Digitized Text Services, OpenURL link resolvers, and Umlaut

Jonathan Rochkind
DLF Fall Forum 2008

rochkind@jhu.edu

http://docs.google.com/Presentation?id=dfd2t3bt_5hkbxrdc

Straight to the demo!

the open source Umlaut link resolver

http://findit.library.jhu.edu/go/752150

- Search Inside from:
  - Google
  - Amazon
  - HathiTrust
- Open Access full text from:
  - Google
  - HathiTrust (only when no Google hit)
  - InternetArchive
- Covers from:
  - Amazon, Google, LibraryThing, OpenLibrary

More demo!

http://findit.library.jhu.edu/go/955948

- Limited excerpts from:
  - Google
  - Amazon

http://findit.library.jhu.edu/go/750567

- Author biographical information
  - Worldcat Identities

And, in the catalog:

http://catalog.library.jhu.edu/ipac20/ipac.jsp?index=BIB&term
29826

- Integrated once using convenient Umlaut APIs and javascript helpers.
  - New plug-ins in Umlaut, automatically show up in OPA

A step back to consider

What is an OpenURL link resolver anyway?

Too Narrow

That thing which gets us full text for articles

Too broad

Anything!

- We will be thinking only of scholarly citation (SAP1/2) OpenURLs and link resolvers, as this is what actually existing products are.
- Do one thing and do it well.

Just Right

Software that takes a scholarly citation formatted as an OpenURL, and provides the user with services.

Umlaut

- Umlaut is a link resolver
- Umlaut does not have it's own knowledge base of license materials
    - Uses SFX (via API) for knowledge base of licensed resources, and article-level linking to those resources SFX can handle.
    - Disentangle knowledge base from interface and value added services.
    - Umlaut could hypothetically work with other link-resolv type knowledge base products with APIs.

What is the role of a link resolver?

In the library infrastructure we can imagine

Role 1: The electronic front-door to the library

When the user finds a known item, possibly in a non-library third party application, and wants to know what her library ca do for her with that item.

- Not just articles.
- Not just from licensed content providers
- With browser extensions like LibX if necessary, from ***anywhere*** our users are

Umlaut designed with electronic front door in mind

Any services for a known-item we can think of and manage t
include

Including print call number and availability and direct links to
OPAC crucial.

- http://findit.library.jhu.edu/go/955948
- Abolish unnecessary extra clicks!

What is the role of a link resolver?

In the library infrastructure we can imagine

Role 2: A known item service provider used by other software

Many interfaces we need known-item services, often for the same type of interfaces.

Why implement these services multiple times?

"Single business" model, and good software engineering design says put this in one place.

Not neccesarily formal "SOA", but the same goals.

Umlaut does that too

With plug-in architecture
Additional external services added in individual components

With full-featured suite of APIs
Including all Umlaut fetched services

- Full API
- Partial HTML API
- Javascript 'widget' style helpers

Interesting Problems

- Wait time for multiple services
  - Solved with background fetching, with AJAX updating
    - Non-AJAX option
    - Background fetching advertised by APIs
- Metadata matching
  - Identifier or keyword are used depending on capabilitie of foreign service, and metadata in possession.
- Lack of sufficient APIs in third parties.
  - Google lack of detail
  - Amazon screen scrape neccesary
  - Some services not possible.

No shortage of interesting problems.

- Bad metadata
  - Eg local; also Google.
  - Local staff/patrons don't care that it's a third party
- Multiple version issue
  - Include alternate versions or not, what's the right answ
  - Can it be achieved anyway?
  - Multi-volume works tricky
- Terms of service
  - Not always clear
  - Amazon just (this week) changed interface/access negatively

Umlaut collaboration?

- Open source, Ruby on Rails.
- Collaborative open source is sustainable open source.
    - interested?
- Why is nobody interested in using it?

http://wiki.code4lib.org/index.php/Umlaut

Jonathan Rochkind
rochkind@jhu.edu