Building a Collaborative Editorial Workbench for Legal Texts with Complex Structures

Nils Geißler and Daniela Schulz

Building a Collaborative Editorial Workbench for Legal Texts with Complex Structures

Nils Geißler and Daniela Schulz

ABSTRACT

This paper presents the work undertaken by the *Capitularia* project to integrate a collaborative editorial workbench into the open-source content management system (CMS) WordPress. It introduces the reasons for selecting WordPress as the project's CMS, the workflows established (including a sophisticated XSL-scripting pipeline), as well as three plug-ins created to integrate certain functionalities. The Cap-X2WP plug-in facilitates XSL transformations of XML files to HTML directly within the WordPress framework. The Cap-PaGer plug-in is used to generate WordPress pages automatically based on the XML files located in specific folders on the server. Their publication status can be administered via a special interface added to the general WordPress dashboard at a moment's notice. Whereas the aforementioned plug-ins facilitate the daily work of the staff members in the general management and enhancement of the project's website, the Cap-

Coll plug-in eases the specific editorial task of collating texts by including the CollateX algorithms in a WordPress plug-in. The report concludes with a brief perspective on the possibilities for further developments.

INDEX

ACKNOWLEDGEMENTS

1. Introduction

1 For scholarly projects with long-term funding, in addition to achieving the project's actual objectives, there is often also an expectation that the project will create a general added value from which other projects or even the entire research community can benefit. This might, for example, be achieved by documenting the experience gained, in order to contribute to the establishment of best practices. Especially in digital humanities projects, a more concrete contribution lies in the development and provision of helpful software applications that allow for reuse.

2 Within the context of the project *Capitularia: Edition of the Frankish Capitularies,*[1] three plug-ins[2] to extend the basic functionalities of the WordPress core have been developed so far. These could be of general interest for similar editing projects. After a quick description of the decision-making processes that led to the utilization of the open-source content management system (CMS) WordPress (section 2), followed by a brief presentation of the project itself (section 3), these plug-ins and the functionalities they provide are presented in more detail (section 4).

2. Why WordPress?

3   In the last few years, a lot of effort has been put into developing specialized infrastructures (such as GAMS,[3] FuD,[4] or TextGrid[5]) to ease the administration and publication of digital humanities data, and thus to avoid insular solutions.[6] Still, the choice of infrastructure is not an easy one, since certain difficulties exist:

- a large number of scholars working with TEI lack the (access to) technical expertise (Burghart and Rehbein 2012) and/or the financial support needed for the proper use of at least some of those;
- for many, especially small-scale ventures, infrastructural projects might even be a little oversized, and/or the familiarization might take too long;
- infrastructures depend on regular funding to ensure maintenance and further improvements, and thus ensure their longstanding viability. Expired funding might block necessary adjustments to the changing technologies of the future, calling into question the sustainability of the software.[7]

4   Especially in a long-term scholarly editorial project such as *Capitularia*, which is supposed to run for sixteen years (2014 to approx. 2029), the choice of the basic technological framework is extremely important, since the resource is supposed to be available for the entire duration of the project and beyond, and certain requirements need to be met as well. Besides more specific demands, one of these requirements was that the website should be up and running immediately after the project started in the spring of 2014. The research community should be allowed to monitor the project's development and start working on the material provided as soon as possible. These requirements ruled out from the beginning a time-consuming in-house development, but suggested the adoption of a different approach that had previously been used in some digital humanities projects: the application and customization of an existing web framework. Digital Scholarly Edition (DSE) ventures such as the *Saint Patrick's Confessio Hypertext Stack Project*[8] have been built using the CMS Drupal.[9] For the same framework, specific extensions like "TEICHI" (Pape et al. 2012) have been developed to enable working with TEI files.

5   Content management systems already contain a lot of functionalities that serve useful purposes in digital humanities projects. Furthermore, the application of already established software, which is not specifically intended for a particular use and has a wide-ranging community participating in its constant development, could also be more sustainable (Stürmer 2015). WordPress was selected as the CMS for *Capitularia* for the following reasons:

- WordPress provides a framework which is easy to use and maintain even for people with limited technical expertise, but also offers a vast number of possible extensions for those having (access to) programming skills. Compared to Drupal, the learning curve is less steep.

