TEI < Text Encoding Initiative >

# Introduction to TEI

Lisa Spiro
October 2013

# Objectives

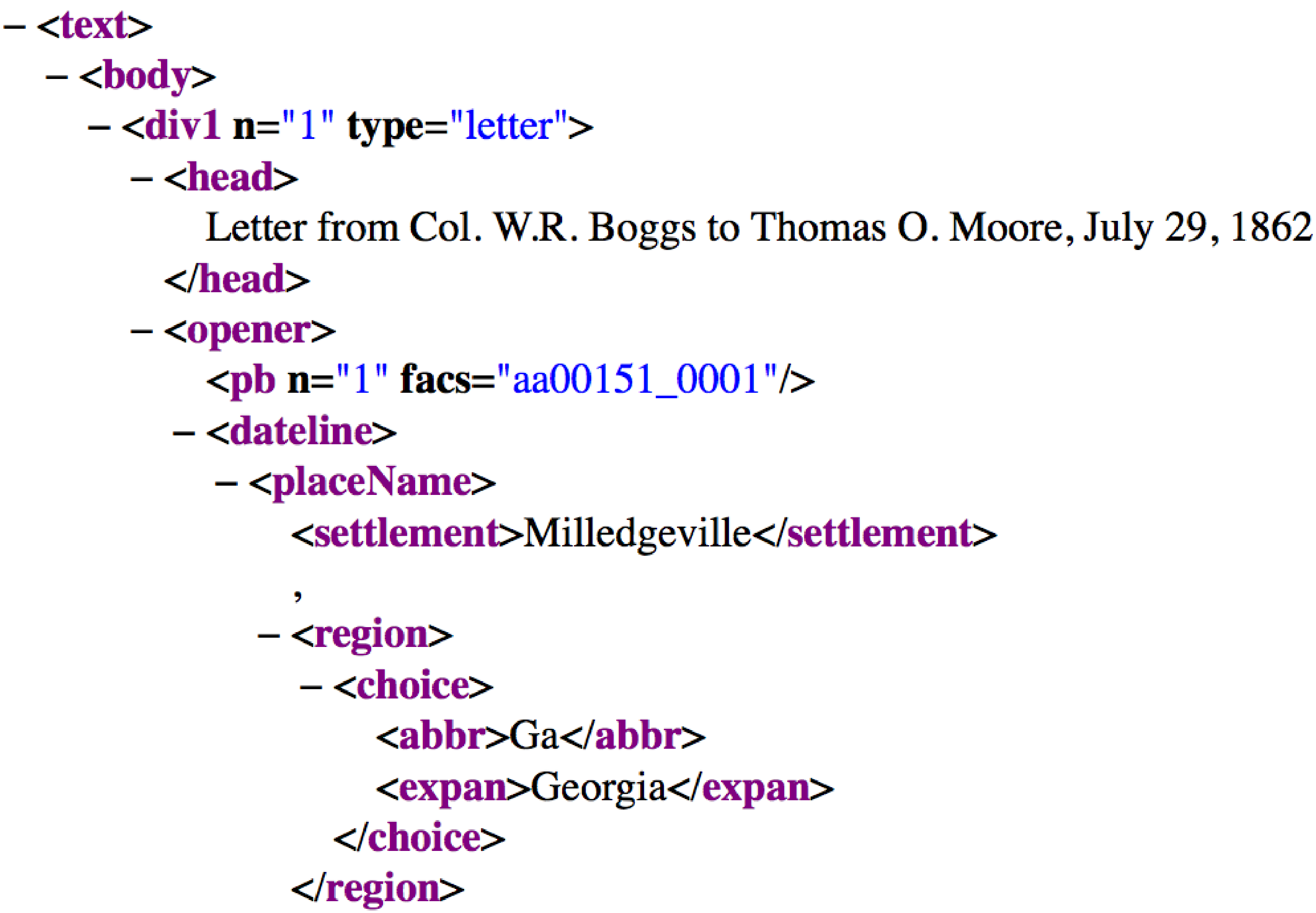
By the end of today’s workshop, you should:

- Know what TEI is and how it is used
- Comprehend the advantages and disadvantages of TEI
- Understand the difference between XML & HTML
- Understand basic rules governing the use of XML
- Understand some of the editorial choices that TEI encoders make
- Know some basic tags for encoding the structure of documents

# Exercise 1: Understanding the Structure of Documents

- View Letter from Letter from Col. W.R. Boggs to Thomas O. Moore, July 29, 1862
- Consider:
  - What do you notice about this document?
  - What features of this document would you want to encode?
  - How is this document structured?
- Mark on the document
  - Structure (paragraphs, etc)
  - Semantic features (names, etc)

# What TEI Looks Like



```
– <text>
   – <body>
      – <div1 n="1" type="letter">
         – <head>
              Letter from Col. W.R. Boggs to Thomas O. Moore, July 29, 1862
           </head>
         – <opener>
              <pb n="1" facs="aa00151_0001"/>
            – <dateline>
               – <placeName>
                    <settlement>Milledgeville</settlement>
                    ,
                  – <region>
                     – <choice>
                          <abbr>Ga</abbr>
                          <expan>Georgia</expan>
                       </choice>
                    </region>
```

What might be the advantages and disadvantages of using TEI as opposed to PDF or HTML?

# What is TEI?

- TEI= Text Encoding Initiative
- Guidelines for representing texts in electronic form.
- Separates content from presentation
- Includes guidelines for marking up:
  - Novels
  - Plays
  - Poems
  - Letters & manuscripts
  - Dictionaries
  - Linguistic corpora
  - Etc.

# Why do we need TEI?

- Support analysis of texts
- Make explicit features of a text so that they can be processed by computer applications
- Support range of output formats (HTML, PDF, Braille reader, etc)
- Adhere to standard for creating scholarly editions
- Preserve documents for the long-term

# TEI Is an XML-Based Markup Standard

- XML, or Extensible Markup Language= a meta-language that
  - provides rules for encoding documents in machine (and human) readable form
  - offers syntax used to define and create markup languages
- XML is…
  - Cross-platform
  - A common, standards based approach for structuring and storing information
  - A group of related technologies for processing and publishing information
- TEI is one of 100s of XML “applications”

# Structure vs. Presentation: XML vs. HTML

**Xavier Xylophone**

*Exuberant XML*

Xpert Boox

# HTML Version

```
<html>
    …
    <body>

        <b> Xavier Xylophone</b><br>
        <i> Exuberant XML</i> <br>
        Xpert Boox

    </body>
</html>
```

# XML Version

```
<?xml version=“1.0” encoding="UTF-8"?>
     <book>

             <author type=“primary”>
             Xavier Xylophone
             </author>
             <title> Exuberant XML</title>
             <publisher>
             Xpert Boox
             </publisher>

     </book>
```

