Title

Three Nex-Generation Catalog Projects. A report on Presentations and a Discussion Hosted by the LITA Next Generation Catalog Interest Group. American Library Association Midwinter Meeting, Philadelphia, January 2008

Author

Shafer, S

Publication Date

2023-12-10

Peer reviewed

THREE NEXT GENERATION CATALOG PROJECTS.

REPORT ON PRESENTATIONS AND A DISCUSSION HOSTED BY THE LITA NEXT GENERATION CATALOG INTEREST GROUP. AMERICAN LIBRARY ASSOCIATION MIDWINTER CONFERENCE, PHILADELPHIA, JANUARY 2008.

Thomas Dowling, Chair of NGIG, launched the discussion by giving historical context to the second ever meeting of the Next Generation Catalog Interest Group. Being that the first meeting of NGCIG addressed commercial alternatives to the traditional library catalog, the second time around was dedicated to non-commercial, open source to see what some of our peers are doing on their own. The meeting held presentations and discussion about three locally developed and/or open source catalog projects. Ross Shanley-Roberts of Miami University discussed their SolrPac project; Bess Sadler of the University of Virginia discussed their project Blacklight; and Chris Barr and Andrew Nagy of Villanova discussed their VuFind project.

SolrPac

http://beta.lib.muohio.edu

Ross Shanley-Roberts (Miami University – Ohio) began with a brief background about the SolrPac project; why it was initiated and who the major players are. Seeking freedom from restrictions imposed by the traditional ILS, a systems librarian and a cataloging librarian joined forces to create SolrPac. Ross specializes in authority control and technical services cataloging while Rob Casson is a computing specialist and reference librarian. Both had many years of scripting, but neither had any formal computer science training. Ross estimated that he and Rob spend between 25% - 50% of their time on this project. While Ross spends most of his time analyzing and mining data from the catalogs, Rob spends his time creating/maintaining the web pages, and optimizing Solr performance. Work is also done to import metadata from new sources and to incorporate user suggestions.

The Library was looking for an easy to adapt solution that would provide a richer experience for patrons with as near as possible to synchronous display of data; especially with regards to check in and check out circulation activities.

The Library wanted to:

- Be able to index whatever and however they wanted.
- Include Web 2.0 features: tagging, faceting, etc...
- Work with something developer friendly and not expensive.
- Maintain consortia membership (thus limiting what ILS could be used)

The Library evaluated options such as Endeca, Primo, Encore, AquaBrowser,
LibraryThing, WorldCat Local, Drupal, Solr and Evergreen. Solr was decided to be the best solution. It is open source, has a moderate learning curve and it has native faceting which is very fast. Some quick experiments with a few thousand records and within a couple of weeks a test platform for the video collection was created. In their current set up, Solr actually resides near the end of their workflow. The other programs that they use are Expect, a couple of Perl modules, PHP, MySQL, hAjax and Drupal. They have a chron job set up that runs some Expect and PHP scripts and also re-indexes SolrPac every two hours so the SolrPac version is only two hours behind what is really going on in their ILS. They have three Expect scripts that export new and updated bibliographic item and checkin records. Another Expect script captures all the non-deleted bibliographic and item number records.

1. Bibliographic records are converted from MARC to text files using a modified version of the Perl module MARC to XML.
2. PHP script interprets bibliographic records and loads resulting values to a mySQL table.
3. Another PHP script loads the item and checkin record information into a 2nd mySQL table.
4. PHP script queries the updated records in mySQL tables and load them for Solr to re-index.

SolrPac Demo

Ross gave a SolrPac demo to illustrate some of the current faceting and social networking features included in SolrPac. These included:

- Drupal modules and themes that SolrPac sits inside of.
- Faceting and tagging.
- Incorporation of digital collections and external metadata.
- RSS feeds and external bookmarks.
- Ability to export to Refworks.
- Spellchecking.
- IM bar and Facebook applications that can access SolrPac. The wrap around is Drupal because it makes development and delivery of social networking tools much more robust.

Examples used to demo features:

- Search for everything in the catalog with a wildcard.
- Facets can be opened without typing a search.
- Advance limiting and sorting options to satisfy user demands such as, “I want the book with the least amount of pages”.
- Use of the user added tag- “grant writing”.

