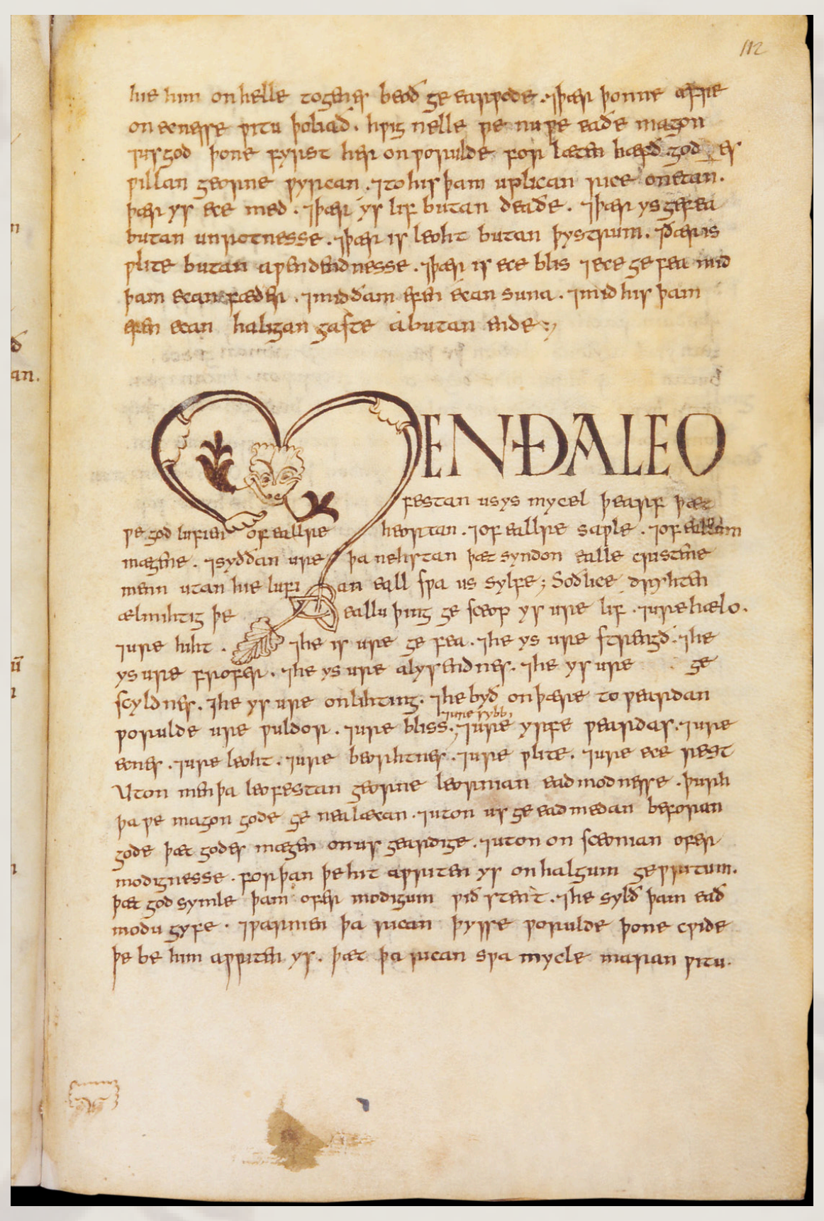
Edition Visualization Technology

A simple tool to visualize TEI-based digital editions

The Problem

The Vercelli Book digital edition

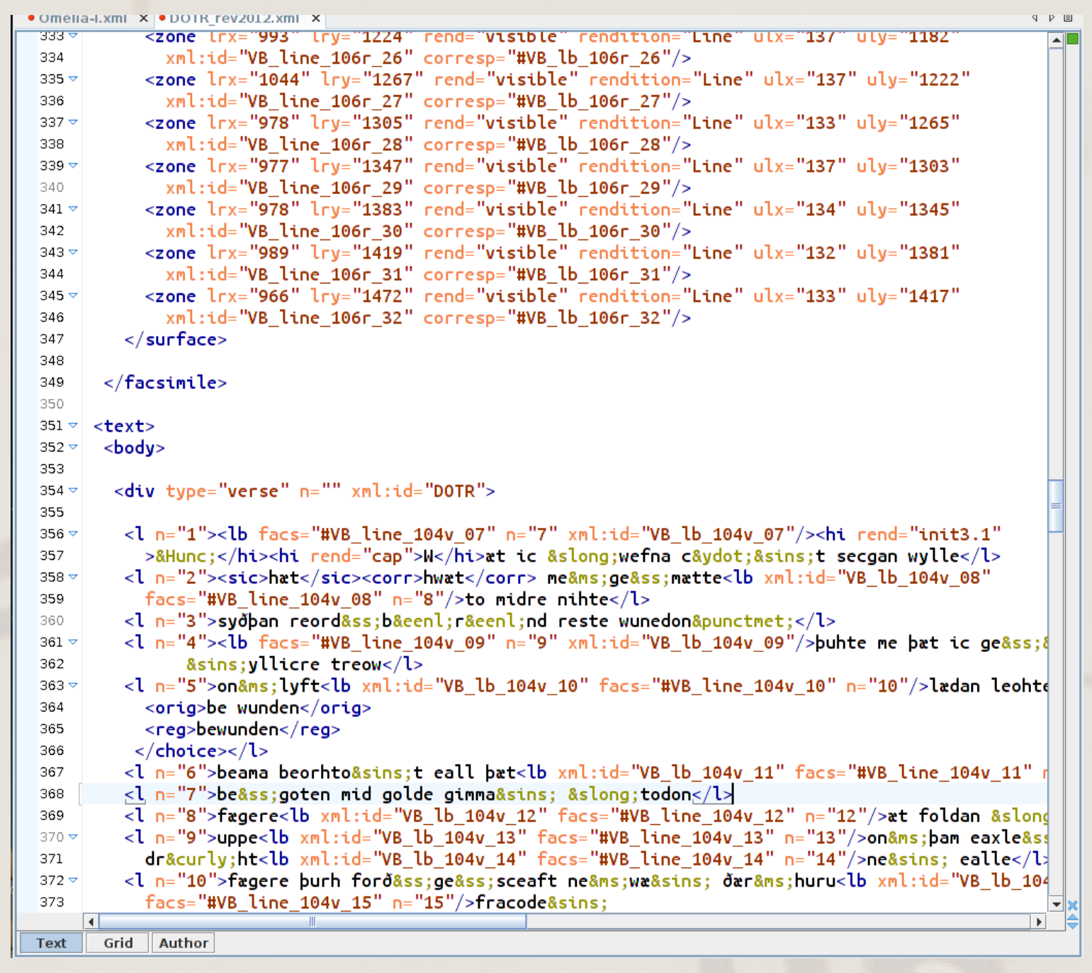
When the transcription of the Vercelli Book manuscript (Archivio e Biblioteca Capitolare di Vercelli CXVII; web site: http://vbd.humnet.unipi.it/) passed the 50% landmark, researchers of the project started to think about the best way to visualize the edition. Thanks to the openness of the Archivio Capitolare, it was decided to abandon the original plan of a CD/DVD publication, largely inspired by digital editions such as the Electronic Beowulf, in favor of a web-based publication. While this decision was critical in selecting the most supported and wide-spread medium, we soon discovered that it didn't make choices any simpler: on the one hand, the XSLT style sheets provided by TEI are great for HTML rendering, but do not include support for image-related features (such as the text-image linking available with the TEI P5 schemas) and tools (zoom in/out, magnifying lens, hot spots etc.) that represent a significant part of a digital facsimile and/or diplomatic edition; other features, such as an XML search engine, would have to be integrated at a later time. On the other hand, there are powerful frameworks based on CMS and other web technologies which looked far too complex and expensive, also considering future maintenance needs, for our project's purposes.



Vercelli Book f. 112R

Standard vs. fragmentation

We had to conclude that, while the TEI schemas and Guidelines are a solid foundation for philological work, this excellent standard is matched by an astounding diversity of publishing tools, which is particularly true when it comes to digital editions, in particular editions including images of manuscripts. As a consequence, a single scholar, or a small group of researchers, can surely work and encode an image-based edition in TEI XML, but will have to look for further support and resources to publish it, which is a serious hurdle towards greater acceptance of digital philology methods and techniques (not to mention publication of a digital edition of the Vercelli Book :). While continuing to inquire about a good, community-tested solution, and in spite of being aware of the dangers coming from the so-called NIH (Not Invented Here) syndrome, we started looking into a project-specific solution that could fulfill our needs.

