An Open Source XML Publishing System
<http://teipublisher.sf.net>

Slavic Digital Text Workshop: July 5th - July 8th
Brown Bag Demonstration: July 8th
University of Illinois
project participants

• Amit Kumar
  – University of Illinois, USA

• Susan Schreibman
  – University of Maryland, USA

• Stewart Arneil, Martin Holmes
  – University of Victoria, Canada

• John Walsh
  – University of Indiana, USA
functionality & features

<http://teipublisher.sf.net>
features I

• upload and delete documents;
• analyze XML document instances to determine elements for search and browse;
• develop consistency in ontology and refine ontology development;
• index and store XML documents for efficient search and retrieval;
features II

• generate search/browse, results and metadata display pages for end users;
• create inter and intra document links;
• partition the repository into collections;
• create repository backups;
• change the look of the interface;
• associate XSL transformation scripts and CSS stylesheets.

<http://teipublisher.sf.net>
architecture

• teiPublisher has two components
  – teiWizard
    • java Application that analyzes the XML documents and with user feedback generate an XML repository.
  – teiRepository
    • a server and web application that allows search/browse and repository administrative functions.

<http://teipublisher.sf.net>
architecture

<teiWizard> client
  Document Validation
  Analysis  Query Generator

<teiRepository> web app
  Add Documents To database
  Generate Browse and Search Pages

Browse  Search  File list
Xml Configuration Files

Lucene and eXist

XSL Style Sheets

<http://teipublisher.sf.net>
<teiWizard>

XML Analysis comprises of:
- Indicating the missing nodes.
- Indicating the ontology mismatches.
- The analysis is based on the set of rules.
- Rules comprise of XML schema fragments for the node.

<http://teipublisher.sf.net>
<teiRepository/>

<http://teipublisher.sf.net>
demo
validating document instances in <teiWizard>
selecting the search function <teiWizard>

Steps:
1. Select Node: clicking on the bullet to the left of the tree node opens up the DTD tree. Continue clicking on bullets to expand the DTD tree to select the desired node(s) which will form the basis of the browse page. The browser page may be made up of multiple elements, but the steps listed below must be carried out for each element.
2. Once you have located the element to be browsed on, click on the element name. The XPath statement that node is automatically generated in the Node XPath box.
3. Unselecting Nodes: To unselect a node, go back to the node tree, right click on the node in question, and select Unselected node from the pop-up box.
4. Constraint Via an Attribute: allows you to add an attribute constraint to the XPath by clicking on the selected element/attribute.

DTD tree structure:
- Starts With
- Contains
- Equals

<http://teipublisher.sf.net>
selecting browse and search nodes in &lt;teiWizard&gt;

**Step 4 of 8: Select Browse Nodes**

1. **Select Node:** clicking on the bullet to the left of the TEI 2 node opens the DTD tree. Continue clicking on bullets to expand the DTD tree to select the element(s) which will form the basis of the browse page (the browse page may be made up of multiple elements, but the steps listed below must be carried out for each element in turn).

2. Once you have located the element to be browsed on, click on the element name. The XPath statement to that node is automatically generated in the Node XPath box.

3. **Unselecting Nodes:** to unselect a node, go back to the node tree, right click on the node in question, and select Unselect Node from the pop-up box.
Ranking search results

1. The next Wizard is the Search Wizard. Before going on to choose search nodes, you must decide how you want search results displayed or ranked. For example, you may allow users to search anywhere in the body of the document for a particular name, such as James Joyce (via a free

Sort by (XPath of the node)

`/TEI.2/teiHeader/fileDesc/titleStmt/title`
generate repository  <teiWizard>

Step 7 of 8: Generate Repository

1. Generate Repository: generates your customized teiRepository. Please be patient, this may take several minutes, particularly if your repository is large.

Click the Generate Repository button to generate your repository.

Generate Repository

<http://teipublisher.sf.net>
run server <teiWizard>

**Step 8 of 8: Run Server**

1. **Run Server**: by pressing this button, you will run a Web server on your computer. After you start the server, open a browser window and type: `http://127.0.0.1:8080/[project name]/admin` to see your repository (the `[project name]` is the name you chose in Step 2, Repository Configuration).

2. Alternatively, you can move the generated teiRepository (located in the `[workspace/project name]`)

   ![Run Server Dialog](<http://teipublisher.sf.net>)

run web application server
web design via
<teiRepository>

<http://teipublisher.sf.net>
database option
lucene
both lucene and eXist

<http://teipublisher.sf.net>
Repository Management

Add Documents

Found 92 matching documents in the database
Select a file to insert into the teiPublisher repository:

Select the database you would like the document to be added to
Add to Lucene Index ▼  □  Validate

store

Add Document Images

File:  Browse  submit

Click here to see list of document images.

