EXPLORING DISCOVERY

THE FRONT DOOR TO YOUR LIBRARY'S LICENSED AND DIGITIZED CONTENT

EDITED BY
KENNETH J. VARNUM

ala editions

An imprint of the American Library Association
Chicago 2016

KENNETH J. VARNUM is senior program manager for discovery, delivery, and learning analytics at the University of Michigan Library. In this role, he is responsible for the library's discovery interfaces (the "MLibrary" single search tool, ArticlesPlus, Search Tools, etc.), link resolution and content delivery interfaces, and the library's evolving and emerging learning analytics infrastructure. Varnum received a master's degree from the University of Michigan's School of Information and a bachelor of arts degree from Grinnell College. Over his two decades working with public-facing technology in academic, corporate, and special libraries, he has gained a deep appreciation and understanding of the need to tailor systems and interfaces to the local user base. In addition to numerous articles and chapters, he wrote *Drupal in Libraries* (2012), compiled the Library Information Technology Association guide *The Top Technologies Every Librarian Needs to Know* (2014), and edited Lorcan Dempsey's *The Network Reshapes the Library* (2014). He blogs at rss4lib.com and can be found on Twitter at @varnum.

ISBN: 978-0-8389-1414-4 (paper)

Library of Congress Cataloging-in-Publication Data

Names: Varnum, Kenneth J., 1967- editor.
Title: Exploring discovery : the front door to your library's licensed and digitized content / Kenneth J. Varnum, editor.
Description: Chicago : ALA Editions, an imprint of the American Library Association, 2016. | Includes bibliographical references and index.
Identifiers: LCCN 2015043425 | ISBN 9780838914144 (paper)
Subjects: LCSH: Electronic information resource searching. | Federated searching. | Information retrieval—Computer programs. | User interfaces (Computer systems) | Libraries—Special collections—Electronic information resources. | Electronic information resources—Management.
Classification: LCC ZA4230 .E97 2016 | DDC 025.5/24—dc23 LC record available at http://lccn.loc.gov/2015043425

Book design in the Bembo Std and Proxima Nova typefaces by Alejandra Diaz.
Imagery © Meowu/Shutterstock, Inc.

∞ This paper meets the requirements of ANSI/NISO Z39.48–1992 (Permanence of Paper).
Printed in the United States of America

20 19 18 17 16 5 4 3 2 1

CHAPTER FIFTEEN

FROM USER STORIES TO WORKING CODE

A Case Study from NYU's Digital Collections Discovery Initiative

DANIEL LOVINS

The New York University (NYU) Division of Libraries (the "division") embarked on a strategic initiative in 2013 to provide unified intellectual access to locally curated digital collections. The need for such an initiative is, naturally, not unique to NYU. The effort to establish bibliographic control over proliferating repositories, resource types, and metadata standards is a defining challenge of cultural heritage organizations today. NYU is one of the more complex organizations of this type, though, with unique constituencies, collections and mandates, and therefore must tailor its approach to meet local resource discovery challenges.[1]

In this chapter, I begin by describing the context of the initiative at NYU, its roots in the strategic planning process, and the kind of cross-departmental collaboration that was necessary for a successful outcome. I then describe how NYU assessed end-user needs and how these assessments were translated into functional requirements and work packages in order to ensure unified access to curated digital collections. Finally, I discuss lessons learned and recommended next steps, both for NYU and for other institutions engaged in similar efforts.

BACKGROUND

The NYU Division of Libraries, like the university it serves, is a complex and rapidly changing organization. It is a kind of consortium unto itself as well as a service provider to an external consortium. Within NYU, the libraries serve more than 40,000 students and over 3,100 faculty at multiple New York sites including Washington Square, the Health Sciences complex, three major special collections (University Archives, the Tamiment/Wagner Archives, and the Fales Library), the NYU Polytechnic School of Engineering, two additional full-fledged campuses in Abu Dhabi and Shanghai, and multiple smaller-scale research institutes and study sites both in New York and around the world (www.nyu.edu/about.html). Externally, the division facilitates shared borrowing and a union catalog for the Research Library Association of South Manhattan (RLA-SM), which includes Cooper Union, the New School for Social Research, the New York School of Interior Design, the New York Historical Society, and the Brooklyn Historical Society. The NYU Libraries system includes fourteen service locations, while the RLA-SM accounts for eight additional locations (library.nyu.edu/about/locations.html). Core bibliographic processes and applications are managed from the Bobst Library at New York's Washington Square, and its annex facility at 20 Cooper Square.