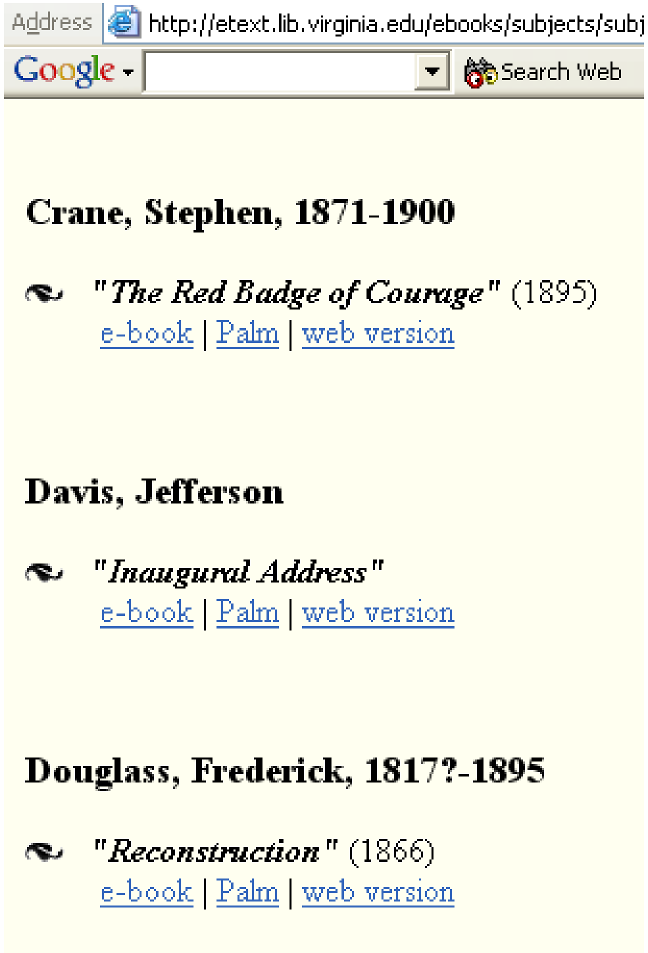
[Note: This is not TEI]

# Advantage of XML: Semantic & Structural Richness

- XML enables one to make explicit the structural features of a document
  - Chapters, paragraphs, etc.
- XML enables one to make explicit the semantic features of a document
  - Personal names, place names, dates
- The XML markup can then be used to search, retrieve, and display information

# Advantage of XML: Reusability

- “Build once, use many”
- Separates presentation from content
- Multiple outputs possible, e.g.:
  - Web
  - e-book
  - Pdf
  - Braille reader
  - Database of personal names
  - Index



**Crane, Stephen, 1871-1900**

*"The Red Badge of Courage"* (1895)
e-book | Palm | web version

**Davis, Jefferson**

*"Inaugural Address"*
e-book | Palm | web version

**Douglass, Frederick, 1817?-1895**

*"Reconstruction"* (1866)
e-book | Palm | web version

# Advantage: Sustainability

- Non-proprietary, open standard
- Well-supported
- Human and machine readable
- Platform and software independent
- Recommended for digital preservation

# Advantage: Information Exchange/ Interoperability

- Supports exchange of data between different systems and applications

- Can easily convert from one standard/ format to another, e.g.
  - TEI ←→ Open Office
  - TEI ←→ ePub

  See for instance OxGarage, http://oxgarage.oucs.ox.ac.uk:8080/ege-webclient/

# How do you produce XML?

- XML permits two kinds of documents:
  - "**well-formed**," which must conform to the rules described above (otherwise the browser or processor will balk).
  - "**valid**," which conform to those rules AND parse against a schema (such as TEI). The schema constrains logical relationships among elements.
- You can create XML files
  - By using an editor such as Oxygen
  - By hand, in a plain text editor (not recommended)
  - Automatically, through software

# XML Rules for Well-Formedness, I

- Elements that contain data must have **start tags and ending tags**:
  <sample> [...] </sample>.

- **Empty tags must be closed**. If a tag contains no data and therefore takes no closing tag (e.g., for a page or line break or an image), then embed the closing within the tag itself:
  <br />
  or provide a closing tag: <br> </br>

- Element and attribute names are **case sensitive**.

# XML Rules II

- **All elements must be properly nested.** Elements should not overlap.

  Bad Nesting

  ```
  <p> Do not <b>improperly nest tags.</p></b>
  ```

  Good Nesting

  ```
  <p> Do not <b>improperly nest tags.</b> </p>
  ```

# XML Rules III

- **All XML documents must have a root element.**

<?xml version="1.0" encoding="UTF-8"?>

- **All attribute values must be wrapped in quotation marks.** For instance, you should use:
  <pb n=“1”>
  rather than
  <pb n=1>

- **No isolated markup-start characters (< or &) can occur in your text data**. They must appear as the entities &lt; and &amp;

# Exercise 2: Find the errors

```
<author type=primary>
Xavier Xylophone

<title> <Exuberant XML
<publisher>
Xpert Boox
</title>
</PubLisheR>
```

1) Must have XML declaration, e.g.

