

EVT User Manual

V. 0.2

Table of Contents

Introduction.....	3
About EVT.....	3
How it Works.....	3
Main Features.....	4
A Quick Guide to EVT.....	4
0. Installation.....	5
1. Images.....	5
1.1 Single Side Images.....	5
1.2 Double Side Images.....	6
1.3 Hotspot Images.....	7
1.4 Other Images.....	7
2. TEI XML Transcription Files.....	8
2.0 TEI Header.....	8
2.1 Facsimile.....	8
2.2 Parallel Transcription.....	9
2.3 Embedded Transcription.....	11
3. Configuring EVT.....	13
4. Step by step instructions.....	13
Known bugs.....	14

Introduction

This guide is aimed at the editor interested in creating an image-based edition using EVT, and not at the reader/user of such edition. To use EVT the editor must have a good knowledge of the TEI XML markup, and must be able to apply XSLT style sheets to an XML document. EVT configuration is done by editing a single .xsl file, note however that the software can be customized as desired modifying the relevant XSLT and CSS style sheets, or adding new ones (f.i. for a different kind of text visualization).

About EVT

EVT (Edition Visualization Technology) is a software for creating and browsing digital editions of manuscripts based on text encoded according to the TEI XML (<http://www.tei-c.org/>) schemas and Guidelines. This tool was born as part of the DVB (Digital Vercelli Book: <http://vbd.humnet.unipi.it/>) project in order to allow the creation of a digital edition of the Vercelli Book, a parchment codex of the late tenth century, now preserved in the Archivio e Biblioteca Capitolare of Vercelli and regarded as one of the four most important manuscripts of the Anglo-Saxon period as regards the transmission of poetic texts in the Old English language. 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, EVT is built on open and standard web technologies such as HTML, CSS and JavaScript. Specific features, such as the magnifying lens, are entrusted to jQuery plug-ins, 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.

How it Works

The basic idea of EVT is very similar to the *modus operandi* which is commonly used to convert TEI XML documents in HTML: when the main style sheet is applied to the document, it starts a processing which ends with a website containing the digital edition of the manuscript. Our ideal goal, in fact, is to have a simple, very user-friendly drop-in tool, requiring little work and/or knowledge of anything beyond XML from the editor. To reach this goal, EVT is based on a modular structure where a single style sheet (`evt_builder.xsl`) starts a chain of XSLT 2.0 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 (the content of each folio which lies between `<pb/>` elements, or the content of each `<surface>` when the Embedded Transcription method is used) 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.

Main 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 TEI *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.

A Quick Guide to EVT

To create an image-based edition EVT requires both images and TEI XML documents as input data. For the XSLT style sheets to work properly, all the input data must follow specific rules:

- images have to be named according to a strict naming convention and have to be available in a few different sizes/resolutions;
- TEI XML documents are likewise bound not only by the encoding schema, which must include the *transcr* module, but also by the use of several attributes whose values must follow rigid conventions.

All these conventions, which will be explained in full detail in the following sections, are necessary because they build a layer of rich metadata information which is accessed by the different XSLT style sheets to find all the relevant data to be extracted, processed and finally presented to the user.

After preparing all the image and textual data according to the conventions that will be explained below (sections *1. Images* and *2. TEI XML Transcription Files*), the user will be required to check and if necessary modify the EVT configuration file to suit her/his needs (see section *3. Configuring EVT*), and finally follow a step-by-step path to put everything in place and start the transformation chain that will result in a web-based edition (see section *4. Step by Step Instructions*).

This guide tries to include examples for each case described, however remember that the zipped archive available on SourceForge (<https://sourceforge.net/projects/evt-project/>) includes all text and image files necessary for the sample editions it holds, so that you can inspect them in case of doubt.

In the `builder_pack/doc` folder you will find a first draft of the technical documentation, automatically generated thanks to the [XSLTdoc](#) system. **[[Incompleta, mancano “short**

descriptions"]]

0. Installation

EVT installation is quite easy: you only have to download the zipped archive from [SourceForge](#) and unpack it in some suitable folder on your hard disk. Everything is already in place, first of all you should browse the included examples clicking on the `index.html` document. You can also browse the available documents in the `data/input_data` folder, especially the TEI XML files, to understand what the building system requires; if you feel particularly adventurous you're welcome to browse the XSLT style sheets in the `builder_pack/` and `builder_pack/modules` folders, note however that you should try to modify these files only if you know what you are doing!

