IntegraL: Lightweight Link-based Integration of Heterogeneous Digital Library Collections and Services in the Deep Web

Min Song
Information Systems Department
New Jersey Institute of Technology
song@njit.edu

Michael Bieber
Information Systems Department
New Jersey Institute of Technology
bieber@njit.edu

Abstract

IntegraL is a digital library system demonstrating a lightweight system integration technique for digital library collections and services. Digital library systems generally require integration with minimal or no changes to their code. IntegraL users see a totally integrated environment. They use their digital library system just as before. They also see extra link anchors. Selecting one generates a list of links to relevant meta-information (structural, content-based and knowledge-sharing relationships, and metadata). IntegraL generates the vast majority of supplemental link anchors and meta-information links automatically through the use of relationship rules. This paper presents the concept of meta-information, describes the IntegraL infrastructure and architecture supported by single sign-on authentication middleware, and explains how systems can integrate into the infrastructure. This research's primary contribution is providing a relatively straightforward, sustainable infrastructure for integrating digital library collections and services.

1. Introduction

The Deep Web refers to the pages on the Internet that are not indexed by conventional search engines such as Google or Yahoo [15]. Most of the deep web is made up of information found in specialized databases. Each of these databases can be searched, much like searching Google, but the results are often delivered to you in web pages that are made just in answer to your search. These pages are not stored anywhere, rather they are created "on the fly." It is easier and cheaper for these databases to spontaneously generate the answer page for each search than to store all the possible pages containing all the possible answers to all the possible searches people could make to the database. To tackle this issue, we propose the IntegraL (Integrated Library) system that provides lightweight digital library integration through automated linking in the deep web.

IntegraL supplements collections by linking them automatically to relevant services and related collections. Users see a totally integrated environment, using their system just as before. However, they will see additional link anchors. When clicking on one, IntegraL generates a set of supplemental links. IntegraL utilizes collaborative filtering [8] to filter and rank order this set to user preferences and tasks. IntegraL provides a systematic approach for integrating digital library systems (and by extension, any other system with a Web interface). Our approach is “lightweight” or “non-intrusive” and relatively “uncoupled” because integration requires little or no change to a system’s source code. Collections and services can still operate independently of IntegraL after integration [1].

A major research goal of the IntegraL project is to develop a structure for providing users with comprehensive meta-information. Meta-information includes the structural relationships (links based on element type), lexical relationships, knowledge-sharing relationships, and meta-information around an element of interest. Combined, the meta-information goes a long way towards establishing the full semantics for (the meaning of and context around) a system’s elements. IntegraL’s approach is entirely different from federated search and metasearch [10]. The vast majority of IntegraL links are not found through searching. Instead they are specified through structural relationships. These are pre-specified through the relationship rules [9] by element type (i.e., link skeletons are pre-specified for any author, any date, any space mission, any document, any stock, etc.)

1530-1354/08 $25.00 © 2008 IEEE
DOI 10.1109/CEC/EEE.2008.32

In many ways, IntegraL is a link resolver service, in that it generates a set of relevant links to information resources, and when the user selects one, IntegraL forwards appropriate commands and parameters to have that information presented to the user. Link resolvers (LinkFinderPlus, SFX, 1Cate, etc.) using OpenURL protocols and Digital Object Identifiers (DOI) are used to bridge the gaps between the "silos of information" of separate library database systems. Link resolvers primarily link citations to accessible copies of the cited resource (although they also may link citations or citation elements to general web searches) [3] [11].

IntegraL's approach provides advantages over the cross-database linking provided by library vendors such as EBSCO, WebFeat, and others. IntegraL provides information seekers with links at the point of need within the native mode of the system or database being used that lead directly to related materials in an entirely independent system, without intervening steps or proprietary software.