   <?xml version="1.0“?>

2) Must have root element, e.g. <book>
3) Must put attributes in quotations
4) Must close tags properly
5) Must nest tags properly
6) Must use consistent case.
7) < and & must be escaped

# Being Valid

- A valid XML document must conform to a DTD (old-school approach) or schema (such as TEI)
- Schemas provide the rules for encoding a document
  - Define the elements & their relationship to each other
  - Facilitate consistency and document interchange

# Valid XML: Examples of XML applications

- Encoded Archival Description
- Text Encoding Initiative
- MathML
- METS
- CNXML
- Mind Reading Markup Language (MRML)

See http://xml.coverpages.org/xmlApplications.html

# Components of an XML Document

- Elements
- Attributes
- Other components that we’ll set aside for now:
  - Entities
  - PCDATA & CDATA
  - Processing instructions

# Elements

Elements serve as the building blocks of XML.

An element consists of an opening and closing tag as well as the content within:

<cowboy>Roy Rogers</cowboy>

Each tag is wrapped in angle brackets; end tags have a forward slash before the element name.

# Attributes

If elements= nouns defining what something is, attributes= adjectives providing additional info about the elements

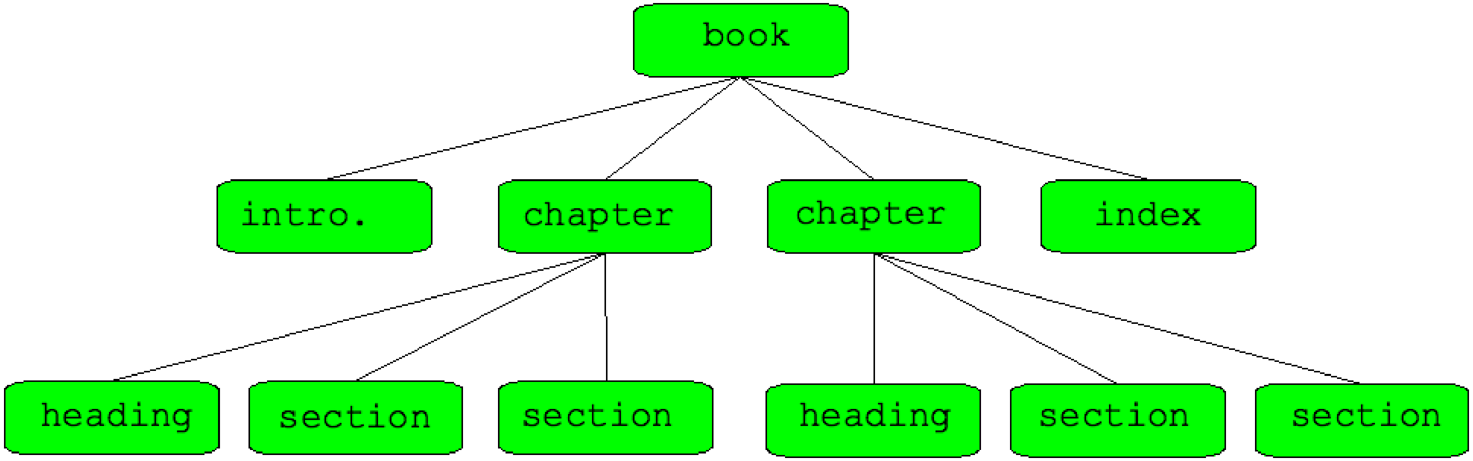
– e.g. distinguishing “singing,” “rodeo” and “urban” cowboys

