Blake/An Illustrated Quarterly
Issue Archive

Basic Technical Structure

= Components =

This resource is composed of the following components:

FILE TYPES
php (hypertext preprocessor, i.e., server side scripting)
xml (extensible markup language)
html (hypertext markup language)
xsl (extensible stylesheet language)
css (cascading style sheets)
js (javascript)
png, jpg, gif, ico (image)
pdf (portable document format)

SEARCHABLE DATABASE
Solr

SERVER CAPABILITIES
running PHP
URL redirection
running Solr (separate server)

= The HTML the User Sees =

HTML is the code which provides the text and structure of a webpage.

In a static website, each URL corresponds to a single HTML file which contains all the code used to display the webpage.

Our website is not static but dynamic: each webpage is generated when the user loads the URL, using multiple resources (about which more later).

= Files Referenced in the HTML: images, PDFs, CSS, Javascript =

The HTML which users see includes references to other files, such as:

1. *image files* which are displayed on the webpage.
   These are in the "img" directory, which has three sub-directories:

“general” (for a few images used throughout the website)
“inline” (for a few images used in place of typed characters, which appear in line with ordinary characters)
“illustrations” (the vast majority of images, used as illustrations in articles)
The filenames for images in the HTML users see are based on
the “n” attributes of “figure” tags (in XML articles)
or the “href” attributes of “img” tags (in HTML articles).

2. *PDF files* which are linked from the webpage but open at a separate URL.
These are in the “pdfs” directory.
The links to PDF files are generated using the filename/id of the corresponding article (46.1.bentley links to 46.1.bentley.pdf).

3. *CSS files* which determine the styling of the webpage (margins, borders, text color/size/font, etc.).
Our site uses one CSS file, “style.css”.
Some of the HTML articles also have their own CSS, which is extracted using PHP and loaded into the “head” tag in the HTML users see.

4. *Javascript files* which can change the page after it has loaded. (For example, Javascript shows and hides issues on the home page when decade headings are clicked.)
These are in the “js” directory.

= The Process that Generates the HTML User See =

PROCESS FOR INDEXES

index.php (home page, URL: “/”) is an index of TOCs (tables of contents, one for each issue).
articles.php (URL: “articles”) is an index of articles.
illustrations.php (not linked from anywhere) is an index of external illustrations (illustrations for an HTML article which are stored in separate HTML files).

XML files (“docs” directory) & HTML files (“html” directory)
⇑
(fields in xml/html files—e.g., volume/issue, date, cover image—are loaded and processed by…)
⇑
PHP (index.php, articles.php, illustrations.php)
⇑
URL redirected
For all websites, the root directory points to an “index.php” or “index.html” file.
We have set up a redirect from the URL “articles” to the file “articles.php”.
No redirect has been set up for “illustrations.php”—to view it, you have to navigate to that URL.

PROCESS FOR SEARCH

Solr database*
⇑
(Solr database accessed using…)
⇑
Solarium PHP (in “lib” directory)
⇑
(Solarium PHP loaded and processed by…)
⇑
PHP file (search.php)
⇑
URL redirected (“search” to “search.php”)

* See BQ-tools instructions for updating data in Solr database — note that BQ and BQ-dev use separate Solr databases.

PROCESS FOR ARTICLES

XML ARTICLES

XML file (in “docs” directory)	⇐ *(transformed by…)* ⇐	XSL (xsl/quarterly.xsl)
⇑		⇑
(loaded and processed by…)		*(loaded and processed by…)*
⇑		⇑
===========	PHP (xdoc.php)	===========
	⇑	
	URL redirected	

HTML ARTICLES

HTML file (in “html” directory)	⇐ *(transformed by…)* ⇐	XSL (xsl/quarterlyHtml.xsl *or* xsl/quarterlyHtmlToc.xsl *or* xsl/bonus.xsl)
⇑		⇑
(loaded and processed by…)		*(loaded and processed by…)*
⇑		⇑
============	PHP (hdoc.php)	===========
	⇑	
	URL redirected	

_____

We have set up a URL redirect based on certain patterns of numbers and letters.

XML articles: Everything from “1.1.druids” to “44.4.toc” (as well as “About,” “Contact,” and “Emend”) is redirected to the PHP file “xdoc.php” with a query string.

bq.blakearchive.org/1.1.druids ⇒ bq.blakearchive.org/xdoc.php?file=1.1.druids
bq.blakearchive.org/44.4.toc ⇒ bq.blakearchive.org/xdoc.php?file=44.4.toc
bq.blakearchive.org/About ⇒ bq.blakearchive.org/xdoc.php?file=About

HMTL articles: Everything from “45.1.bentley” onward (as well as everything starting with “bonus”) is redirected to the PHP file “hdoc.php” with a query string.

bq.blakearchive.org/45.1.bentley ⇒ bq.blakearchive.org/hdoc.php?file=45.1.bentley
bq.blakearchive.org/bonus.bentley ⇒ bq.blakearchive.org/hdoc.php?file=bonus.bentley

The PHP file takes the “file” value from the query string and looks it up in the appropriate directory and (if there is such a file and it has been approved to be published) loads the correct file. For example:

1.1.druids ⇒ xdoc.php *loads…* ⇒ docs/1.1.druids.xml
45.1.bentley ⇒ hdoc.php *loads...* ⇒ html/45.1.bentley.html

The XML/HTML file is then *transformed* using XSL. (The correct XSL file is loaded by the PHP.)

HTML articles require only a few changes, such as changing the filepaths for images and the targeted URLs for links, but the changes required depend on the original format, so there are different XSL files for different kinds of HMTL files.

Regular HTML articles (e.g., “45.1.bentley.html”) *transformed by*: quarterlyHtml.xsl
HTML tables of contents (e.g., “45.1.toc.html”) *transformed by*: quarterlyHtmlToc.xsl
Bonus HTML articles (e.g., “bonus.bentley.html”) *transformed by*: bonus.xsl

XML articles need to be converted from XML to HTML.

All XML *transformed by*: quarterly.xsl

= PHP includes =

One of the features of PHP is that when one PHP file is loaded, it can load and incorporate another PHP file. This is useful whenever you want to use the same code on multiple pages. Our “include” directory contains PHP files which are included by other files.

header.php
footer.php

Provide header and footer (logo, search bar, main navigation, copyright information, etc.)

head.php

Not to be confused with the visible header, the head provides metadata for the page and points to CSS and Javascript files.

analyticstracking.php

If on the public site, tracks users on Google Analytics.

functions.php

Provides complex commands (“functions”) used throughout the site. Also gives the values of key variables, including which issues should be published.

simple_html_dom.php

Used to load and process HTML files.

notfound.php

Included when a file is not found.