Digital library systems should find several benefits to integrating through IntegraL:

- IntegraL virtually enlarges the size of a collection and the "feature set" or services that a system provides through links to related information and relevant services that external systems provide.
- Users become aware of other systems through seeing links to them within other systems. Similarly, IntegraL causes users to be aware of the kinds of information and services that are available, because of these links.
- IntegraL streamlines individual systems by providing direct access through links among a single system's information and functions, sparing the user from navigating through a possible series of menus.
- IntegraL's lightweight approach is cheaper in time and resources than other integration approaches, with minimal or no changes to a system's documents or code.

The rest of paper consists of the following chapters: Section 2 describes meta-information. Section 3 depicts the overall architecture of IntegraL. Section 4 demonstrates the web interface of IntegraL. Section 5 concludes the paper.

2. Meta-information

The notion of meta-information expands on what people typically consider metadata. Whereas metadata often describes characteristics of an element of interest, surrounding relationships often point to other entities or documents, as well as to functions (services) that can be executed over aspects of that element. Meta-information includes structural relationships, lexical relationships, user-declared knowledge-sharing relationships, as well as the metadata around an element of interest [6] [9].

Structural relationships apply to an entire class of elements in an information domain. Structural relationships are inherent to the design or "structure" of the system. A database entity-relationship diagram, for example, contains structural links. Structural links can connect the equivalent element, such as the same author or subject. They also can connect related elements (such as teaching materials on a particular subject or documents with a common author) or characteristics of an element (such as an author's address and background). The connected elements can be in the same or different systems. Thus a user may follow a link from an element in a specific system to a related element in a completely different system [13].

Lexical relationships contribute to understanding the context around an element of interest. While equally important, they are not necessarily fixed in the structure of the related systems. Instead they are based on the display content. For textual content, lexical relationships typically are found using one or more of the many lexical analysis techniques ranging from simple keyword search to cluster analysis.

Knowledge-sharing relationships can be provided by authors within digital library systems. But they especially encourage users within the digital library's community to interact with each other and participate within the digital library, thus promoting the ideal from the hypermedia research community of the "reader as author" [2] [4] [12]

One begins by identifying the elements of interest for which one wants to provide structural relationships and meta-data. For existing systems, one can look at screen shots to identify the elements of interest that a user might want to request meta-information about. When designing a new system, one can identify elements of interest (entities) from the use cases, a standard part of a formal systems analysis. For each element of interest, one asks a domain expert a series of questions to elicit characteristics about it and the relationships around it.

3. System Architecture

In this section we describe the general steps for integrating a system with the IntegraL infrastructure, and present an overview of our architecture.

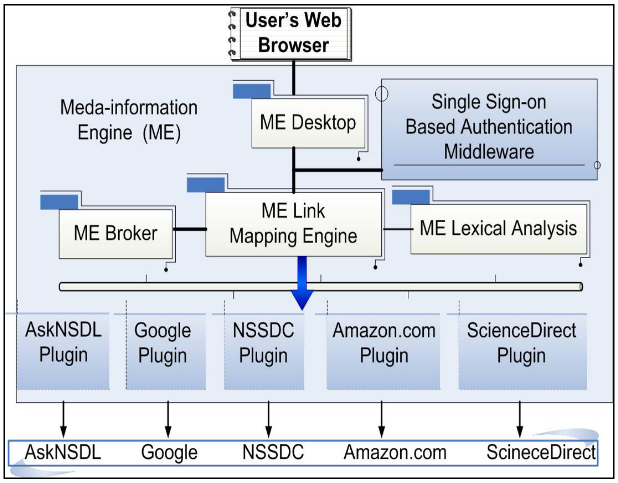


Figure 1: IntegraL High-level Architectural Overview

IntegraL is a loosely coupled system, where various components communicate with each other via messages that conform to a well-defined standardized internal protocol. This approach allows new components to be developed and added without affecting existing components and functionality.

