

Princeton University Library Drupal Conversion

Antonio Barrera

Kevin Reiss



Princeton University
Library

Outline

- Project Scope and Goal
- Project Status
- Drupal Basics
- Content & Features
- Layouts
- Search
- Future Directions

Library Basics

- Academic Research Library
- 13 Branch Libraries
- Stakeholders
 - Students - Undergraduate, Graduate, PHD, Alumni
 - Staff - Both Users and Editors
 - Faculty
 - Global Researchers, Local Community
- Current site: library.princeton.edu
- Coming site: librarybeta.princeton.edu

Guiding Principles

- Management/Staff
 - Modern Design
 - "User" centered rebuilding of our web content
 - Focus on connecting users to our core services
- Developers
 - A web platform for today and tomorrow
 - Rapid development platform

Current Web Content

- Current site - circa 2006
 - Dreamweaver HTML Templates
 - Custom PHP
 - Legacy Apps
- Numerous sub-sites
 - Varying designs in static html
 - Most with own domain
 - Two major departmental Drupal sites
 - Seven small exhibition and special project Drupal sites
 - Twenty Wordpress sites

Why Drupal?

- Flexibility and Modularity
- Growing campus adoption of Drupal
- Drupal is widely used in the library community
- Get content back in the hands of our content managers
 - Drupal's authoring experience is improving.....
 - Easily build editor workflows with permission
- Drush - easily script site management tasks

Project Status

- Planning Phase - March - June 2012
 - Team of staff and developers
 - Contract with Campus IT for Design
 - Content Planning
 - Draw Wireframes - <http://balsamiq.com/>
- Code and Project Management
 - Two Developers
 - Github - Code - <http://github.com>
 - Open Atrium - Content - <http://openatrium.com/>
 - Self Hosted - Princeton Data Center
 - Virtual Machine
 - Full Control

Project Status

- Implementation
 - Content Migration - June
 - Design - July
 - Project Lead Left - October / ...Sandy
 - Steering Group - November
- Public Preview
 - <http://librarybeta.princeton.edu/>
- Usability Studies
 - Preliminary Studies - informal - September
 - Staff preview - December-January
 - Usability tests with key user groups - February

Our Core Drupal Stack

- Omega Theme - Responsive
- Views & Panels
- Drupal's [entities](#) & [entity reference](#)
- [Search API](#) and Apache Solr
- Formed:
 - 26 Content Types with 10 [Feeds](#) Importers
 - 18 Taxonomies
 - Over 150 Fields & 10 user roles
 - 56 Views

Site Migration & Content Building

- Some successful recipes
- Problems we had along the way
- Future proof our site by sticking with entities
 - A generic wrapper around all content units in Drupal
 - Nodes, Taxonomy Terms, Users are all entities
 - They are all "fieldable"
 - A lot of Drupal's main module families are entity-aware (Panels, Organic Groups, Views)
 - All entities aren't created equal in Drupal 7
 - Drupal 8 is bringing all entity functions into core

Hand to Hand Combat

- Reviewing and Migrating Static Data
- A number of our subsites needed major revisions and rewrites
- We could have done a better with job content inventory
- Get your editors into Drupal as soon as you can
- Minimize Word Time

Library Resource List

- Migrated from a custom LAMP application
- List core library research tools
- Built with Panels and Views
- Feeds Module use for basic content import
 - [Entity Field Query](#) used for advanced content mapping
 - Execute queries with drush scripts
- Editorial toolbar and dashboard
- Hindsight is 20/20
 - Migrate module would have been a better option
 - Nodes vs. Taxonomies

Staff Database

- Imported from a back office Access application
- Regularly scheduled Feeds Importer
- Entity reference fields are used to relate staff to relevant content
- Solr index of staff and their related entities
- Authentication via CAS - University Single Sign-on system
- The entire Princeton Community is potentially a Drupal user

Library Hours

- Local Vendor Built, Legacy
 - 20+ Calendars
- Calendar vs FullCalendar
- A Little Bit of Everything

Putting the Pieces Together

- Panels for Layout
 - Custom pages - overridden content types, taxonomies
- View Panes versus Blocks
- Responsive Design
- Responsive Panels
- Webforms
 - Mollom Captcha

Example:

<http://librarybeta.princeton.edu/libraries/firestone>

Services

- Share data with other applications
- Manage with Drupal
- Experimental - For us
 - <http://librarybeta.princeton.edu/locations/catalog>
 - <http://librarybeta.princeton.edu/services/locations>

Editor Workflows

- Make sure to use hints and field descriptions effectively
- Managing files and images is a challenge
- Toolbars by user role
- Dashboards by user role
- Contextual Edit options where appropriate
- Leverage search to allow editors to avoid the noise at #admin/content

Search - A Library Problem

- A focal part of our site design
- It's our gateway user activity
- Content Stores
 - Local collections (books, images, archives)
 - Web-scale scholarly materials search
 - Subjects and resource list database
 - Library web content
- Tab for each content store
- Users search for the right content in the wrong silo

Search - A Core Drupal Problem

- Drupal's core search module
 - Doesn't scale very well
 - Doesn't offer a lot of granularity or indexing options
- Two robust solutions for full-text search exist
 - [Solr](#) with [Facet API](#)!
- Apachesolr
 - Oriented towards an enterprise site search
- Search API
 - More flexible
 - Good integration with views and fully entity aware

New Approach - For Princeton

- Build a search that serves as a router
- Users get the item they want or to a better place
- One year's worth of search data indicates three primary usage patterns
 - Known titles and articles
 - Conceptual keyword searching
 - Library "things-entities" - buildings, hours, people, events, help, requests, etc.

All Search

- Panels, Views, and Custom Data Sources
- 10 different buckets
 - 5 external web services
 - Come via AJAX through a Silex application
 - 5 internal indexes via view panes
 - Each is a separate search api index
- Some buckets disappear without results
- Continue to improve with regular user testing and data analysis
- Buckets show up on other pages too

Moving Forward

- How are we going to maintain this?
 - Drush
 - Github
 - Features module where appropriate
 - Teamwork with our core editors and sub-site managers
- Responsive Improvements
 - Better menus
 - Improve search integration
 - Tabular data

Finish Building Editing Tools

- Explore in place editing for Panels - [Spark project](#)
- Get your editors/content managers into Drupal as fast as you can
 - They will surprise you
 - Most take to it relatively easily
 - Even the web-averse among our staff have been pleased
- Show authors how metadata drives the dynamic nature of a drupal site

Search Improvements

- As more library drupal sites come online
- Use feeds to aggregate all eligible sites
- Implement unified tagging scheme
- Bring in our external data sources as first class Drupal entities
- Start leveraging Drupal support for linked data with [schemaorg](http://schema.org)
 - Rich site snippets

Words of Wisdom

- Drupal is...
 - Big
 - Complex
 - Addictive
- Don't Rush
 - Can be a long learning curve
 - Take the time to map out "features"
 - Plan for content reuse
 - Start building your site for editors early on
- Drupal Community
 - Learn from others - their experiences, modules etc