Customizing Discovery Interfaces: Understanding Users' Behaviors and Providing Better Service

Rachel Vacek, Head of Design & Discovery
Ken Varnum, Senior Program Manager
University of Michigan Library

ER&L - March 5, 2018

LIBRARY

@varnum @vacekrae @UMichLibrary

Explain Rachel's absence (flu)

Overview

1. Background
   a. Our landscape (where we started)
      i. Apps/systems
      ii. Wireframes and UX research
   b. What we set out to do
      i. Goals for the project
      ii. Communication plan
      iii. Resources (people, groups, tools)
2. Project
   a. How we made progress
      i. Prototype for acceptance
      ii. Library-wide project team (oversight/direction)
      iii. Development team / UX, design, front-end, back-end
      iv. Iterative development (agile methods)
   b. Where we are & what's still undone
   c. DEMO if time

Rachel was to have talked about the 'background' section; I'll be filling in, and hope to do cover the points with the detail she would have brought.

Where We Started

An organic set of discovery tools...

...with poor consistency...

...with different user experiences…

...that we realized had to go.

RACHEL

Organic growth (where we've been) compared to planned growth (where we want to be)

- Each tool served a need
- It's now time -- well, past time, really -- to fix this UX mess

We’ve all been there.

RACHEL

Current starting points

They are many.

- Mirlyn: catalog
- ArticlesPlus: licensed content
- Search Tools: database finder and journal finder
- MLibrary: catalog, licensed content, database and journal finders, subject experts, website content, research guides

RACHEL
On our current library website, we have many different starting points:

One discovery interface is the **catalog (Mirlyn)**
- VUFind Version .6, though we built on it heavily
- The catalog metadata is abstracted from Aleph and indexed in Solr.

A second discovery interface is the **ArticlesPlus layer**
- It uses the Summon API and a custom Drupal module.
    - Our traditional catalog content is not combined with our licensed content (a decision from 2009) and there is WAY too much content in each area. Adjusting the relevancy ranking with that much data would have been a nightmare.
    - Deep Blue Docs (DSpace) = Institutional Repository → is available as a “database” in Search Tools. Individual records are harvested in Summon and available through ArticlesPlus.

A third discovery interface is **MLibrary**
- This provides a quick overview of everything.
- This is the default tab that presents a bento box interface
    - Results are from: ArticlesPlus, Mirlyn, Database Finder + Journal Finder (custom Drupal module manages them together), Subject Experts, Website content (managed in Drupal, thrown into Solr), and LibGuides.
    - Search Tools = DB Finder + Journal Finder
    - Digital library materials are currently included at the collection level and listed as “databases” within Search Tools.

Clicking on anything in the search results takes the user out of our website and into another system (catalog, database, online journal, etc.)

Not currently included in any of our discovery interfaces:
- DLXS (custom software written in 90s) = Digital Library = digitized images, text
- Deep Blue Data (Hydra) = Data Repository

UX Research -- How we got started

- “Old” and new research
  - Reviewed user research from previous 4 years
  - Partnership with Deirdre Costello and EBSCO UX Team to conduct contextual inquiries
  - Conducted additional user testing on existing search interface
  - Analytics data and search logs
- Created tons of wireframes
- Conducted usability testing on new wireframes

RACHEL
User research from previous years:
- User journeys, user personas,
- lots of UX research prior to half the team even being hired

Partnership with EBSCO
- collaboration on contextual inquiry. We observed the habits and patterns of current users.
- Provided a baseline around their designs for their new interface, and could better understand effectiveness of them

One thing we learned after all that research: We needed to design a uniformed search across multiple systems so it’s less fractured across the systems.

Usability Testing

- Utilized task-based, guerilla, and contextual methods
  - Initially with paper and clickable prototypes, then on developed interfaces
  - Focused on specific feature sets and data stores
- Recruited from different user groups
    - Advanced, beginner
    - Students, faculty, staff
    - Student organizations
    - Allies across campus
- Subject experts helped identify participants
- Used candy, gift cards, and swag as incentives
- Student interns helped run tests and analyze results

RACHEL

- Utilized task based, guerilla, and contextual methods
  - Initially with paper and clickable prototypes, then on developed interfaces
  - Focused on specific feature sets and data stores
- Recruited from different user groups
    - Advanced, beginner
    - Students, faculty, staff
    - Student organizations
    - Allies across campus - Services for Students with Disabilities
- Subject experts helped identify participants
- Candy, gift cards, and swag as incentives
- Student interns helped run tests and analyze results

Resources (people)

It has taken us a lot of FTE over the past 9 months!

- 2 front-end developers
- 1 back-end developer
- 1 designer
- 1 UX researcher
- 2 UX interns
- 1 content expert
- 1 project manager

These people weren't as free to take on many other projects.

RACHEL
This has been a big project. Yes, our library has an exception design and development team — but like any project, we have worked just a bit beyond the edge of where rationality would have indicated. We've moved other projects forward — collaborating with others in the division on a data repository, online exhibits (Omeka) tool, other Samvera (formerly Hydra) work, and more.

Project Team (our department)

Ken Varnum
Project Management

Heidi Burkhardt
UX and Content Strategy

Ben Howell
UX and Accessibility

Rachel Vacek
Advocate, Cheerleader, Remover of Barriers

Albert Bertram
Back-End Development

Jon Earley
Front-End Development and Accessibility

Bridget Burke
Front-End Development and Accessibility

Trevor Dobias
User Interface Design

LIBRARY

@varnum @vacekrae @UMichLibrary

KEN

Library Participants

Kristen Castellana
Martha Conway
Scott Dennis
Gabriel Duque
Kat Hagedorn
Martin Knott
Shannon Moreno
Catherine Morse
Jon Rothman
Whitney Townsend

Plus….

Colleagues in LIT, Document Delivery, & Library Operations

And many others in the library

And a number of others, with formal and informal work responsiblitieS

Resources (tools, etc.)

- Confluence
- JIRA Service Desk
- JIRA Software
- GitHub
- InVision
- Slack

RACHEL

If you were to do this yourself, using our software, it would take you less -- we spent a lot of effort on infrastructure.

Jira Service Desk - ticketing system
Jira Software - project management tool

Politics & Governance

- Spoiler alert: Everyone has an interest in discovery
- Library created a project “steering team” with about a dozen members from across the library
- The active “project team” is part of the steering team, but is the group of us that is doing the research & development

RACHEL

And not just an interest in the service -- strong opinions firmly based in personal history.

Communication

- Alpha “roadshow” -- met with roughly 50% of library staff to explain approach & build support
- Regular all-staff emails about progress
- Frequent solicitations for feedback
- Campus communication
  - Links on current search tools
  - Campus daily news (week of 11/6)

RACHEL

Goals of New Discovery Environment

- Enable detailed usage information to contribute to library and learning analytics
- Establish a better understanding of how our resources are used
- Improve our ability to expose resources and expertise with the campus
- Free up instructional librarians to spend more time on information literacy and less on how to use the tool

RACHEL

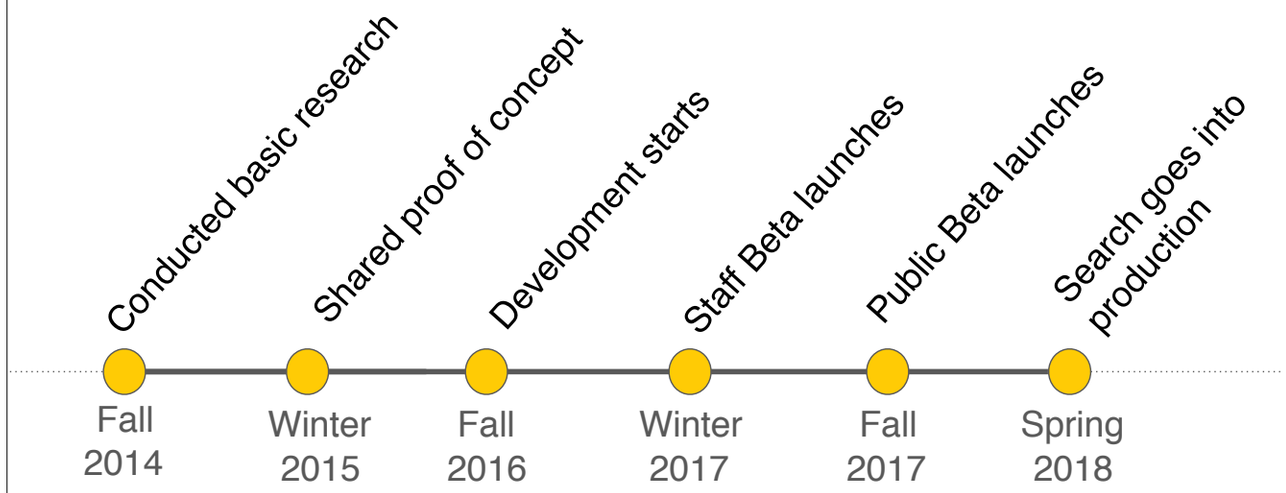
Goals of New Discovery Environment (cont.)

- Improve accessibility and responsive design for all users
- Allow for personalized search pages -- build your own bento box from anything we offer
- Allow us to more easily pull stats from each datastore for library and learning analytics
- Enable us to better understand usage patterns of our resources

RACHEL

And now I'm going to switch gears a bit, to talk in more detail about the software development project itself, rather than the environment and background. This is where I would have taken over from Rachel.

Timeline



KEN

Did basic research in 2014
Proof of concept shared with library winter 2015
Development work started fall 2016
Staff beta April 2017
Public beta November 2017
Launch May 2018

From Vision to Prototype

We started with a working prototype

Grayscale functioning tool that demonstrated the core functionality

Took it on a road show -- about 20 meetings with 175 of 450 library staff

Validation of direction

So then we scrapped it and remade it

KEN START

Search Project Team

Established a team to guide the tool's development

Drawn from across the library

Roles

- Set functional specifications
- Provided feedback on designs & early developments
- Brought library voice to our user-driven research

KEN

UX Successes

- Iterative process
  - Put prototypes in front of stakeholders and users more quickly to inform design changes
  - Set up feedback mechanisms for library staff to provide feedback

- Relationships
  - Improved credibility with other library staff by including them in usability tests and sharing results frequently
  - Created a shared understanding on project team
  - Recruited repeat participants for different studies - they 💖 libraries

LIBRARY

@varnum @vacekrae @UMichLibrary

KEN

- Iterative process
  - Put prototypes in front of stakeholders and users more quickly to inform design changes
  - Set up feedback mechanisms for library staff to provide feedback

- Relationships
  - Improved credibility with other library staff by including them in usability tests and sharing results frequently
  - Created a shared understanding on project team
  - Recruited repeat participants for different studies - they 💖 libraries

Students and staff saying that they'll actually use search now.

Agile Development Process

Two-week long sprints

- Start with a planning meeting
- End with a retrospective
- Merged into one meeting

Regular “grooming” sessions

- 60-90 minute meeting a week
- Do we have right work described to get the next goal done?
- Regular reprioritization

Daily Slack stand-ups

It’s a learning process (at least for me)

Design, Build, Design, Build, Repeat

Features

- Work with search results
- Guide the user
- Alternate displays
- Favorites

UX Research

- Interfaces & interactions
- Mobile interface
- Accessibility
- Analytics

KEN

- Select and act on search results
- Subject & Service Expertise
- Browse journals & browse databases
- Favorites application

What We Aim For

Sprint 1	Sprint 2	Sprint 3
Test Feature A	Test Feature C	Test Feature B
Design Feature B	Design Feature D	Design Feature E
Develop Feature C	Develop Feature B	Develop Feature D

Technical Specs

Tools

- Ruby
  - Rails
  - Blacklight *
- Front-end JavaScript
  - React
  - Pride **

APIs

- Summon
- Solr
- Custom APIs for Aleph holdings

Search application: https://github.com/mlibrary/search

* https://github.com/cul/clio-spectrum
** https://github.com/mlibrary/pride

KEN
Main **framework candidates** were: React (is not a framework, only a library for creating UIs), Mithril (micro-framework, very light weight), Ember, and Angular. And if you have specific questions about these, you probably know more detail than I do— but I'd be happy to connect you with my colleagues who know more.

React and Angular seem to have some of the strongest communities behind them. Partially because their respective developers Facebook and Google have deep pockets.

React was attractive because it only handles rendering, while in contrast **Angular** is a full MVC framework that made assumptions about the models that were untrue of our environment.
- We went with React because it has momentum, and it didn't bring along extra baggage.
- React makes it very easy to manage the UI application state, something that was traditionally difficult and can cause issues.
- We prefer to use smaller libraries (Like React) over monolithic frameworks (Angular, Ember). It allows : us to be flexible, write less code, and make unique applications for our own specific needs.
- React has a thin, but powerful and easy to remember API. Developers end up writing more application specific code (good), rather than framework boilerplate (not so good).

Pride is a small library we developed in-house (available on our U-M Library GitHub) that communicates with our Ruby back-end.
- Pride makes smart decisions about what resources to request or cache and make available to the UI. This improves the speed and performance when interacting with the interface.
- For example: navigation between datastores with the same search term is instant.
- And as a bonus we now have a decoupled reusable library for easily accessing our resources.

We also use **Blacklight** inside of the Ruby part.
- We based that code off of **Columbia University's Clio-Spectrum** project. (Their Unified Search & Discovery tool available on Github).
- We do not, however, use Blacklight for rendering html. That's where React comes in.

A Demo is Worth 1000 Words (or 100 well-chosen curses)

https://search.lib.umich.edu/

KEN

https://search.lib.umich.edu/

Roadmap -- Post-Launch

Features

- New books tool
- Additional sources
- Personalized bento box results
- Discovery of additional resources as well as relevant locations, services, and events tied to search results
- Allow for build-your-own bento box from anything we offer

...and ongoing UX research

KEN

Features
- New books tool
- Additional sources
- Personalized bento box results
- Discovery of additional resources as well as relevant locations, services, and events tied to search results
- Allow for build-your-own bento box from anything we offer

...and ongoing UX research

Potential personalization features

If logged in locally:

- Customizable search panel
- View Favorites
- View additional datastores that display relevant materials based on course registrations
- Receive relevant current awareness services

If logged in elsewhere:

- Access widgets in Canvas that displays relevant library resources for each course

Investing Time Now to Save Time Later

Designing up front slows incremental development…

But saves long-term maintenance and re-designs

My team keeps me honest

KEN
It has been well documented in many disciplines that fixing errors in later phases of design can be expensive. Best to take the time to plan ahead of time. Want to do it right, not at the speed of light.

Yes, we developed this tool iteratively, but we spent lots of time doing the initial planning, research, and making sure what we were planning really was what users were wanting and needing.

Questions?

Ken Varnum
Senior Program Manager
varnum@umich.edu

Rachel Vacek
Head of Design & Discovery
rvacek@umich.edu

University of Michigan Library GitHub site
https://github.com/mlibrary

LIBRARY

@varnum @vacekrae @UMichLibrary

RACHEL