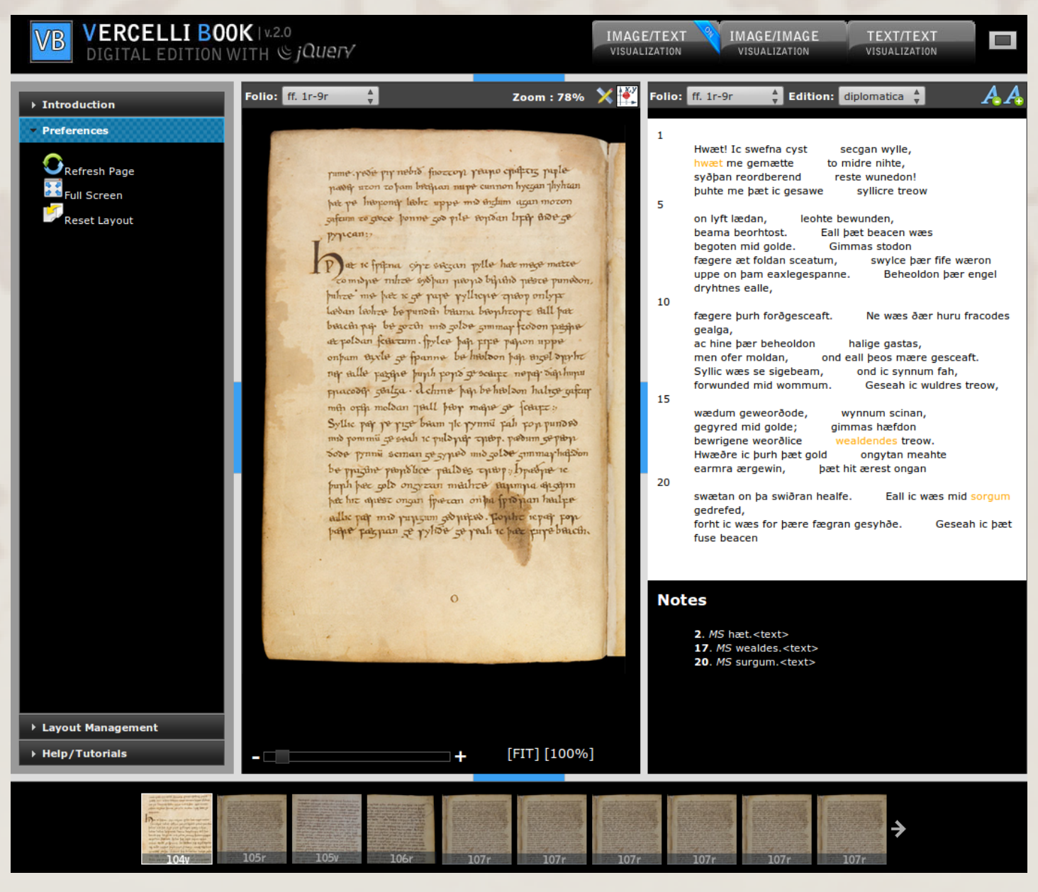


```
    <zone lrx="993" lry="1224" rend="visible" rendition="Line" ulx="137" uly="1182"
      xml:id="VB_line_106r_26" corresp="#VB_lb_106r_26"/>
    <zone lrx="1044" lry="1267" rend="visible" rendition="Line" ulx="137" uly="1222"
      xml:id="VB_line_106r_27" corresp="#VB_lb_106r_27"/>
    <zone lrx="978" lry="1305" rend="visible" rendition="Line" ulx="133" uly="1265"
      xml:id="VB_line_106r_28" corresp="#VB_lb_106r_28"/>
    <zone lrx="977" lry="1347" rend="visible" rendition="Line" ulx="137" uly="1303"
      xml:id="VB_line_106r_29" corresp="#VB_lb_106r_29"/>
    <zone lrx="978" lry="1383" rend="visible" rendition="Line" ulx="134" uly="1345"
      xml:id="VB_line_106r_30" corresp="#VB_lb_106r_30"/>
    <zone lrx="989" lry="1419" rend="visible" rendition="Line" ulx="132" uly="1381"
      xml:id="VB_line_106r_31" corresp="#VB_lb_106r_31"/>
    <zone lrx="966" lry="1472" rend="visible" rendition="Line" ulx="133" uly="1417"
      xml:id="VB_line_106r_32" corresp="#VB_lb_106r_32"/>
  </surface>

 </facsimile>

<text>
 <body>

  <div type="verse" n="" xml:id="DOTR">

   <l n="1"><lb facs="#VB_line_104v_07" n="7" xml:id="VB_lb_104v_07"/><hi rend="init3.1"
     >&Hunc;</hi><hi rend="cap">W</hi>æt ic &slong;wefna c&ydot;&sins;t secgan wylle</l>
   <l n="2"><sic>hæt</sic><corr>hwæt</corr> me&ms;ge&ss;mætte<lb xml:id="VB_lb_104v_08"
     facs="#VB_line_104v_08" n="8"/>to midre nihte</l>
   <l n="3">syðþan reord&ss;b&eenl;r&eenl;nd reste wunedon&punctmet;</l>
   <l n="4"><lb facs="#VB_line_104v_09" n="9" xml:id="VB_lb_104v_09"/>þuhte me þæt ic ge&ss;
      &sins;yllicre treow</l>
   <l n="5">on&ms;lyft<lb xml:id="VB_lb_104v_10" facs="#VB_line_104v_10" n="10"/>lædan leoht
    <orig>be wunden</orig>
    <reg>bewunden</reg>
   </choice></l>
   <l n="6">beama beorhto&sins;t eall þæt<lb xml:id="VB_lb_104v_11" facs="#VB_line_104v_11"
   <l n="7">be&ss;goten mid golde gimma&sins; &slong;todon</l>
   <l n="8">fægere<lb xml:id="VB_lb_104v_12" facs="#VB_line_104v_12" n="12"/>æt foldan &slong
   <l n="9">uppe<lb xml:id="VB_lb_104v_13" facs="#VB_line_104v_13" n="13"/>on&ms;þam eaxle&s
    dr&curly;ht<lb xml:id="VB_lb_104v_14" facs="#VB_line_104v_14" n="14"/>ne&sins; ealle</l
   <l n="10">fægere þurh forð&ss;ge&ss;sceaft ne&ms;wæ&sins; ðær&ms;huru<lb xml:id="VB_lb_104
    facs="#VB_line_104v_15" n="15"/>fracode&sins;
```