Solr has added a tagging tool which limits tag creation to current staff and students, but anyone can view the tags. The

Library decided not to pre-populate the tag set because they wanted their community to do the tagging.

Findings and conclusions so far:

- SolrPac can be used as a federated search.
- Creating your own indexes is key to this endeavor.
- Inconsistent data and coding is exposed.
- MARC redundancy is a marvelous thing.
- Having a combination of cataloger with some scripting experience and a programmer with some MARC experience was beneficial.
- Currently, they do not have an author facet. They index and facet the authors, but do not display them at this time because there are 316,000 distinct authors and it causes a very slow display so they turned it off for now.

Identified needs and actions:

- Need more controlled fields.
- Need catalogers to put the related codes back in so the system can tell if this is an Illustrator, Director, Violist, etc...
- Hired a new staff member to clean up the catalog.
- Need a uniform title spec.
- Need to Put in LCC and LCSH authority records.
- Need to expand the hierarchies, languages, coverage dates, etc...
- Implement negative searching.
- Experiment with larger data sets.
- Limit the out of synch time period from two hours to one hour.
- Work out the details to open source the work.

By producing the interface to the catalog in-house, it means a quick turnaround for implementing new things. Ross mentioned that he attended a RUSA meeting in the morning where someone mentioned that they can send the call number to a cell phone. Ross exclaimed,“Oh, that’s next!”

Blacklight

http://blacklight.betech.virginia.edu

Bess Sadler (University of Virginia) is a Research and Development Librarian and gave her presentation on the open source OPAC replacement known as Blacklight. Bess pointed out that having Eric Hatcher (one of the main authors of the Lucene indexing library under Penn Solr) on staff was a tremendous help on the project. Blacklight’s creation was due in part to a workshop that Bess and Eric gave last year at code4lib. The workshop was organized around Solr and

Lucene and got a lot of people interested in how these could be used to replace front ends for OPACS. That effort resulted in their first prototype of about 3.5 million MARC records. To add to the experiment, they included all of the Etext Center's Chang Dynasty Chinese poems. Erik happened to be learning Chinese and he wanted to see how well the Chinese would index. It turns out that Lucene and Solr are completely Unicode compliant and searches work beautifully in Chinese, Arabic, Japanese and even Tibetan. Eric Hatcher can also be credited with the name "Blacklight" which is solar radiation in the UVA spectrum.

Why did University of Virginia Library decide to go this route?

- Unhappy with the front end of their ILS (they are a SirsiDynix site).
- Interested in faceted browsing.
- Impressed with NCSU's Endeca rollout, but it was too expensive.
- They had Eric Hatcher on staff.

A prototype came up in six weeks but to bring something scalable and stable enough to a production environment for the public is much more difficult.
Project Blacklight is currently working on that very task.

University of Virginia is a Google Library and project Blacklight is trying to bring those scanned items in to the index as well to the catalog so users can get to items directly. Another key area of focus for project Blacklight is the creation of specialized interfaces for specialized collections. This year, the project focused on the music collection. In the field of music, there is a need for particular searches (E.g. what musical instruments are used in a piece? I've found a score. Now, how do I get musical recordings for that score?)
Technical process notes:

- Eric Soleberg (now working at NCSU) was instrumental in engaging cataloging staff to map MARC records to a Solr index.
- Blacklight does not convert MARC to MARCXML; it uses Ruby to index MARC directly in to Solr.
- Front end is written in Ruby on Rails.

Technical pros:

- Scalability of Solr is great. UVA has about four million records and there has never been a performance bottleneck. According to people on the Solr List, there are instances of using it with billions of records with no performance difficulties.
- Good in clustered environments. UVA is looking at a production implementation that will have a cluster of five web servers. One will be the master node and then they can send indexing

requests to it. It doesn't have to worry about receiving requests. Once it has done all the grunt work of indexing something, it can very quickly and transparently (there is no interruption to service) send that down to all the other nodes. The advantage to that is flexibility. If you add another million records or decide to become part of a consortium and host for several other universities all you need to do is add nodes to that cluster. If you have a load balancer in front of it you don't need to buy a super computer; you just need to have a lot of commodity hardware linked together. Bess commented that it was, "A nice metaphor for your open resource community".

- Error alerts.