Figure 1 shows an overview of the IntegraL architecture. The core IntegraL "meta-information engine" (ME) resides on a server, which users access automatically through a proxy setting in their web browser. The ME consists of five primary components:
1) The Desktop translates the displayable portion of IntegraL's internal messages, from the standard internal XML format to a format that can be displayed to a user via a Web browser and vice versa.
2) The Broker enables the communication between the IntegraL ME modules. All IntegraL messages pass through the Broker, which then redirects them to the appropriate component.
3) The Relationship Engine maps the system data and relationships to links at run-time. The Relationship Engine maintains a repository of meta-information rules. When a screen is being sent to the IntegraL Desktop for display, the Relationship Engine retrieves all relevant rules for each element in that screen. The Desktop then converts the elements to link anchors and the relationships to links. The other three components will be described the following sections.

3.1 Single Sign-on based Authentication

IntegraL seeks to seamlessly integrate traditional and digital libraries, as well as bring relative content directly to users through an authenticated manner. The concept of Single Sign-On (SSO) plays a very important part in this, since IntegraL relies on a lightweight method of integration that minimizes user intervention. The followings are contributions made by our SSO authentication middleware:

- Ability to enforce uniform authentication across the organization deploying IntegraL
- End to end user audit sessions to improve security reporting
- Removes developers of plugins from having to understand and implement identity security in the private digital libraries

We employ Shibboleth for the authentication middleware for our system. Shibboleth is an open source solution to Single Sign-On [7]. It allows browser users to access online resources across several domains that would normally require multiple usernames and passwords. Shibboleth preserves the privacy of its users browsing and the security of the resources accessed. Shibboleth's implementation of Single Sign-On allows users who fall within IntegraL's federation to take advantage of the resources and services that are linked [7]. When a user requests a Shibboleth-protected resource, IntegraL will initialize authentication by redirecting the user to the Shibboleth WAYF (Where Are You From) service. From there, the user selects the institution for which he or she belongs to, and logs in. Shibboleth's purpose of inter-institutional web resource sharing has a somewhat custom implementation in the IntegraL project by protecting authentication scripts on the Service Provider (SP) side. For non Shibboleth-protected resources, the system provides an option for users securely to store their log-in information with IntegraL. Since IntegraL acts as an unobtrusive proxy between the browser-user and the web, access information must be carried with the user and exchanged with the proxy while browsing. This access information is carried and exchanged in the form of a session ID (SID). The SID guarantees that the user has been authenticated through Shibboleth and that he or she is authorized to browse the web.

Shibboleth plays two distinct roles in IntegraL. Perhaps the more important role is that which grants access to protected resources throughout the federation. In this situation, there is no true interaction, or integration, between Shibboleth and IntegraL's proxy. Instead, IntegraL's implementation of Shibboleth initializes the Single Sign-On process, which allows for seamless browsing of Shibboleth protected resources within the federation; once a user authenticates with his or her corresponding Identity Provider (IdP), access to content protected by Service Providers within the federation is granted.

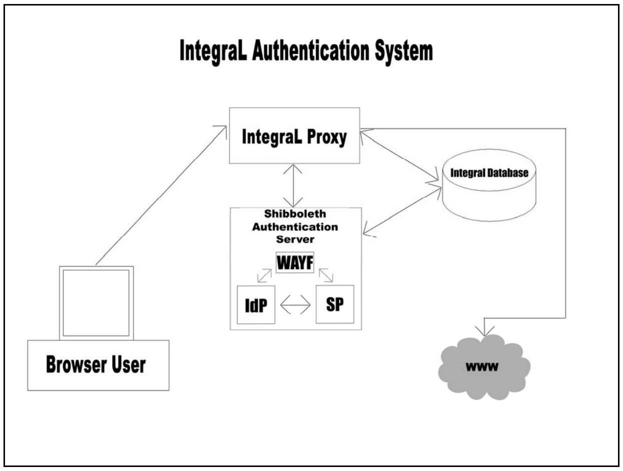

IntegraL Authentication System
IntegraL Proxy
Integral Database
Shibboleth Authentication Server
WAYF
IdP
SP
Browser User
www