TEI XML markup for The Dream of the Rood poem

First experiments

At first, however, EVT was born more as an experimental research project for students at the Informatica Umanistica course of the University of Pisa than as a real attempt to solve the digital edition viewer problem. We aimed at investigating some UI-related aspects of such a viewer, in particular certain usability problems that are often underestimated by similar projects, and to push the use of standards so to ensure the maximum longevity for the edition; we also considered releasing the source code as free software right from the start. The first implementations of our concepts were promising, but in the end we felt we reached a dead end: in spite of all our good intentions the UI looked cluttered, the software sported several secondary features (such as a rich text editing widget), but lacked critical ones (text-image linking), and wasn't fully stable, possibly as a result of too many widgets of different origins being used at the same time. On the architectural side, data was loaded straight into the web code, which had to be done by hand, with no options for configuration at all. In conclusion: not bad as a first attempt, but falling way short of fundamental goals we had in mind.



1 Hwæt! Ic swefna cyst secgan wylle,
hwæt me gemætte to midre nihte,
syðþan reordberend reste wunedon!
þuhte me þæt ic gesawe syllicre treow

5 on lyft lædan, leohte bewunden,
beama beorhtost. Eall þæt beacen wæs
begoten mid golde. Gimmas stodon
fægere æt foldan sceatum, swylce þær fife wæron
uppe on þam eaxlegespanne. Beheoldon þær engel
dryhtnes ealle,

10 fægere þurh forðgesceaft. Ne wæs ðær huru fracodes
gealga,
ac hine þær beheoldon halige gastas,
men ofer moldan, ond eall þeos mære gesceaft.
Syllic wæs se sigebeam, ond ic synnum fah,
forwunded mid wommum. Geseah ic wuldres treow,

15 wædum geweorðode, wynnum scinan,
gegyred mid golde; gimmas hæfdon
bewrigene weorðlice wealdendes treow.
Hwæðre ic þurh þæt gold ongytan meahte
earmra ærgewin, þæt hit ærest ongan

20 swætan on þa swiðran healfe. Eall ic wæs mid sorgum
gedrefed,
forht ic wæs for þære fægran gesyhðe. Geseah ic þæt
fuse beacen

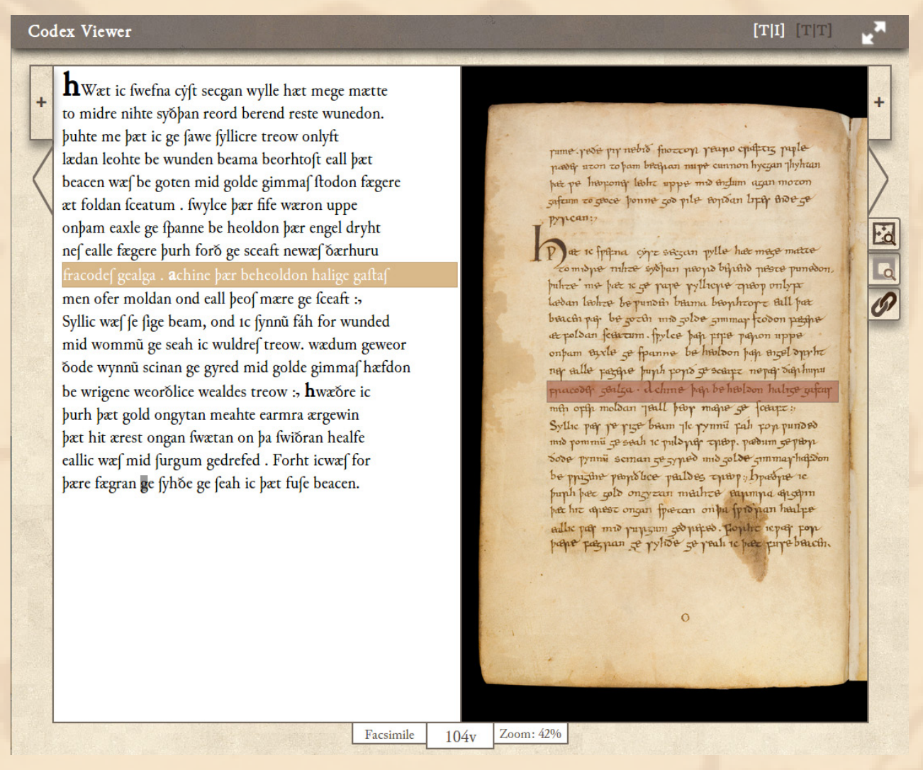
2. MS hæt.<text>
17. MS wealdes.<text>
20. MS surgum.<text>

The last version of EVT before the "reboot"