  Blacklight sends an email to Bess every time an error is created. Although Bess needs to fine tune that (she doesn't want to know about every time a user misspells), it enables quick fixes to problems.

Data Cons:

- Found that the character encoding in the MARC code was inconsistent.

  They can force it all in to UTF 8. There are open source tools for doing that; Yaz by Index Data is excellent. However, there is a problem if it actually wasn't in UTF 8 all you get out is gibberish. Because there is redundant data all over the MARC records, the team found that if it is gibberish in one place there might actually be real data in another place in the record. They are trying to figure out ways to harvest that without having someone manually go in to the record. UVA is working on a method which will utilize a programmatic data clean up. They are also working on a production workflow for keeping the catalog updated daily. In the spirit of open source, UVA folks are having conversations with VuFind folks regarding these methods.
- Blacklight highlights how inconsistent UVA's records are.

  Bess echoed her co-presenters and stated that data cleanup is needed. However, the upside to exposing the need for data clean up has resulted in an improvement of the workflow; errors are discovered and cleaned up right away.

Blacklight Demo

Bess gave a demo to illustrate some of the current faceting and search features of Blacklight's specialized interface to the music collection. Bess showed how it is beneficial to take the same index and put a different interface on the front of it which exposed different possibilities for action that are specialized to music. Focus groups of music graduate students revealed they wanted:

- To quickly find anything that was a recording or a score.

- To know the composition era (E.g. Jazz recorded in the 20th century).
- To know the instrument(s) in the piece.
- Combine topic and genre together.

Bess demonstrated how all of these requests could be fulfilled via Blacklight searching and browsing features.
Identified actions for the future:

- Show customized interface per record type. (If it is an image show image. If it is a recording show click to listen)
- Experiment with an advanced search that allows for addition of "ands" and "ors" instead of going with a bunch of different fields and choices.
- Cataloging system for images based around the work. The image would have a contextual menu and it would display metadata.
- Continue the process of getting permission from administration to engage with the open source community. (Permission was just granted to make Blacklight open source).

Bess commented that she was, "very excited about launching a larger, more community driven project around this topic".

VuFind

Vudind.org/demo

Chris Barr (designer) and Andrew Nagy (programmer) of Villanova discussed the VuFind project. They emphasized that it was made plain looking on purpose to ease customization to your library template. Andrew pointed out that VuFind also runs on Solr and has all the faceted browsing features of the first two presentations. They decided to focus on some of the different features that VuFind has compared to the first two presentations.
Tagging:

- They built a tagging system very similar to Del.i.cious; where a user can organize their favorites by tags.
- Use tags to organize people's favorites and at the same time populate the global system so other people can benefit from the ability to search the tags.

When you first install VuFind you will have to import your entire MARC collection. From then on there is a synchronization process that will keep it up to date with your catalog. There is a delay in the synchronization process for when new items are created or deleted, but there is no delay on holdings information. In VuFind, you are getting that information live. Andrew explained that when viewing the "available" tag, what is happening is that we're searching the Solr

index for the search results first and then we're using an Ajax query after that search is run to get the live status of the books afterwards. So, we get the searches quickly and then we query the actual database of the ILS to get the live status. That way, it is not on a delay, we have the live holdings status. Benefits of having the opac directly extracted from the ILS:

- Migration of OPAC (user will never know the backend changes)
- Portable from ILS to ILS

Highlighted features:

- Algorithmically generated list of similar items.
- Ability to pull in Amazon.com or Bookreview.com or other book review sources.
- Allows for commenting by patrons.
- Citation widget can get APA or MLA citation format on that record. They recently introduced a Zotero implementation.
- Accounts system can either be hooked up to an ildac directory or also authenticated by remote db. VuFind has a mySQL db shipped with it to keep track of tags and comments but can also be used locally for user authentication and can be tied in to your ILS patron record.

Work in progress:

- Experimental browse features akin to itunes where the idea is that a user can select a topic/genre or something else and then narrow down by region.
- Facets incorporating the LC Subject and Call number. First letter of the call number is displayed next to the subject label. It will be possible to drill down more and more. Currently, they have two levels and they've labeled the first level. They hinted at a giant XML file headed their way that would increase the number of drill downs.
- User interface which is "anyspace" design; meaning that a zoom-in takes the whole page with it. Nothing is squished.
- Providing Wikipedia biographies for popular authors and they also plan to tie in with WorldCat Identities.
- In the planning stage for a federated search being part of VuFind.