- One of the huge advantages of WordPress is that—owing to its widespread use—a large community participates in its further development and documentation. Therefore, more or less ready-made solutions already exist for numerous problems; when they do not, one can easily develop one's own plug-ins with the help of the good documentation.

- WordPress is PHP-based and uses a MySQL database—both standard technologies.

- WordPress can be regarded as sustainable open source software (Stürmer 2015).

- WordPress allows for a multilingual interface.

- WordPress had already been used in the *Bibliotheca legum* project,[10] a database of Carolingian secular law texts, and therefore one could build upon experiences. In contrast with *Capitularia*, the *Bibliotheca legum* relies solely on the use of already existing plug-ins without any specific in-house developments.

3. About the Project

3.1 Scope

6   The *Capitularia* project is concerned with the hybrid edition of decrees by Frankish rulers. These legal texts are an important source for various aspects of early medieval European history. Capitularies originated as individual texts from deliberations and assemblies at court, but hardly any original has survived. Mostly they were transmitted in sundry collections compiled by attendants of these assemblies, or based on copies sent to bishops or other office holders, which created a vast variety of different versions of the texts.[11] What most capitularies have in common is that they appear as a list of chapters, with different capitularies often amalgamated with one

another. This outward appearance also explains why they are commonly called "capitularies." Most often capitularies mention neither date, not place, nor the "issuer." Some appear to have been official documents; others might have been private notes, drafts, or extracts. They were rearranged, modified, or extracted by the compilers, sometimes with individual titles, vague titles, or no titles at all. This wide spectrum makes it hard to judge the status of a particular text. The texts also differ significantly in length and number of extant witnesses, ranging from unique up to more than thirty. All in all, there are about three hundred texts in more than three hundred extant manuscripts. The characteristics of the source material raise particular issues that affect the TEI encoding (for example, the identification of certain passages, overlaps, and contaminations), and consequently also the work of the editors. The modeling of an overarching structure to depict and reference the single textual units in their various manifestations within the manuscripts posed one of the biggest challenges. And it still does.

3.2 Objectives

7   A traditional critical print edition will be published in the Leges series of the Monumenta Germaniae Historica (MGH)[12] containing the reconstructed capitularies with full commentary and a German translation, but with a simplified critical apparatus that includes only the variants considered relevant by the editors. The digital edition (figure 1) is meant as its complement with a focus on the transmission and tradition of the texts, enhanced by further resources such as a comprehensive, annotated bibliography and various overviews. Here, each text considered as capitulary is presented as it appears in the respective manuscript source, with annotations. In addition to the TEI-compliant transcriptions, thorough manuscript descriptions are taken from the seminal study *Bibliotheca capitularium regum Francorum manuscripta* by Hubert Mordek (1995). They are enhanced by meticulous observations carried out by the project team. Hence, interested scholars have access to all relevant information on each codex and the text(s) it contains, but the print edition is not overburdened by an oversized critical apparatus.

Figure 1. The *Capitularia* web page (http://capitularia.uni-koeln.de/en).

3.3 Logistics

8   *Capitularia* is funded by the North Rhine–Westphalian Academy of Sciences, Humanities and the Arts, and is being prepared in close collaboration with the Cologne Center for eHumanities (CCeH),[13] the MGH, and other partners.[14] The digital edition is overseen by a team based in Cologne. The print edition is being prepared jointly by a group of editors. Since the collaborators are scattered, a central platform for internal communication as well as for the distribution of resources among staff is essential to facilitate successful cooperation. Hence, WordPress not only provides the web publication and thus the outward presentation of the project to the public, but also serves as a means of exchange and a collaborative editorial workbench.