Within Bobst and 20 Cooper, there are some nonobvious divisions of responsibility worth mentioning here. Knowledge Access Design & Development, or "KADD," the department led by the current author, includes experts in original cataloging, metadata services, and bibliographic systems administration. As the name suggests, the department is responsible for designing new strategies and workflows, and to provide access to new forms of knowledge.[2] This involves going beyond the traditional integrated library system (ILS) and making available websites, blogs, research data, social media archives, full text corpora, or any other objects of interest to our students and researchers.

The concentration of systems and metadata expertise in KADD enabled an unprecedented level of collaboration with other technical units in the libraries, notably Web Services and Digital Library Technology Services (DLTS). Web Services is the department responsible for the division's web presence, including the look and feel of its content management system, integration among diverse software applications, integration with university administrative systems, and custom software development. Among other services (dlib.nyu.edu/dlts/services/), DLTS is responsible for reformatting, preservation, and publication of NYU and affiliated digital collections, as well as grant-funded and custom software development.

Goal Four (out of eight) of the libraries' *Strategic Plan 2013–2017* (library.nyu.edu/about/Strategic_Plan.pdf) is to "establish processes and support structures that ensure we can select, acquire, preserve, and provide access to the full

spectrum of research materials." Subsumed under this goal is Initiative Three (out of three), henceforth, "the initiative:" "a plan to provide intellectual access to NYU-curated digital collections via the library's primary discovery-and-access interfaces."

By identifying this work as a strategic priority, the division has been able to leverage the five-year strategic planning cycle to ensure long-term institutional buy-in and regular communication across a wide range of colleagues and stakeholders, including collection curators, metadata specialists, web developers, and usability experts. These communications include internal project meetings (as described below) as well as regular exchanges with the Department Managers Group and with library administrators. Moreover, the strategic plan itself was preceded and informed by environmental scans, department-wide interviews, all-staff forums, and a "Synthesis" report, all of which pointed to collaboration, agile development, and user-centered design as high priorities.

IMPLEMENTING A STRATEGIC INITIATIVE

In July 2013, the initiative charge was drafted and two working groups were established: (1) Collections and Functional Requirements, and (2) Technical Specifications and Prototyping. The first group included representatives from the Fales and Tamiment special collections, along with repository curators and website archivists; namely, individuals who were able to help identify "hidden collections" and submit user stories to inform functional requirements for their discovery. Also in this group was the head of the User Experience (UX) Department, who offered guidance on the efficacy of new features. The current author was a member of both groups, and therefore could facilitate communications between them. Target collections in a first phase included geospatial data sets, numeric data sets, archived websites, Drupal-based digital collections, and files in NYU's DSpace-powered Faculty Digital Archive (FDA).

The second group identified tools and procedures for aggregating, normalizing, enhancing, and disseminating metadata from these digital collections, based, in part, on earlier work on a "Union Catalog for Digital Projects" (2011) and a prototype developed at an NYU Libraries hackfest (2013). This group included developers from Web Services, DLTS, and KADD. It is this second group that settled on the Hydra technology stack and agile project management for implementing user stories submitted by the first group. It is worth noting that, for the initiative, we did not ask students and scholars to submit user stories directly. Rather, we relied on curators and usability specialists to represent their needs and interests.

The shorter-term goal for the initiative was to provide lightweight normalization and remediation for newly received metadata before integrating them

with NYU's primary discovery portal, BobCat. This would solve an immediate known problem, namely, that it is often hard to know where to find information on curated digital collections. At the end of this initiative, our patrons would be able to discover many additional NYU digital resources in BobCat, records for which would provide immediate access to resources as well as back-links to the original host repository for richer descriptions and retrieval options. The longer-term goal was to implement a robust internal data model that exposes highly structured, semantically rich metadata to aggregators such as DPLA and WorldCat. Additionally, we see a role for curated library data as part of the "backbone of trust" for the Semantic Web: assertions about authorship and subjects, for example, in knowledge bases like the Google Knowledge Vault.

Like several other projects described in *Exploring Discovery,* ours benefited from having access to the Hydra technology stack and support community. "Ichabod," as we call our discovery solution, is based on Fedora, Solr, Blacklight, and Hydra itself, and is being used at NYU to normalize incoming metadata to a centralized "NyuCore" element set.

In Ichabod we are using the Fedora repository only to store metadata, not the digital objects (for which we have other solutions). Solr is a widely used Lucene-based indexing and searching tool. Blacklight is a discovery layer, maintained by the same community that developed Hydra. The Hydra "gem" (a module of Ruby code) allows web-based management of Fedora objects. NyuCore is a locally defined metadata application profile based on Dublin Core. We are using a "headless" instance of Hydra (hence the name, "Ichabod") as a staging area before exporting metadata to BobCat.[3] We are not (currently) using Blacklight or a Hydra head as a public interface, since we have the Primo discovery tool already embedded in BobCat. We do, however, retain the Blacklight "head" for staff interaction with the Fedora repository.

