

Integration von Python in L^AT_EX am Beispiel von Katalogeinträgen

Uwe Ziegenhagen, Lukas C. Bossert

Viele Dissertationen in der Archäologie enthalten am Ende der Arbeit einen Katalog, in dem die untersuchten Daten in einem bestimmten System aufgeschlüsselt präsentiert werden. Ein solcher Katalog kann aus Bildern, Bohrproben, Architekturen etc. bestehen.

Eine händische Erstellung der einzelnen Einträge bspw. über `\section` oder `\subsection` und anschließend in einer `items`-Umgebung ist nicht effizient, fehleranfällig und nur bei wenigen Katalogeinträgen einsetzbar. Es muss zudem berücksichtigt werden, dass nicht auf alle Katalogeinträge alle vorgegebene Kategorien passen, sodass Kategorien leer bleiben und dann im Katalogeintrag nicht auftauchen sollen. Dabei soll der Code möglichst viel redundante Tipp-Arbeit abnehmen, wie sie bspw. bei Maßeinheiten vorkommt. Darüberhinaus sollen alle Einträge immer gleich formatiert sein und ihr Aussehen global verändert werden können.

Nachdem eine Lösung gefunden wurde, die allen bisher genannten Anforderungen entspricht (siehe unten), sollte in einer Kategorie alle Seitenzahlen enthalten, auf denen der Katalogeintrag im Haupttext erwähnt wurde. Ein Lösungsansatz sah die Nutzung von `glossaries` vor: Dafür musste allerdings für jeden Katalogeintrag ein eigenes Glossar angelegt werden, was nicht nur viel händische Arbeit bedeutet, sondern auch die Zählerkapazitäten von T_EX überforderten.

Es musste also eine andere Lösung her und ich (Lukas) erinnerte mich an einen Beitrag, den Uwe Ziegenhagen in dtk 1/2015 vorgestellt hatte: Damals ging es darum, bei einem Werkkatalog die Erwähnung des Stückes im Haupttext anzugeben. Prima, genau das, was ich gesucht habe und ich fand den damals verwendeten Code auf seiner Webseite.¹ Allerdings sah ich nun, dass die Aufgabe damals darin bestand, nur *eine* Erwähnung im dem Haupttext anzugeben, was mit `label` und `ref` bewerkstelligt werden konnte. Ich brauche jedoch alle Erwähnungen im Haupttext. Meine Emailanfrage bei Uwe nach einer Möglichkeit mehr als nur eine Erwähnung anzeigen zu lassen, beantwortete er mit dem Vorschlag mittels Python das Problem zu lösen.

Um gemeinsam an dem hybriden Konstrukt von L^AT_EX und Python wurde ein Repository auf der Plattform `github`² erstellt, die unabhängig der Programmiersprache einen exzellenten Austausch und eine detaillierte Versionskontrolle bereithält, sodass sich kollaboriertes Arbeiten sehr gut konzipieren lässt.

¹<http://uweziegenhagen.de/?p=3020>

²<https://github.com/LukasCBossert/DTK-TeX-Python>

Katalogeintrag

Die gewählte Lösung für die Katalogeinträge arbeitet mit dem `keyval`.³ Dafür werden in der Präambel verschiedene keys definiert. In der Grundversion sieht ein Eintrag wie folgt aus:⁴

```
\define@key{family}{key}{#1}
```

Für das konkrete Beispiel wird die key-Familie (family) mit `catalogue` angegeben, der Schlüssel (Kategorie im Katalog) als `house` bezeichnet und als Resultat soll der Wert im Makro `\KVhouse` gespeichert werden:

```
\define@key{catalogue}{house}{\def\KVhouse{#1}}
```

Entsprechend dieses Beispiels können alle Kategorien als Schlüssel angelegt werden:

```
1 \makeatletter
2 \define@key{catalogue}{house}{\def\KVhouse{#1}}
3 \define@key{catalogue}{label}{\def\KVlabel{#1}}
4 \define@key{catalogue}{description}{\def\KVdescription{#1}}
5 \define@key{catalogue}{location}{\def\KVlocation{#1}}
6 \define@key{catalogue}{size}{\def\KVsize{#1}}
7 \define@key{catalogue}{interior}{\def\KVinterior{#1}}
8 \define@key{catalogue}{interiorM}{\def\KVinteriorM{#1}}
9 \define@key{catalogue}{interiorW}{\def\KVinteriorW{#1}}
10 \define@key{catalogue}{interiorS}{\def\KVinteriorS{#1}}
11 \define@key{catalogue}{literature}{\def\KVLiterature{#1}}
12 \makeatother
```

Das Aussehen eines Katalogeintrages wird separat in einem `\newcommand` definiert.

```
1 \newcommand\catalogueentry[1]{%
2 \RaggedRight\begin{group}
3 \setkeys{catalogue}{#1}
```

In der Definition des Befehls `\catalogueentry` wird festgelegt, dass die Einträge nicht nur im Flattersatz ausgegeben, sondern auch, dass hierfür die Einträge aus der key-Familie `catalogue` ausgelesen werden. Es folgt in der Definition die Verarbeitung der einzelnen keys:

```
1 \ifdef{\KVhouse}{\section{\KVhouse
2 \ifdef{\KVlabel}{\label{\KVlabel}}{}}{}}
3 \begin{labeling}{Beschreibung}
4 \ifdef{\KVdescription}{\item[Beschreibung] \KVdescription}{}}
```

³ Basierend auf der Idee vorgestellt auf: <http://tex.stackexchange.com/a/254336/98739>

