste abgespeichert.
- 3) Erstellen einer Liste aller Systemnummern für die eine XML-Datei angelegt werden muss.

- ⇒ Diese Liste kann erzeugt werden, indem über die Liste mit allen Systemnummern iteriert wird, und für jede geprüft wird, ob sie in den bereits existierenden Systemnummern vorkommt; ist dies nicht der Fall, so ist für die Systemnummer eine XML-Datei anzulegen. Diese Liste wird anschliessend gespeichert.
- 4) Abfragen aller relevanten Metadaten vom HAN-Katalog.
  - ⇒ Für jede zu bearbeitende Systemnummer wird eine Online-Anfrage an den HAN-Katalog gemacht. Die Antwort wird als Binärdatei abgespeichert. Daraus können die bestehenden Metadaten ausgelesen werden.
  - ⇒ Programmiererisch ist dieser Schritt relativ einfach, aufgrund der bestehenden Infrastruktur gestaltet er sich aber eher mühsam: Von HAN können Daten nur über das Protokoll Z39.50 abgefragt werden; dieses kann mit Python nur über ein altes, längst nicht mehr unterstütztes Modul angesprochen werden, was einen zwingt, das komplette Programm in einer veralteten Version von Python zu schreiben und laufen zu lassen. Deshalb muss, damit das Programm verwendet werden kann, erst auch ein kompatibles Python-Environment mit allen verwendeten Modulen erstellt werden; die präsentierte Lösung kann also als nicht als vollkommen nachhaltig angesehen werden, unter den vorliegenden Gegebenheiten ist eine nachhaltigere Lösung aber wohl nicht mit angemessenem Aufwand zu bewerkstelligen.
- 5) Erstellung der tatsächlichen XML-Dateien.
  - ⇒ Für jede bearbeitete Systemnummer wird eine XML-Datei erstellt:
    - ➔ Der Name der Datei muss den Konventionen der BEBB entsprechen.
    - ➔ Die Struktur der Datei muss der Struktur der bestehenden XML-Dateien entsprechen.
    - ➔ Die relevanten Metadaten müssen aus der Binärdatei ausgelesen und korrekt ins XML geschrieben werden.
  - ⇒ Die Dateien werden in einem Ordner gespeichert und können zum Import in BEOL verwendet werden.

## **Aktueller Stand**

Die Schritte 1) – 5) konnten erfolgreich umgesetzt werden.

Schritt 6) ist derzeit in Bearbeitung.

Für technische Fragen wurde Tobias Schweizer konsultiert; nach Fertigstellung wird nochmals mit ihm anzuschauen sein, ob die Umsetzung technisch angemessen und die Dokumentation ausreichend ist.

Für inhaltliche Fragen wurde Fabienne Schwizer konsultiert. Nach Fertigstellung muss mit Sulamith Gehr angeschaut werden, ob das Resultat ihren Vorstellungen entspricht oder ggf. Anpassungen vorgenommen werden müssen.

Arbeitsaufwand bisher: ca. 20-25h