AGILE DEVELOPMENT AND THE SCRUM FRAMEWORK

"Agile" denotes a way for groups to work together with maximum transparency, accountability, and responsiveness to user needs. It is hard to find a single succinct definition in the literature, but practitioners emphasize the value of face-to-face contact, short-term, iterative development cycles, the relative importance of working code (over complex planning), and treating customers as development partners.[4]

"Scrum" is a particular type of agile development. The official *Scrum Guide* (2013) defines it as "a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value" (p. 3). Scrum involves special designations of "roles" and

"events." The roles include a "product owner," a "development team," and a "scrum master" (www.scrumguides.org/scrum-guide.html#team). In our case we had two product owners: the current author in his capacity as initiative chair, and the senior manager of NYU's Digital Library Infrastructure. In theory this could pose a problem if the product owners had conflicting agendas. In practice, though, there was no conflict, and the arrangement helped ensure that both digital discovery and digital publishing would inform the Ichabod development process. The two product owners assumed organizational and political responsibilities for prioritizing functional requirements, establishing priorities, and managing relationships with stakeholders. They collected user stories, stored them in a PivotalTracker knowledge base (www.pivotaltracker.com), placed some stories in the "backlog" for short-term assignment, and others in the "icebox" for long-term reference. By having product owners serve as gatekeepers to the backlog and as buffers from outside pressure, developers could be freed up to work on high-value features that had already been vetted.

The development team included two developers from Web Services, three from DLTS, and two from KADD. The Web Services contingent kicked off the project by setting up the initial Hydra instance and sharing their development workflows and infrastructure. DLTS developers had deep knowledge of NYU digitization projects and digital collections and were especially helpful in designing ingest mechanisms into Ichabod. KADD members implemented the NyuCore schema and set up the initial data loaders. They also shared their knowledge of systems underlying BobCat, especially Aleph, which powers our ILS, and Primo, our discovery layer.

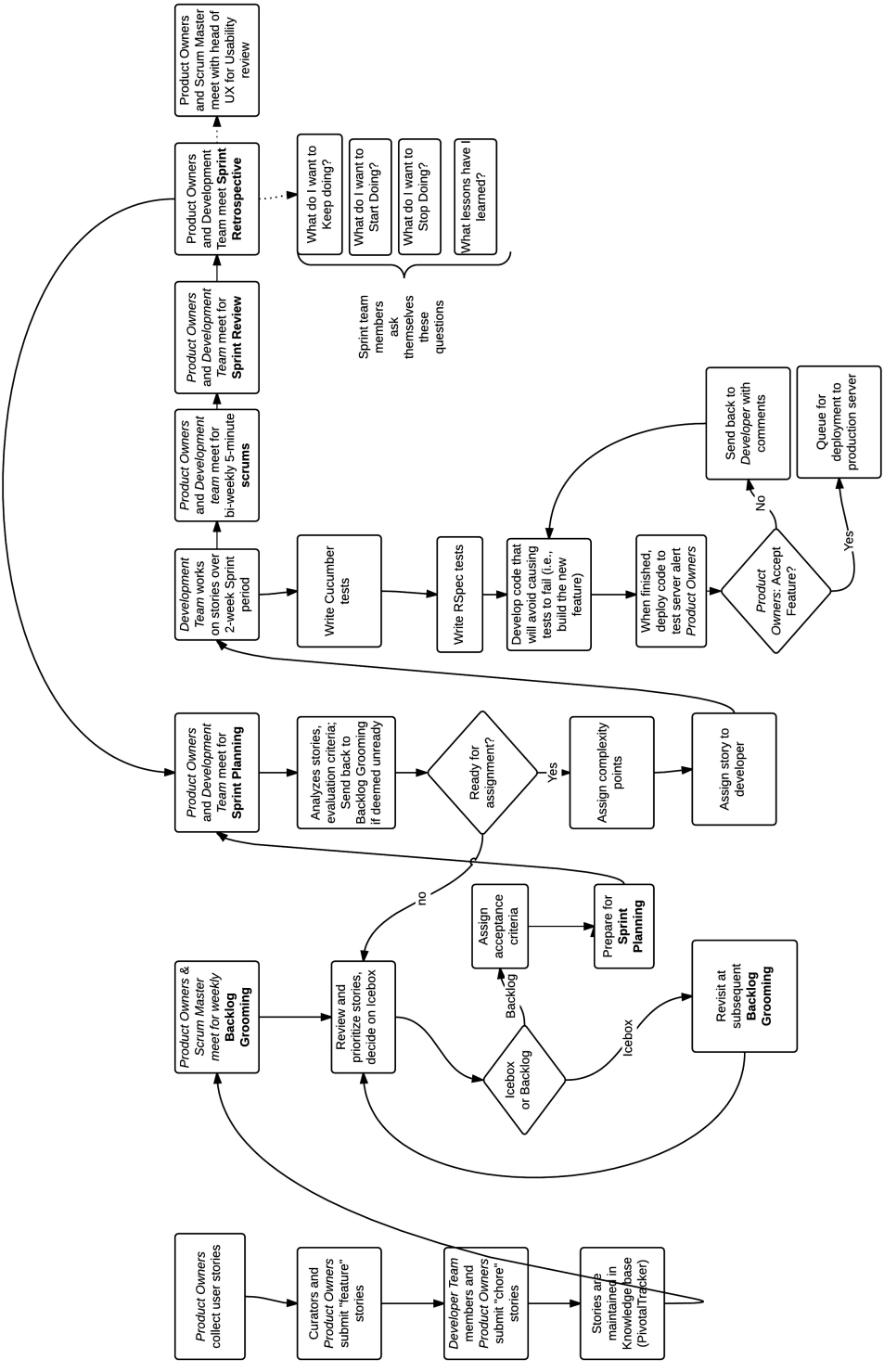
"Events" in Scrum include "backlog grooming," "sprint planning," "daily scrums," "sprint reviews," and "sprint retrospectives" (www.scrumguides.org/scrum-guide.html#events). In the case of Ichabod, we did not implement every aspect of Scrum, but only as much as was practicable given our resources and needs. In our case, the product owners met each week with the scrum master in a backlog grooming session to review user stories, document prerequisites and dependencies, and decide which stories to move from the icebox to the backlog before the next sprint planning meeting. At the biweekly sprint planning sessions, the product owners and the development team analyzed user stories, sought clarification on details and acceptance criteria, assigned complexity points, and discussed who might volunteer to undertake a given story for the upcoming sprint period.