9   There is an internal workspace in WordPress for the project staff and the editorial team. It allows the participants to access data, resources, manuals, and tools. Recently, GitHub has been introduced for the overall administration of the project's technical developments.[15] To ensure the long-term availability of the resource, the *Capitularia* project relies on a combination of suitable technical infrastructure and strong institutional ties. The server space is provided by the University of Cologne's Regional Computing Centre (RRZK), with which both the CCeH and

the Cologne Data Center for the Humanities (DCH),[16] an institute specifically dedicated to the sustainability of humanities data, maintain close contact. In addition, *Capitularia* also participates in the web archiving program of the Bavarian State Library (BSB).[17]

3.4 Workflows

10   The workflow for creating *Capitularia* web content is as follows (figure 2): as has been mentioned before, the main source for the manuscript pages as well as most index pages (such as lists of manuscripts or capitularies) is Mordek's *Bibliotheca capitularium* (1995). He provided descriptions of all witnesses bearing capitularies, but died in 2006 before he could provide a new edition of the material. His book was digitized by means of optical character recognition (OCR) and marked up with XML.[18] Further markup was then added to this corpus file to enable the automated creation of TEI-compliant manuscript descriptions that are stored in `<msDesc>` elements inside the `<teiHeader>`.

11   The diplomatic transcriptions of the individual capitularies are mostly based on digital facsimiles. Whenever possible, the originals are also consulted. Preceded by an editorial preface by the *Capitularia* staff members, the encoded transcriptions form the `<body>` of the file. The TEI-compliant encoding is carried out in the oXygen XML Editor,[19] which is connected to the server by means of Web-based Distributed Authoring and Versioning (WebDAV).[20] This ensures that all employees have access to the latest versions at all times. GitHub is used for managing the files. In addition to that, older versions are manually saved in a special archive folder.

12   The transcriptions are checked by a very strict schema– a project specific customization in RELAX NG – supplemented by Schematron,[21] and checked manually by the staff members. One person is responsible for transcribing and encoding a particular capitulary of a manuscript. The transcription is then reviewed twice by other staff members before the original transcriber incorporates their annotations or corrections to finalize the transcription. Before publishing the manuscript page on the web, the HTML version of the file is proofread once more.

Figure 2. The *Capitularia* workflow for the creation of the digital edition.
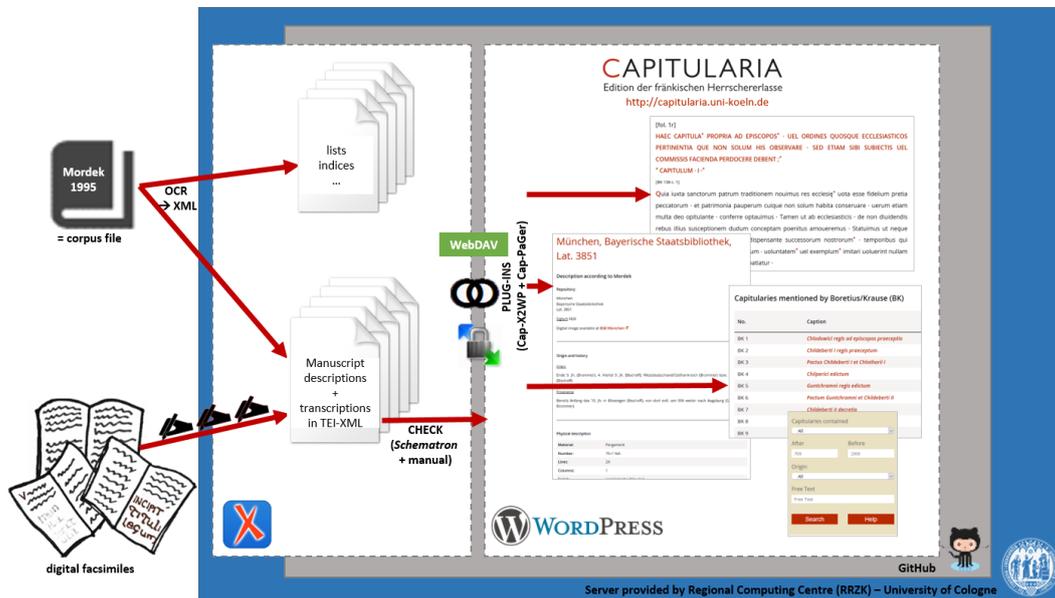
13 The transcriptions form the basis of the print edition which is being prepared by the team of editors using the Classical Text Editor (CTE),[22] a program the editors were familiar with. Their work is facilitated by computer-aided collation (see section 4.3). The TEI output provided by the CTE is then also uploaded onto the *Capitularia* website to publish preprint versions of the reconstructed capitularies (figure 3).