<cowboy type="singing">Roy Rogers</cowboy>

attribute value

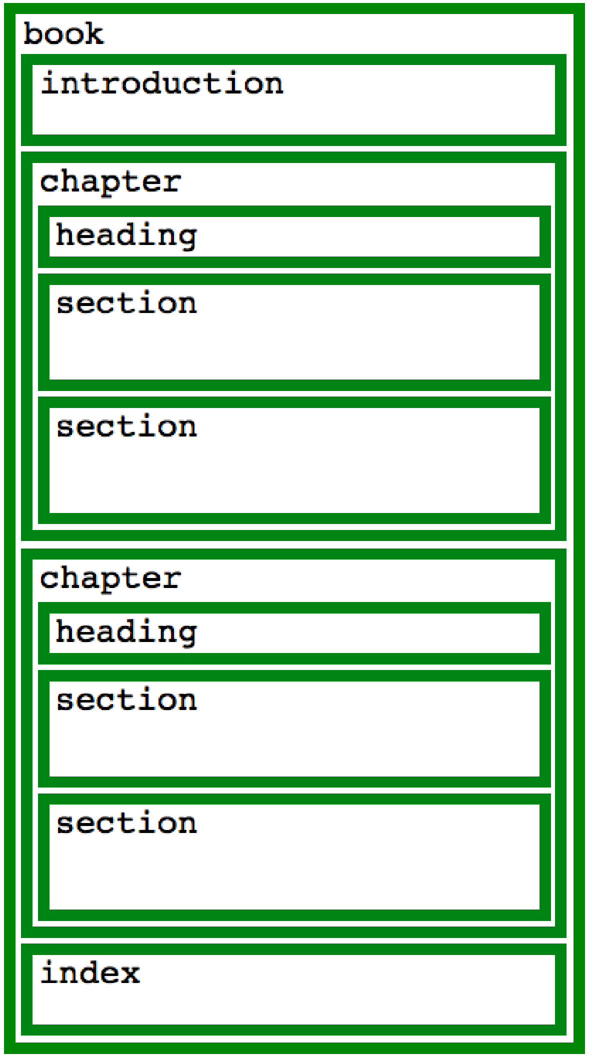
http://www.flickr.com/photos/alan-light/210260023/

# Structure: XML as a Tree



Bauman, Introduction to XML

# Structure: XML as “Boxes in Boxes”



Bauman, Introduction to XML

# Default structure of TEI document

```
<TEI>
    <teiHeader>
    [e.g. metadata about the digital file & source]
    </teiHeader>
    <text>
        <front> [e.g. preface, Table of Contents]
        </front>
        <body> [e.g. the main part of the text]
        </body>
        <back> [e.g. the appendix, index]
        </back>
    </text>
</TEI>
```

# Structural Markup

Divides text into meaningful parts to facilitate analysis, e.g.:

- Chapter: <div1 type=“chapter”>
- Section: <div2 type=“section”>
- Paragraph: <p>
- Stanza: <lg type=“stanza”>