From that point on, the developers had two weeks to work on their assigned stories. Two or three times each week the full Scrum team would check in for five-minute Google Hangout meetings ("scrums"), presided over by the scrum master. Members reported on what they accomplished since the last scrum, what they planned to do next, and what, if anything, was blocking them. At

FIGURE 15.1

Evolution of user stories into product features

Roles are in *italics*; Events are in **boldface**



the end of a development cycle the scrum master would sometimes convene a combined sprint review and retrospective, where members would share what they want to keep doing, what they want to stop doing, lessons learned, and any other reflections. Then the cycle would start over with a new sprint planning meeting, continued backlog grooming sessions, and regular scrums. If a developer were unable to finish a story within the two-week sprint, he or she would simply carry it over into the next sprint. Figure 15.1 shows how user stories start as plain English-language requests and evolve into actionable feature specifications and then running code.

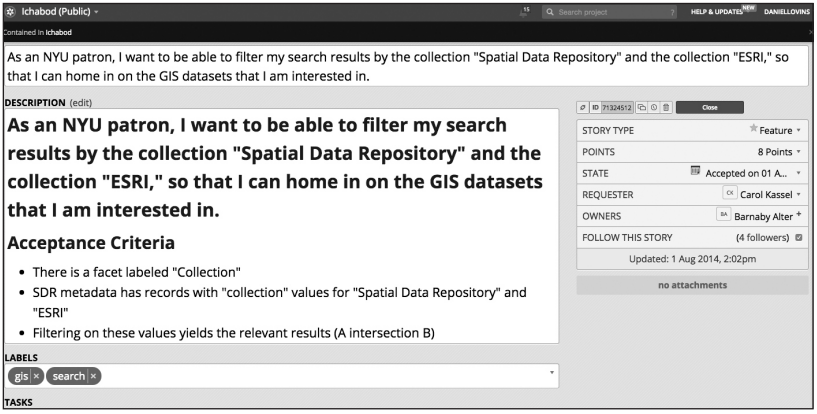
FROM USER STORIES TO FEATURES

Here is an example of how a user story was turned into an Ichabod feature: product owners met with managers of the NYU Spatial Data Repository (SDR), who are also members of the Collections and Functional Requirements group. In our discussion on functional requirements, we all understood the need for collocating SDR records within Ichabod, both by original repository (SDR) and by collection (in this case, the name of the GIS software company and platform "ESRI").[5] Toward that end, we documented this user story:

"As an NYU patron, I want to be able to filter my search results by the collection 'Spatial Data Repository' and the collection 'ESRI,' so that I can home in on the GIS datasets that I am interested in."