Generate HTML

Selected Style Sheet: tei2html

Generate All

transform xml to html

<http://teipublisher.sf.net>
### Repository Status

<table>
<thead>
<tr>
<th>Select</th>
<th>File Name</th>
<th>Lucene Index</th>
<th>eXist Index</th>
<th>Html Transformed</th>
<th>Transform</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑</td>
<td>acrelius_founding.xml{html}{xml}</td>
<td>L</td>
<td>☑</td>
<td>no</td>
<td>Transform</td>
</tr>
<tr>
<td>☑</td>
<td>arabic.xml{html}{xml}</td>
<td>L</td>
<td>☑</td>
<td>no</td>
<td>Transform</td>
</tr>
<tr>
<td>☑</td>
<td>ashbridge_account.xml{html}{xml}</td>
<td>L</td>
<td>☑</td>
<td>no</td>
<td>Transform</td>
</tr>
</tbody>
</table>

Available XSLT stylesheets

Available XSLT stylesheets for documents in the repository.

Visit [http://teipublisher.sf.net](http://teipublisher.sf.net)
Customize <teiRepository> 

The Archive

<http://teipublisher.sf.net>
### Generate website pages

```html
<?
Generate Index Page.
Generate Browse Page.
Generate Search Page.
```

### Upload Images for use in website pages

```html
<?
Upload Image
To view the list of images available for site design. Click Here.
```

---

<http://teipublisher.sf.net>
advanced functions <teiRepository>

Back Up Documents

Back up teiPublisher documents on a regular basis.

Back up Now

Generate a zip file

Stylesheet Management

The following stylesheets are available:

<table>
<thead>
<tr>
<th>Label:</th>
<th>Stylesheet path:</th>
</tr>
</thead>
<tbody>
<tr>
<td>tei2html (Currently Selected)</td>
<td>xsl/tei2html/xslt/tei2html.xsl</td>
</tr>
</tbody>
</table>

Change the Default Stylesheet:

| tei2html | change |

Upload New Stylesheet:

<table>
<thead>
<tr>
<th>Label:</th>
<th>Set as Default:</th>
<th>Browse...</th>
</tr>
</thead>
</table>

Upload custom XSL stylesheets

<http://teipublisher.sf.net>
limitations
<teiPublisher>
• Search results return documents, not XML fragment.
  – i.e. can not return div[n] of a document.

• Requires Multiple document. Can not search for XML fragments in one big document.
  – i.e. if you have only one (big) XML document, teiPublisher is not useful.
search engine
<teiPublisher>
<http://teipublisher.sf.net>
• teiPublisher provides Lucene and eXist as search engine choice.

Lucene: Full Text Search engine and stemmer.
eXist: Native XML Database search engine.
• Lucene is Full Text Search Engine.
  – Lucene is not XPATH/XML aware.
  – teiPublisher indexes the XML nodes.
  – Allows compound searches like
    • (title="Time") AND (author="Emily Dickinson")
      AND (text="burial")
  – Use Lucene if the document structure is too complex.
  – Use Lucene if the number of documents is large.
• eXist provides XPATH search support.
  – XPATH based searches can be more complex, and return XML fragments
    • /TEI.2[./teiHeader/titleStmt[contains(.//title='Time')]]/teiHeader/titleStmt/author
      » Return the author node of the XML document where the title node contains ‘Time’.
  – XPATH searches can get XML fragments across the document boundaries
    – //p[../persname="Thomas Macgreevy"]
      » Return all the paragraphs from any document that have a person with a name “Thomas Macgreevy”.

<http://teipublisher.sf.net>
modify results page.
<teiPublisher>
• The results page can be modified to display custom element or attribute
  – By default it displays “title” by “author”.

Number of Results Matching your Query: 7

- First Voyage to Virginia (1584) By Arthur Barlowe 1550-1620. {View Document}
- Florida of the Inca By Garcilaso de la Vega, el Inca 1539-1616. {View Document}
- Founding of New Sweden By Israel Acrelius 18th Century. {View Document}
- Founding of St. Augustine By Francisco López de Mendoza Grajales 16th Cen
- Four Ages of Man By Anne Bradstreet 1612-1672. {View Document} {View XML}
- Francis Yeardley’s Narrative of Excursions into Carolina, 1654 By Francis Yea
- Fruits of Solitude By William Penn 1644-1718. {View Document} {View XML}
<xsl:template name="singleresult">
  <xsl:param name="fname" select="" /> 
  <div class="singleresult">
    <xsl:value-of select="document(string($fname))/TEI.2/teiHeader/fileDesc/titleStmt/title" />
    By
    <xsl:value-of select="document(string($fname))/TEI.2/teiHeader/fileDesc/titleStmt/author" />
  </div>
  <a href="#" onclick="showDocument('');">
    View Document
  </a>
  <a href="#" onclick="showXML('');">
    View XML
  </a>
</xsl:template>
For Installation Instructions; Documentation and tutorial visit http://teipublisher.sf.net