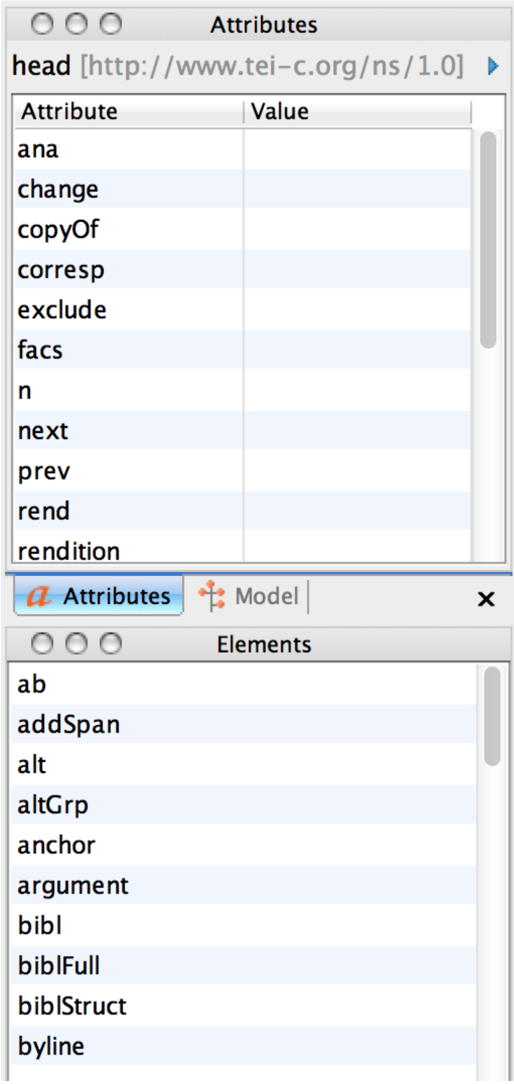
# Encoding <div>

- <div>: “text division”
- Can take the attribute “type,” e.g. type=“chapter” or type=“letter”
- Can be numbered, e.g. n=“1” for chapter 1
- Usually accompanied by a <head> (heading)
- Example:

```
<div1 type=“chapter” n=”1">
   <head>Chapter 1</head>
   <p>It was the best of times, it was the worst of
      times...</p>
</div1>
```

# Getting Started in Oxygen

- Launch Oxygen
- Go to File> New Document
- Go to the “From Template” tab and choose TEI P5– All. Hit OK
- To add an element or attribute, you can
  - select it from the sidebar menu
  - begin typing it; Oxygen will autocomplete



head [http://www.tei-c.org/ns/1.0]

| Attribute | Value |
|---|---|
| ana | |
| change | |
| copyOf | |
| corresp | |
| exclude | |
| facs | |
| n | |
| next | |
| prev | |
| rend | |
| rendition | |

ab
addSpan
alt
altGrp
anchor
argument
bibl
biblFull
biblStruct
byline

# Oxygen Basics

- To **indent lines** (so they don’t go on forever), select the format & indent button [second line of toolbar, middle/right] (or Just hit CTRL-Shift-p)

- To **place content inside an element**, highlight the phrase, CTRL click, choose “Refactoring,” then “Surround” (or just hit CTL E). Find the element you need from the menu.

# Exercise 3: Structure the Document

Let’s set up the basic structure for the letter.

- Replace <p>Some text here.</p> with <div1> (don’t forget to close it)
- Add attributes
  - Click on the attributes menu
  - Select n and type 1 in the box on the right
  - Select type and enter letter
- Add a <head>, e.g.

  <head> Letter from Col. W.R. Boggs to Thomas O. Moore</head>

# Marking Up the Beginning of Letters

- <opener> often contains
  - <dateline> may contain
    - <placeName>Houston</placeName>
    - <date when="2013-10-17">October 17</date>
    - <name type="monster">Oscar the Grouch</name>
- <salute> Dear Santa</salute>

# Exercise 4: Add the <opener>

- Open TEIWorkshopBoggsLetter.txt (source document)
- Add an <opener>, <placeName>, <date>, <name type=“person”> and <salute> to the Boggs letter
  - Copy and paste the first part of the text
- Note that you will need to escape out the & (an entity) by typing amp; after it, e.g. &amp;

# Milestone elements

- **<pb/>** (page break) marks the boundary between one page of a text and the next

  <pb> often takes these attributes:

  – facs="imagename" to associate page w/ facsimile image

  – n="page#" to designate page #

- **<lb/>** (line break) marks the start of a new line

- These are empty elements

# Exercise 5: Add paragraphs and milestones

- Add <p>s to your XML file to mark paragraphs

- Add <lb>s to mark line breaks (do just a few)

- Add <pb>s to mark page breaks, e.g.

```
<pb n="1" facs="001"/>
```

facs provides the filename for the page facsimile

# Handling Abbreviations

- **<choice>** **“**groups a number of alternative encodings for the same point in a text”
- **<abbr> “**(abbreviation) contains an abbreviation of any sort.”
- **<expan> “**(expansion) contains the expansion of an abbreviation.”