FIGURE 15.2

SDR story in PivotalTracker



At a backlog grooming session, we developed the following acceptance criteria:

- There is a facet labeled “Collection”
- SDR metadata has records with “collection” values for (A) “Spatial Data Repository” and (B) “ESRI”
- Filtering on these values yields the relevant results (A intersection B)

At a subsequent sprint planning session, developers asked questions and sought consensus on the complexity level of this story. Developers would discretely write down a complexity score (a kind of educated guess, represented by a Fibonacci number between zero and twenty-one), and then share them with the group all at once. This would then lead to discussion on why some developers assigned a higher score than others, and they would negotiate until consensus was reached, thus affording a more nuanced understanding of the nature of the challenge. We used the Fibonacci sequence (en.wikipedia.org/wiki/Fibonacci_number) to remind ourselves that the more highly scored stories were often *significantly* more complex than the less highly scored ones. Finally, a member of the group volunteered to take on the story (which, incidentally, had received a complexity score of 8). In this case, it involved modifying a Ruby-based data loader, aligning incoming records with NyuCore, assigning the collection name as a new element, and testing the feature with RSpec (www.rspec.info/) and Cucumber (www.cukes.info/). This testing step is critical in the evolution of plain English user stories into working code. As one project member put it, “With Cucumber, you may describe a feature in plain English, then define the steps in Ruby (with RSpec), and finally let it fail. This will have the added benefit of documenting what your code is doing and what you are testing for.”[6] See figures 15.3–15.6 for screenshots of an SDR record as it appears and moves through three different systems.

FIGURE 15.3

SDR record in SDR

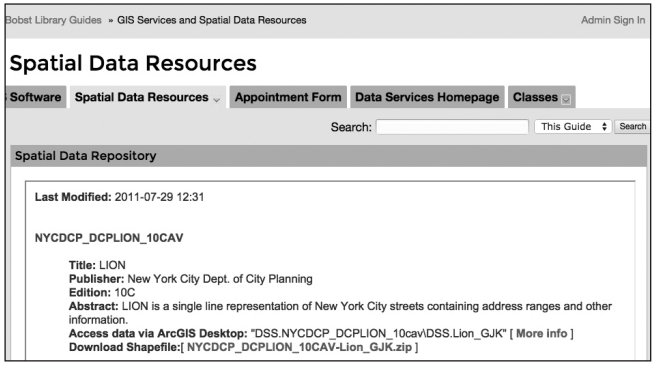


FIGURE 15.4

SDR record in Ichabod

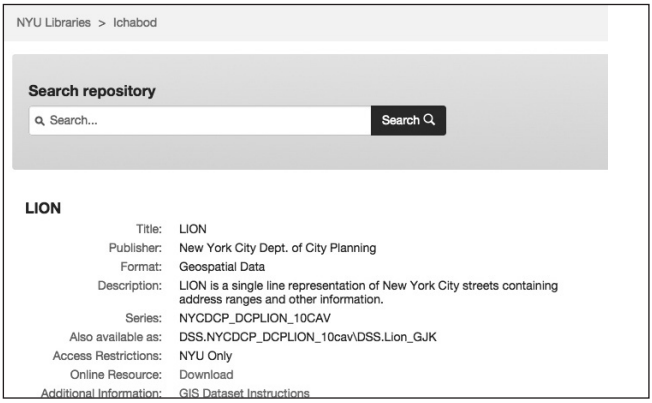
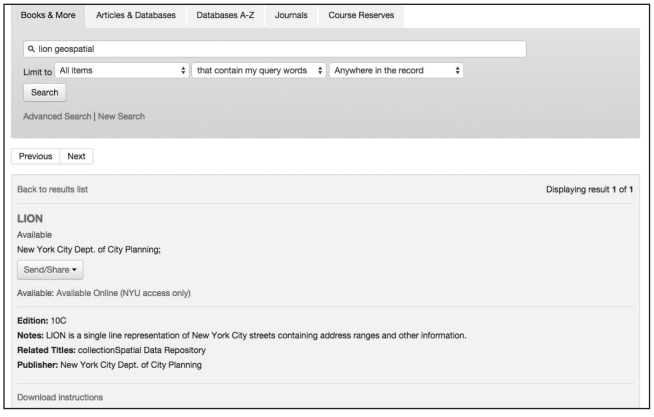


FIGURE 15.5

SDR record edit in Ichabod

FIGURE 15.6

SDR record in BobCat



This was a fairly straightforward example, since we had complete control over collection names and could assign them as a kind of rubber stamp at ingest time. More challenging user stories go beyond the insertion of text strings like collection names, though, and require careful thinking about levels of organization. For example, here are two other user stories related to the first:

(1) "As a curator, I want to be able to describe existing resources at the collection level, so that I can add information to them."

