Outsourcing Complex Digitization: Lessons Learned

John Carlson, Mary Ann Lugo, and David Sewell
Electronic Imprint, University of Virginia Press

Converting a letterpress documentary edition:
- What tasks can be successfully outsourced to a conversion vendor?
- What tasks are better done in-house?
- What problems are typical, and what solutions address them?

The Tasks
The University of Virginia Press’s ROTUNDA imprint for digital scholarship has completed one large digitization of a printed documentary edition (*The Papers of George Washington*, 52 volumes, over 30,000 pages), and is in the midst of a second (*The Papers of Thomas Jefferson*, 33 volumes). The editions are being converted to P5 TEI-XML with minor schema customization. We outsource the digitization to data conversion vendors who are responsible for both rekeying and basic TEI tagging.

The Goals
Our *Papers of George Washington Digital Edition* (PGWDE) and the editions that will follow in our American Founding Era collection are intended as richly hypertextual digital editions offering many features not found in the letterpress editions. These rely on what we call “second-level tagging”: TEI markup that goes beyond capturing basic structure to include various types of metadata and linking: bibliographic data, document cross-references, expansions of abbreviations and short titles, etc. We therefore require both letter-for-letter accuracy in document rekeying, and close adherence to our guidelines in XML tagging.

Our Experience
In digitizing PGWDE, we assigned major responsibility to our conversion vendor not only for primary transcription but also for interpretive markup. This turned out to be a mistake, as error rates for tasks involving interpretation were invariably higher than those for straightforward document capture.

Our experience with PGWDE has led us to modify our procedures for the *Papers of Thomas Jefferson* (PTJ) and editions that will follow. We are simplifying our tagging demands on the vendor; moving much of the interpretive work in-house; and automating as many procedures as possible.

This Poster Presentation
The following pages outline
- areas of vendor success and vendor difficulty in our PGWDE conversion process
- solutions we have adopted aimed at minimizing errors in future digitization, and procedures we have used to identify errors in both transcription and markup in XML files received from conversion vendors
## Document Conversion Tasks

### Conversion Workflow

<table>
<thead>
<tr>
<th>Steps</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vendor Tagging / In-house Proofing (PGWDE)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Transcription</strong></td>
<td>Digitize primary text via double-rekeying to 99.995% accuracy</td>
</tr>
<tr>
<td><strong>Basic Markup</strong></td>
<td>Add TEI-XML tagging of structural units</td>
</tr>
<tr>
<td><strong>Internal Metadata and Linking</strong></td>
<td>Add source notes, standard abbreviation, and cross-reference markup</td>
</tr>
<tr>
<td><strong>Document Metadata</strong></td>
<td>Record author(s), recipient(s), dates for each document</td>
</tr>
</tbody>
</table>

---

## Distribution of Conversion Tasks (Original and Revised)

### Vendor Tagging / In-house Proofing (PGWDE)

- Transcription
- Markup
- Internal Metadata
- Document Metadata

### In-house Tagging (PTJ)

- Vendor Tagging / In-house Proofing (PGWDE)

---

## Hierarchy of Conversion Tasks

<table>
<thead>
<tr>
<th>Level</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>First-Level Features</strong></td>
<td><strong>Transcription</strong>: Digitize primary text via double-rekeying to 99.995% accuracy</td>
</tr>
<tr>
<td></td>
<td><strong>Basic Markup</strong>: Add TEI-XML tagging of structural units</td>
</tr>
<tr>
<td><strong>Second-Level Features</strong></td>
<td><strong>Internal Metadata and Linking</strong>: Add source notes, standard abbreviation, and cross-reference markup</td>
</tr>
<tr>
<td></td>
<td><strong>Document Metadata</strong>: Record author(s), recipient(s), dates for each document</td>
</tr>
</tbody>
</table>
Vendor Successes: Markup and Metadata

Transcription

Primary data capture of standard English text and numeric data has been excellent, meeting 99.995% target accuracy. This includes complex or repetitive data such as tables and back-of-the-book indexes.

Basic Structural Tagging

Vendor successfully tagged the main structural features of the print edition with minimal errors. Including:

- Front matter of volumes, including editorial apparatus and short-title and repository symbol lists.
- Document openers, closers, and body.
- Document back matter, including editorial source notes and editorial footnotes.
- Back matter of volume, including appendices and indexes.
- Unique structural features like document groups or enclosures (nested documents).