Figure 2: Shibboleth's based Single Sign-on Authentication

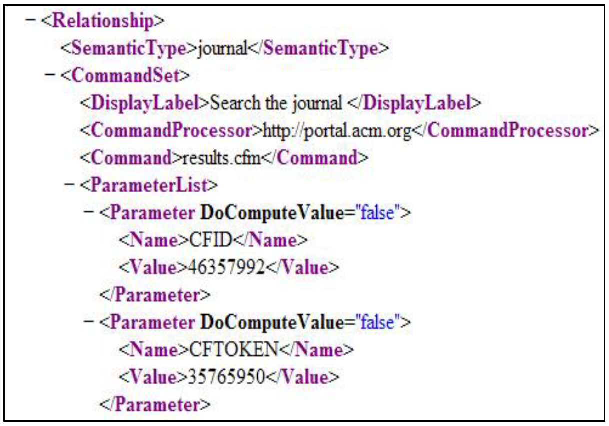
The second role that Shibboleth plays in the IntegraL project is that which allows for access through the IntegraL proxy, as well as seamless authentication to non-Shibboleth protected resources. As mentioned above, scripts protected by IntegraL's Service Provider generate a SID in the form of *username:random hash:timestamp*. The SID is carried with the user by way of a query variable embedded in the URLs of websites that the user visits. Each request is inspected for the SID by the IntegraL proxy and is matched up with an entry in the IntegraL database. If a user is not authenticated, he or she is sent to the WAYF to log in. However, if it has been determined that the user is indeed authenticated through Shibboleth and that the URL simply did not contain the SID (i.e. the user typed in a new URL in the web browser), server-side scripts reinsert the SID and Shibboleth's support for a "lazy session" creates a seamless transition without the user ever noticing the technical behind-the-scenes work. Shibboleth's custom implementation in IntegraL makes it possible for users to access Shibboleth protected resources while guaranteeing that users have authenticated with IntegraL. While the description of IntegraL's authentication system and its use of Shibboleth may seem overwhelmingly complex, containing many "moving parts", the methods by which IntegraL implements allows for an enjoyable web browsing experience with seamless transitions between authentication scripts and authenticated resources with minimal user intervention.

3.2 ME Link Mapping Engine

The link mapping engine is the core layer to link related library resources "on the fly." The strengths of our link mapping engine include 1) the XML rule-driven content parsing and 2) the Object-oriented plugin development.

In the mapping engine, we separate the business logic as to instruction of content parsing from actual implementation specifics. A mapping rule kept in the XML rule file governs how to parse the web pages of the target digital resources.

Figure 3 shows the snippet of the mapping rules for the journal search at ACM Digital Library.



```
- <Relationship>
    <SemanticType>journal</SemanticType>
  - <CommandSet>
      <DisplayLabel>Search the journal </DisplayLabel>
      <CommandProcessor>http://portal.acm.org</CommandProcessor>
      <Command>results.cfm</Command>
    - <ParameterList>
      - <Parameter DoComputeValue="false">
          <Name>CFID</Name>
          <Value>46357992</Value>
        </Parameter>
      - <Parameter DoComputeValue="false">
          <Name>CFTOKEN</Name>
          <Value>35765950</Value>
        </Parameter>
```

Figure 3: Mapping Rule for Journal Search at ACM Digital Library

Upon receiving the response to the user's link request, IntegraL feeds it into so called "Plugin Manager" whose responsibility is to delegate it to appropriate plugins (wrappers) which perform content parsing (See Figure 4). With the object-oriented plugin architecture, this provides for a great flexibility, and makes the system extremely robust. Once the core was developed, any new functionality could then be introduced into the system by designing and implementing a piece of code, and this piece of code is referred to as a *"plugin"*. Not all requests are being handled by all plugins. Each plugin, before it registers with the system, announces a set of conditions to the plugin manager. When the response is finally delivered to the manager it checks that condition string against the response header and if they match, the response is forwarded to that plugin.