The `data/input_data` folder is where your own data will go before you can use EVT to produce a web edition of your manuscript.

1. Images

Digitized images of the manuscript(s) should be provided in three different formats:

- single side images
- double side images
- detail (hot spot) images

and the first two formats must be made available in three different resolutions:

- standard
- high (mandatory for the Magnifier tool to work properly)
- thumbnail (not yet used for double side images)

Note that, although only single side images are actually necessary, it is recommended to provide double side ones too, so that the Bookreader view is available to browse the manuscript showing verso-recto folios. If for some reason the double side images aren't available, however, the Bookreader view can be disabled in the configuration file (see below).

1.1 Single Side Images

These are single page/folio images of the manuscript:

- **standard** resolution: each image must be named "[folio_number].jpg" (e.g. "104v.jpg"),

where "[folio_number]" is the value of the @xml:id attribute of the <pb/> element (if using the Parallel Transcription method) or the <surface> element (if using the Embedded Transcription method) that is used to encode the corresponding folio;

- **high** resolution images: these are necessary for the magnifier (the proportion must be the same as the normal ones) and are named "[folio_number]_big.jpg" (e.g. "104v_big.jpg"), where "[folio_number]" is the name of the corresponding image in standard resolution (note the "big" suffix);
- **thumbnails** of the standard images must be named "[folio_number]_small.jpg" (e.g. "104v_small.jpg"), where "[folio_number]" is the name of the corresponding image (note the "small" suffix).

Important: image files must end with a ".jpg" extension; also note that it is recommended to use a meaningful prefix including, f.i., a reference to the manuscript (e.g. "VB_fol_104v", which means "folio 104v of the Vercelli Book manuscript"), essential when dealing with more than one manuscript.

Examples:

VB_fol_104v.jpg	263,0 kB
VB_fol_104v_big.jpg	2,0 MB
VB_fol_104v_small.jpg	23,7 kB
VB_fol_105r.jpg	309,5 kB
VB_fol_105r_big.jpg	2,2 MB
VB_fol_105r_small.jpg	26,9 kB
VB_fol_105v.jpg	332,5 kB
VB_fol_105v_big.jpg	2,2 MB
VB_fol_105v_small.jpg	25,1 kB
VB_fol_106r.jpg	300,2 kB
VB_fol_106r_big.jpg	2,1 MB
VB_fol_106r_small.jpg	25,8 kB

Note that you will have to find a suitable compromise with regard to image size/resolution and actual performance of the edition on the Web: bigger images may be better for detail inquiries, but they take longer to load and to navigate, especially on low speed Internet connections. The current beta of the Digital Vercelli Book is geared more on navigation speed than image detail.

1.2 Double Side Images

These are double side images of the manuscript, showing the verso of a folio and the recto of the following one:





- **standard** resolution: these images must be named "[left folio number]-[right folio number].jpg" (e.g. "104v-105r.jpg"), where "[left_folio_number]" and

"[right_folio_number]" are the names of images of the single facing folios, respectively the "verso" (on the left hand side) and the "recto" (on the right hand side); the double image name is therefore composed by the names assigned to the image of each single folio;

- **high** resolution double folios images: these are necessary to activate the magnifier in the Bookreader view (the proportion must be the same as the normal ones) and must be named "[left_folio_number]-[right_folio_number]_big.jpg" (e.g. "104v-105r_big.jpg"), where "[left_folio_number]-[right_folio_number]" is the name of the corresponding double folio normal image.

Important: for the first and last page of the manuscript and for all the pages that do not have a "partner", it is still mandatory to have the images for the double view; in this case they must be named [left_folio_number]-.jpg (e.g. "135v-.jpg") if it doesn't have the right partner, and -[right_folio_number].jpg (e.g. "-103r.jpg") if it doesn't have the left partner (note the presence of hyphens). The same rule applies to the high resolution images, they must be named [left_folio_number]-_big.jpg (e.g. "135v-_big") and -[right_folio_number]_big.jpg (e.g. "-103v_big.jpg").

