Collection and Exhibition Platforms

A number of software solutions are either in use at Harvard, being considered for use at Harvard, or being developed as new platform options for creating web based presentations and exhibitions for digital collections. On this page we try to capture that list, and provide a forum for adding pros and cons of these options - what are they good for and what are they not so good for.

We are collecting requirements here!

An architecture diagram putting platforms in ecological context can be found here: Data Provider Architecture

Notes from the April 9th, 2013 meeting are here.

Platforms compared on this page:

- Omeka
- Harvard Virtual Collections
- 3D
- Viewshare
- XTF
- Blacklight
- Exhibit 3

Omeka

http://omeka.org/

Omeka is a free, flexible, and open source web-publishing platform for the display of library, museum, archives, and scholarly collections and exhibitions.

Installations at Harvard

- Center for the History of Medicine - http://collections.countway.harvard.edu/onview
- Ruhleben: A Digital Exhibit - http://library.law.harvard.edu/digitalexhibits/ruhleben/exhibits/show/ruhleben/

Under Development

- Colonial North American Digital Collection - http://colonialnorthamerican.library.harvard.edu

Other Installations

- Universite de Namur: http://neptun.unamur.be
- More available at: http://omeka.org/showcase

Pros

- open source, supported and under continuous development
- easy for IT staff to install and maintain
- provides a modern visual experience, with search and browse capability
- funded IMLS grant to create an Omeka Curator Dashboard

Cons

- no current automatic import mechanism for Harvard collection metadata
- open source code is complex and not easy to extend
- Last build does not work with EADImporter plugin

Implementation

- Apache, PHP, MYSQL, open source

Plug-ins developed at Harvard

- Plug-in written by Jonathan Kennedy at HMS to enable linking to content external to Omeka
  - Extended by Dee Crema to enable linking to DRS images, video and documents

Other available plug-ins for Omeka:
2nd Generation Virtual Collections

http://hul.harvard.edu/ois/systems/ctools/overview-vc.html

Virtual Collections (VC) is a Harvard Library developed and supported service that can harvest descriptions and links from Harvard Library union catalogs (HOLLIS, VIA, and HGL) and provide a web-based search and display interface of these materials for the user.

Installations at Harvard

- Latin American Pamphlet Digital Collection (stand alone - basic UI)
- Loeb Music Library Digital Scores and Libretti (stand alone - basic UI)
- Expeditions and Discoveries: Sponsored Exploration and Scientific Discovery in the Modern Age (embedded - custom HTML)
- Daguerreotypes at Harvard (embedded - custom HTML)
- For a complete list of Virtual Collections, see - http://hul.harvard.edu/ois/systems/ctools/collections-list.html

Pros

- supported by Harvard Library
- search and browse capability
- automatically imports metadata from HOLLIS, VIA, and HGL, and re-harvests as necessary to keep collection up to date
- stand alone collection web sites or ability to embed search and browse within other sites
- built in OAI-PMH data provider supports bulk export and harvesting of metadata. Regularly harvested by OCLC, and optionally, DPLA
- Web based staff maintenance tool to delete items from a collection, and add subject terms.
- PIN login for staff

Cons

- These cons would be addressed by 2nd generation upgrades/replacement
  - outdated visual experience
  - underlying technology is aging
  - not open source
  - no current investment in enhancements
- minimal startup charge and on-going charge to library (?)
- This con needs to be addressed for multiple upcoming projects:
  - does not work with finding aid metadata

Implementation

- service operated by Library Technology Services, hosted on load balanced HUIT servers
  - Linux, Java, Tomcat, Tamino
- nightly automated loads of updated data from source records in Harvard catalogs
- fee for 2-3 days work to set up a new collection

Suite Spot (formerly 3D)

About 3D

3D (Discovery and Delivery of Digital collections) is a new exhibition tool developed at the Harvard Law School library that overcomes some of the perceived limitations of Omeka and Virtual Collections.

Installations at Harvard

- Oliver Wendell Holmes Digital Suite- http://library.law.harvard.edu/suites/owh/

Pros

- compatibility with external image servers (can display DRS images without uploading to platform)
- will be open source soon, supported by HLS
- supports all P1 law school collection discovery and display requirements:
  - Full text search of OCR text plus metadata plus tags
  - Faceted browse
  - User tagging of PDS objects at the page level
  - Discussion forum (vanilla forums open source project)
  - Will have an OAI data provider in the future to export data (not implemented yet)
  - Granularity of tagging, search, etc at the page level
  - Parses EAD finding aids and merges with DRS load reports to create 3D records with links to DRS files (with manual code tuning
for each EAD)
  • Can be customized for branding in new 3D instance

**Cons**

  • no current automatic import mechanism for Harvard collection metadata
  • authoritative metadata grows in 3D - gets out of sync with HOLLIS/VIA
  • Start up cost for a new collection = 50 hours design time ($3,000?) + 1-2 weeks week data wrangling ($3,000-$6,000)
  • Brand new system with only one production installation
  • local authentication/registration (no PIN login for admin tasks)

**Implementation**

**Web App**

  • apache/PHP/css/linux/MYSQL 5.x
  • laravel framework
  • vanilla forums
  • solr index (java), solr php library

**Operations**

  • ingest scripts for FA and DRS load report, includes MYSQL DB update, and solr indexing (manually run)
  • deploy new data files by MySQL dump, SCP to prod, run solr index on prod
  • operations support - restart apache, mysql, solr
  • needs data model documentation and training manual (not done yet)
  • may not work well with IE
  • deployment of a new collection requires hiring a designer (50 hours)
  • EAD mapping, or can do manual data entry of FA data (2 days to a week), depends on complexity of FA or # items in collection.