The current EVT version

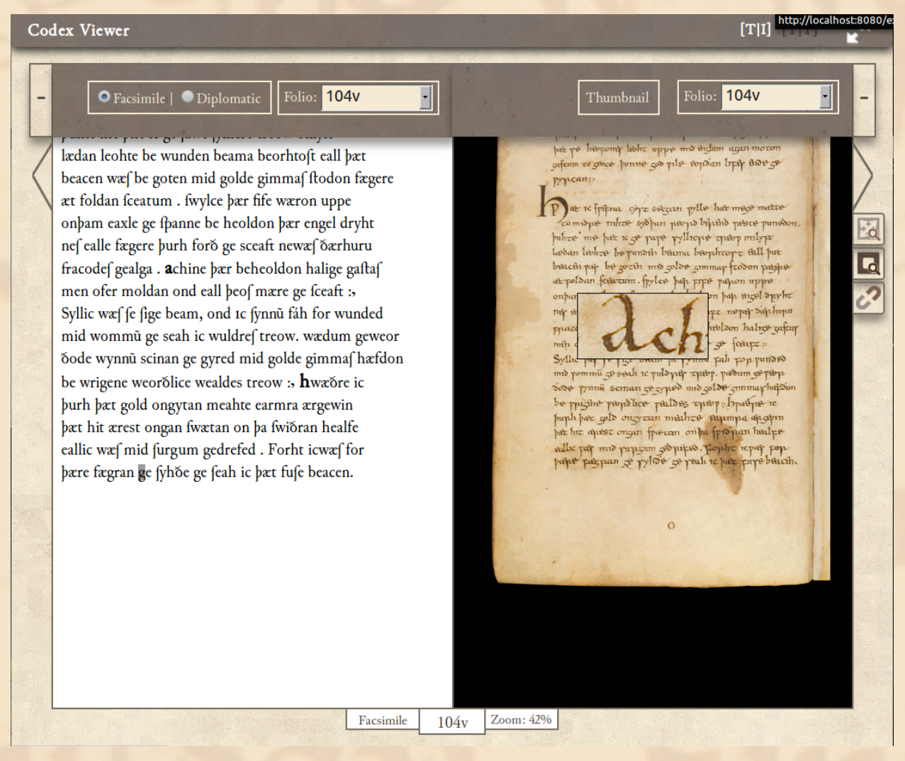
EVT v. 0.2.0, rebooting the project

To get out of the impasse we decided to completely reboot the project, removing secondary features and giving priority to essential ones. We also found a solution for the data-loading problem: instead of finding a way to load the data into the software we decided to build the edition around the data itself. Making the TEI XML files the starting point means that the editor can focus on his work, marking up the edition, with very little configuration needed to create the edition. Also this approach allowed us to quickly test XML files belonging to other edition projects, to check if EVT could go beyond being a project-specific tool. The inspiration for these changes came from TEI Boilerplate and other recent and interesting tools being developed within the TEI community. EVT is built on open and standard web technologies, such as HTML, CSS and Javascript, to ensure that it will be working on all the most recent web browsers, and for as long as possible on the World Wide Web itself; specific features, such as the magnifying lens, are entrusted to jQuery plugins, again chosen among the open source and best supported ones to reduce the risk of future incompatibilities; the general architecture of the software, in any case, is modular, so that any component which may cause trouble or turn out to be not completely up to the task can be replaced easily.



hWæt ic ſwefna cẏſt secgan wylle hæt mege mætte
to midre nihte syðþan reord berend reste wunedon.
þuhte me þæt ic ge ſawe ſyllicre treow onlyft
lædan leohte be wunden beama beorhtoſt eall þæt
beacen wæſ be goten mid golde gimmaſ ſtodon fægere
æt foldan ſceatum . ſwylce þær fife wæron uppe
onþam eaxle ge ſpanne be heoldon þær engel dryht
neſ ealle fægere þurh forð ge sceaft newæſ ðærhuru
fracodeſ gealga . achine þær beheoldon halige gaſtaſ
men ofer moldan ond eall þeoſ mære ge ſceaft :-
Syllic wæſ ſe ſige beam, ond ıc ſynnū fāh for wunded
mid wommū ge seah ic wuldreſ treow. wædum geweor
ðode wynnū scinan ge gyred mid golde gimmaſ hæfdon
be wrigene weorðlice wealdes treow :- hwæðre ic
þurh þæt gold ongytan meahte earmra ærgewin
þæt hit ærest ongan ſwætan on þa ſwiðran healfe
eallic wæſ mid ſurgum gedrefed . Forht icwæſ for
þære fægran ge ſyhðe ge ſeah ic þæt fuſe beacen.

