More than just TEI…

The use of text technologies within the SSRQ

Beni Ruef <bernhard.ruef@ssrq-sds-fds.ch>

Law Sources Foundation of the Swiss Lawyers Society (SSRQ)

Seminar «Language Technology for Legal Documents»
University of Zürich, 22.10.2020

Outline

Who/What is the Swiss Law Sources Foundation?

research institution, founded in 1898 by the Swiss Lawyers Society

- publishes critical editions of Swiss law sources from the early Middle Ages until 1798 in its “Collection of Swiss Law Sources” (aka SSRQ: **S**ammlung **S**chweizerischer **R**echts**q**uellen)
- law sources: sources of law like town charters, constitutions, rights of use, catalogues of goods, court decisions etc. (“dusty legal documents” ;-)
- 120 (logical) volumes with a total of 84’000 pages completed
- 17 ongoing projects (i.e. volumes), whereof 13 TEI-based; projects can take 10 years…

Characteristics of the Collection

- different text types, diachronic, and doubly multilingual, both source texts (Latin, French, Italian, German and Romansh) and editorial content (German, French and Italian)
  in other words: a very heterogeneous corpus
- mirrors the technical development within the last 120 years:
  - until ca. 1995: lead type and phototypesetting
  - from ca. 1995 until 2010: *FrameMaker*
  - from 2010 until 2018: *InDesign*
  - since 2011: LaTeX (desperate because of *InDesign* and deciding to do everything ourselves…)
  - TEI since 2010 (retrodigitalisation) / 2012 (digitally born)

Anatomy of an SSRQ Volume

- catalogue of source texts
- introduction
- lists (list of archives, bibliography, list of abbreviations etc.)
- source texts
  - title
  - date
  - introductory remarks (e.g. setting)
  - transcripton
  - manuscript description (archive, substrate, format etc.)
  - annotations
  - further remarks (e.g. related texts or aftermath)
- indices
  - persons, families and organisations
  - places
  - lemmas
  - keywords

Retrodigitalisation (SSRQ online)

- Phase 1 (2008–2011)
  - all pages scanned (G4 compressed TIFF, i.e. bitonal, 600 dpi) and pimped up (wiping margins and deskewing)
  - catalogues of source texts OCRed and manually corrected, enabling search in titles
  - development of a viewer
- Complete OCR (2017)
  - analogue volumes (1898–1996) OCRed
    - using ABBYY FineReader 14
    - language and period specific training (typography changed over time)
    - correct recognition of ſ (LATIN SMALL LETTER LONG S) and characters like ů
    - result provided as PDF
  - post-1995 (“digital”) volumes also provided as PDFs

Automatic Generation of Indices

- Problem
  - our indices contain information about their entries, not just page numbers
  - they are traditionally accurate to the line
  - they were created manually (sic!) even with *FrameMaker* and *InDesign*…
- Solution
  - LaTeX macros (`\persname` etc.) tag index entries and create lists of occurrences
  - information on entries is stored in a database
  - a Perl script creates the LaTeX code for the index from the occurrences and the database

What's so Special about SSRQ's TEI Portal?

- entirely based on the TEI Processing Model (and the TEI Publisher Libraries, respectively)
- complex critical apparatus with highly nested TEI elements, e.g. `<app><lem/><rdg><unclear/></rdg></app>`
- rich semantic annotations (`<date>`, `<measure>`, `<persName>`, `<placeName>` etc.)
- complete schema and validation down to attribute values documented in the ODD
- multilingual: labels and attribute values translated to German, French and English

Why/How to Produce a Print Version in the TEI Age

- yes, a print version is still needed!
  - as an immutable publication suitable for reference
  - as long-time storage *(don't laugh)*
- requirements
  - well-defined, clear structure and clean layout
  - supporting typographic complexity (indices, hyphenation, line numbers etc.) and quality (ligatures, kerning etc.)

Choices

XSL-FO

- falls short on formatting a complex critical apparatus
- typesetting quality does not live up to high demands unless using a commercial engine

LaTeX

- has a long tradition of formatting scholarly editions, providing packages like *reledmac*
- LaTeX code can preserve most of the TEI semantics: readable for humans, easier to debug
- SSRQ has some LaTeX expertise :-)

How to Convert XML/TEI to LaTeX

- traditionally done with XSLT (entailing a lot of redundancy)
- advantages of using the TEI ODD and Processing Model
  - a single ODD describes transformations to web (HTML) **and** print (LaTeX)
  - many TEI elements use the same `<model>` for both web and print, resulting in much less code
  - very fast implementation (directly translated into XQuery functions)

Handling Graph Data

- today, index entries (cf. above) are stored in three different databases using two different technologies: XML (*eXist-db*) and RDF (*Apache Jena Fuseki*)
- however, these index entries ($\equiv$ entities) are highly interconnected (related)
- in other words: they can be represented as a *graph*
- how do you store graph data?
- our goal: combining the elegance of a graph database with the stability of an RDBMS
- our approach
  - Entity-attribute-value model with classes and relationships (EAV/CR) implemented on top of *PostgreSQL*: simple, stable relational schema independent of ontology changes
  - searching with (our own implementation of) Gremlin
  - used in two projects

The Importance of Metadata and How to Handle It

- metadata are absolutely crucial!
- there is no access to DH data without metadata; full-text search is never a substitute
- many types and levels of metadata
  - about the source (cf. above)
  - about the transcription
  - index data (cf. above)
  - metadata about index data (e.g. when was person X born?), i.e. metadata about metadata
  - metadata about metadata about index data (reference for birth date of person X), i.e. metadata^3 :-)
- where to store metadata?
- how to make them available?

Technologies Used

(rather incomplete list…)

- HTML5
- Perl
- LaTeX (actually XeLaTeX and a bit of TeX, too)
- XML, XSLT, XQuery, Schematron
- Python, RDF, SPARQL
- TEI ODD, TEI Processing Model
- EAV/CR, Gremlin, SQL
- and last but not least: everything is held together by the UNIX shell!

Links

- Law Sources Foundation of the Swiss Lawyers Society
- SSRQ online (retrodigitalisation)
- SSRQ Index Databases
- SSRQ's TEI Portal
- Getting Started with TEI ODDs
- The TEI Processing Model in the Guidelines
- TEI Publisher
- Creating High-Quality Print from TEI Documents
- Entity-Attribute-Value Model
- Gremlin (Apache TinkerPop)
- histHub