Examples:

 VB_fol_104v-VB_fol_105r.jpg	584,8 kB
 VB_fol_104v-VB_fol_105r_big.jpg	1,1 MB
 VB_fol_105v-VB_fol_106r.jpg	632,5 kB
 VB_fol_105v-VB_fol_106r_big.jpg	1,4 MB

1.3 Hotspot Images

These alternative images for specific areas of a manuscript folio, e.g. a more detailed version produced thanks to virtual restoration, possibly to be accompanied by a textual note. The only requirement is that they are named in the same way as they are referred to in the XML file, it is recommended that they are included in the <back> section of a text:

```
<back>
  <div type="hotspot">
    <div facs="VB_hs_106r_01" xml:id="VB_div_hs_106r_01">
      <p>The scribe here wrote a word, 'prowode', that he
        later corrected in 'wolde' by erasing the first
        three letters and adding an 'l'. In this image
you
        can see the contours of the erased letters.</p>
      <figure>
        <graphic url="VB_106r_HS_01.jpg"/>
      </figure>
```

```

        </div>
    </div>
</back>

```

1.4 Other Images

It is not impossible that you have a set of images created for a specific purpose: in the Codice Pelavicino Digitale project, for example, all the notary graphic symbols have been collected to form a *signa tabellionis* database, and have been placed in a separate `images/signum` folder. Again this is easily done, provided that the correct links to images are provided in the XML documents.

2. TEI XML Transcription Files

2.0 TEI Header

[coming soon]

2.1 Facsimile

The `<facsimile>` element holds as many `<surface>`s as there are single side images. Each `<surface>` includes:

- a `@corresp` attribute pointing to the `@xml:id` of the corresponding `<pb/>` element

```
<surface xml:id="VB_surf_104v" corresp="#VB_fol_104v">
```

- a `<graphic>` element: this can include an `@url` attribute pointing to a specific file location, but at the present moment only `@height` and `@width` are used by EVT;

```
<graphic height="1800px" url="../../images/Vercelli-
Book_104V_S_300dpi.jpg" width="1200px"/>
```

- a number of `<zone>` elements: these are used to record the coordinates of image areas corresponding to text elements and other metadata information, to do so they require use of several attributes:
 - **ulx, uly, lrx, lry** cartesian coordinates of the image area
 - **rendition** two possible values: **Line** (for line-by-line text-image linking) and

- **HotSpot** (for image areas to be used in hotspots)
- **xml:id** the unique ID for the current <zone>
- **corresp** points to the corresponding <lb/> ID (i.e. line) in the text transcription

(OPTIONAL)

```
<zone corresp="#VB_lb_104v_01" lrx="1052" lry="211"
      rend="visible" rendition="Line" ulx="261" uly="156"
      xml:id="VB_line_104v_01"/>
```

Complete example:

```
<facsimile xml:id="VB_fac_dotr">

<surface xml:id="VB_surf_104v" corresp="#VB_fol_104v">
  <graphic height="1800px" url="immagini\Vercelli-
    Book_104V_S_300dpi.jpg" width="1200px"/>
  <zone corresp="#VB_lb_104v_01" lrx="1052" lry="211"
    rend="visible" rendition="Line" ulx="261" uly="156"
    xml:id="VB_line_104v_01"/>
  <zone corresp="#VB_lb_104v_02" lrx="1072" lry="263"
    rend="visible" rendition="Line" ulx="257" uly="209"
    xml:id="VB_line_104v_02"/>
  [...]
  <zone corresp="#VB_lb_104v_23" lrx="1084" lry="1318"
    rend="visible" rendition="Line" ulx="262" uly="1269"
    xml:id="VB_line_104v_23"/>
  <zone corresp="#VB_lb_104v_24" lrx="1104" lry="1372"
    end="visible" rendition="Line" ulx="260" uly="1316"
    xml:id="VB_line_104v_24"/>
</surface>
```

For more information about digital facsimiles in TEI XML see chapter [11 Representation of Primary Sources](#) of the Guidelines.

2.2 Parallel Transcription

The [Parallel Transcription method](#) is the most popular and recommended way to couple a (semi-)diplomatic transcription with digitized manuscript images. This is what EVT expects to find in a TEI XML document created according to this method:

- <text> follows the <facsimile> element and holds the following items:

- `<front>` front matter (information about a text, regesto (diplomatic text summary), etc.)
- `<body>` actual text
- `<back>` back matter (notes, comments, etc.)
- a `<div>` is used to hold a whole text inside a `<body>`: it can hold further sub-`<div>`s, but it is essential that the parent one include the following attributes with the specified values:
 - `type` "prose" or "verse"
 - `subtype` "edition_text" (for the transcription text); note that this is mandatory even if it is the only available value at the moment
 - `n` the label to be showed in the viewer
 - `xml:id` unique ID of the text