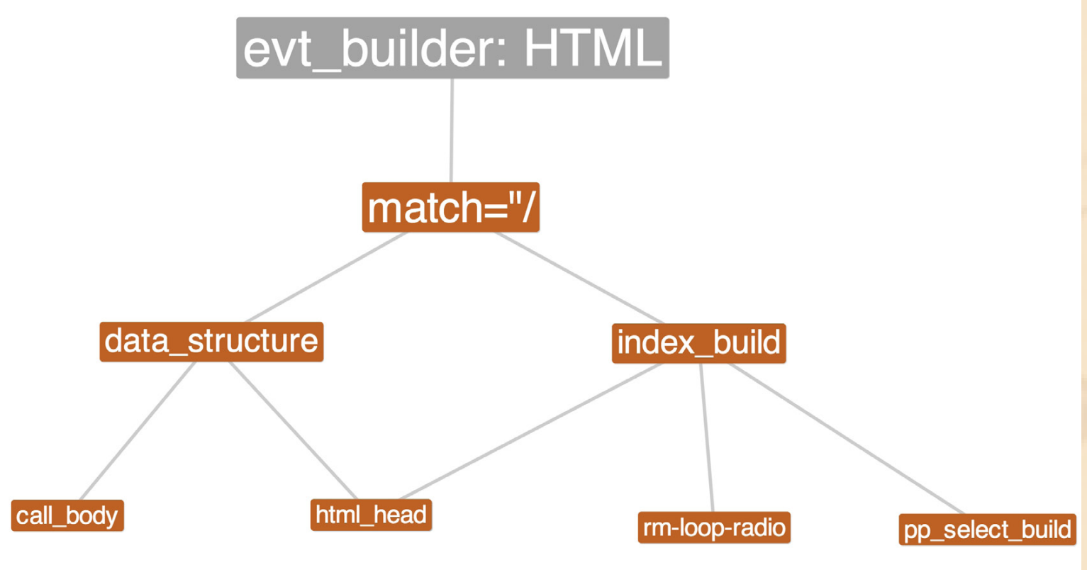
The current version of EVT showing the image-text linking feature. Original concept by R. Masotti, developed together with J. Kenny



The current version of EVT with the side menus open

How it works

The basic idea of EVT is very similar to how TEI Boilerplate works, i.e. you apply an XSLT style sheet to your already marked-up file(s), and you're presented with a web-ready document. The ideal goal would be to have a simple, very user-friendly drop-in tool, requiring little work and/or knowledge of anything beyond XML from the scholar. To reach this goal, EVT is based on a modular structure where a single style sheet (evt_builder.xsl) starts a chain of transformations calling in turn all the other modules; the latter belong to two general categories: those devoted to building the HTML site, and the XML processing ones, which extract the edition text lying between folios using the <pb/> element and format it according to the edition level. All XSLT modules live inside the builder_pack folder, in order to have a clean and well organized directory hierarchy.



HTML generation by the XSLT style sheets

As a result, assuming the available formatting style sheets meet your project's criteria, to create a digital edition you only have to follow three simple steps:

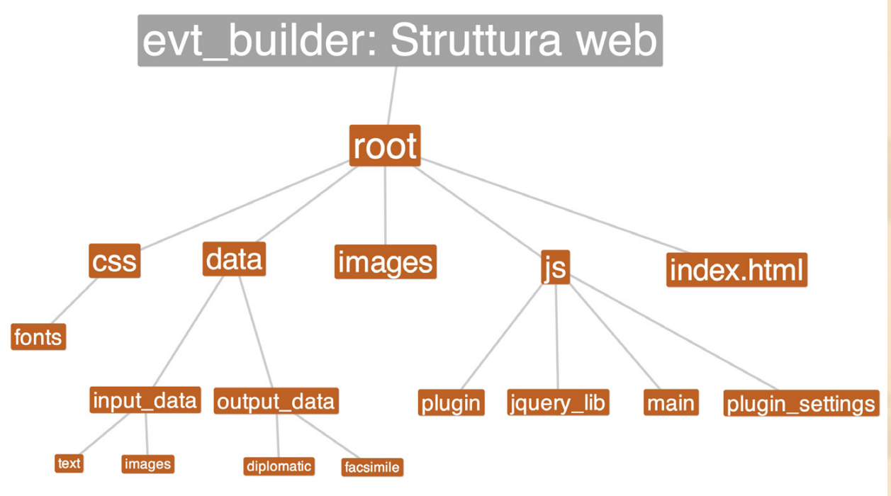
- copy the edition data in the data/input_data folder, there are different sub-folders for text (TEI XML documents) and images (these have to follow a specific naming convention);
- optionally you can modify transformation options editing evt_builder-conf.xsl, e.g. the number of edition levels, presence of images, etc.;
- you can then apply the evt_builder.xsl style sheet to your TEI XML document using Oxygen or another XSLT 2 compliant engine.

When the XSLT processing is finished, the starting point for the edition is the index.html file in the root directory, and all the HTML pages resulting from the transformations will be stored in the output_data folder. You can delete everything in this latter folder (and the index.html file), modify the configuration options and start again, everything will be re-created in the assigned places.

Features

At the present moment EVT can be used to create image-based editions with two possible edition levels: diplomatic and diplomatic-interpretative; this means that a transcription encoded using elements of the TEI transcr module (see chapter 11 Representation of Primary Sources in the Guidelines) should already be compatible with EVT, or require only minor changes to be made compatible. The Vercelli Book transcription is following the standard TEI schemas with no custom elements or attributes added: our tests with similarly encoded texts showed a high grade of compatibility. A critical edition level is being studied and it will be added in the future.

On the image side, several features such as a magnifying lens, a general zoom, image-text linking and more are already available. The image-text feature is inspired by Martin Holmes' Image Markup Tool software and was implemented in XSLT and CSS by one of the students collaborating to the project; all other features are achieved by using jQuery plugins.



Directory structure of the EVT root folder

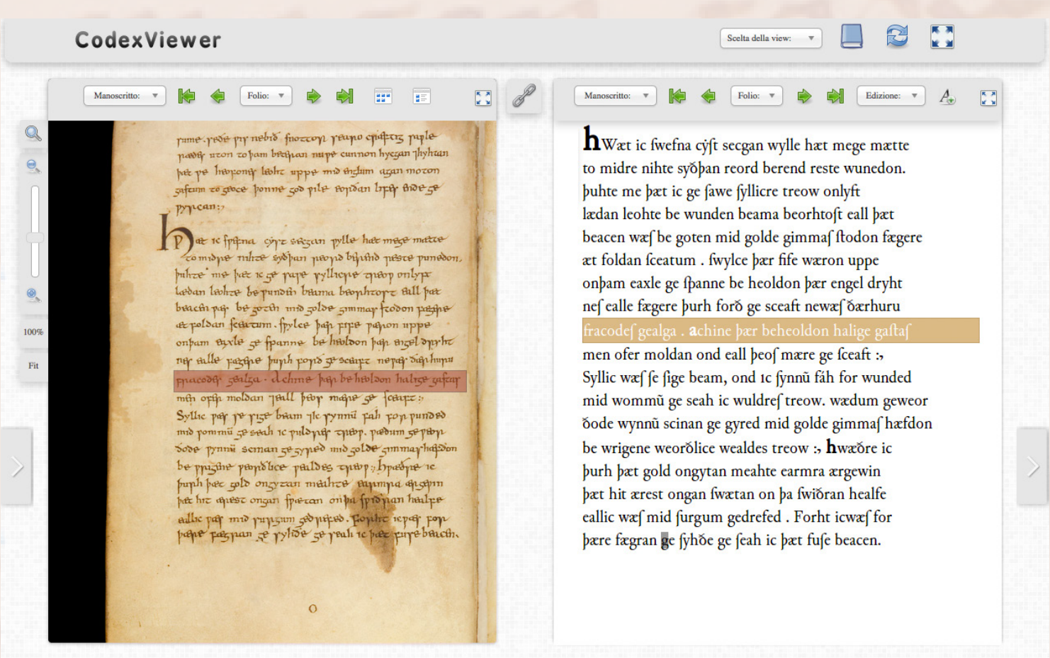
The XSLT style sheets

EVT builder's transformation system divides an XML file holding the manuscript transcription into smaller portions, each one corresponding to a folio side (recto or verso), and for each of these portions of text it creates as many output files as requested by the file settings. Using XSLT modes it is possible to separate the rules for different transformations of a TEI element and to recall other XSLT style sheets in order to manage the transformations or send different parts of a document to different parts of the transformation chain. This allows to extract different texts required for different edition levels (diplomatic, diplomatic-interpretative, critical) processing the same XML file, and to save them in the HTML site structure which is available as a separate XSLT module. If the TEI elements that are processed are placed in an HTML element with the class edition_level- TEI_ element's_name (e.g. for the element <abbr> in the transformation to the diplomatic edition that would be dipl-abbr) it is possible to keep the semantic information contained in the markup and, if necessary, associate the element with the corresponding class in the CSS rules so as to specify the visualization and highlighting of the item.

The Future

New layout

One important aspect that will be introduced in the next version of EVT is a completely revised layout: the current user interface is the same temporary UI introduced to test the new builder system and hasn't changed since then, it is now time to design and implement a new interface that will include all the features which will be available in EVT and that are currently not implemented (such as the possibility to maximize frames, to access a search engine, to browse the introduction and other documentation, etc.). Some of the new features planned for the next version(s) will require considerable changes in the general appearance and layout of the software.

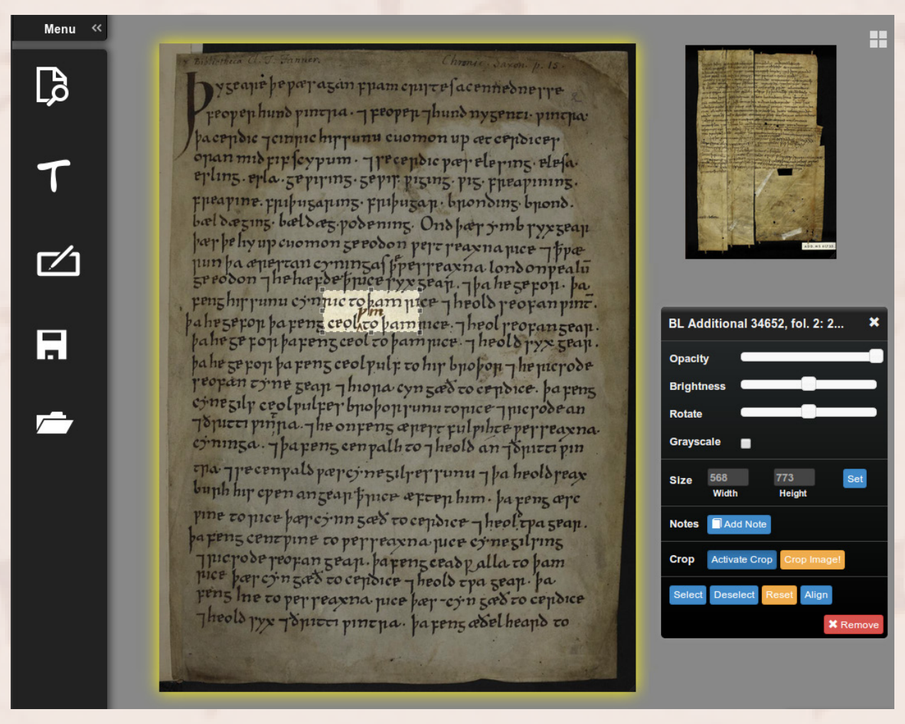


The new layout under development by C. Leoni

New tools / features

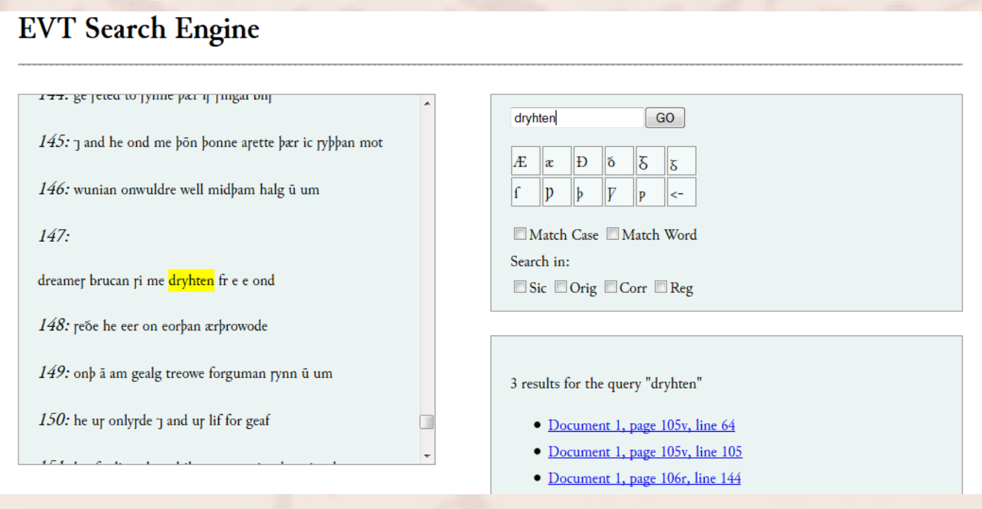
Speaking of features, there are several that we are currently considering for inclusion in EVT:

- first and foremost, an XML-based search engine and database such as eXist (http://exist-db.org/): work on this is already started and ongoing, but it will require a lot of time and resources before this task will be finished;
- an XSLT style sheet for a critical edition: this is a very important feature, but also one that depends on future developments of the TEI Critical Apparatus module (unless we decide to write it for the current CA module);
- a new "view" dedicated to manuscript browsing: a much simpler layout which will work as a bookreader for digitized images;
- a "digital lightbox" view for advanced image manipulation and study: a separate view where it is possible to load one or more images, crop and annotate them, apply graphic filters, etc.



The Digital Lightbox view currently developed by G. Buomprisco

Don't forget that this is a free software project: the software is already available on SourceForge, if you are interested in using it and/or helping with its development just let us know! You can already download all the code and experiment with it.



145: ⁊ and he ond me þōn þonne aſette þær ic ſyþþan mot

146: wunian onwuldre well midþam halg ū um

147:

dreameſ brucan ſi me dryhten fr e e ond

148: ſeðe he eer on eorþan ærþrowode

149: onþ ā am gealg treowe forguman ſynn ū um

150: he uſ onlyſde ⁊ and uſ lif for geaf

3 results for the query "dryhten"

- Document 1, page 105v, line 64
- Document 1, page 105v, line 105
- Document 1, page 106r, line 144

The experimental UI for eXist integration (J. Pugliese)

New architecture

The integration of EVT with eXist will imply a very important change in EVT: so far everything, from XSLT processing to browsing of the resulting web site, is done on the client side, but use of eXist will require a move to the more complex client-server architecture. We will try to make the move as painless as possible and to preserve the basic simplicity and flexibility that has been a major feature of EVT so far.

The EVT Team

Project lead: Roberto Rosselli Del Turco (Università di Torino)
XSLT framework, graphic plugins: Julia Kenny and Raffaele Masotti (Università di Pisa)
eXist integration: Jacopo Pugliese (Università di Pisa)
Digital Lightbox: Giancarlo Buomprisco (King's College London)
UI design, HTML programming: Chiara Leoni (Università di Pisa)

SourceForge repository:
https://sourceforge.net/projects/evt-project/

Poster credits: Chiara Leoni