⁴ Vgl. <http://www.tug.org/tugboat/tb30-1/tb94wright-keyval.pdf>

```

5 \ifdef{\KVlocation}{\item[Verortung] \KVlocation}{}
6 \ifdef{\KVinterior}{%
7   \item[Ausstattung] \KVinterior
8   \begin{labeling}{Wandgemälde}
9     \ifdef{\KVinteriorM}{\item[Mosaike] \KVinteriorM}{}
10    \ifdef{\KVinteriorW}{\item[Wandgemälde] \KVinteriorW}{}
11    \ifdef{\KVinteriorS}{\item[Statuen] \KVinteriorS}{}
12    \end{labeling}
13  }{}
14 \ifdef{\KVsize}{\item[Größe] \SI{\KVsize}{\meter\squared}}{}
15 \ifdef{\KVliterature}{\item[Erwähnungen] \KVliterature}{}
16 \end{labeling}
17 \endgroup
18 }

```

Mit dieser Einstellung lassen sich die Katalogeinträge schon sehr gut darstellen. So wird aus den folgenden Angaben

```

\catalogueentry{%
  house={Haus des M. Fabius Rufus},
  label={haus:M-Fabius-Rufus},
  size={172},
  description={Haus besteht aus mehreren Einzelgebäuden.},
  location={Regio VII, Insula 16, Eingang 17--22.},
  interior={Reicher Fundkomplex.},
  interiorM={S/W-Mosaik},
  interiorW={Dionysius mit einer Mänade, Narzissus und ein Cupido, Hercules und
↵Deinira, etc.},
  interiorS={Bronzene Statue eines Epheben},
}

```

ein Katalogeintrag, der so aussieht:

Haus des M. Fabius Rufus

Beschreibung Haus besteht aus mehreren Einzelgebäuden.

Verortung Regio VII, Insula 16, Eingang 17–22.

Ausstattung Reicher Fundkomplex.

Mosaike S/W-Mosaik

Wandgemälde Dionysius mit einer Mänade, Narzissus und ein Cupido, Hercules und Deinira, etc.

Statuen Bronzene Statue eines Epheben

Größe 172 m²

Wie man sieht, ist die Reihenfolge in der die keys angegeben werden irrelevant. Der Wert bei size wird intern sogleich an das vordefinierte \SI-Makro mit entsprechender Einheit (m²) übergeben. Ähnlich kann auch mit anderen Angaben verfahren werden (bspw. mehrere \label an \cref, etc.).

Bei diesem Eintrag sind alle Kategorien ausgefüllt. Wenn eine Kategorie nicht ausgefüllt ist, wird sie nicht ausgegeben. Allerdings besteht die Kategorie interior [Ausstattung] zusätzlich aus drei Unterkategorien (interiorM [Mosaike], interiorW [Wandgemälde], interiorS [Statuen]). Diese sollen angezeigt werden auch wenn interior selbst nicht definiert ist.

```
\catalogueentry{%
  house={Haus des Wilden Ebers},
  label={haus:Wilden-Ebers},
  size={54},
  description={Renovierung nach Erdbeben 62\,n.\,Chr.},
  location={Regio VII, Insula 4, Eingang 48, 43},
  interiorM={S/W-Mosaik},
  interiorW={Venus, Leda und der Schwan, Ariadne und Theseus},
}
```

So wird daraus:

Haus des Wilden Ebers

Beschreibung Renovierung nach Erdbeben 62 n. Chr.

Verortung Regio VII, Insula 4, Eingang 48, 43

Ausstattung Mosaike S/W-Mosaik

Wandgemälde Venus, Leda und der Schwan, Ariadne und Theseus

Größe 54 m²

Um zu diesem Ergebnis zu kommen, ist es notwendig mit Booleschen Operatoren zu arbeiten. Dafür müssen bei den key-Definitionen drei Operatoren eingeführt werden:

```
1 \newbool{@KVinteriorM}%Mosaik
2 \newbool{@KVinteriorW}%Wandgemälde
3 \newbool{@KVinteriorS}%Statue
4 ...
5 \define@key{catalogue}{interiorM}{\def\KVinteriorM{#1}\booltrue{@KVinteriorM}}
6 \define@key{catalogue}{interiorW}{\def\KVinteriorW{#1}\booltrue{@KVinteriorW}}
```

```
7 \define@key{catalogue}{interiorS}{\def\KVinteriorS{#1}\booltrue{@KVinteriorS}}
```

Diese können dann bei der Ausgabe der Katalogeinträge eingesetzt werden.

```
1 \ifdef{\KVinterior}{%
2   \item[Ausstattung] \KVinterior
3   \ifboolexpr{bool{@KVinteriorM} or bool {@KVinteriorW} or bool {@KVinteriorS}
4     \begin{labeling}{Wandgemälde}
5       \ifdef{\KVinteriorM}{\item[Mosaike] \KVinteriorM}{}
6       \ifdef{\KVinteriorW}{\item[Wandgemälde] \KVinteriorW}{}
7       \ifdef{\KVinteriorS}{\item[Statuen] \KVinteriorS}{}
8       \end{labeling}
9     }{}}
10  {\ifboolexpr{bool{@KVinteriorM} or bool {@KVinteriorW} or bool {@KVinteriorS}
11    \item[Ausstattung]%
12    \begin{labeling}{Wandgemälde}
13      \ifdef{\KVinteriorM}{\item[Mosaike] \KVinteriorM}{}
14      \ifdef{\KVinteriorW}{\item[Wandgemälde] \KVinteriorW}{}
15      \ifdef{\KVinteriorS}{\item[Statuen] \KVinteriorS}{}
16      \end{labeling}
17    }{}}
```

Dadurch werden die Unterkategorien von interior ausgegeben, auch wenn interior selbst nicht definiert ist.

Was fehlt ist die Ausgabe der Seitenzahlen, wofür der literature bereit gestellt wird.

python

Integration