Straightforward Metadata

Straightforward or regular metadata was handled properly:

- Document dates were tagged with minimal errors.
  - Example: <dateline>Mount Vernon <date when="1772-01-27">Jany 27th 1772.</date></dateline>
- Regular internal metadata was also captured well, including tagging of repository abbreviations and short titles in the source notes.

In general, tagging of regular or well-structured features was accurate.
Vendor Problems: Markup and Metadata

**Transcription**

Anomalous errors included global transforms on particular words or names, or odd capitalization possibly the result of global search-and-replace or ad hoc macros:

- MacIver → MacIver; Van Horne → Van Home [similar names/words not affected]
- [from an editorial note]: “Lewis Replied On 27 Aug. 1788 That He Was Reluctant To Act As Gw’S Agent Because He Was Having Difficulty Obtaining Tenants For His Own Land…”

(These look like OCR errors. Problem: how to assure that vendor is actually using promised double-rekeying method?)

**Markup and Internal Metadata**

Vendor errors clustered in markup that required editorial interpretation. For instance, patterns of errors cropped up in tagging of internal document cross-references:

- Vendor mistakenly tagged some short titles as document cross-references.
- Vendor repeatedly tagged references to external sources as internal document cross-references.
- Vendor overgeneralized: “Cty” for “county” was globally converted to repository abbreviation <ref>Cty</ref>

**Document Metadata**

Vendor was not attuned to documentary editing conventions and metadata for authors/recipientsof correspondence showed distinct patterns of errors:

- Atypical names presented a problem. For instance, single corporate authors such as “Constable, Rucker, & Company” were incorrectly broken up and tagged as multiple authors. A letter “From Thomas, Lord Fairfax” might be tagged as by two authors.
- Third-party correspondence was not always handled properly. For instance, vendor overgeneralized by assigning “George Washington” as the default author or recipient for most documents without an obvious author or recipient.

In general, markup requiring editorial know-how presented some problems.
**Spot-checking to determine accuracy rate**

Unless you can afford to do (or pay for) a character-by-character proofreading of your digitized file, you will need to extrapolate accuracy based on a sample of the digitized text.

Our target accuracy rate of 99.995% equates to 5 errors per 100,000 characters, and that is the minimum quantity of text that must be sampled for a statistically valid estimate of whether the accuracy meets the target. The following table shows the 95% confidence interval for the number of errors that may occur in sampled text with an accuracy of 99.995% (calculation courtesy T. Finney, using GNU R stats package):

<table>
<thead>
<tr>
<th>n (characters sampled)</th>
<th>min</th>
<th>target</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>100,000</td>
<td>2</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>200,000</td>
<td>4</td>
<td>10</td>
<td>17</td>
</tr>
<tr>
<td>500,000</td>
<td>16</td>
<td>25</td>
<td>35</td>
</tr>
</tbody>
</table>

In other words, in a sample of 100,000 characters (~35 pages of the letterpress *Papers of George Washington*), more than 11 errors would suggest a vendor is not meeting stated accuracy.

**Foreign language**

If you have used @xml:lang on TEI elements in your data, extract that text and run it through a spell-checker corresponding to the language (may be less than useful for premodern data).

**Ad hoc errors**

A `grep`-type program or, ideally, XQuery, is useful for finding instances of a particular error pattern you have noticed in your data. For example, we located instances of the bizarre capitalization transform in Washington files (“He Was Reluctant To Act As Gw’S Agent Because He Was Having Difficulty”) using an XQuery similar to

```xml
for $par in collection("PGWDE")//tei:note/tei:p
 [count(tokenize(., "[A-Z][a-z]\d+\s[p{P}]")) gt 7]
 [not(matches(., "s[a-z]"))]
return $par
```

which searches for paragraphs containing more than 7 instances of strings like *Was* or *Gw’S* and no word tokens beginning with a lowercase letter. Obviously, thorough knowledge of XPath 2.0 and regular expressions is helpful in constructing searches of this type.
Document Metadata: Solutions

Generating XML Document Templates

1. Project editors enter document metadata into OpenOffice spreadsheet; File saved in Excel XML format
2. Custom XSL transformation converts Excel XML into TEI P5 XML, creating template (or “skeleton”) file
3. Vendor completes the template with document content with basic markup and linking

Advantages
1. Fewer errors since project editors, unlike vendors, possess specialized knowledge of content material
2. Tabular formatting of metadata facilitates proofing
3. Predetermined structure and identifiers allow vendors to insert cross-references with confidence
Effective Guidelines and Vendor Tools

Clear Specifications
Detailed markup instructions that use examples drawn from the text are essential for ensuring vendor accuracy. This is especially true when addressing metadata encoding.