Figure 3. Workflow for the critical print edition with the PDF output as generated by the Classical Text Editor (CTE), as well as the reprocessed CTE-TEI output of the "Prooemium generale" issued by Emperor Louis the Pious (Capit. LdF no. 5, olim Boretius-Krause no. 137) including the translation into German.
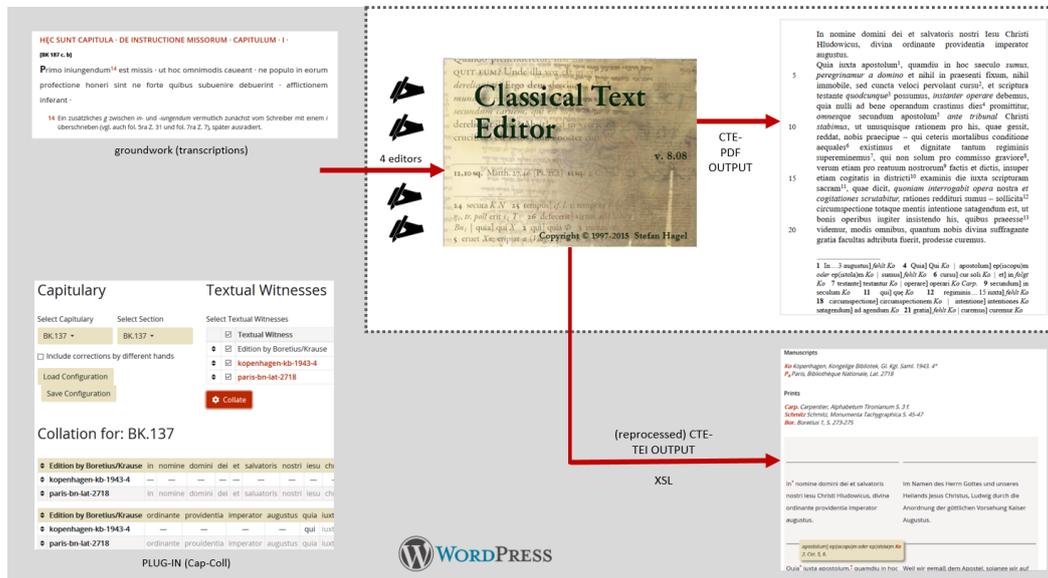
4. Tool Development

14   WordPress distinguishes between different user roles which are given different authorization levels. This distinction allows WordPress to display certain content and areas of the website only to staff members or users who are logged in. The front end is usually visible to all users. The dashboard, which is part of the WordPress back end, is the main administration area where most of the site's settings are managed. It is therefore only accessible to users with appropriate administrative privileges. The plug-ins are also adjusted and administered here.

15   Three main plug-ins have been developed and implemented within the context of the *Capitularia* project so far: (1) an XSL Processor plug-in which enables transforming the TEI XML files with XSL into HTML (Cap-X2WP) to be outputted into a WordPress page; (2) a page generator plug-in (Cap-PaGer) that allows one to generate numerous pages in bulk; and (3) the collation plug-in (Cap-Coll). Collation, the process of comparing the texts of various witnesses to investigate their textual variance, is one of the main tasks in the field of textual criticism.

4.1 Cap-X2WP

16 The Cap-X2WP plug-in facilitates XSL transformations of XML files within the WordPress framework on the server side. It is based on the XSLT Processor plug-in[23] that had been deployed in the *Bibliotheca legum* project. Both use shortcode, which is a WordPress-specific way to include shortcut commands into the content area (the actual body of a page or post).[24] Shortcode is written in square brackets. In its simplest form it consists only of the name of the shortcode (here `cap_xsl`), but additional information can be provided by using attributes. In the case of *Capitularia*, these attributes contain the path to an XML file and an XSL file in order to display the result of the transformation on a WordPress page (figure 4).

Figure 4. At left, the shortcode in square brackets starting with its name, `cap_xsl`, is followed by `<div class="xsl-output">`, which contains the transformation's output. At right, the result as displayed in a user's browser.
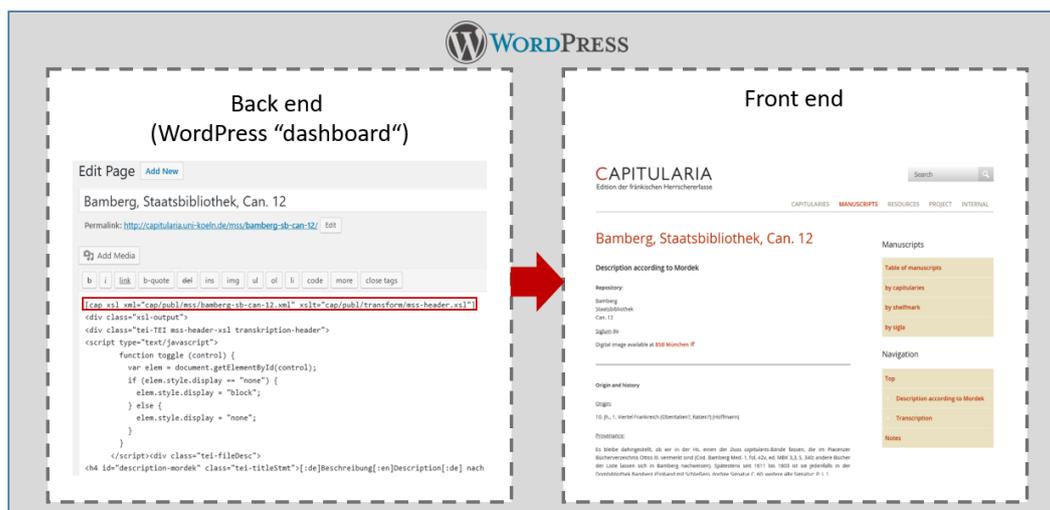
17 Redevelopment became necessary because the XSLT Processor plug-in is no longer maintained and performance issues have arisen as the complexity of the files increased. To prevent the transformation process from starting again on every page request, which would result in longer loading times, Cap-X2WP checks whether the underlying files have been edited in the meantime. The core idea of this plug-in is to cache the result of a transformation in its WordPress page, and only retransform if either the XML or the XSL file has changed. Each transformation's result is stored in a `<div class="xsl-output">` element during this process, which is replaced when a new transformation is triggered. Writing the generated content simultaneously into the WordPress

page itself has the additional advantage that the regular full-text search already included in the WordPress core can also be used in this context. General settings can be configured easily via the options interface included in the WordPress dashboard (figure 5).

Figure 5. The interface for the configuration of the Cap-X2WP plug-in.

4.2 Cap-PaGer

18   The Cap-PaGer plug-in is used to generate WordPress pages automatically based on the XML files located in specific folders on the server. Their publication status can be administered via a special interface added to the general WordPress dashboard with a single click.

19   Cap-PaGer was developed to enable the automated generation of the numerous manuscript pages (one page for each codex that contains the manuscript description as well as the transcriptions of the texts contained) in WordPress, instead of creating them manually. The general structure of a WordPress page is quite simple: it consists of a title (to be displayed as its heading), a page slug, information on its location within the overall hierarchy of the website, and its content. Since in the case of *Capitularia* most of the pages are composed of the requests to the respective XML and XSL files in shortcode as mentioned above (figure 4), it seemed obvious that we should build those pages in a generic and reproducible way. Within the course of development, the plug-in was extended to enable building all WordPress pages that are based on XML files, and its administration became increasingly sophisticated. By now the Cap-PaGer is fully implemented for use in the back end of WordPress.

Figure 6. The options interface for the general settings (left), which allow the creation of tabs for the different categories such as manuscripts or capitularies that date before 814 CE, and the specific options for the category "manuscripts" (right).

20 Figure 6 shows how tabs for the different sections can be created and administered by the project staff. Here, general settings (such as the root directories in which the XML files are stored) can be entered. Further options and more detailed configurations for the individual sections are available on their respective tabs. In conjunction with the multilingual plug-in qTranslate-X,[25] different translations for different language settings are also possible. The directories mentioned here correspond to the actual directory structure on the server. This means that all files located within the mss (manuscripts) directory on the server will be displayed as a list within the manuscript section. A schema can also be adjoined, as well as one or more XSL files used for the transformation. For example, there are three different transformations associated with the section "manuscripts": first, a transformation to display the comprehensive manuscript description taken from Mordek (1995) as mentioned above; second, a transformation for the main transcription; and finally, a transformation for a footer that attaches some additional notes such as how to cite this particular page, a hyperlink to the XML source file available for download,[26] and the revision history. This modular approach was deliberately chosen to reduce the complexity of the single XSL files and thus to facilitate their maintenance. Despite the complex processing pipeline working in the background, the interface enables the project staff to connect the different parts and determine what will be displayed on the page in an easy and clear way.

21   The administration of the pages takes place within the page generator dashboard (figure 7). The staff members have a synoptic view of all pages belonging to a particular category as well as their publication states. They can easily select publishing, publishing privately, or unpublishing, as well as further options (such as the extraction of metadata), enhancing functionalities originally implemented in the WordPress core ("private" vs. "published"). These synopses enable collaboration among staff as different researchers can work on many manuscripts at the same time without getting lost in the process.

Figure 7. Publicly published manuscripts are indicated in green, privately published in yellow. Not shown here: Unpublished items are indicated in red.

4.3 Cap-Coll

22   In order to facilitate the editorial tasks involved with the numerous textual witnesses, collation is supported by alignment tables. This functionality is based on CollateX[27] with the algorithms included in the *Capitularia* Collation Tool. Before the actual collation takes place, the XML files containing the TEI-compliant transcriptions are preprocessed and normalized by some simple XSL

transformations to eliminate surplus information that would otherwise complicate the collation process. Each manuscript can be included in or excluded from the collation with a single click. Dragging and dropping changes its position within the default (alphabetical) order. Various settings can be chosen to customize the automated collation, such as the alignment algorithm applied (Dekker, Needleman-Wunsch, or MEDITE[28]), the Levenshtein distance[29] score, as well as other options. The configuration can then be saved to replicate the run (figure 8). Some of the options are only available to staff members, while for public display and usage, only those settings that have proven to lead to best results are available.

Figure 8. The *Capitularia* Collation (Cap-Coll) Dashboard showing the available manuscripts as well as the different settings.

In the collation output, corresponding text passages are displayed aligned with each other (figure 9). At the same time, deviations from a base text are highlighted, providing more clarity and transparency when comparing the textual witnesses. Cap-Coll allows the editors (and the users) to compare each chapter of a capitulary in any (sub)set of manuscripts either by using the old nineteenth-century edition of Boretius and Krause (1883/1897) as the basic text, or by selecting another version from any manuscript for the role of base text. Therefore, it allows for the testing

of hypotheses regarding the textual tradition, and makes similarities/differences more explicit. Based on the collation output, the editors investigate the filiation and reconstruct, annotate, and translate the single capitularies for the print edition.

Figure 9. Collation output for chapter 4 of the "Capitula legibus addenda" (Boretius-Krause no. 139).

Collation for: BK.139_4

	Edition by Boretius/Krause	qui	viduam	intra	primos	—	triginta	dies	viduitatis	suae	vel	invitam	—
⇅	berlin-sb-phill-1737	qui	uiduam	infra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	cava-dei-tirreni-bdb-4	qui	uidua	intra	primos	—	triginta	dierum	uiduitatis	suę	uel	inuita	sua
⇅	gotha-flb-memb-i-84	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	ivrea-bc-xxxiv	qui	uiduam	intra	primos	—	xxx	dies	uiduetatis	suę	uel	inuitam	—
⇅	kopenhagen-kb-1943-4	qui	uiduam	intra	primus	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	london-bl-egerton-2832	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	montpellier-biu-h-136	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	muenster-sa-msc-vii-5201	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	paris-bn-lat-2718	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suae	uel	inuitam	—
⇅	paris-bn-lat-4280a	qui	uiduam	intra	primos	—	xxx	dies	uiduitatis	suę	uel	inuitam	—
⇅	paris-bn-lat-4626	qui	uiduam	intra	primos	—	xxx	dies	uiduitati	suę	uel	inuitem	—
⇅	paris-bn-lat-4628	qui	uiduam	intra	primus	—	xxx	dies	uiduitatis	suae	uel	inuita	—
⇅	paris-bn-lat-4628a	qui	uiduam	intra	primos	—	triginta	dies	uiduitatis	suę	uel	inuitam	—
⇅	paris-bn-lat-4631	qvi	viduam	intra	primos	—	triginta	dies	viduitatis	sue	vel	invitam	—

5. Conclusion and Further Prospects

24   In the light of the experience gained so far, WordPress has proven an effective and easy-to-maintain framework for *Capitularia.* Its simple extensibility and adaptability are especially strong arguments for the adoption of WordPress in digital humanities projects working with TEI files.

25   One of the main problems of using existing tools was that, for the most part, they were developed to meet the needs of a specific project, and so adaptions to other material are difficult, time consuming and resource-intensive. Often infrastructure maintenance is limited by the project's funding. That is at least not the case for the Cap-X2WP and Cap-PaGer plug-ins, since their functionalities are so general that they are of use even beyond the domain of digital humanities, and thus universally applicable to WordPress websites. In its current state, the code is optimized for usage within the *Capitularia* project, but it can easily be adjusted to others' particular needs and is accessible via GitHub. By making the code available on public repositories, others can build upon previous work.

26   The Cap-Coll plug-in has a specific field of application, but still collation is an essential task in nearly all DSE ventures. The most recent development in the *Capitularia* project was, to make Cap-Coll available in the front end as well. Before, the collation tool could only be used by researchers with the appropriate WordPress account, accessing it in the back end. By modifying the plug-in, now all visitors to the website are allowed to make their own collations. The inclusion of further visualizations would be a desirable next step to enhance it.

27   Slightly connected to the further development of the collation plug-in is the long-term plan to improve the work routines, so that the preparation of the critical print edition becomes fully integrated into the *Capitularia* XML workflow, making it more efficient and seamless. The prerequisites for such an enhancement are: first, to provide a decent print output based on XML, which would be comparable to that provided by the CTE; and second, the willingness of all participants to familiarize themselves with new software applications.

BIBLIOGRAPHY

Burghart, Marjorie, and Malte Rehbein. 2012. "The Present and Future of the TEI Community for Manuscript Encoding." *Journal of the Text Encoding Initiative* 2. http://journals.openedition.org/jtei/372; doi:10.4000/jtei.372.

Mordek, Hubert. 1995. *Bibliotheca capitularium regum Francorum manuscripta: Überlieferung und Traditionszusammenhang der fränkischen Herrschererlasse.* Munich: Monumenta Germaniae Historica.

Pape, Sebastian, Christof Schöch, and Lutz Wegner. 2012. "TEICHI and the Tools Paradox: Developing a Publishing Framework for Digital Editions." *Journal of the Text Encoding Initiative* 2. http://journals.openedition.org/jtei/432; doi:10.4000/jtei.432.

Stürmer, Matthias. 2015. "Wann sind Open Source Projekte digital nachhaltig?" In *Open Source Studie: Schweiz 2015*, edited by swissICT and Swiss Open Systems User Group, 36–37. https://www.ossdirectory.com/knowhow/160_OSSStudieSchweiz2015.pdf.

NOTES

1   Accessed April 1, 2020, https://capitularia.uni-koeln.de/en/.

2   Plug-ins are modular packages of code written in PHP that introduce new functionalities to WordPress without changing the actual core. They make it easy to customize and enhance a website to meet one's specific needs. For a general introduction to WordPress plug-ins, see "Plugins," WordPress.org, accessed March 18, 2020, https://wordpress.org/support/article/plugins/.

3   Geisteswissenschaftliches Asset Management System (Humanities' Asset Management System), accessed March 18, 2020, http://gams.uni-graz.at.

4   Die virtuelle Forschungsumgebung für die Geistes- und Sozialwissenschaften, accessed March 18, 2020, https://fud.uni-trier.de.

5   TextGrid: Virtuelle Forschungsumgebung für die Geisteswissenschaften, accessed March 18, 2020, https://textgrid.de/.

6   TEI Publisher (accessed March 18, 2020, http://tei-publisher.com), a very promising attempt arising directly out of the TEI community itself, has only been published recently and hence could not be considered as an option when the *Capitularia* project started.

7   Stürmer (2015) establishes four different criteria for evaluating the digital sustainability of open source software products: (3a) heterogeneous communities participate in the development to ensure distributed knowledge; (3b) an ecosystem of commercial vendors also exists; (3c) nonprofit organizations coordinate the development as well as distribution; and (3d) it is possible for users to adapt the software to their own needs.

8   Royal Irish Academy 2009–2017, accessed March 18, 2020, http://www.confessio.ie.

9   Accessed March 18, 2020, https://www.drupal.org/.

10   Edited by Karl Ubl, assisted by Dominik Trump and Daniela Schulz (Cologne, 2012), accessed March 18, 2020, http://www.leges.uni-koeln.de. The *Bibliotheca legum* can be seen as complementary to *Capitularia* as it deals with the transmission of the so-called *leges* (such as the *Lex Salica*), which are another important source for early medieval legal knowledge. *Leges* and capitularies often appear together within the same codices.

11   For a concise overview of the different attempts to define capitularies, see "What Are Capitularies?," accessed March 18, 2020, https://capitularia.uni-koeln.de/en/project/definition/.

12   Accessed March 18, 2020, http://www.mgh.de.

13   Accessed March 18, 2020, http://cceh.uni-koeln.de.

14   Accessed April 1, 2020, https://capitularia.uni-koeln.de/en/project/.

15 CCeH GitHub repository, accessed March 18, 2020, https://github.com/cceh/capitularia.

16 Kölner Datenzentrum für die Geisteswissenschaften, accessed March 18, 2020, http://dch.phil-fak.uni-koeln.de/.

17 BABS (Bibliothekarisches Archivierungs- und Bereitstellungssystem): Long-term Preservation at the Bavarian State Library (Bayerische StaatsBibliothek), accessed March 18, 2020, https://www.babs-muenchen.de/index.html?c=&l=en.

18 The MGH kindly granted permission to provide and prepare the material.

19 Accessed March 18, 2020, https://www.oxygenxml.com.

20 Jim Whitehead, "Welcome to WebDAV Resources," last modified April 21, 2010, http://webdav.org/.

21 Accessed March 18, 2020, http://schematron.com/.

22 Stefan Hagel, Classical Text Editor version 10.1, last updated February 28, 2020, http://cte.oeaw.ac.at/.

23 Hakre Chryzo, "xslprocessor," accessed March 28, 2020 (plug-in closed as of August 30, 2019), https://wordpress.org/plugins/xslprocessor/.

24 "Shortcode," WordPress Codex (online manual), accessed March 18, 2020, https://codex.wordpress.org/shortcode.

25 For details see qTranslate X, version 3.4.6.8, https://wordpress.org/plugins/qtranslate-x/.

26 When looking for a readable version, users can just right-click on the transcription and choose "print" to access a version that is specially formatted for printing.

27 CollateX – Software for Collating Textual Sources, accessed March 18, 2020, https://collatex.net/.

28 A very brief overview of the three algorithms implemented in CollateX is given in "Alignment Algorithms," CollateX documentation, accessed March 18, 2020, https://collatex.net/doc/#alignment-algorithms.

29 A metric for the sameness of/difference between two strings, which counts the insertions, deletions, and substitutions needed to convert one string to the other.

AUTHORS

NILS GEIßLER

Nils Geißler is a student at the University of Cologne, working towards a MA in "Informationsverarbeitung" (humanities computer science). He has worked as a student assistant at the Cologne Center for eHumanities since May 2014.

DANIELA SCHULZ

Daniela Schulz is currently working at the Duke August Library Wolfenbüttel (HAB) in the context of the CLARIAH-DE project. While studying history and English at the University of Cologne, she participated in a number of different projects in institutions such as the Cologne Center for eHumanities (CCeH), the Royal Irish Academy (RIA), and the Duke August Library, and also received specific training in digital humanities.