Here are the acceptance criteria:

- All existing collections in Ichabod are changed to use Collection object rather than Resource Set to store collection information.
- Collection Abstract field exists, is editable.
- UI (user interface) is unchanged as a result of this modification.

(2) "As a collection curator, I want to see my resources organized in Ichabod by collection, because that's the way I think."

And here is the acceptance criterion:

- There is a Collection object in Rails with these fields: Collection Name, Abstract, Rights, Discoverability Flag, Provider, Department.

Curators have told us they need to be able to modify aspects of the Collection class and have them inherited by members of that class. In this example, the access rights for the entire collection might change, so rather than having to revise access restrictions item-by-item, we want the Collection class to contain the information and cascade down to all of its instances. Having a Collection object with properties like "rights," along with editing forms and instance inheritance, makes this possible.

Here is another user story that illustrates the need for a well-defined collection object.

"As a curator, I want to request that a set of new records be hidden from view in Ichabod, so that I can fix the metadata before making them discoverable." This feature was requested by curators who want time to improve the quality of resource descriptions before exposing a web archives collection called "Composer Web Sites" in Ichabod and BobCat. Here are the acceptance criteria that were determined through consultation with the curator, the product owners, and the development team:

- On load, the Composer Web Sites records are marked as non-discoverable.
- A search for "louiskarchin.com" (i.e., the URL for one of the curated composer sites) shows 0 results when not logged in.
- A search for "louiskarchin.com" shows a result when logged in as (curator).
- The ability exists to mark the records as discoverable from Jenkins (our continuous integration server).[7]

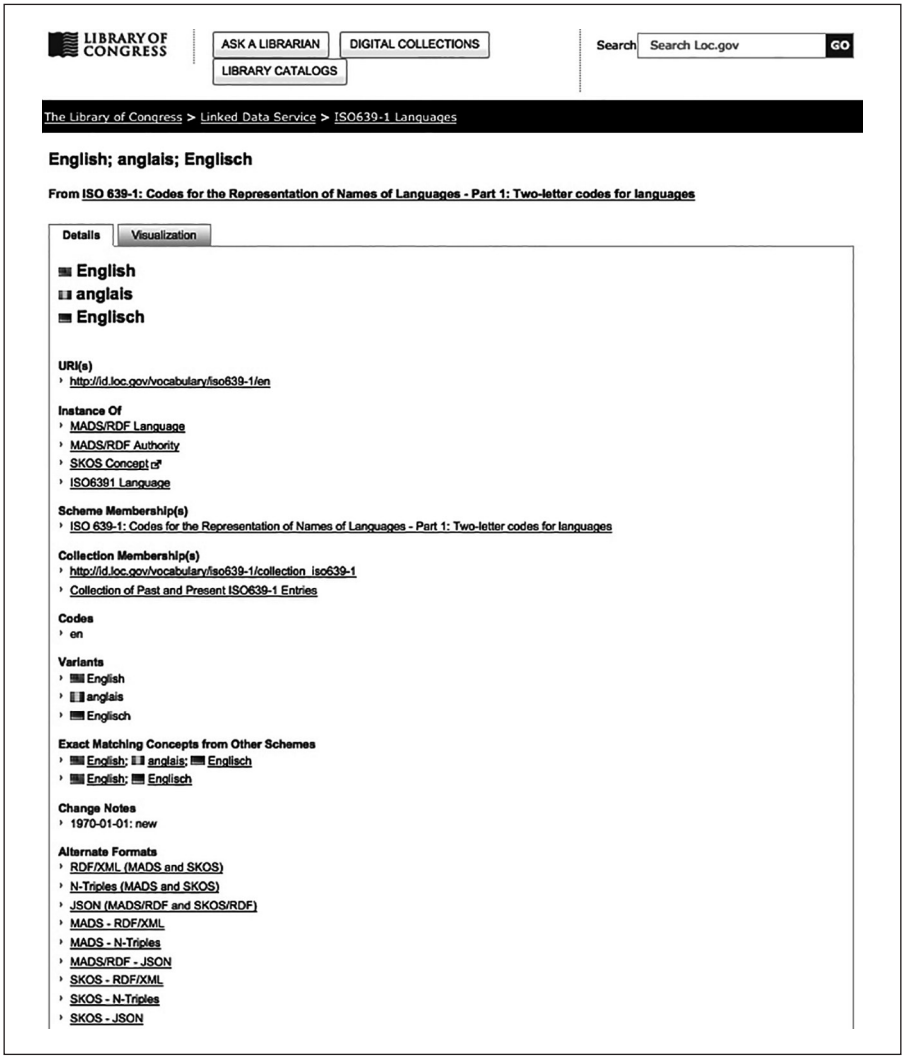
These user stories require specific implementation decisions in Fedora and Rails, but even before that happens, they require a consensus on our data definitions, namely, how we define the "collection" object and enumerate its attributes. We do not want to make these decisions just for Ichabod and BobCat, however, since a longer term goal is to enable interoperability with higher-level aggregations. Thus, we note that the Digital Public Library of America's (DPLA's) *Metadata Application Profile* (*MAP*) uses the label "Collection" for a "collection or aggregation of which described resource is a part,"[8] while the *Europeana Data Model* (*EDM*) uses "Is Part Of" for "a related resource in which the described resource is physically or logically included," and both are reusing the DCMI Metadata property: "dcterms:isPartOf" (dublincore.org/documents/dcmi-terms/).