Cross References
Within the editorial notes following each document or preceding document groups, two distinct types of cross-references regularly appear: external references that link to additional material and internal references that provide additional context or detail.

In general, the two types of cross references can be distinguished by their presence in various more typical citations giving details about the source, library, newspaper, article, or book in which the material originally appeared. Internal cross references will generally be less likely to include information like author, recipient, and (invariably) date in the text.

Elements, please use the document spreadsheet or look-up tool to ensure the accuracy of the target attribute values; these should correspond to the actual document. Example (from volume 13, pg. 158):

- The copy of this text volume that contains this document is deposited in the Library of Congress, Washington, D.C., D.C. However, this example:

  - The attribute given in this example would be replaced with the numbers in the actual encoding. When the target of such a cross is the print editor's fail to differentiate between two letters, to ensure the target attribute is not taken.

Document Look-Up Tool
Using the XML template files, which contain document metadata, it is possible to design XQuery tools to help vendors accurately encode cross-references in the text.

Jefferson Document Date Search
Date range for documents in database: 1760-01-14 to 1816-12-31

You searched for "1787-12-25" (1787-12-25) as exact search.

<table>
<thead>
<tr>
<th>Title</th>
<th>Author(s)</th>
<th>Recipient(s)</th>
<th>Document date</th>
<th>Doc ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>From John Adams</td>
<td>Adams, John</td>
<td>TJ</td>
<td>1787-12-25</td>
<td>TSN-01-02-0490</td>
</tr>
<tr>
<td>From Maria Cosway</td>
<td>Cosway, Maria</td>
<td>TJ</td>
<td>1787-12-25</td>
<td>TSN-01-02-0471</td>
</tr>
<tr>
<td>From Lafayette</td>
<td>Lafayette</td>
<td>TJ</td>
<td>1787-12-25</td>
<td>TSN-01-02-0472</td>
</tr>
<tr>
<td>From John Rutledge</td>
<td>Rutledge, John</td>
<td>TJ</td>
<td>1787-12-25</td>
<td>TSN-01-02-0473</td>
</tr>
</tbody>
</table>

Enter a date to search in one of these formats:
- YYYY-MM-DD
- Year Mon. Year
- Example: 1776-06-03 or 1778-03-12
- Examples: 5 Jun. 1776 or 12 March 1778 or 7 Mch. 1792
- Month should only be letters: March, August, Aug., Aug., Aug. may all work.
- Mch may be used for "March" (abbreviation used in Jefferson Papers)
- Month may be indicated with digit: 3 6 1776 = 3 June 1776
- Select "Exact date" (the default) to return only documents dated to that precise date.
- To find documents falling in a date range, enter a date within the range and choose the date range search.

1787-12-25
- Exact date
- Date range includes date

Submit date
Basic Structural Markup

Validating against a TEI schema is usually not enough to insure that vendors are following tagging specifications, unless it is highly customized to restrict options. We have found it more useful to validate against supplemental Schematron schemas. We created one for vendor use and others for checking returned files.

For example, we asked vendors to remove the trailing periods (.) from short-title abbreviations in the PGW front-matter lists. Here is our Schematron rule to check for violations of the rule (using an XPath 2.0-aware Schematron processor):

```
<sch:rule context="tei:label[@type='shorttitle']">
  <sch:report test="matches(., '.\.$')">
    There should be no period at the end of a short-title label!
  </sch:report>
</sch:rule>
```

Simple ad hoc XQuery scripts can be used for quick checks on structural markup. Here is code to extract all elements in the digitized files tagged as French (using @xml:lang="fr"):

```
declare namespace tei = "http://www.tei-c.org/ns/1.0";
for $french in collection("PGWDE")/*[@xml:lang="fr"]
return $french
```

Internal Metadata

As with structural markup, we use Schematron schemas supplemented by XQuery scripts to check that metadata such as short-title references and document cross-references is tagged properly. For example, here is a Schematron rule to warn if a PGWDE document cross-reference in a note points to the document it occurs in (which is almost never proper):

```
<sch:rule context="tei:ref[@type eq 'document']">
  <sch:report test="@target eq ancestor::*[starts-with(name(), 'div')]/@xml:id[1]/@xml:id">
    Warning: self-referential document target!
  </sch:report>
</sch:rule>
```