



## Journal of the Text Encoding Initiative

Issue 8 | December 2014 - December 2015  
Selected Papers from the 2013 TEI Conference

---

# Bridging the Gap: Greater Usability for TEI encoding

Stefan Dumont and Martin Fechner

---



### Electronic version

URL: <http://journals.openedition.org/jtei/1242>

DOI: 10.4000/jtei.1242

ISSN: 2162-5603

### Publisher

TEI Consortium

### Electronic reference

Stefan Dumont and Martin Fechner, « Bridging the Gap: Greater Usability for TEI encoding », *Journal of the Text Encoding Initiative* [Online], Issue 8 | December 2014 - December 2015, Online since 09 June 2015, connection on 20 April 2019. URL : <http://journals.openedition.org/jtei/1242> ; DOI : 10.4000/jtei.1242

---

For this publication a Creative Commons Attribution 4.0 International license has been granted by the author(s) who retain full copyright.

---

# *Bridging the Gap: Greater Usability for TEI encoding*

Stefan Dumont and Martin Fechner

---

## 1. The Situation

### 1.1 A Gap Between User and Possibility

- 1 For a reader of *jTEI* it might be self-evident that in this day and age a historical-critical edition should be created with established technologies (e.g., XML). The benefits are well known: texts encoded in TEI-XML can be used to produce both Web and print publications and at the same time ensure long-term accessibility and reusability. Furthermore, both XML and TEI are well-established technologies—the TEI Guidelines having been in use for over 25 years. Nevertheless, in Germany as well as internationally, edition projects continue to use programs such as Microsoft Word to edit their texts. These texts are used to create print editions and may be digitized in a simple form such as PDF.
- 2 There is thus a gap between the available, useful technologies and their actual adoption. The cause of this is in many cases economic circumstances. Academic institutions and publishing houses are reluctant to alter their “tried-and-true” workflows, in part because any changes cost time

and money. Moreover, the know-how is often not available to correctly encode transcriptions into TEI. An important additional factor is that despite the many advantages of a Web publication, the printed edition—especially for veteran edition projects—remains the bar of what serious results should look like. Thus a new technology must justify itself not only economically, but academically as well (see [Sahle 2013, 109–10](#)).

- 3 Another cause of this gap is the dearth of programs to help researchers who lack the technical knowledge to transcribe and annotate their texts in TEI. While other prevalent text-editing programs offer an intuitive user interface, this is not the case for most XML editors.<sup>1</sup> A factor contributing to the lack of such a user interface could be the high complexity of a TEI-encoded edition.

## 1.2 The Bridge: Greater Usability

- 4 The solution that is repeatedly suggested—that researchers should learn TEI-XML or even learn how to program—does not seem viable, and in any case simply shifts the problem to the side of the user.<sup>2</sup> The lack of knowledge of how to encode a manuscript in TEI should not hinder the use of TEI. On the contrary, an easy method to adopt TEI for using should be made available. The center of editorial work should remain the evaluation and examination of manuscripts.<sup>3</sup>
- 5 When one considers the developments in graphical user interfaces over the last few years, one observes that programs are offering simpler and more intuitive interfaces—even with sometimes more complex program tasks and functions behind them. It is not without reason that there are now entire fields that concern themselves with the “ergonomics” of software.
- 6 In our opinion, the Digital Humanities community should not avoid such developments but instead should take advantage of them. A simple and intuitive user interface would help reduce the fear and frustration that often meets new technologies ([Nielsen 1993, 24–25](#)).<sup>4</sup> User-friendly programs create a positive user experience and would increase interest in using TEI for the transcription and annotation of manuscripts.
- 7 A fundamental part of user-friendly software is providing the user with visual feedback for an executed action ([Nielsen 1993, 134](#)). For example, after the appropriate markup for a crossed-out piece of text is entered, this text can then be displayed as crossed out. Although this type of feedback might seem trivial, another type certainly is not: print output. As mentioned above,