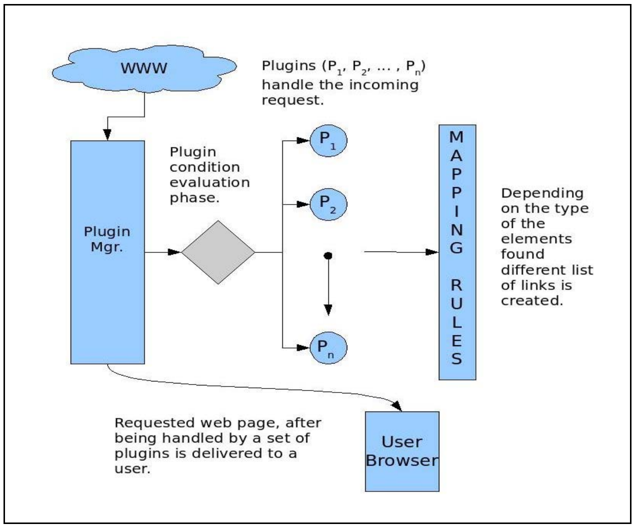

WWW
Plugins (P_1, P_2, … , P_n) handle the incoming request.
Plugin Mgr.
Plugin condition evaluation phase.
P_1
P_2
P_n
MAPPING RULES
Depending on the type of the elements found different list of links is created.
Requested web page, after being handled by a set of plugins is delivered to a user.
User Browser

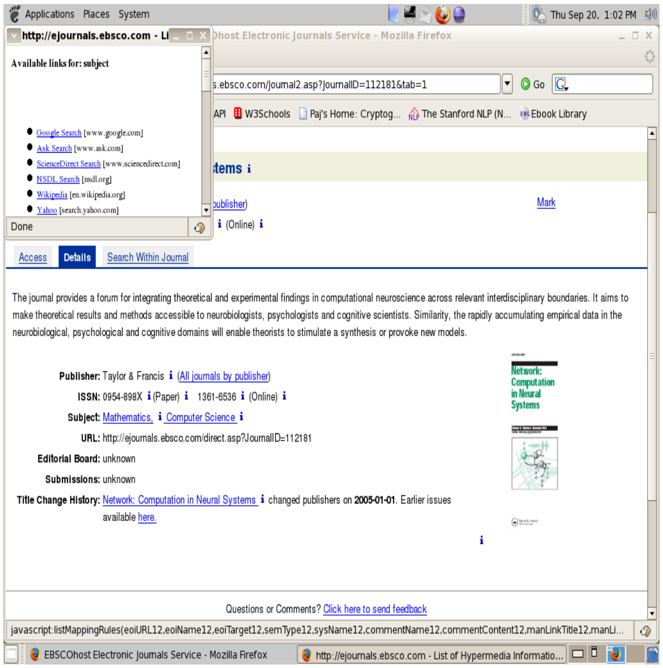
Figure 4: Link Mapping Procedure

3.3 ME Lexical Analysis

Our lexical analysis employed two novel techniques: 1) phrase extraction [14] and 2) named entity extraction techniques [5]. Based upon these two techniques, we have designed our plugins to parse and detect elements of interest (EoIs) on pages that are forwarded to them in a form of a HTTP response from the plugin manager. EoIs can be a multitude of things ranging from author names, titles, to publish dates, publishers, and many more. Each is elilgible to be the nexus of a wealth of meta-information for a group of users. We have utilized two techniques for detection of these special interest elements. One involves HTML tag parsing, which is aided in great deal by Scone Java library, which allows us to interpret the incoming response as a stream of HTML tokens. Based on a predefined sequence of tokens we are able to identify where, and what type of an element that is. The second technique is based on the idea of "phrase extraction" and "named entity recognition" in a sense that it does not require a separate plugins for different web pages. We have used Stanford Natural Language Processing libraries extensively to aid in a more automatic way of detecting EoIs [5]. Both of these techniques have their strengths and weaknesses, but to discuss those would require more in depth description of the system. Once the EoI has been detected, an information icon is inserted into the response, so that when viewed by the user, the user will see a small "i" icon next to it. When clicked, a small window is displayed, and a list of meta-information links for this EoI is shown. If we assume the EoI is of type "author", then every entry in the list will point to databases on the Internet that carry some information about this particular author. Links are created on the fly from a predefined template, and this whole operation is executed by the "mapping rule engine". The mapping engine, upon being delegated to produce a list for a specific EoI type, consults the template, and based on the content of the EoI, it produces desired links.