**Viewshare**

[http://viewshare.org/](http://viewshare.org/)

Viewshare is a free platform created by the Library of Congress for generating and customizing views (interactive maps, timelines, facets, tag clouds) that allow users to experience your digital collections.

**Installations at Harvard**

  • Museum of Comparative Zoology - Thayer Expedition Fish Drawings

**Pros**

  • free and hosted by LC, or locally installable from the open source Recollection software package
  • ingest collections from spreadsheets or MODS records. Upload from your desktop or import them from a URL.
  • generate interactive visual interfaces, including maps and timelines, and faceted navigation.
  • copy-paste to embed your interface in any webpage.
  • ease of use

**Cons**

  • no automatic harvest from Harvard catalogs
  • limited to 1000 records per collection on hosted installation
  • editing of records requires re-upload of entire dataset
  • links to external resources open in the same window - causes some navigability issues
  • development updates are infrequent

**Implementation**

  • option 1 - hosted by LC
  • option 2 - deploy Recollection locally: Apache, Python, Django

**XTF**

[http://xtf.cdlib.org](http://xtf.cdlib.org)

The eXtensible Text Framework (XTF) is a powerful open source platform for providing access to digital content. Developed and maintained by the California Digital Library (CDL), XTF functions as the primary access technology for the CDL's digital collections and other digital projects worldwide.
Blacklight

http://projectblacklight.org

Blacklight is an open source Ruby on Rails gem that provides a discovery interface for any Solr index. Blacklight provides a default user interface which is customizable via the standard Rails (templating) mechanisms. Blacklight accommodates heterogeneous data, allowing different information displays for different types of objects.

Installations
http://levysheetmusic.mse.jhu.edu/about
http://bv.stanford.edu
http://d.lib.ncsu.edu/collections

More examples: https://github.com/projectblacklight/blacklight/wiki/Examples

Pros
- faceted browsing
- configurable relevance based searching
- bookmarkable items
- permanent URLs for every item
- user tagging of items

Cons
- Is this getting new development attention?

Implementation
- Linux
- Ruby on Rails
- SOLR


Exhibit 3

http://www.simile-widgets.org/exhibit3/

“Exhibit 3.0 is a publishing framework for large-scale data-rich interactive Web pages. Exhibit lets you easily create Web pages with advanced text search and filtering functionalities, with interactive maps, timelines, and other visualizations. The Exhibit 3.0 software has two separate modes: Scripted for building smaller in-browser Exhibits, and Staged for bigger server-based Exhibits.” (Source: Exhibit 3 website)

Installations

Exhibit 3 is currently being tested with content from the Colonial North American digitization project at the Harvard Library. See the screenshots below:

Screenshots
- Simple layout
- Simple layout with details
Exhibit 3 offers two architectures that support different project needs and customer skill-sets: 1) Scripted - JavaScript-based, runs in the client; 2) Staged (currently in beta) - Java-based, runs server-side. Please note: the Scripted client-side architecture is evaluated here. The Staged architecture has not yet been evaluated.

**Fit-to-Requirements**
This spreadsheet shows the degree-of-fit for Exhibit 3 to the set of requirements proposed by the working group and by the needs of the Colonial North American digitization project: Web Presentation Requirements (29 July 2013).

**Pros**
- Set-up is very easy: a user can simply copy and modify one of the demo Websites and get up-and-running quickly
- Uses HTML and CSS and therefore it is very easy to create custom layouts without having an extensive Web-development skill set (e.g. people who do not know how to write software, but who can create simple Web-pages and CSS stylesheets)
- The widget model allows users to easily choose from a stock set of feature-full components and views to include in their HTML document
- Desirable features like search (of the content in metadata fields) and faceting are stock widgets that can simply be added to the HTML document
- Easy integration with the SIMILIE timeline widget create a unique view on data having date fields
- A number of pre-packaged data importers (e.g. CSV to JSON, Google spreadsheet to JSON, RDF to JSON, Excel spreadsheet to JSON) exist
- Modifications to the HTML DOM are straightforward and easy to manipulate using JavaScript libraries like jQuery and underscore, so advanced users can customize the look/feel of their Web-presentation very easily
- The base functionality offered by the stock widgets dovetails nicely with a minimal set of the digital collection user requirements
- There are quite a few demos that show off the features of Exhibit 3 and can be used as templates upon which users having few Web-development skills can base their own sites
- The software is Open Source and has an active development community
- Does not require a database installation; JSON data is contained in a single file on a file system (which is also a 'con')
- Supports bookmarks, persistent browsing, history, and on-the-fly export of data into a variety of formats

**Cons**
- Source data must be in a format that offers a conversion path to JSON via one of Exhibit 3's importers. (Or, a user must have the skills to convert the data to JSON on their own.)
- End-user documentation is sparse (although developer documentation for the API is fairly comprehensive)
- The Scripted version runs client-side, using a single data file, and therefore may not scale to very large datasets (however, the Staged version claims to support huge datasets)
- Extending Exhibit 3 to support new views/functionality is most definitely possible but requires programming expertise

**Implementation**
- Scripted: JavaScript, HTML5, client-side
- Staged: Java, server-side