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

- `<pb/>` elements are used to mark up folio sides: they must include the `@n` and `@xml:id` attributes, respectively to show the correct folio number and to enable text-image linking at the page level (see above *1.1 Single Side Images at standard resolution*). If you want to activate the Bookreader view, the `@n` attribute values must end with "r" (for "recto") or "v" (for "verso").

```
<pb n="104v" xml:id="VB_fol_104v"/>
```

- `<lb/>` elements are used to mark up manuscript lines: they must include the `@n` and `@xml:id` attributes, respectively for line numbering and text-image linking at the line level.

```
<lb facs="#VB_line_104v_07" n="7" xml:id="VB_lb_104v_07"/>
```

- different **edition levels** in the same TEI document are managed through a combination of transcriptional and editorial elements:
 - the **diplomatic** level is encoded using the `<hi>`, `<abbr>`+`<am>` and `<orig>` elements; at the character level, if a `<charDecl>` is present, using the `<mapping type="diplomatic">` character values inside each `<char>` (or `<glyph>`) element;
 - the interpretative level is encoded using the `<expan>`+`<ex>` and `<reg>` elements; at the character level, if a `<charDecl>` is present, using the `<mapping type="normalized">` character values inside each `<char>` (or `<glyph>`) element; `<sic>`/`<corr>` pairs are possible, but not necessary if a full critical edition is envisaged.

Note that all the editorial elements are usually inserted in <choice> elements: this is mandatory for word-level pairs.

Full example:

```
<text>
<body>

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

  <pb n="104v" xml:id="VB_fol_104v"/>

  <l n="1"><lb facs="#VB_line_104v_07" n="7" xml:id="VB_lb_104v_07"/><hi
rend="init3.1"><g ref="#Hunc"/></hi><hi rend="cap">W</hi>æt ic <g
ref="#slong"/>wefna c<g ref="#ydot"/><g ref="#sins"/>t secgan wylle</l>
  <l n="2"><choice>
    <sic>hæt</sic>
    <corr resp="Grein">hwæt</corr>
  </choice>
  <choice>
    <orig>mege mætte</orig>
    <reg>me gemætte</reg>
  </choice><lb facs="#VB_line_104v_08" n="8" xml:id="VB_lb_104v_08"/>to
midre nihte</l>
  <l n="3">syðpan <choice>
    <orig>reord b<g ref="#eenl"/>r<g ref="#eenl"/>nd</orig>
    <reg>reordberend</reg>
  </choice> reste wunedon<orig><pc type="metrical">.</pc></orig></l>
  <l n="4"><lb facs="#VB_line_104v_09" n="9" xml:id="VB_lb_104v_09"/>puhte
me þæt ic <choice>
    <orig>ge <g ref="#sins"/>awe</orig>
    <reg>gesawe</reg>
  </choice>
  <g ref="#sins"/>yllicre treow</l>

  [...]

  <l n="156">ælmihtig <name type="religion"><choice>
    <orig>god</orig>
    <reg>God</reg>
  </choice></name> þær hi<g ref="#sins"/> eðel wæ<g ref="#sins"/>
<g ref="#colmidcomposit"/></l>

</div>
</body>
</text>
```

2.3 Embedded Transcription

The [Embedded Transcription method](#), for which EVT provides experimental support thanks to an EADH small grant, is an alternative way to attach a transcription to an image, be it on parchment or other types of support. This is what EVT expects to find in a TEI XML document created according to this method:

- the TEI document must be based on one or more `<sourceDoc>` elements, with no `<facsimile>` or `<text>`;
- cartesian coordinates for text-image link at line level are once again placed in the relevant attributes of `<zone>` elements (see the preceding section);
- folio sides marked with the `<surface>` element that are direct children of `<sourceDoc>` or `<surfaceGrp>` must include the `@xml:id` attribute. If you want to activate the Bookreader view, the `@xml:id` attribute values end with "r" (for "recto") or "v" (for "to");
- transcription text goes into `<line>` elements within or at the same hierarchy level of `<zone>` elements.