# Exercise 6: Abbreviations

- Encode an abbreviation in the document

```
<choice>
		<abbr>obt. serv</abbr>
		<expan>obedient servant</expan>
</choice>
```

# Closing a Letter

- <closer>: “groups together salutations, datelines, and similar phrases appearing as a final group at the end of a division, especially of a letter.”
  - <dateline>
  - <salute>
  - <signed>: signature

# Exercise 7: The Closer

- Add a closer to the letter

```
<closer>
        <salute>I remain your Excellency’s<lb/>
        <choice>
               <abbr>obt. serv</abbr>
                <expan>obedient servant</expan>
        </choice>
       </salute><lb/>
       <name type="person"> W. R. Boggs
       </name><lb/>
       Col of Arty C.S. Army
   </closer>
```

# Exercise 8: Validating TEI

- To make sure that the document validates against the TEI schema:
  - Select Validate document ☑ or go to Document > Validate > Validate
  - Correct any errors and re-validate

# How Do We Go From XML to HTML?

Apply an XSLT (Extensible Stylesheet Language Transformations) stylesheet, e.g. http://www.tei-c.org/Tools/Stylesheets/

# Exercise 9: Transforming the File

Let’s convert the file to XHTML

- Transform the file by hitting “Configure Transformation Scenario(s)” or go to Document > Transformation > Configure…
- Choose TEI P5 XHTML from the menu.
- Select “transform now”
- Voila-- the HTML file should open in a browser

# What is the TEI Header?

- Provides metadata about the TEI file (like a cataloging record or title page)
- Documents how the electronic text was created, its source, and any revisions.
- Important to provide this info for scholars, software processing texts, and cataloguers.
- Contains four parts:
  - **fileDesc:** the electronic file itself (required)
  - **encodingDesc**: how the text was encoded
  - **profileDesc**: classification information, e.g. keywords
  - **revisionDesc:** history of revisions to etext

# Exercise 10: Understanding TEI Headers

- Inspect the TEI Header in the sample file, TEIExerciseBoggsLetter.xml

# How to Publish TEI on the Web

- Render TEI for the web using:
  - CSS (Cascading Style Sheets)
  - XSL stylesheets
- Present online
  - Using capabilities of browser (limited)
  - Using an XML processor (e.g. Cocoon)
  - Using an XML database (e.g. exist)
  - Using an XML publishing platform (e.g. XTF)

# Disadvantages of TEI

- Time consuming and thus relatively expensive to create (although the work can be automated to some extent)
- It can be complex to publish TEI
- Markup can be inconsistent across and even within projects
- Imposes constraints (which is also an advantage)

# TEI in Action

- Search texts for words or phrases: http://www.marktwainproject.org/
- Create diplomatic & normalized transcriptions: http://tinyurl.com/4ycj3cv
- Make available different versions of a text: http://etext.virginia.edu/users/spiro/index.html
- Collate texts: http://v-machine.org/samples.php
- Create scholarly apparatus around text: http://tinyurl.com/43m3zck
- Analyze texts: http://tinyurl.com/4yf9x5g

# This Presentation Was Built On…

- TEI Guidelines: http://www.tei-c.org/release/doc/tei-p5-doc/en/html/index-toc.html
- Guidelines' Appendix on Elements: http://www.tei-c.org/release/doc/tei-p5-doc/en/html/REF-ELEMENTS.html
- TEI by Example: http://tbe.kantl.be/TBE/TBE.htm
- Brown Women Writer's Project's Resources: http://www.wwp.brown.edu/outreach/resources.html
- Oxford University training materials: http://tei.oucs.ox.ac.uk/Oxford/2009-04-galway/

  and http://tei.oucs.ox.ac.uk/Oxford/2009-07-dublin/