4. Integral System

In this section, we describe the web interface of the IntegraL system. Figure 5 presents a screenshot from our current system, which can be accessed from the project web site along with the pop-up window listing other external resources. IntegraL supplements collections by linking them automatically to relevant services and related collections. IntegraL supplements services by automatically giving relevant objects in collections (and other services) direct access to these services.


Applications Places System
Thu Sep 20, 1:02 PM
http://ejournals.ebsco.com - Li
Ohost Electronic Journals Service - Mozilla Firefox
s.ebsco.com/Journal2.asp?JournalID=112181&tab=1
Go
API
W3Schools
Paj's Home: Cryptog...
The Stanford NLP (N...
Ebook Library
Done
Access
Details
Search Within Journal
EBSCOhost Electronic Journals Service - Mozilla Firefox
http://ejournals.ebsco.com - List of Hypermedia Informatio...

Available links for: subject

- Google Search [www.google.com]
- Ask Search [www.ask.com]
- ScienceDirect Search [www.sciencedirect.com]
- NSDL Search [nsdl.org]
- Wikipedia [en.wikipedia.org]
- Yahoo [search.yahoo.com]

stems i

ublisher)

Mark

i (Online) i

The journal provides a forum for integrating theoretical and experimental findings in computational neuroscience across relevant interdisciplinary boundaries. It aims to make theoretical results and methods accessible to neurobiologists, psychologists and cognitive scientists. Similarity, the rapidly accumulating empirical data in the neurobiological, psychological and cognitive domains will enable theorists to stimulate a synthesis or provoke new models.

Publisher: Taylor & Francis i (All journals by publisher)
ISSN: 0954-898X i (Paper) i 1361-6536 i (Online) i
Subject: Mathematics, i Computer Science i
URL: http://ejournals.ebsco.com/direct.asp?JournalID=112181
Editorial Board: unknown
Submissions: unknown
Title Change History: Network: Computation in Neural Systems i changed publishers on **2005-01-01**. Earlier issues available here.

Network: Computation in Neural Systems

i

Questions or Comments? Click here to send feedback

javascript:listMappingRules(eoiURL12,eoiName12,eoiTarget12,semType12,sysName12,commentName12,commentContent12,manLinkTitle12,manLi...

Figure 5: Example of Searching EBSCO with a Pop-up Window to Link to External Resources

Users see a totally integrated environment, using their system just as before. However, they see additional link anchors, and when clicking on one, IntegraL will present a list of supplemental links. Users make queries into the EBSCO database from a query form. The EBSCO "plugin" (one possible approach to integration) parses the query result, identifying EBSCO documents and launch date elements. IntegraL added supplemental anchors on the document for these elements (indicated by the circled "i" in the publisher, ISSN, and subject field). The user clicked on a document anchor. IntegraL then inferred the list of links shown from its base of relationship rules for the kind of element selected. When the user clicks on the document identifier "Subject," IntegraL generates a list of links for document identifiers (for elements of type "document") such as Google, Ask, ScienceDirect, and NSDL.

There are several advantages of using the Integral system to deep web. First Integral provides users with various access points to dynamic content which are returned in response to a submitted query or accessed only through a form, especially if open-domain input elements (such as text fields) are used

Second, the Integral system enables users to navigate unlinked content which are not linked to by other pages. This content is referred to as pages without backlinks. Third, the Integral system provides links to private web which consists of sites that require registration and login by the Shibboleth-based SSO.

5. Conclusion

In this paper, we introduce the IntegraL system, designed and developed to support the principle of lightweight system integration through linking of heterogeneous digital collections.