We have similar user stories around the concept of language. At the application level, we want to avoid literal character strings like "English" or codes like "eng." We prefer an RDF-type object with a unique persistent identifier, a preferred label, alternate labels, position within its language family, and so on. The Library of Congress maintains a list of ISO language terms, where the URL that identifies the English language (id.loc.gov/vocabulary/is0639–1/en.html) represents a complex object (see figure 15.7). We want Ichabod to point to this kind of URL and benefit from its persistence, its cross-references, and its RDF serializations.

Ichabod is an aggregation of discrete digital library collections curated at NYU. We do not know exactly what new collections will be added or what will happen as Ichabod collections get combined with BobCat (another aggregation), or WorldCat or DPLA. By building linked data concepts into Ichabod and aligning our data model with those of the wider web and cultural heritage communities, we can fulfill the need for data consistency and collocation as reflected in our user stories, while also benefiting other constituencies through improved interoperability, shared identifier schemes, and a lower maintenance burden on any one institution. Additionally, the Ichabod team is contributing its own work back to the wider Hydra community and helping refine a common data model.

FIGURE 15.7

English-language object as linked data



LESSONS LEARNED AND NEXT STEPS

The strategic planning process cannot prevent conflicting priorities and competition over scarce resources, but it can help ensure that only the highest-priority initiatives become part of that competition. And in the case of new discovery services, projects can be mutually reinforcing. For example, overlapping with the

Ichabod project at NYU has been an effort to rebuild our archives discovery portal to improve indexing, display, system logging, responsive design elements, and other features. This, too, involved Scrum methodology, Ruby on Rails, and Blacklight development, and many of the same developers and specialists. Since much of the infrastructure was the same, the project members continue to build up common NYU infrastructure and skills and can be even more effective when returning to Ichabod. Moreover, the archives portal was able to ramp up quickly because of the precedent set by Ichabod.

We need to continuously revisit what it means for something to be a "curated collection," as specified in the initiative. For example, the NYU Faculty Digital Repository is largely self-service. Some of the contributions may appear out of scope or inappropriate for wide dissemination. Some metadata may be irregular or absent, and fail to map to our data model and therefore fail to generate useful access points and collocation in the discovery interface. The former challenge (i.e., what counts as "curated") will remain a grey area and require case-by-case evaluation before being included in Ichabod. The latter (i.e., inadequate metadata) can be mitigated through quality assurance routines, metadata remediation, and vocabulary reconciliation.

While we have in place a basic data model for NYU resources, we envision future iterations that are more fully articulated. Moving forward, we intend to align ourselves with larger-scale aggregators and discovery systems, including DPLA.

As the number of collections grows, and the Ichabod mapping becomes more complex, we want to incorporate usability review more tightly in the development process. This will become even more important if we decide to make Ichabod a discovery platform in its own right, that is, in addition to its role as an aggregator, remediation platform, and staging area. We also should consider expanding the range of assessment inputs to include application transaction logs, web analytics, and interviews directly with students and researchers.

CONCLUSION

NYU has undertaken a strategic initiative to make its distributed digital collections discoverable through the libraries' central portal, BobCat, and has completed its short-term goal of being able to represent high-priority curated digital collections in BobCat via Ichabod. The "user story," a component of agile development, has proven an effective means of understanding patrons' needs on their own terms, and allowing them to remain at the center of conversation among developers, product owners, and stakeholders. In the case of Ichabod, we see how plain English feature requests are captured in PivotalTracker, assigned

priority level and acceptance criteria by product owners, then assigned complexity points by the developer team. The developers instantiate the acceptance criteria in the form of natural-language Cucumber and Ruby RSpec tests and proceed to write the code. Once the features have been realized within the application, curators and usability specialists help us determine how well they satisfy the original stories. Product owners decide whether to "accept" the story as complete, and, if so, whether the implemented feature warrants future changes or enhancements (in which case they will need new user stories).