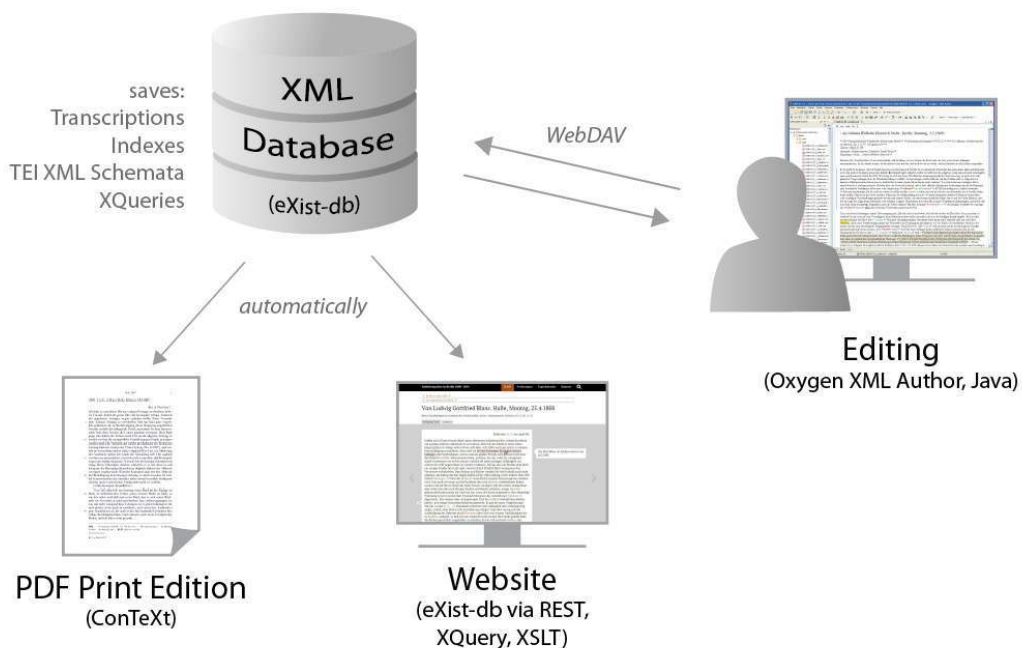
the printed edition is still the main goal for most scholarly edition projects, even when it is supplemented with a Web publication (Sahle 2013, 111–112). When the formatting process begins after the encoding is completed, the researcher can only check their work in an abstract manner. When the publication and creation of the edition is not separated, but happens simultaneously in one work environment, the researcher has the benefit of being able to see his or her markup with one click displayed properly in the desired medium. This not only improves the supervision of the work in progress, but also increases trust in and satisfaction with the technology—in this case the TEI encoding of manuscripts.

## 2. The Pilot Project

- 8 [TELOTA](#),<sup>5</sup> the digital humanities initiative at the [Berlin-Brandenburg Academy of Sciences and Humanities](#) (BBAW),<sup>6</sup> had to confront these issues as they were given the task to develop a digital work environment for the edition project [Schleiermacher in Berlin 1808–1834](#).<sup>7</sup> This environment was intended to assist the researchers in transcribing, encoding, and annotating the letters, lectures, and daily calendar of the Protestant theologian Friedrich Schleiermacher. In addition to creating a user interface for transcribing and editing, the environment had to be able to produce both a Web and a print edition from the TEI-XML data. Ideally this would happen automatically, so that the editors would be able to see their results immediately.
- 9 To develop such a solution from scratch requires significant resources. To reduce time and effort of development it is preferable to build on existing software. After evaluating the available software, the team chose the following programs:
  - [eXistdb](#)<sup>8</sup> for the database in which the TEI files would be stored. A decisive factor was the ability to retrieve and run XQuery and XSLT scripts from within the eXistdb and not just externally.
  - [Oxygen XML Author](#)<sup>9</sup> as the user interface for entering and editing the centrally stored TEI files. Decisive were the extensive functions of Oxygen XML Author that allowed visual editing of XML for persons without technical knowledge.

- **ConText**<sup>10</sup> for the formatting language to generate an automatic print-ready proof from the XML files. Among its benefits were its use of the long-established formatting language TeX, the possibility of processing XML directly, and the option to add one's own functions with the help of the programming language Lua.
- 10 Although eXistdb and ConText are open source software, this is not the case for Oxygen XML Author. A completely open-source solution would have been preferable, but there is no software currently available that offers such extensive functions for end users.
- 11 Because of the solution's modular organization, the software components can be replaced if necessary (for example when a tool is no longer supported by the community or a company). Essentially every piece of software can and will become outdated at some point; what is important is that the data is in a format that can still be reopened and reused.

Figure 1. Workflow in the digital work environment.



- 12 The three selected software components were combined, and scripts and additional functions were written. The goal was to create a software solution that was not just generally suitable for the research project, but customized to increase the user-friendliness of the work environment

despite the complexity behind a critical edition. Besides the creation of multiple TEI P5-compatible schemata, the functions and toolbar of the Oxygen XML Author were also configured and supplemented with additional custom functions. XQuery and XSLT scripts were written for the Web publication and its presentation was designed in HTML and CSS. The automatic generation of the print edition was programmed with the formatting language ConTeXt.

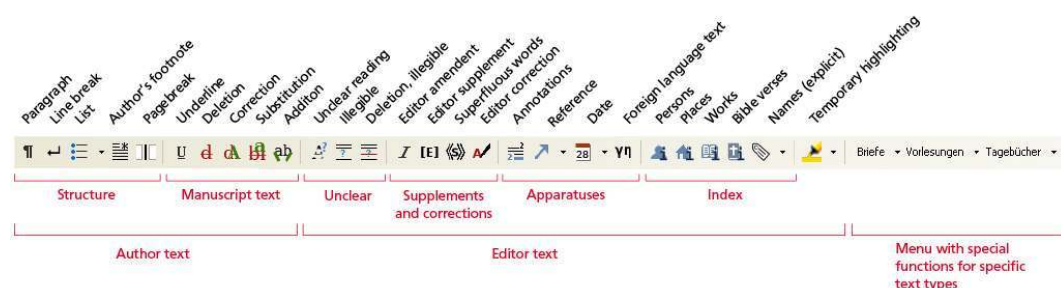
## 2.1 A Central Database Enables Collaborative Work

- 13 The digital work environment uses the open-source XML database eXistdb as its central repository for the TEI documents. The database is installed on a server and available online. This gives all the researchers on the project access to the same data collection, allowing them to work collaboratively. The contents of the database are directly available and searchable for users as a collection of files in Oxygen XML Author. Collaborative access is made possible through the WebDAV protocol, which locks files that are in use to avoid conflicts and unintended overwrites.

## 2.2 A User-Friendly Environment with Oxygen XML Author

- 14 Oxygen XML Author was chosen to create the user interface with which the editors transcribe and edit in the work environment.<sup>11</sup> While using this program, the editors usually do not see the code, but instead are in a user-friendly “author view” that is styled with CSS. More than one sub-view was made for the project within this author view, so that editors can choose the one most fitting for their current phase of work. In the various views, only certain TEI markup is shown or highlighted in the transcription. For example, when an editor only wants to examine the indexed passages of a text, then he or she can choose the view which only highlights the <i>index</i> elements.

Figure 2. Toolbar for the Schleiermacher edition with functions for inserting TEI markup.



- 15 Above all, however, the end user can enter markup into the manuscript or into the TEI Header with one click in the toolbar. In this way, text phenomena like struck-out text can be documented, editorial comments can be added, and the creation date of a manuscript can be noted. For every type of editorial markup there is a special button that enters the appropriate valid XML. If needed not only single elements but complex XML fragments will be inserted. Furthermore, the buttons for the metadata insert proper elements in the correct place of the `<teiHeader>`. If necessary, the editor is asked via an extra dialog box for the value of an attribute. The editor is thus not concerned with the name of the actual TEI element, whether it needs an attribute, or whether in this context there are structural variations to be considered. The entire text can thus be quickly and simply marked up with TEI conformant XML. The researcher can also, if desired, view the XML code at any time for more precise editing or simply to follow the process more closely.

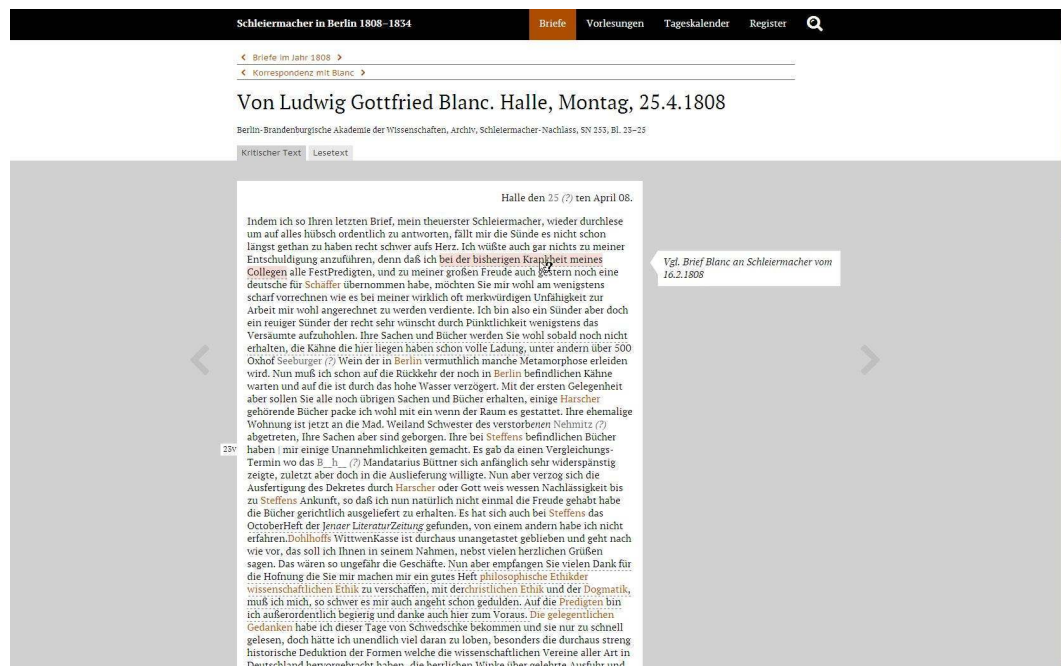
### 2.3 Central Indexes

- 16 Besides the transcription of the text, the database can also include central indexes (for persons, places, works, etc.) in the form of XML files. These can be edited and linked through the texts. If, for example, one wishes to mark up a person's name, one selects the word in the text to be indexed, and a dropdown list appears with all the names from the person index, from which the editor can choose the appropriate one. Selecting the name causes both the correct element `<persName>` and the correct ID of the person (as an attribute) to be entered. The index with the actual references to the correct text passages is then automatically created for both the print and Web publications from the marked-up TEI document. This index function is created through a specially-programmed Java operation for Oxygen XML Author and with various XQuery scripts in the database. When programming the Java operation, we used parameters and not constants in the source code. Thus, the parameter values could be comfortably entered in Oxygen XML Author and changed if necessary.





Figure 4. Website for the Schleiermacher edition.

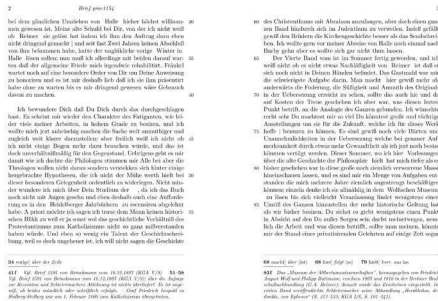


- 18 From a technical viewpoint, the website consists primarily of XQuery scripts, whose request results are presented as HTML5, through an XSL transformation. The scripts and transformations are requested from eXist through a REST interface and processed with the eXistdb internal parser—a great advantage of this database. The website is thus generated completely server-side.