Full example:

```
<sourceDoc xml:id="DOTR_ET">
  <surface xml:id="VB_fol_104v" n="104v">
    <graphic height="1800px" url="immagini\Vercelli-Book_104V_S_300dpi.jpg"
      width="1200px"/>
    <zone lrx="1108" lry="560" rend="visible" rendition="Line" ulx="210"
      uly="459">
      <line n="7" xml:id="VB_txtline_104v_07"><hi rend="init3.1">
        <g ref="#Hunc"/></hi><hi rend="cap">W</hi>æt ic <g
          ref="#slong"/>wefna c<g ref="#ydot"/><g ref="#sins"/>t secgan
          wylle
        <choice>
          <sic>hæt</sic>
          <corr resp="Grein">hwæt</corr>
        </choice>
        <choice>
          <orig>mege mætte</orig>
          <reg>me gemætte</reg>
        </choice></line>
      <line n="8" xml:id="VB_txtline_104v_08">to midre nihte syðþan <choice>
        <orig>reord b<g ref="#eenl"/>r<g ref="#eenl"/>nd</orig>
        <reg>reordberend</reg>
      </choice> reste wunedon<orig><pc type="metrical">.</pc></orig></line>
    </zone>
    <zone corresp="#VB_txtline_104v_09" lrx="1031" lry="610" rend="visible"
      rendition="Line" ulx="257" uly="558" xml:id="VB_msline_104v_09"/>
```

```

<line facs="#VB_msline_104v_09" n="9" xml:id="VB_txtline_104v_09">þuhte me
    þæt ic <choice>
      <orig>ge <g ref="#sins"/>awe</orig>
      <reg>gesawe</reg>
    </choice>
    <g ref="#sins"/>yllicre treow<choice>
      <orig>onlyft</orig>
      <reg>on lyft</reg>
    </choice></line>

[...]

<zone corresp="#VB_txtline_105r_32" lrx="987" lry="1476" rend="visible"
  rendition="Line" ulx="121" uly="1422"
xml:id="VB_msline_105r_32"/>
<line facs="#VB_msline_105r_32" n="32"
  xml:id="VB_txtline_105r_32"><choice>
  <orig>ge namon</orig>
  <reg>genamon</reg>
</choice>
  <damage type="faded">hie</damage> þær ælmihtigne <choice>
    <orig>god</orig>
    <reg>God</reg>
  </choice> <g ref="#aacute"/>hofon hine of ðam</line>
</surface>

</sourceDoc>

```

3. Configuring EVT

The Configuration File

If you open the `evt_builder-conf.xml` file that is inside the `builder_pack/modules` folder, you will be able to configure the existing parameters to customize the output: for instance, it is possible to disable the Bookreader view setting the `double_view` value to `false()`; or to indicate the different edition levels for the current edition modifying the variable `edition_array` as indicated.

All options are briefly described by means of comments in the file itself and should be understandable for anyone having a little experience with XML/XSLT documents.

Fine-tuning the results

The Bookreader navigation needs the identification of page pairs, which is based on the `@n values` of the `<pb/>`/`<surface>` elements. If you want EVT Builder to properly recognize the page pairs, you need to use numbers instead of letters for such values, followed by “v” for *verso* and “r” for *recto*. We do realize that this is a bit restrictive, but we are working on a better way to recognize

the pairs. It is possible, in any case, to manually modify the `structure.xml` file (in the `output_data` folder) to correct possible mistakes: you'll find a `<pages>` element inside that file, with `<pair>`(s) containing two `<pb>`(s), one for the image in the left-hand-side, one for the image in the right-hand-side. Note that every time you transform the XML file with EVT Builder, the structure file will be overwritten and you will lose your changes.

4. Step by step instructions

How to use the EVT builder pack, step by step:

1. Put the images files of every manuscript folio in the folder "`data/input_data/images`". The images must be organized as explained above:

- single side images go in the `data/input_data/images/single` folder
- double side images go in the `data/input_data/images/double` folder
- hotspot images go in the `data/input_data/images/hotspot` folder

2. Put the TEI XML file of the transcription and all other related files if necessary (e.g. schema, entities) in the `data/input_data/text` folder. You can use a sub-folder e.g. for the schema, provided that the correct path to the schema is included in the TEI XML documents.

3. Delete, or move into another folder on your hard drive, all content of the `data/output_data/` folder and the `index.html` file.

4. Start the edition creation process by applying the `evt_builder.xsl` stylesheet, which is in the `builder_pack` folder, to the TEI XML transcription document. To perform the XSLT transformation using Oxygen XML Editor:

- Open the XML file with Oxygen;
- Select the "Document/Transformation/Configure a transformation scenario" menu voice;
- Choose the "XML transformation with XSLT" scenario;
- Clone a basic scenario such as "TEI P5 XHTML";
- Type or select the path to the "`evt_builder.xsl`" file;
- Choose one of the **9.x** versions of the Saxon XSLT transformer engine (for example Saxon-EE 9.1.5.1)
- Save and apply the transformation scenario.

Known bugs

[[premesssa generale]]

[[bug sul <div>]]