White Paper Report

Report ID: 98611

Application Number: PK5002707

Project Director: Michael Buckland (buckland@ischool.berkeley.edu)

Institution: University of California, Berkeley

Reporting Period: 10/1/2007-9/30/2010

Report Due: 12/31/2010

Date Submitted: 2/7/2011

Final Performance Report

Award Number: PK-50027-07

Project title: Context and Relationships: Ireland and Irish Studies

Project Director: Michael Buckland

Grantee Institution: Regents of the University of California
Berkeley campus: Electronic Cultural Atlas Initiative,
School of Information,

Date submitted: February 7, 2011.

Final Performance Report

Context and Relationships: Ireland and Irish Studies

A project supported by the National Endowment for the Humanities and the Institute of Museum and Library Services through the Advancing Knowledge Program award number: PK-50027-07

Grantee Institution: Regents of the University of California
Berkeley campus: Electronic Cultural Atlas Initiative,
School of Information,

Project Director: Michael K. Buckland

February 7, 2011.

INTRODUCTION

The difference between seeing and understanding anything lies in knowing the context. Who were the people involved? Where was this place? What else was going on at or before this time? Why was this event significant? . . . and so on. Large reference collections reveal the massive cumulative investment of effort and expertise in rich genres of reference resources to provide contextualizing explanations. But the benefit reaped, the return on this investment, depends on the use actually made of these expensive resources. Even effective tools will be little used unless they are extremely easy to use and they are conveniently accessible within users' working habits.

The move to a digital environment offers exciting new technical options for the publication, access, and use of trustworthy, specialized scholarly reference resources, but the impact has instead been mainly to focus attention and use on the resources that are easiest to use, notably Google and Wikipedia. The rich resources of the reference library have become less visible and the extreme convenience of Google and Wikipedia provide a seductive and very popular alternative. The reference resources and the resource guides and pathfinders crafted by librarians and scholars remain underutilized. The return on investment in reference resources will remain much less than it could be without a dramatic increase in use. Use is likely to increase significantly only if there is a substantial improvement in ease of use. And, as with any retail or service situation, substantially increased demand can be handled economically only by moving customers towards increased self-service. The design challenge we have addressed is this: How can the use of trustworthy, scholarly reference resources be made as easy and convenient to use as Google and Wikipedia?

The Student in the Dorm

A student struggles to complete an assignment in her dorm room at 3:00 a.m. Closely related texts exist, as do relevant reference resources and library guides that could help her. She may be aware they exist, but unaware of the details and, in practice, she makes no use of them.

Usually, just a few reference resources suffice if they are the optimal selection for that particular purpose and for that particular person. In a print on paper environment her situation would be transformed if she were working in a good reference library or if, just when she needed them, a librarian delivered to her the few best resources for her for this assignment and a different set for a different assignment and different again for another student with different knowledge and skills. The central question for this project was: *How could we provide that amenity in a digital environment?*

Our iconic student working at 3:00 a.m. is mythic, but the design challenge is of great significance because the need is pervasive. For most people in most places most of the time finding the few best reference resources for any task is not conveniently practical on paper or online, but that is what is needed and what should be provided. We will demonstrate how the reader can become connected with pertinent resources

THE CONTEXT FINDER

The central concept of this project is an innovation that we have called the "Context Finder" envisioned as a tool for quickly accessing reference resources while reading any unfamiliar text. During the course of this project, we implemented three prototype Context Finders.

Prototype I: Finding Context in Web Pages

Using the first Context Finder prototype, one could access encyclopedia articles from Wikipedia and Citizendium through a Web proxy. When the Context Finder proxy received a request for an article, it first checked its database to see if information about the article was stored there. If not, it fetched the article content from Wikipedia or Citizendium and analyzed it for names of people and places using the OpenCalais web service. These names and their locations were stored in the Context Finder database. The prototype then generated and displayed an interface showing the original article text with a sidebar on the left listing the person and place names that had been identified (see Figure 1).

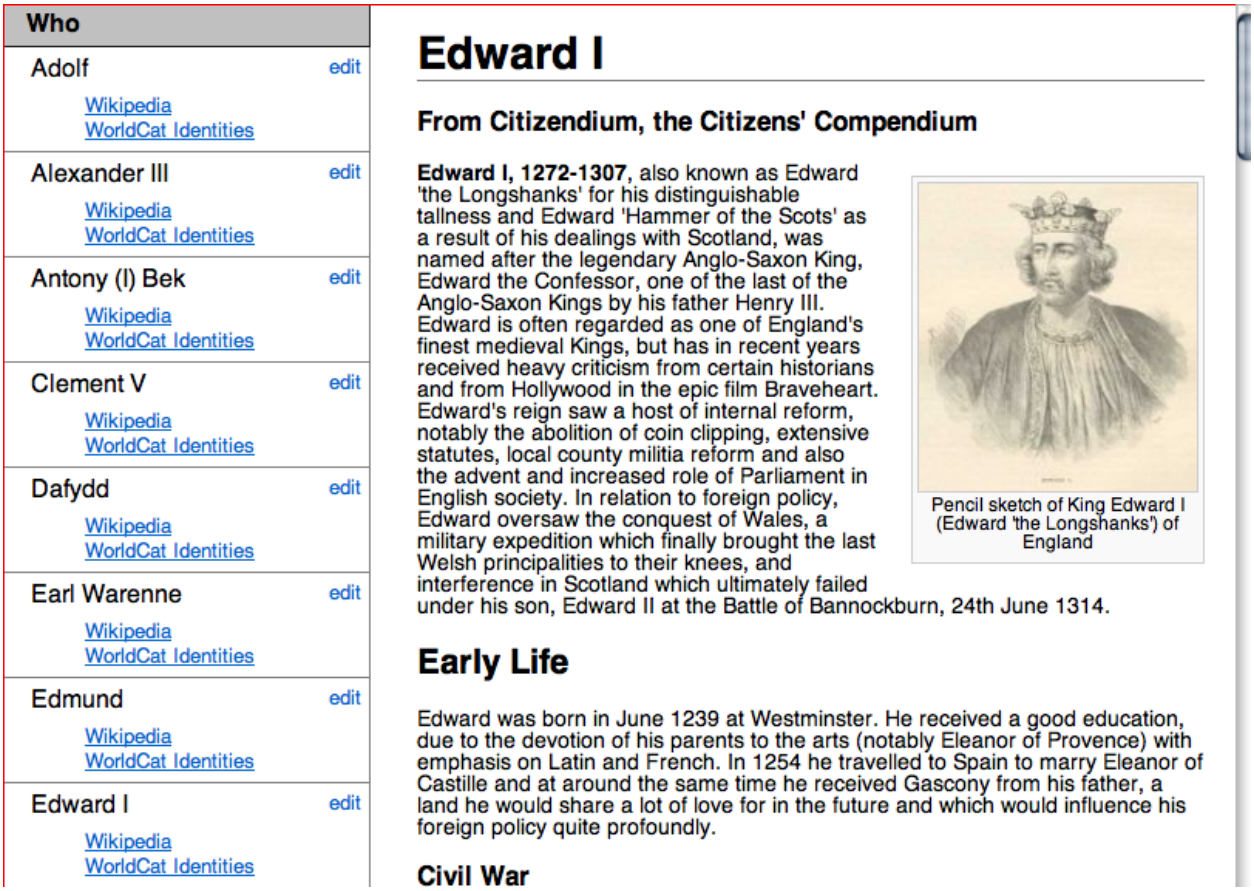

Figure 1: Displaying named entities identified in a Citizendium article.

Hovering over items in the sidebar highlighted all the locations in the article where that name (or a variant name or pronoun referring to that name) appeared (see Figure 2).

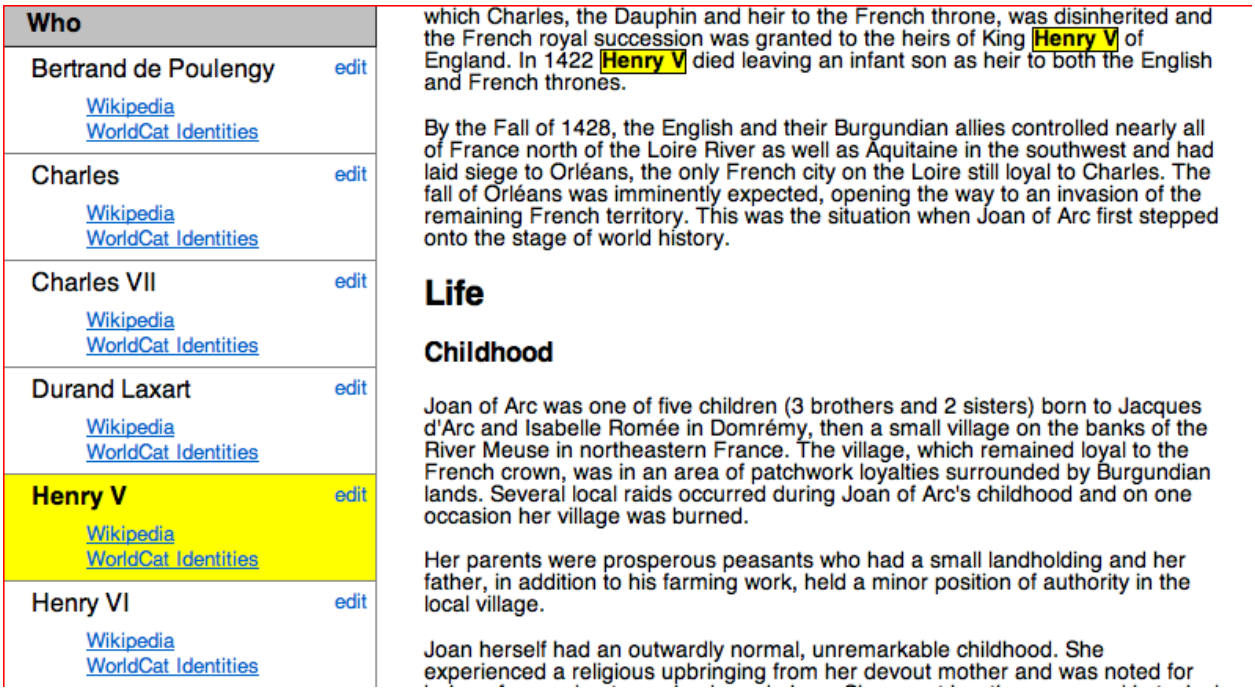

Figure 2: Highlighting person names in the text of an article.

Each name in the sidebar also included links to resources where more information about that name could be found. Personal names were linked to Wikipedia and WorldCat Identities, and place names were linked to Wikipedia and the GeoNames gazetteer. Initially, these links were simply "search" links for the resource: following them would execute a keyword search for the name within the specified resource. For example, following the WorldCat Identities link from the "Henry V" item, one would be presented with a list of search results for "Henry V", including both *Henry V King of England 1387–1422* and *Henry V Holy Roman Emperor 1086–1125* (see Figure 3).

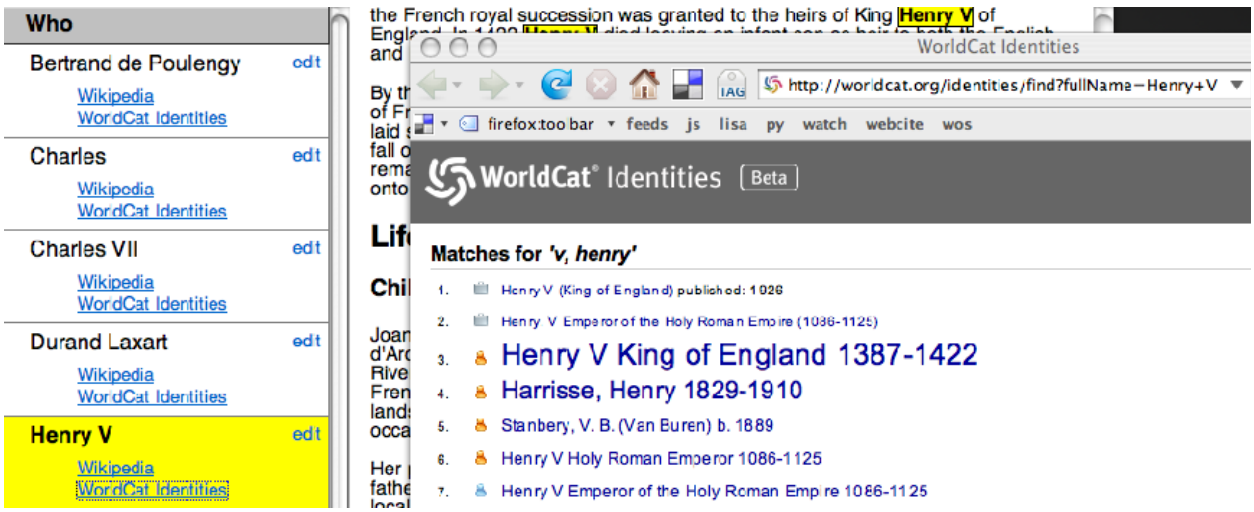

Figure 3: Searching WorldCat for information about "Henry V."

However, by clicking the "edit" link on an item in the sidebar, one could disambiguate the name by choosing a specific identifier. Persons' names were disambiguated using identifiers from WorldCat Identities, and place names were disambiguated using identifiers from GeoNames. To continue the example above, one could disambiguate "Henry V" by selecting *Henry V King of England 1387–1422* from a list of identifiers (see Figure 4). The selected identifier would then be stored in the Context Finder database and associated with the URL of the article. The next time someone viewed that article, the disambiguating identifier for that name would be used to construct more precise links to related resources. In this case, the link to WorldCat identities would lead directly to the page about Henry V King of England 1387–1422 rather than to a page of keyword search results.

Figure 4: Disambiguating a person name by linking it to a WorldCat identity.

Prototype II: Finding Context in Scanned Texts

The second iteration of the Context Finder prototype extended the context finding functionality from Web pages to scanned texts such as books and journal articles. Because the documents being annotated were not simple Web pages, but bundles of page images and OCR data, the primary challenge for this prototype was the interface. Since the quality of the OCR text was often quite poor, we wanted users to see the page images rather than the OCR text. Yet we needed to enable the highlighting and selection of arbitrary locations in the text. To achieve this, we created an interface which could dynamically overlay transparent highlights over the displayed page image. Using the page coordinate information from the OCR data, the interface could generate overlays for any word or sequence of words. The interface could also allow users to select regions in the page image, and then translate their selection into a sequence of words from the underlying OCR data. Figure 5 shows an example of the Context Finder interface for a scanned text.

A button labeled Find names in current page initiates a program to identify place names and personal names in the page displayed.

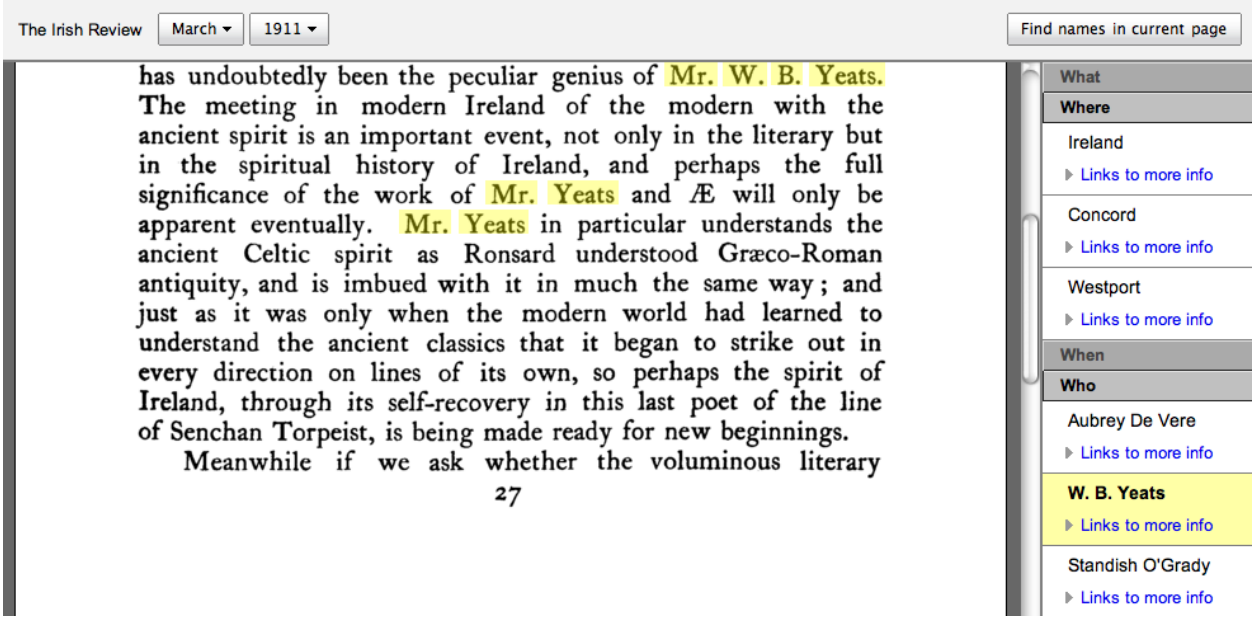

Figure 5: Finding context for named entities in a scanned text.

The interface of the second prototype was similar to the interface of the first, except that the sidebar was moved to the right to be adjacent to the page image scrollbar. We also took advantage of the expanded capabilities of the OpenCalais web service to add "What" (topic) and "When" (event) names to the sidebar. Since the collection of texts was focused on Ireland, we also expanded the menu of related resources shown with each item to include Ireland-specific reference resources. See Figure 6 in which clicking on the Links to more info line has opened a menu of searchable online resources relating to Ireland.

Then clicking on the first-listed resource Ask about Ireland has triggered a search of that resource with the query W. B. Yeats. It is important to note that this specific search was generated by the Context Finder, which is a significant improvement over merely taking the user to the resources home page. Because the Context Finder interface sent a specific query for W. B. Yeats, material relating to Yeats is what the viewer immediately sees in a new window in which the user can also explore the Ask about Ireland site at will. (Note that resources are not stable. Since this screen shot was made, this site has been changed and although the same resources can still be found, they are now not quite so quickly retrieved.)

Clicking on any other of the resources listed under Links to more info will yield comparable resources at each site if and as available.

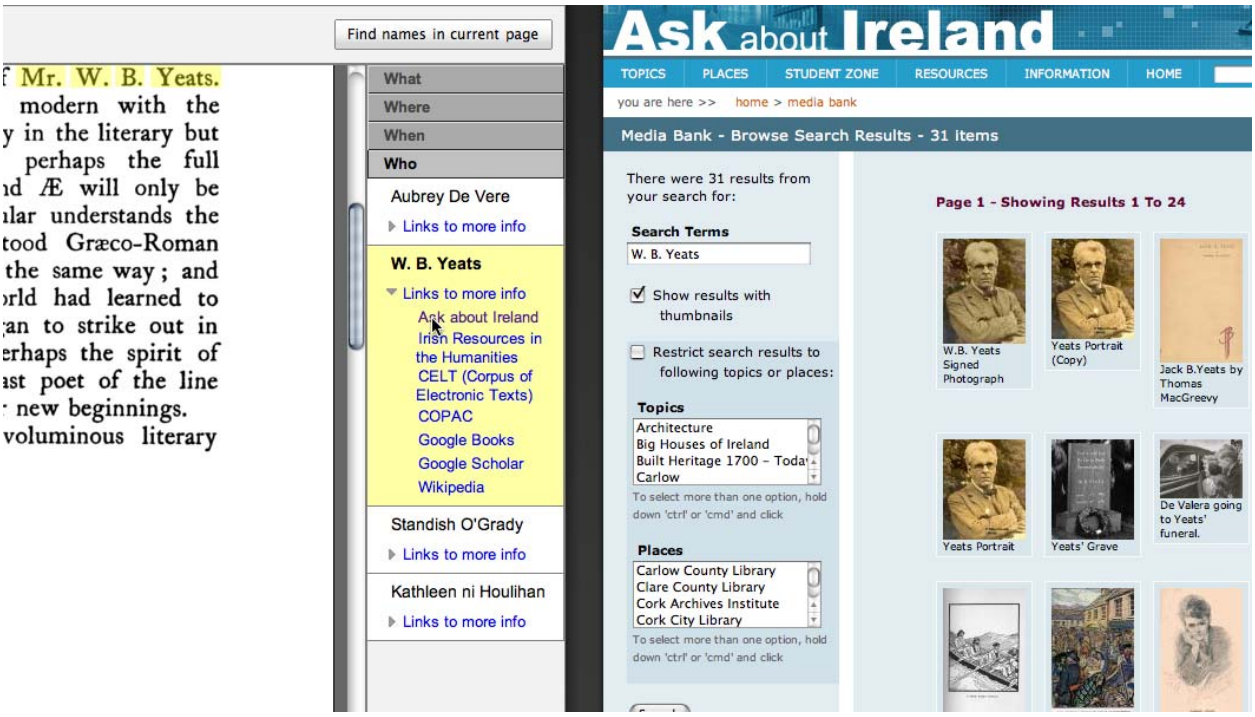

Figure 6: Linking from a textual reference to W. B. Yeats to library items related to him.

Because our interface supported the selection of a location within a page by clicking on the scanned page image, we also added the ability to add additional named entities that had not been identified by OpenCalais (see Figure 7). These additions were stored in the Context Finder database to be displayed to future users.

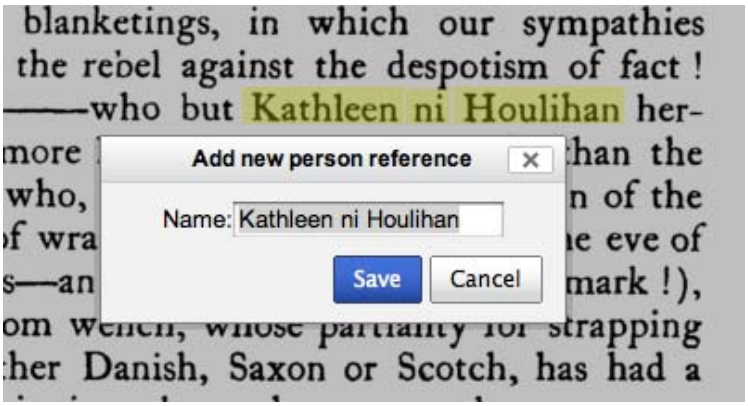

Figure 7: Adding a textual reference to a person.

To explore the scalability of our approach to dealing with scanned texts, we built a small digital library using the full text of one hundred books on Irish topics from the Internet Archive's Ebook and Texts archive and several hundred articles scanned at the Centre for Data Digitisation and Analysis at the Queen's University, Belfast. (This Belfast material subsequently became JSTOR's Ireland Collection). We found that we could process the source page images and OCR

data to generate the context-finding interface in a reasonable amount of time, and that the interface was fast and responsive even with large books of several hundred pages. However, we also found that simply highlighting the appearances of a named entity in the text when it was selected in the sidebar was no longer sufficient for long texts such as book chapters or journal articles, where appearances could be many pages apart. To deal with this, we changed the sidebar interface to show the total number of times that entity appeared in the text, and the index of the currently displayed appearance. Clicking repeatedly on the menu item cycles through the appearances, scrolling pages to the location of each (see Figure 8).

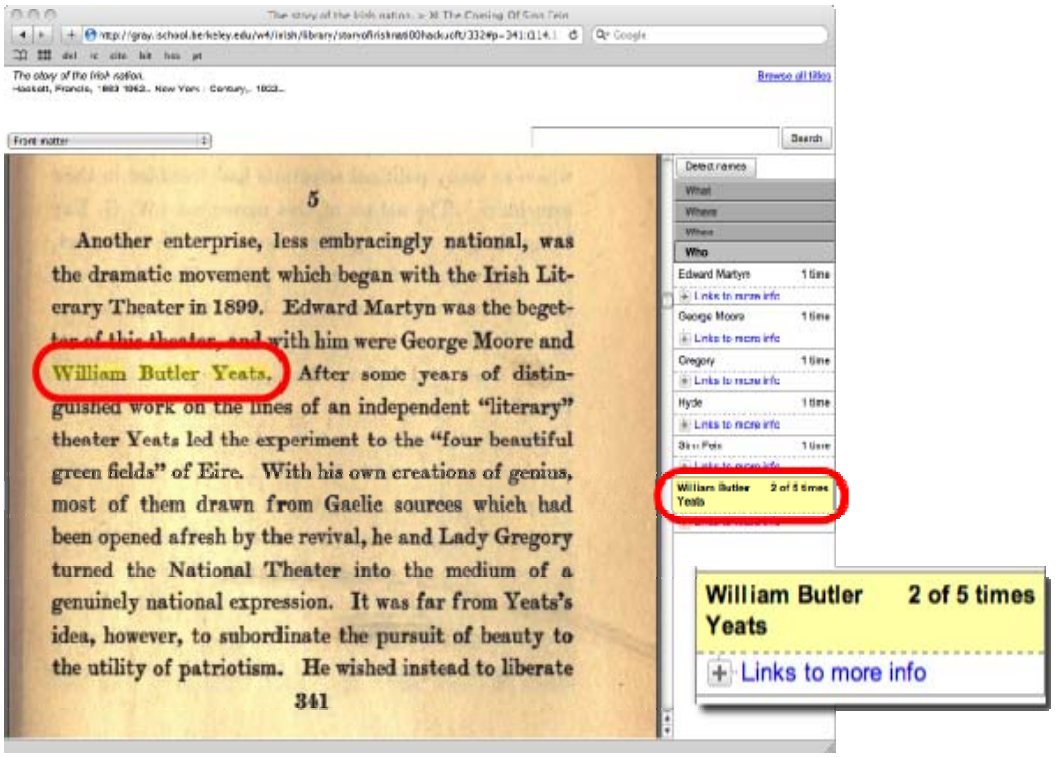

Figure 8: Iterating through the appearances of a name in a longer text.

Prototype III: Describing and Packaging Reference Resources

A significant weakness of the first two prototypes was the fact that the menu of related reference resources shown with each item was hard-coded. Ideally, reference resources could be dynamically selected by knowledgeable librarians, teachers, and experts and personalized by individual users. For our third iteration of the prototype, we sought to realize this vision.

Based on our experiences using the first two prototypes, we also made a decision to significantly change our approach to context finding with the third prototype. First, we decided that automated named entity recognition, while technically impressive, was not really necessary for what we wanted to achieve. Human users are better than machines at recognizing names in a text and do not need automated assistance. Furthermore, there is no reason to limit context-finding to named entities. Any word or sequence of words in a text could need explanation. While named entity recognition still has a role to play in the batch analysis of entire text

collections, e.g. for automated linking to reference resources, we decided that for the user-driven linking we were interested in it was a distraction within our limited resources.

Second, we decided that a general-purpose Context Finder needed to work with arbitrary text content on the Web without any special processing of that content, as was required by the earlier prototypes. Thus the third prototype was implemented not as a standalone Web site as the earlier prototypes were, but as a Firefox browser extension.

The third prototype of the Context Finder had three components:

-- a *Web service* providing read/write access to database of reference resource search interface descriptions,
-- a user interface for creating reference resource search interface descriptions and bundling them into menus of "packages" (the *package builder*), and
-- the Firefox *browser extension.*

The following sub-sections describe and explain each of these components.

Web service

Many reference resources provide some means for searching their contents, i.e. search interfaces. Typically, these search interfaces accept query terms as HTTP GET parameters. But parameter names are not standardized across the search interfaces provided by different resources. One interface may use the parameter name "au" for an author name, while another may use "auth". This makes it impossible for a third party to submit searches to different resources in an automated fashion. An OpenSearch description is a solution to this problem which allows anyone to use XML to describe a search interface, the parameters it accepts, and how they can be combined into valid URIs for querying the interface.

OpenSearch was developed by Amazon in order to add metasearch functionality to their A9 search engine and has been released to the public under a Creative Commons license. The Firefox, Internet Explorer, and Google Chrome Web browsers use OpenSearch to enable customizable search functionality. Users can browse databases of OpenSearch description documents (such as the Mycroft database) and download these descriptions to add custom search functionality to their browsers. Being XML, OpenSearch is extensible, allowing specific communities to add additional fields to description documents beyond what is included in the core specification. In particular, the Library of Congress has developed a SRU (Search and Retrieval via URL) extension in coordination with the SRU 2.0 specification. (SRU is the Web-based successor to Z39.50.)

We stored OpenSearch descriptions of reference resources in a PostgreSQL database, using PostgreSQL's support for native XML storage. This gave us the power, speed and efficiency of relational querying while allowing us to store arbitrary OpenSearch description documents, including those with extensions unknown to us.

Description documents were organized into "packages" (menus), along with descriptive metadata such as the name and purpose of the package and the name of its creator. A package was simply a set of OpenSearch descriptions that described similar, related, or complementary reference resources. For example, one package could bundle a number of resources which contain information related to Irish history, while another could bundle resources related to Indian Classical Music. Clients could read from and write to the database via a Web service. Any

client could retrieve individual search descriptions as OpenSearch XML documents and could retrieve packages and other sets of descriptions as Atom XML documents. (Atom is a widely used open standard for publishing lists of Web resources.) Clients that had authenticated with the Web service could also add new search descriptions or packages using standard HTTP methods. Although XML was the standard format for the OpenSearch and Atom documents read from and written to the Web service, the documents could also be serialized as JSON (JavaScript Object Notation) for easier integration with browser-based software.

Package builder

The Web service simply provided a standard interface through which search descriptions and packages could be read from and written to the database. It did not provide any authoring support for creating search descriptions or packages. This was the function provided by the package builder interface. The package builder was a server-side Web application, accessed through a browser, that enabled users to easily create new search descriptions and packages of search descriptions.

Databases like Mycroft already contain thousands of search descriptions that can be used as-is with the Context Finder. However, there are always new search engines and databases being created for which OpenSearch descriptions may not yet exist. The package builder allowed users to create new search descriptions without having to author XML by hand. The user simply executed a search on the resource she would like to create a description for, and then copied into the interface the URL of the search results page along with the query term for which she had searched. The package builder then generated an OpenSearch description document and stored it in the database.

To create a package of search descriptions, the user could search the database of description documents to quickly add the desired ones. If the desired descriptions were not in the database, they could be loaded from an external database such as Mycroft or manually added as described above. Users could quickly and easily edit existing packages, adding or removing descriptions and editing descriptive metadata.

Browser extension

The browser extension communicated with the Context Finder Web service to load a specific package and provide an interface for searching in the context of a text being read in the browser. The extension was developed using Jetpack, a toolkit from Mozilla Labs for quickly developing Firefox extensions using standard user interface patterns. Jetpack extensions are written in standard JavaScript and HTML.

The browser extension enhanced the browser's context menu, which is typically displayed when the user right-clicks on a web page. If the user had selected some text in the page, a "Query [package name] for [text]..." item appeared in the context menu, with the name of the currently loaded package and the selected text substituted in the expected places (see Figure 9).

Figure 9: Selecting text and choosing a package of resources to query.

If the user selected this menu item, a sidebar appeared showing all the resources described by the currently loaded package. At this point the browser extension would begin silently querying each resource in the background, using the selected text as its query parameter (the other parameters being provided by the search description). Using a custom extension to the OpenSearch description standard, the extension tried to determine whether a given query had returned zero results. If the background query returned one or more results, the resource was highlighted in green. If the background query returned zero results, the resource was highlighted in red in the sidebar. If the extension could not determine whether the query returned any results or not (for example, because the search description did not include the custom extension), the resource was not highlighted. Using color-coded highlights to indicate resources with zero results helps the user to avoid wasting time opening tabs without useful information. In Figure 10 we see that three resources (the *Onomasticon Goedelicum*, *Ireland's History and Maps*, and the *Corpus of Electronic Texts*) have matching results, while the other resources do not. By clicking on individual resources in the sidebar, the user could open tabs displaying the search results (or lack thereof) from each resource.

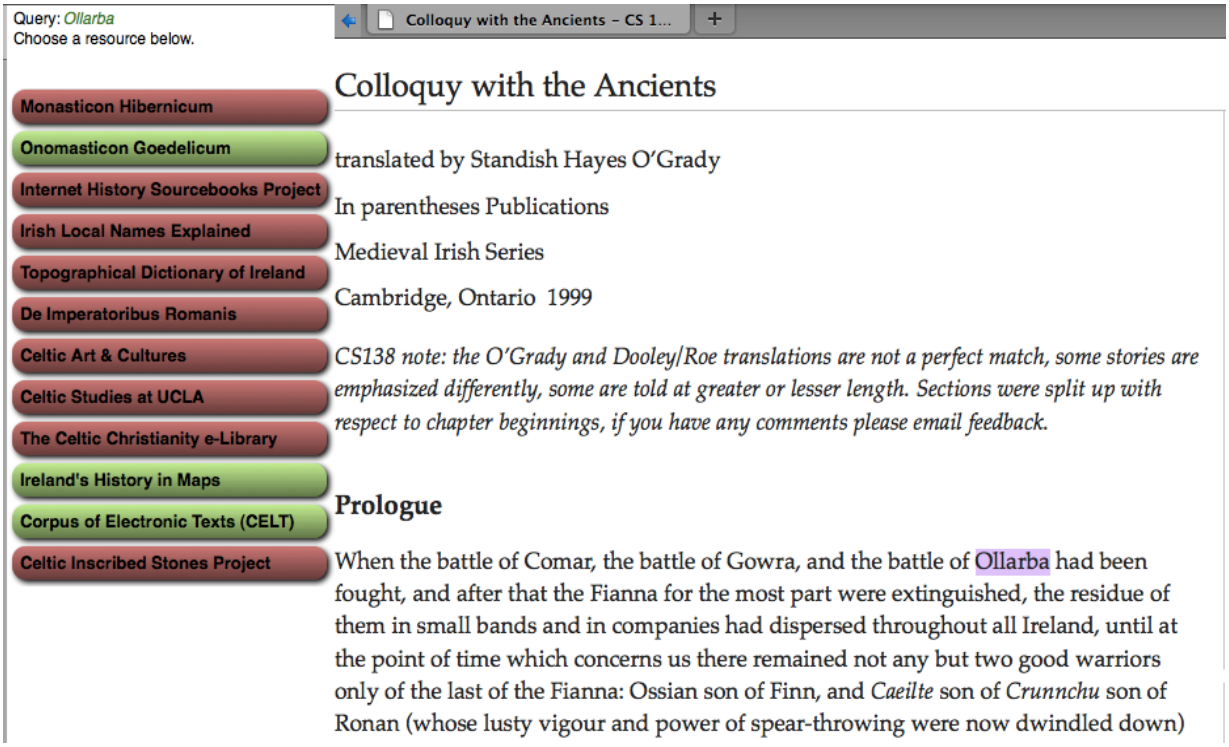

Figure 10: Showing which of the Celtic Studies resources have information about "Ollarba

Clicking in the first green item, the *Onomasticon Goedelicum* immediately displays a relevant extract from that source in a new window. See Figure 11.

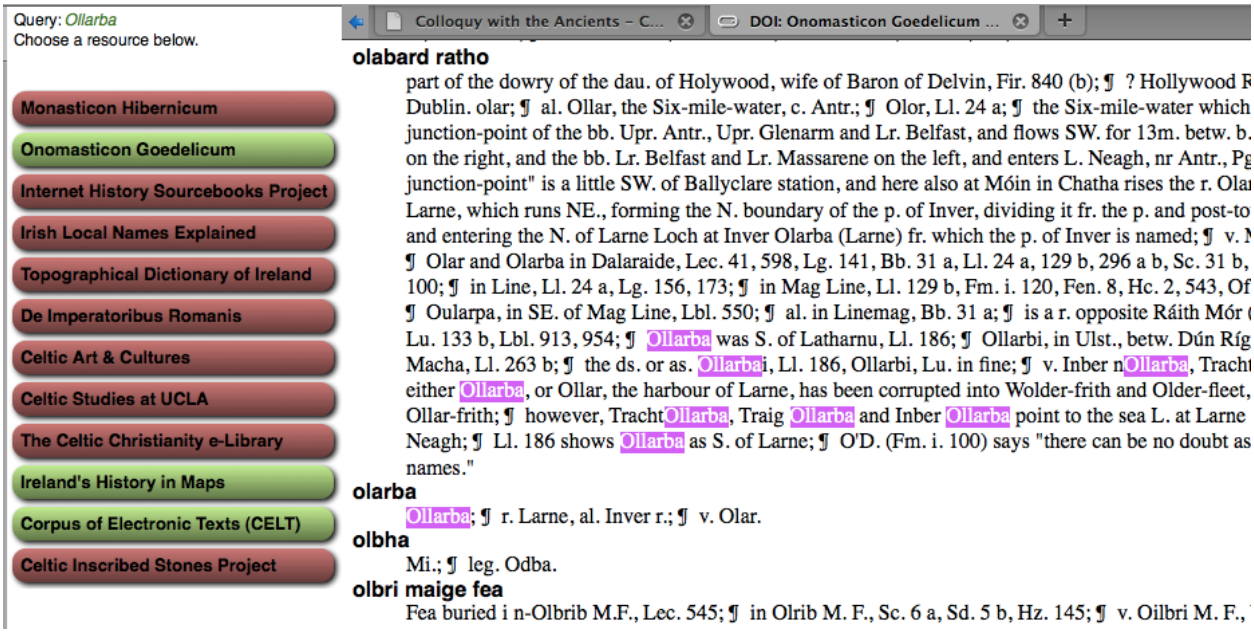

Figure 11: Displaying information from the Onomasticon Goedelicum *about "Ollarba."*

Clicking on the second green item in the menu, *Ireland's History in Maps* shows what that resource has to say about "Ollarba." See Figure 12.

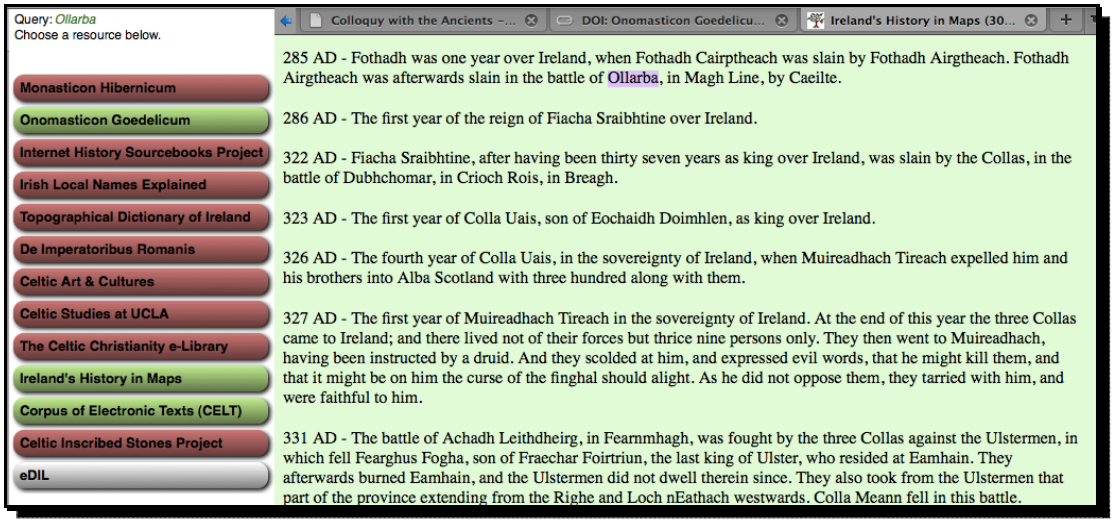

Figure 12: Clicking on the second green resource (Ireland's History in Maps (fetches an explanation of the chronological and political context of the battle of Ollarba and presents it in a new window.

In this example the geographical and historical context of *Ollarba* has been provided from two of the most respected academic resources with just four clicks, no keying, and no real prior knowledge of what the best resources would be. With such an amenity the student working in the dorm in might well start using the best scholarly resources rather than only the most popular familiar resources.

The examples given above derive from the deployment of the Context Builder for an upper division undergraduate course offered by Berkeley's Celtic Studies Program taught by Co-P.I. Professor Daniel F. Melia: Celtic Studies 138: Irish Literature, which covers Gaelic literature 700-1800 in translation, including study of the prose saga-cycles, satire, classical lyric poetry, and bardic poetry, that developed the mythological and traditional background of modern Irish literature. An attempt was made to evaluate the third prototype though its use by students taking this course. The requisite permission from the campus Committee for the Protection of Human Subjects was received after three months of negotiation and the imposition of so many restrictions that, in the end, little was learned from the evaluation.

FUTURE DEVELOPMENT OF THE CONTEXT FINDER

Based on our experience we have identified three areas on which we believe future work should focus.

Package switching and customization

Changing the package used by the browser extension involved re-installing a new version of the extension. Obviously it would be preferable if users could change the currently loaded

package at will, by browsing a gallery of packages and choosing the desired one. Even better would be if users would customize any currently loaded package without having to use the full package builder interface. Customization would involve adding new descriptions to or removing descriptions from a package directly from the browser extension. This would be a simpler operation than using the package builder interface, as the user would be working from an existing package rather than building "from scratch." Ideally, customizations could be tracked via "derived from" links among packages, enabling package authors to see how others have modified their packages and offering them the option of incorporating these changes back into their original package.

Shared annotations

It is possible for a community of users using (customized versions of) the same Context Finder package to share information with one another about relevant resources that they have found by writing information back to a shared database. In our project, this kind of functionality was used to implement logging for research purposes, and the shared database was visible only to researchers. However, there is no reason to limit the shared database to this function. For example, the browser extension could allow users to indicate that they "like" (i.e., are happy with) particular results. Information about liked results could be aggregated in the shared database, and statistics displayed to package users in the browser extension. So, for example, a class of students using the same Celtic Studies package could easily see which resources have proved most useful to their fellow classmates. Arbitrary variations of this idea are possible, enabling users to annotate packages, resources, search results, and web pages with any kind of information to be shared with other package users.

Description and package building

Although the package builder was fairly easy to use, a number of improvements could be made. The technique of pasting a search results page into the interface to generate a search description did not work with all sites, particularly those that did not follow Web standards by using HTTP GET for their search interfaces. In these cases a more comprehensive "wizard" interface for helping users author search descriptions may be necessary.

While the browser extension provided a useful interface for using Context Finder packages within the browser, there is no reason that packages could not be used in other ways as well. For example, an instructor may wish to embed a package in her course Web site so that it can be used to query resources even by students who have not installed the browser extension.

An advanced package builder interface could allow users to specify how such a package "widget" would be displayed and assist them in embedding the widget in a Web page.

Finally, we would like to experiment with ways to extend search descriptions to include structured metadata describing the content of individual resources. For example, in our third prototype a user could specify in a *textual* description that a resource was a database containing biographical articles about 19th century German philosophers. A system that allowed descriptions to be expressed as structured metadata might include descriptors such as "entity type: Person," "time period: 19th Century," and "place: Germany." These descriptors would enable enhanced searching for resources, so that (for example) one could search the Context Finder database for all descriptions of resources with information on 19th-century people.

The Context Finder was at the center of our project and the primary focus of our attention and resources, but two derivative developments, which we called the *Context Builder* and the *Context Provider*, were always intended. Although the design complexities of the Context Finder dominated our efforts, we have also provided proof-of-concept of the Context Builder and the Context Provider.

THE CONTEXT BUILDER

The Context Finder is designed to enable anyone to go find explanatory descriptions of anything at any time and to find an explanation as up-to-date as the selected resource is at the time it is queried. This emphasis on live, real-time searching has a significant advantage over the inherent tendency to obsolescence in designs that depend on caching and storing resources.

There is, however, benefit in noting any resources that proved satisfactory for some particular query so that it is available for the next reading and/or the next reader. The ability to accumulate notes on what resources proved useful we call the *Context Builder*. See Figure 13 where

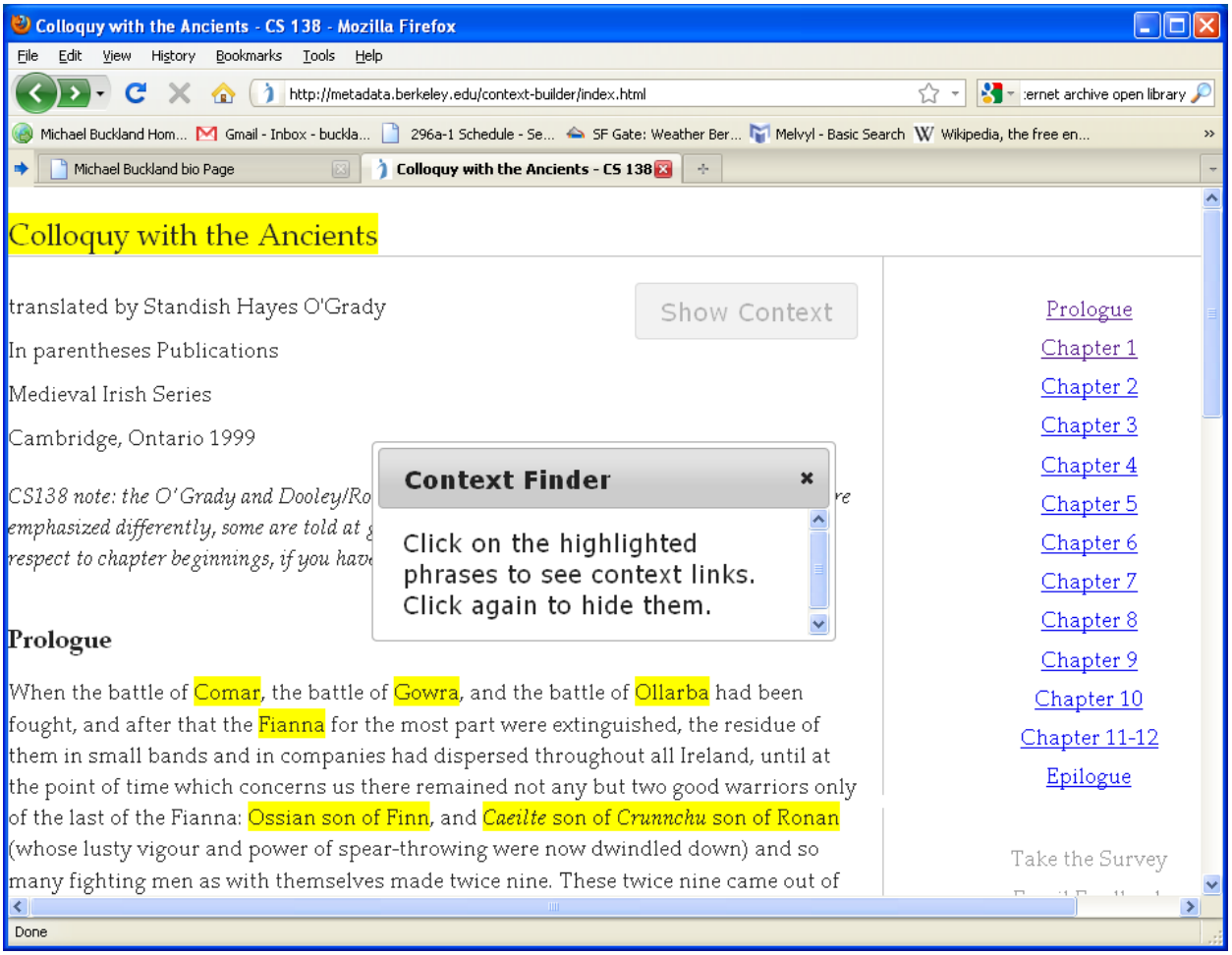

Figure 13: The Context Builder

So, for example, clicking on the first highlighted text, Colloquy with the Ancients, opens a menu of contextualizing resources that have "built into" the text, as shown in Figure 14.

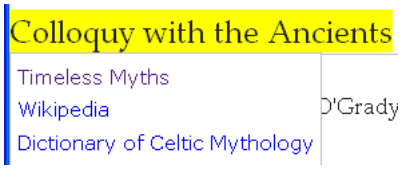

Figure 14: A Context Builder menu.

Clicking on the first link, to Timeless Myths, causes the interface to find and display in a new window the relevant text from that resource. See Figure 15.

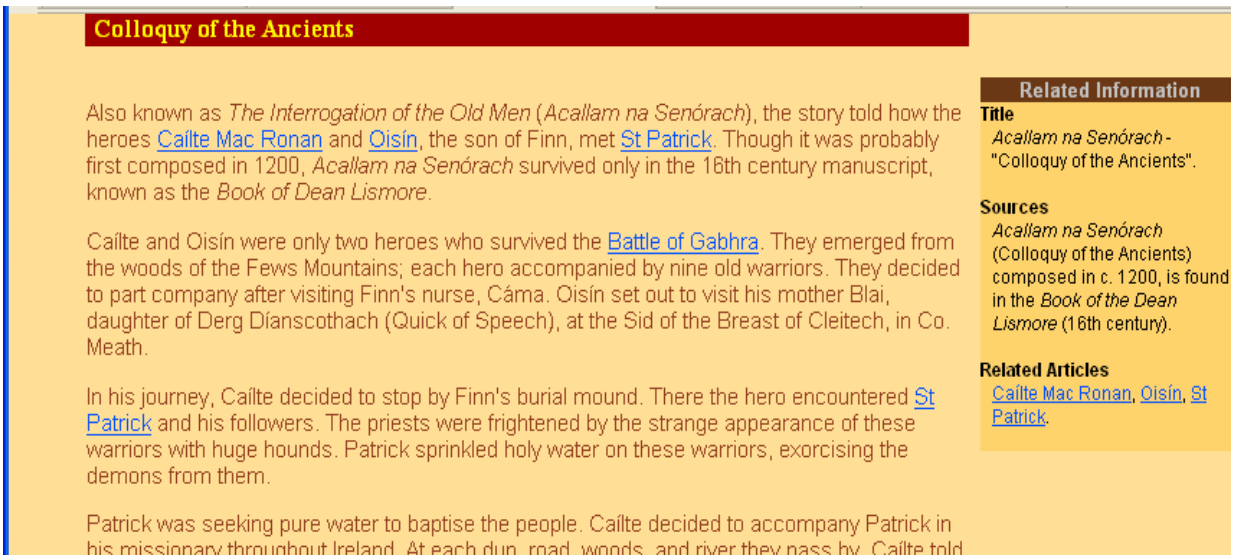

Figure 15. Explanation provided by the Context Builder.

THE CONTEXT PROVIDER

The Context Builder enables a text to be enriched with numerous links to one or more explanatory, contextualizing resources. Each link is from a specific word or phrase in the text and points to a specific point in a specific resource. What if these links are also provided in the reverse direction? All the needed ingredients are present, always assuming that permission to enrich the resource(s) is granted or not needed. One can readily imagine procedures for extracting from the text the links inserted by the Context Builder that point to one particular resource, sorting them into the right order, then embedding them into the resource so that they point back to the words mentioning the name or topic in the original text. This second derivative of the Context Finder we called the *Context Provider*. Because of the priority given to the Context Builder within the project, we were unable to spend much time on the Context Provider beyond demonstrating proof-of-concept.

Figure 16 shows the entry in the *Dictionary of Celtic Mythology* for *Sliab Fuait* (aka Sliabh Fuiad). We have added the Mentions in the Text box as an implementation of the Context Provider.

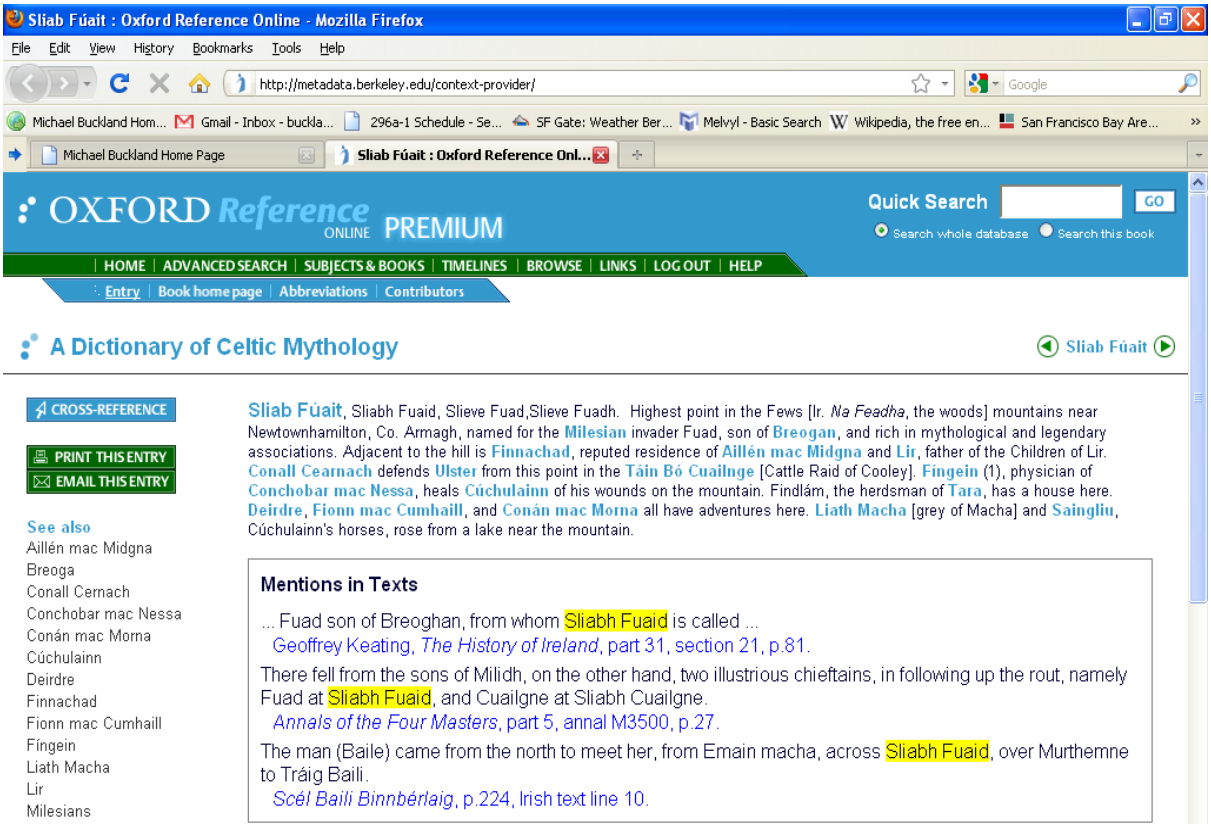

Figure 16: The Context Provider. Proof-of-concept using an entry from A Dictionary of Celtic Mythology. http://metadata.berkeley.edu/context-provider/ In the "Mentions" box the sources in blue italics are links to the texts where Sliabh Fuaid is mentioned.

The practice of citing sources in scholarly reference works such as place name gazeteers is long-established. Edmund Hogan's *Onomasticon Goedelicum*, shown in Figure 11, is noteworthy for its many (greatly abbreviated) references to mentions in of each place name in any of the hundreds of early texts that he examined. Our adoption of this practice of providing links is not new, but rather a matter of enabling the continuation of that practice as an economical by-product of the reading of texts for other purposes. The adding of such links can be done cumulatively and gradually a web of myriad links can be built up in a way that would be difficult by other methods.

DOCUMENTATION AND DISSEMINATION

The project is documented at a prject website http://ecai.org/neh2007/. This project website includes the original proposal, acknowledgment of the funding, a link to openly accessible prototypes, and details of numerous presentations and reports.

Because of the perceived strategic importance of the work performed in this project an effort was made to disseminate the work widely. The list of "Related publications" includes presentations in nine different states and twelve foreign countries on four continents, including one presented and published in French. Of these, five have been selected and are listed below. Additional publications are intended.

CONCLUSIONS AND FUTURE WORK

Here we add some comments on what we see as implications of our work.

1. In the research and development advancing the transition of library services to a digital environment the emphasis on using new technology to *empower reference librarians* to use reference resources deftly needs to be complemented by an increased effort to *empower library users* to assist themselves. Library users tend to prefer selfhelp and self-service scales more cost-effectively than does provision of a human intermediary.

2. The move from isolated silos with bespoke search support to network-accessible resources with standard search protocols has been far slower for reference works than for library catalogs. We suspect a decline in the provision of local search engines in favor of keyword access by internet search engines restricted to the local domain. This provides a useful, economical service, but more effective discovery requires facetted search support. The OpenSearch and Mycroft approach offers promise. We started to examine the addition of search term recommender services which will be needed to get beyond the use of keywords.

3. Librarians have a long and strong tradition of providing guides to reference resources, not only in person but also through published guides, some general, some specialized, and some tailored to local needs. These guides have been moved online. They typically provide a live link to each resources' home page, but much more could be done to couple individual words in any text being read with search results specific to that word in trustworthy resources appropriate to both the topic and the knowledge and cognitive skills of the individual reader.

4. There are significant opportunities to improve the presentation of geographical contexts because now, as never before, place names can be combined with map displays and geographically-based time series of historical statistical data, such as successive censuses of population. We explored this aspect through preliminary work on dynamic map displays of data from successive Irish population censuses.

5. Discussion of the potential usefulness of the Context Finder for editors preparing scholarly editions of historically-important texts led to a new multi-institutional project entitled "Editorial Practices and the Web" supported by a $457,000 grant from the Andrew W. Mellon Foundation. The project is concerned to evolve editorial work practices in ways that will increase the return on investment by reducing duplicative

research and through the quicker, fuller, and wider publication of editors' explanatory notes. Details are at http://ecai.org/mellon2010/

6. Only services that are closely integrated with individuals' personal work environments are likely to increase use substantially. Search support incorporated into browsers and the ability to create personal virtual "reference libraries" with an ease comparable with book marking are required. There are significant differences between *usable*, *useful*, and *actually used.* It is only actual use that brings benefits and constitutes the return on the large investments made.

We believe that the work performed during this project provides the basis for a significant change in digital support for scholarship and for enabling learners to understand the context and relationships of whatever they are interested in. This project, with its emphasis on establishing links to related records as a way to illuminate context, converges on the more useful aspects of the Semantic Web. We are seeking to continue and to extend this work.

Acknowledgments

We are very grateful for the support from the National Endowment for the Humanities, the Institute of Museum and Library Resources, and the University of California, Berkeley that enabled us to undertake this work.

Many people worked on this project including Co-P.I.s Professor Daniel Melia and Fredric Gey. Ryan Shaw played a central role in software development and project coordination. Others who contributed included Matthew Holmberg, Krishna Janakiraman, Noah Kersey, Thejo Kote, Ray Larson, Longhao Wang, and Jeanette L. Zernicke (all at Berkeley) and Paul Ell (The Queens' University, Belfast).

Selected Project-related Publications

Michael K. Buckland, Ryan B. Shaw, & Daniel F. Melia. *Rapid Access to Reference Resources.* Poster presentation at the American Society for Information Science and Technology Annual Meeting, Pittsburgh, Oct 25, 2010. Textual summary in Proceedings
http://metadata.berkeley.edu/asistcontext.pdf

Daniel F. Melia. An On-line Context Finding System for Irish Materials. Seventh Australian Conference of Celtic Studies, University of Sydney, Sept 30, 2010.
http://people.ischool.berkeley.edu/~buckland/sydney10paperprint.pdf

Michael Buckland, Ryan Shaw, & Daniel Melia. Contexte et connexions: L'Irlande et le patrimoine irlandais. Pages 15-28. In: Patrimoine 3.0. Actes du douzieme colloque international sur le document electronique, 21-23 octobre 2009, Universite de Montreal, Canada. Paris: Europia, 2009. ISBN 978-2-909285054-4. [In French]
http://people.ischool.berkeley.edu/~buckland/cidebuckland.pdf
http://people.ischool.berkeley.edu/~buckland/cidebuckland.ppt

Michael Buckland. The Educational Role of the Library in a Digital Environment. Forum for Advanced Studies in Arts, Languages and Theology (SALT), Uppsala University, Oct 3, 2008. Part I: Library Reference Resources when Digital; Part II: Design for Learning.
http://metadata.berkeley.edu/SALT108.ppt
http://metadata.berkeley.edu/SALT208.ppt

Michael Buckland. Reference library service in a digital environment, *Library and Information Science Research* 30, no 2 (2008): 81-85. Preprint at
http://people.ischool.berkeley.edu/~buckland/libref.pdf