## 2.5 Print Edition

- 19 A further output option is the print edition, implemented with the help of ConTeXt, which automatically generates a PDF (at any point in the workflow) from a TEI document. With the correct configuration, the format and presentation of the PDF can meet the research project's exact needs for printed editions. Each TEI element is given a specific formatting command through a configuration file. In this way the different apparatuses can, for example, appear as footnotes that refer to the main text with the help of line numbers and lemmata. The print edition can also provide the correct index for each transcription and resolve any cross-references between manuscripts.

Figure 5. Print edition for the Schliermacher edition generated via ConTeXt.



- 20 The print edition helps the researchers to check their TEI-encoded transcriptions; additionally, it obviates the need for a separate typesetting process.

### 2.6 Feedback

- 21 The researchers of *Schliermacher in Berlin 1808–1834* had a two-hour training session introducing them to the work environment. They were also given an extensive handbook, a quick start guide, and a “cheat sheet” with an overview of all the available functions.<sup>12</sup>
- 22 After working with the environment for two months, the researchers were asked to share their opinions. The feedback was overwhelmingly positive, and they confirmed that the environment eased their work on the edition and saved them time. The ability to check their work immediately on the website or in the print edition was seen as a great advantage. The researchers were also thankful that they were not expected to work directly in the XML code, but instead could mark up the texts with TEI conform XML in a simple graphical user interface. At the same time, some researchers had familiarized themselves with the TEI XML behind the interface and did sometimes edit directly in the XML, especially in the beginning when some functions were not yet available.

## 3. From the Bottom Up

- 23 In the end the pilot run was so successful that other projects within the Berlin-Brandenburg Academy of the Sciences and Humanities expressed interest. In the spring of 2013 the work environment was implemented for the Academy’s project *Commentaria in Aristotelem Graeca et Byzantina* (CAGB)<sup>13</sup> and in the summer of 2013 for *Regesta Imperii—Friedrich III.*<sup>14</sup> This was done

using the same technical concept, but with customized functions for each project for the text entry and for the publication (Web and print). This was due to the fact that the TEI schemata were in both cases very different from the pilot project. For example, CAGB is not an edition of manuscript transcriptions but of manuscript descriptions. In contrast, the time and effort for implementation would be minimal if one were using a schema from a project for which the work environment is already implemented. The amount of work does not depend on the software or technology, but on the complexity and diversity of TEI-encoded editions.

- 24 With every new implementation, the work environment developed further and was supplemented with new functions. This applied not just to the Java operations but also to the XQuery scripts, which are responsible for various fundamental tasks such as making the index file available for the Java operations. For these frequently used XQuery and XSLT scripts and for further configurations, an eXistdb application<sup>15</sup> was developed in 2013 that allows for a simple and fast installation into the database.<sup>16</sup> New functions are then, when suitable, integrated into the already existing work environment so that after the launch of the work environment the operability continually improves.
- 25 After we published a report about ediarum,<sup>17</sup> it caught the attention of other research institutions. During this time TELOTA introduced and explained the environment to interested institutions in various presentations. As a result, for example, the Academy of Science and Literature Mainz adopted the concept of ediarum and built their own digital work environment based on it. Other interested persons can read about the concept in a series of blog entries that are written as tutorials.<sup>18</sup> In addition, the custom Java functions were made available by TELOTA on [GitHub](#) for the TEI community.<sup>19</sup> There the functions can be downloaded and used directly in Oxygen Frameworks or, if necessary, changed in accordance with the GNU LGPL (Lesser General Public License).
- 26 At the end of 2013, TELOTA began implementing ediarum for the [historical-critical edition of Jeremias Gotthelf](#).<sup>20</sup> In this collaboration with Bern University, new functions are being developed and existing ones improved. For instance, we plan to build a function to allow end users to insert overlapping markup (using, e.g., <anchor> elements and @spanTo) in Oxygen XML Author.

## 4. Summary and Prospects

- 27 The bottom-up development of ediarum with its strong focus on user-friendliness brings with it certain implications.
- 28 It has required a lot of work during the initial implementation and customization of the environment, especially to generate the website and print edition. TELOTA is now considering simplifying the creation of the Web and print editions, so that less time is needed for programming. The goal, however, will be not to develop a complete plug-and-play solution. That would be impossible, considering the complexity and diversity of critical editions. The aim for a future version of ediarum will be rather to create a solution for encoding editions in TEI which involves the lowest possible amount of time and effort for programming while being customized to the specific needs of a project.
- 29 The concept of ediarum's development with its tailored solution has various advantages:
- Time and effort for programming is spared through the consistent use of existing software and the elimination of extraneous functionality not required by a particular project. In this way a functional software solution is reached more quickly and practically.
  - The work environment is customized to the exact needs of the project. Due to the user-friendly interface, researchers enjoy operating it and use it in their daily work.
  - The automatic generation of the Web and print edition from the XML file allows the researcher to immediately check their TEI encoding. This eases the publication process, leaving more time for the actual editorial work.
- 30 Thus, through addressing the needs of a specific project, we found a solution that can be used by many: ediarum helps bridge the gap between hesitant users and the many possibilities and advantages of TEI encoding.

---

## 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 (February). <http://jtei.revues.org/372>. doi:10.4000/jtei.372

- Nielsen, Jakob. 1993. *Usability Engineering*. Boston: Academic Press.
- Pape, Sebastian, Christof Schöch, and Lutz Wegner. 2012. "TEICHI and the Tools Paradox." *Journal of the Text Encoding Initiative 2* (February). <http://jtei.revues.org/432>. doi:10.4000/jtei.432.
- Rockwell, Geoffrey, Susan Brown, James Chartrand, and Susan Hesemeier. 2012. "CWRC-Writer: An In-Browser XML Editor." In *Digital Humanities 2012: Conference Abstracts*, edited by Jan Christoph Meister, 508–11. Hamburg: Hamburg University Press. [http://www.dh2012.uni-hamburg.de/wp-content/uploads/2012/07/HamburgUP\\_dh2012\\_BoA.pdf](http://www.dh2012.uni-hamburg.de/wp-content/uploads/2012/07/HamburgUP_dh2012_BoA.pdf).
- Sahle, Patrick. 2013. *Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels. Teil 2: Befunde, Theorie und Methodik*. Schriften des Instituts für Dokumentologie und Editorik, 8. Norderstedt: BoD. <http://kups.ub.uni-koeln.de/id/eprint/5352>.
- Vertan, Cristina, and Stefanie Reimers. 2012. "A TEI-based Application for Editing Manuscript Descriptions." *Journal of the Text Encoding Initiative 2* (February). <http://jtei.revues.org/392>. doi:10.4000/jtei.392.

## NOTES

- 1 In the TEI community one can find multiple tools. However, almost all of them support only the publication of TEI documents and not their creation (see, e.g., [Pape, Schöch, and Wegner 2012](#)). Tools that aim to support the actual transcription and editing of texts are less common. One such tool is from the Teuchos project, which provides a basic user interface for certain parts of the manuscript description ([Vertan and Reimers 2012](#)). In this case the data is not edited directly in XML but saved in a text format. In addition, it is not possible to enter inline markup. A more functional solution is the CWRCWriter, which still requires the user to have a good knowledge of TEI ([Rockwell et al. 2012](#) or <http://cwrctc.artsrn.ualberta.ca/>).
- 2 This point calls for a more thorough discussion that does not fit within the parameters of this article. It can nevertheless be determined that the lack of user-friendly tools is an obstacle for new TEI users ([Burghart and Rehbein 2012](#), § 32).
- 3 Of course the creation of a TEI-conformant XML schema is also an editorial process and is thus done in collaboration with the editors, when not done by them directly.
- 4 Sahle (2013, 111) notes that editors are often afraid of the technical challenges of digital editions.
- 5 <http://www.bbaw.de/en/telota>.
- 6 <http://www.bbaw.de/en>.
- 7 <http://www.bbaw.de/en/research/schleiermacherII>.

8 <http://exist-db.org/>.

9 [http://www.oxygenxml.com/xml\\_author.html](http://www.oxygenxml.com/xml_author.html).

10 [http://wiki.contextgarden.net/Main\\_Page](http://wiki.contextgarden.net/Main_Page).

11 To this end we used “Oxygen Frameworks”: that is to say, we developed additional frameworks within the context offered by the program. Oxygen XML delivers a standard framework for TEI documents that provides a toolbar with a few basic TEI elements. However, because of the large number of TEI elements and attributes, the development of an all-inclusive TEI Framework would not be practical. The editor would be overwhelmed with such a large number of superfluous functions.

12 <http://www.bbaw.de/en/telota/software/ediarum>.

13 <http://www.bbaw.de/en/research/cagb>.

14 <http://www.bbaw.de/forschung/regestaimperii> and <http://www.regesta-imperii.de/unternehmen/abteilungen/xiii-friedrich-iii.html>.

15 For further information about apps in eXistdb, see “Getting Started with Web Application Development,” <http://exist-db.org/exist/apps/doc/development-starter.xml>.

16 This eXistdb app will soon be available online.

17 Stefan Dumont and Martin Fechner, “Digitale Arbeitsumgebung für das Editionsprojekt ‘Schleiermacher in Berlin 1808–1834’,” *digiversity: Webmagazin für Informationstechnologie in den Geisteswissenschaften* (October 11, 2012), <http://digiversity.net/2012/digitale-arbeitsumgebung-fur-das-editionsprojekt-schleiermacher-in-berlin-1808-1834/>.

18 Stefan Dumont, “Tutorial: Wie baue ich ein eigenes Framework für Oxygen XML?,” *digiversity: Webmagazin für Informationstechnologie in den Geisteswissenschaften* (October 30, 2013), <http://digiversity.net/2013/tutorial-wie-baue-ich-ein-oxygen-xml-framework/>, and Stefan Dumont, “Tutorial: Indexfunktionen für Oxygen XML Frameworks,” *digiversity: Webmagazin für Informationstechnologie in den Geisteswissenschaften* (December 16, 2013), <http://digiversity.net/2013/tutorial-indexfunktionen-fuer-oxygen-xml-frameworks/>.

19 [https://github.com/telota/ediarum/tree/master/oxygen\\_java](https://github.com/telota/ediarum/tree/master/oxygen_java); for documentation, see [https://github.com/telota/ediarum/wiki/Documentation-ediarum.jar-\(english\)](https://github.com/telota/ediarum/wiki/Documentation-ediarum.jar-(english)).

20 *Jeremias Gotthelf: Historisch-kritische Gesamtausgabe (HKG)*, Forschungsstelle Jeremias Gotthelf, Universität Bern, [http://www.gotthelf.unibe.ch/content/index\\_ger.html](http://www.gotthelf.unibe.ch/content/index_ger.html).

---

## ABSTRACT

This article shows how, with a focus on user-friendliness and a bottom-up approach, the digital work environment ediarum was successfully developed. With ediarum, researchers can comfortably encode and edit in TEI, as well as publish their results in an online or print edition. This solution, developed by the TELOTA initiative of Berlin-Brandenburg Academy of Sciences and Humanities, is based on three software components: eXistdb, Oxygen XML Author, and ConTeXt. These components are combined, supplemented with additional functions, and tailored to fit a project's needs. After a pilot run, ediarum has been implemented for multiple internal and external research projects. The experience that was gained and our self developed program components are available to the Digital Humanities community.

## INDEX

**Keywords:** tools, usability, bottom up, usability, graphical user interface, ediarum

## AUTHORS

### STEFAN DUMONT

Stefan Dumont is a research associate in the TELOTA working group for digital humanities at the Berlin-Brandenburg Academy of Sciences and Humanities.

### MARTIN FECHNER

Martin Fechner is a research associate in the TELOTA working group for digital humanities at the Berlin-Brandenburg Academy of Sciences and Humanities.