IntegraL research makes several contributions:
• an architecture for lightweight integration of digital library systems through linking
• a systematic approach to integrating digital library collections and services, with other appropriate collections and services applying the concept of meta-information to digital libraries for specifying the structural, lexical and knowledge-sharing relationships (as well as metadata) around elements of interest

This research provides a new means for making digital library services interoperable. IntegraL facilitates a virtual restructuring of public web spaces and services, with authenticated digital libraries into broad "federated" digital library spaces constructed from numerous interrelationships. Elements reside within a rich context of meta-information that helps users understand and work with them. This provides a ripe environment for organizations and individual people to develop small, specialized collections and services, which automatically become part of the federated space and accessible to those they can benefit. IntegraL extends the boundaries of how we think about and interact with digital libraries. As future studies, we are undertaking user interface studies to determine the best ways to display meta-information, as well as link anchors, links and lists of links to meta-information. In addition, we will explore creating new services from existing ones. For example, we could string together a document summarizer service with a standard Web language translation service to create a new "summarize this document in my language" service.

6. Acknowledgements

Partial support for this research was provided by the National Science Foundation under grants DUE-0434581 and DUE-0434998, by the Institute for Museum and Library Services under grant LG-02-04-0002-04, and by the New Jersey Institute of Technology.

7. References

[1] Barrett, D.J., et al. (1996). A Framework for Event-Based Software Integration, ACM Transactions on Software Engineering and Methodology, 5(4).

[2] Chin, A. and Chignell, M. (2006) A social hypertext model for finding community in blogs Full text, Proceedings of the seventeenth conference on Hypertext and hypermedia table of contents Odense, Denmark, 11 - 2

[3] Collins, Maria D. and Christine L. Ferguson (2002). "Context-sensitive Linking: It's a Small World After All," Serials Review, 28(4), 267-282.

[4] Cotkin, George (1996). 'Hyping the Text': Hypertext, Postmodernism and the Historian. American Studies 37: 103 116.

[5] Finkel, J. R. Grenager, T. and Manning, C. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 363-370.

[6] Galnares, R. (2001). Augmenting Applications with Hypermedia Functionality and Metainformation. Ph.D. Thesis, New Jersey Institute of Technology, Newark, NJ 07102.

[7] Gourley, D. (2003) Library Portal Roles in a Shibboleth® Federation, http://shibboleth.internet2.edu/docs/gourley-shibboleth-library-portals-200310.html

[8] Im, Il and Hars., A. (2007). Does a One-Size Recommendation System Fit All?: The Effectiveness of Collaborative Filtering Technology Across Different Domains, User Groups, and Search Modes. ACM Transactions on Information Systems 26 (1).

[9] Joseph, C. Nkechi Nnadi, N., Zhang, L., Bieber, M., and Galnares, R. (2004) Ubiquitous Metainformation and the WYWWYWI* Principle, Journal of Digital Information, 5(1), April 2004.

[10] Liu, K., Meng, W., Qiu, J., Yu, C., Raghavan, V., Wu, Z., Lu, Y., He, H., Zhao, H. (2007). AllInOneNews: Development and Evaluation of a Large-Scale News Metasearch Engine. 26th ACM SIGMOD International Conference on Management of Data ACM (SIGMOD 2007), Industrial track, pp.1017-1028, Beijing, China, June 2007.

[11] Munson, D. M. (2006) Link Resolvers: An Overview for Reference Librarians, Internet Reference Services Quarterly, 136: 7 – 28.

[12] Nielsen, Jakob. (1995). Multimedia and Hypertext: The Internet and Beyond. Boston: AP Professional.

[13] Rubart, J. (2007). Architecting structure-aware applications. Hypertext 2007: 185-188.

[14] Song, M., Song, I.-Y., Hu, X. (2003). KPSpotter: a flexible information gain-based keyphrase extraction system. WIDM 2003: 50-53.

[15] Bergman, M. (2001). The Deep Web: Surfacing Hidden Value. Journal of Electronic Publishing.