Open source status:

- Have the approval to make it open source. It is at version 0.7. There is a source code repository for people who want to download and tinker with it.
- Have an agreement with Powelnet to provide service & support.

QUESTIONS & DISCUSSION

Authority Records & Data Normalization

There was an interesting revelation that none of the demonstrated projects currently employ authority records. All of the speakers indicated interest in incorporating authority records in the future. Bess (Blacklight) indicated that they were talking with their Art & Architecture Librarian on how to integrate the AAT. Ross (SolrPac) is currently in the midst of making hierarchies for LC classification automatic.

The flexibility to easily bring in, index and search all sorts of collections (digitized collections, professor's bibliographies, librarian's website bibliographies) was highlighted. A question was raised from the audience asking if the digitized collections were benefiting from full MARC records or other descriptive formats. Andrew (VuFind) builds some import modules (OAI harvesters) so as to bring in OAI data. His library has METS XML data so it is harvestable. Bess (Blacklight) indicated that UVA indexes all kinds of different formats. They index EED records natively and all the TIA documents they're doing have both the metadata and indexing of the full text. They also have some custom formats (GDMS) that they index as well. There was a question from the audience about a possible problem of normalizing against the faceted results. Bess (Blacklight) indicated that it is not a problem because they just create what they call "crosswalks" for those. Whatever the xml format is, the crosswalk defines how to find the title, year, etc… And then they map it to their Solr schema so it does not create a conflict in left fields. Andrew (VuFind) explained how his institution is not using MARC for the searching. The index is using different field names. So they are translating MARC in to their own schema. They can translate MARC, XML, OAI, Dublin Core, METS XML, MODS and any kind of defined format in to their own local schema. Ross (SolrPac) added that it seems like it is necessary to make all the Dublin Core data conform to MARC, but one can actually put in extra things. It is not necessary to make everything go in to one, limited lowest common denominator. They are pulling things in and finding out that content DM is just as bad as their catalog.

Relevance Algorithm

Andrew (VuFind) was asked about the "similar items" algorithm. He explained how it uses six fields which are weighted differently. It tries to use call number range to keep everything close in. It also tries to look for similar authors and formats and titles. It uses a collection of fields and each field is ranked with a different weight. Andrew indicated that the project is working on an administrative

module so that library staff could adjust the weighting within those algorithms.

Data Synchronization and Commercial ILS Limitations

The speakers were asked about how various statuses (on order, checked out…) were synched with the ILS and how accommodating the ILS vendors have been with that process. It came to light that many commercial ILS systems present license challenges such as forbidding customers to write to the database.
Andrew (VuFind) pointed out that each speaker's project synchs differently. VuFind's goal is to get the status as live, but they can't do that with Triple I. It is not possible to query the system through an API or through XML or through a command line. It has to be done via a screen scrape. For certain ILSs, a batch process gets that information. VuFind is trying with as many ILSs as possible to make statuses live. With Voyager it is a direct Oracle call. With Sirsi Dynix it is a command line. With Aleph it is through the XML gateway. Ross (SolrPac) currently has a clickable status updated within the last two hours. Bess (Blacklight) has a system of doing batch dumps and re-indexing as often as possible. Because VuFind is working on a plug-in for SirsiDynix, Beth hopes to employ that solution to improve Blacklight's data synchronization.

Statistics

It was pointed out that all the speakers use Solr as their indexing search engine and it does has some statistics in it. Andrew (VuFind) indicated that they are working on a statistics module that would allow one to see how many searches were made per day and popular searches.

Overall, the three next generation catalog projects overcame many limitations imposed by the traditional ILS and provided flexibility to quickly and easily index whatever and however they wanted to. Data enrichment was seen via tagging, faceted searches and specialized search portals were easy to create to meet the specialized search needs such as seen in music research. All the speakers found that the creation of a prototype came within weeks, but the difficulties lie within producing a scalable and ready for production product. With the recent hurdles of institutional legalities overcome, these projects are on the brink of sharing and collaborating in the true spirit of open source software and it will be exciting to see what the near future brings.