Our longer-term goal involves enhancing the NyuCore application profile to exploit its value in a linked open data environment. At the same time, we are following data modeling efforts at Hydra, DCMI, DPLA, Europeana, Schema .org, W3C, and other communities to ensure that NYU collections data can be aggregated effectively for discovery at any level. This work will need to continue being rooted in the needs of our real users, so user stories will remain essential. Here is a made-up (but plausible) example of a long-term goal user story: "As a researcher, in order to build up a collection of images as quickly and comprehensively as possible, I want to be able to search DPLA for images of Indian Ocean postcards, and find NYU's contributions interfiled with those of other institutions."

The path from user story to product feature is supported by the commitment to a common development environment and work culture: the Hydra framework for its community and shared code libraries, Cucumber and RSpec for test-driven development, NyuCore for a local metadata hub, and Scrum for communication and project management. By embracing a shared vision and infrastructure, we are able to take plain English user stories and find a path to implementing them as tangible product features.

Notes

1. For example, the division oversees rapidly growing physical and electronic resource collections, extensive special collections, a strong digitization program, NYU TV channel operations, and the university press. All of these include important intellectual assets that need to be made available to the NYU community and beyond.
2. Carlen Ruschoff, "Reality Check: A New Framework for Technical Services: Interview with New York University's Carol A. Mandel and Martin Kurth," *Technicalities* 33, no. 5 (September/October 2013): 1ff. In 2010 the division created a new directorate called Knowledge Access and Resource Management Services (KARMS). KARMS comprises three new departments: Resource Management (RM), Metadata Production & Management (MPM), and KADD. RM manages, among other things, workflows around budgets, vendors, and e-resource knowledge bases. MPM "is responsible for providing intellectual access to a continuing flow of physical materials acquired from all over the

world, along with maintaining the bibliographic database and doing the necessary work to keep information accurate, current, and fresh."

3. The reference is to Ichabod Crane, the schoolteacher who engages the Headless Horseman in Washington Irving's *Legend of Sleepy Hollow.*
4. "Manifesto for Agile Software Development": http://agilemanifesto.org/; Agile Alliance, "What Is Agile?" www.agilealliance.org/the-alliance/what-is-agile/; Martin Fowler, "The New Methodology": http://martinfowler.com/articles/newMethodology.html.
5. That is, the "Environmental Systems Research Institute" (ESRI), which, in 1982, released "the first commercial GIS," the ARC/INFO system. See www.esri.com/about-esri/history/history-more.
6. Based on a previous user story, there was already a workflow in place to ingest data from source, map them to NyuCore elements, persist (i.e., store) metadata values to Fedora, index them in Solr, and convert Ichabod's HTTP JSON output to Primo Normalized XML (PNX), with could then generate resource descriptions and delivery options in BobCat's current discovery layer, Primo.
7. "About Jenkins CI": https://jenkins-ci.org/content/about-jenkins-ci.
8. "DPLA Metadata Application Profile, version 4.0": http://dp.la/info/wp-content/uploads/2015/03/MAPv4.pdf.

References

Agile Alliance. "What Is Agile Software Development?" www.agilealliance.org/the-alliance/what-is-agile/.

Definition of the Europeana Data Model v5.2.6. http://pro.europeana.eu/files/Europeana_Professional/Share_your_data/Technical_requirements/EDM_Documentation//EDM%20Definition%20v5.2.6_01032015.pdf.

DPLA Metadata Application Profile. Version 4. https://docs.google.com/a/nyu.edu/document/d/1Jh8ULpw0jb8kyxV-Ygw9U0n-XqXkM6_V3jxfmnkebqo/edit.

Fowler, Martin. "The New Methodology." http://martinfowler.com/articles/newMethodology.html.

"Manifesto for Agile Software Development." 2001. http://agilemanifesto.org.

NYU Libraries. "Report on Strategic Initiative to Provide Enhanced Intellectual Access to NYU Curated Digital Collections." https://archive.nyu.edu/handle/2451/33869.

———. *Strategic Plan 2013–2017*. http://library.nyu.edu/about/Strategic_Plan.pdf.

Pivotal Labs. "PivotalTracker." https://www.pivotaltracker.com.

Ruschoff, Carlen. 2013. "Reality Check: A New Framework for Technical Services: Interview with New York University's Carol A. Mandel and Martin Kurth." *Technicalities* 33, no. 5 (September/October): 1–5.

Schema.org Data Model. http://schema.org/docs/datamodel.html.

Schwaber, Ken, and Jeff Sutherland. 2013. "The Scrum Guide." www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf.