MEDDLEing with Digital Library Searches: Surmounting User Model and System Misalignments through Lightweight Bespoke Proxying

David Bainbridge, Annika Hinze, Sally Jo Cunningham
University of Waikato
Hamilton, New Zealand
{davidb,hinze,sallyjo}@waikato.ac.nz

J. Stephen Downie
University of Illinois
Urbana-Champaign, USA
jdownie@illinois.edu

ABSTRACT

We document how surprisingly easy it is for user misconceptions to arise when using digital library search interfaces, and the significant unseen impact this can have on the user's interpretation of search results. Further, we detail a bespoke proxying technique we have devised called Meddle—for ModifiED Digital Library Environment—which is a lightweight agile technique that helps address identified pitfalls in a DL search interface that operates independently of the originating digital library.

CCS CONCEPTS

• **Human-centered computing → Web-based interaction**; • **Applied computing → Digital libraries and archives**;

KEYWORDS

Digital Library Search, Usability Issues, Bespoke Proxying

ACM Reference Format:
David Bainbridge, Annika Hinze, Sally Jo Cunningham and J. Stephen Downie. 2018. MEDDLEing with Digital Library Searches: Surmounting User Model and System Misalignments through Lightweight Bespoke Proxying. In *JCDL '18: The 18th ACM/IEEE Joint Conference on Digital Libraries, June 3–7, 2018, Fort Worth, TX, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3197026.3203873

INTRODUCTION

Like it or not, the phenomenal success of web search engines such as Google impacts how users interact with any search box they encounter—including the ones in our digital libraries. Given what we know about the users' "folk models" for searching [3] (i.e., expected similarity to Web search engines), in this paper we demonstrate that more can be done to align our digital libraries with this type of mental model. Extending our previous work [1], we start by presenting some worked examples to highlight the ways in which user expectation and search interfaces can disconnect. Following this, we provide implementation details of Meddle, the approach we have devised to help offset the effects of such misalignments. For a broader discussion of the issues, see [1].

JCDL '18, June 3–7, 2018, Fort Worth, TX, USA

ACM ISBN 978-1-4503-5178-2/18/06.
https://doi.org/10.1145/3197026.3203873

Example 1: Known-item search

Known-item searches are a common activity [2]. Using the home page of the ACM DL, however, there is a strong chance of a folk model disconnect with the interface, because the search performed is metadata only. To highlight the impact of this, consider Table 1, which shows the results of searching for articles that mention some well known DL systems.

Using the ACM DL's home page means matches are found only if the authors included the name of the system in a metadata field such as title. In the case of the "ContentDM" query, for instance, this has never occurred, so no matches are returned. Compare this to a user visiting the DL via Meddle—which unobtrusively redirects the query to use the full-text index—resulting in 16 articles being located. Similar improvements can be seen for the other example queries in Table 1.

Example 2: Issues of Word-wrap

Building a full-text indexed DL from PDF documents presents many challenges. In the ACM DL one such difficulty that affects some of the documents are line-wraps, where a word at the end of one line is incorrectly joined with the word that starts the next line. To illustrate the problem, we exploited the fact that we knew that many authors publishing in the ACM DL work in a "Department of Computer **Science**" at the "**University** of ..." which might place the malformed string 'ScienceUniversity" in the index.

Searching for "Department of Computer Science" as a full-text phrase returned 38,892 matches, and "Department of Computer ScienceUniversity" returned 883 matches, indicating that up to 2.22% of relevant documents are missed. Repeating the experiment with the phrase "Department of" returned 100,082 matches, in comparison to "Department of" "ScienceUniversity" which returned 2,191 matches. From this larger sample size the error rate remains broadly the same, this time 2.19%.

Employing Meddle to offset the word-wrap problem a user's query is intercepted, inspected, and potentially adjusted, prior to it being sent to the ACM DL server. When multiple terms are entered, Meddle changes the query to includes additional terms formed by concatenating adjacent words together, thereby increasing the chances of locating documents hindered by the word-wrap problem. Notwithstanding its simplicity, this heuristic works well in practice because the conjoined terms generated are unlikely to appear in

Query term	Original	Adjusted via MEDDLE
"American Memory"	1	28
"ContentDM"	0	16
"DSpace"	27	493
"Omeka"	3	22

Table 1: Comparing known-item searching in the ACM DL with and without Meddle

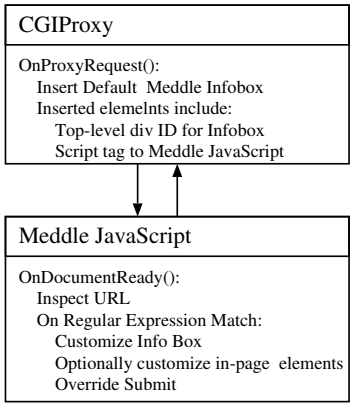

Figure 1: Overview of the Meddle software design

regular prose (with the exception of an article talking about the word-wrap problem!)

IMPLEMENTATION

Figure 1 gives an overview of the MEDDLE software design. It consists of two modules: CGIProxy and the MEDDLE JavaScript library. CGIProxy is an existing Perl module.[1] We chose it because it is a highly configurable —but more importantly—programmable proxy server solution. Used in a default installation, CGIProxy provides a RESTful interface for serving up URLs to users that have been processed through the server CGIProxy provides. A relational database (SQLite by default) is used behind the scenes to cache information to make subsequent operations more efficient.

We have customization CGIProxy to insert an HTML table—the MEDDLE Infobox—at the top of the web page that gets served up when an *OnProxyRequest()* is initiated. Initially the Infobox is rather plain, and mostly serves to highlight the fact that the page has been accessed through the MEDDLE system. The portion of HTML inserted is also marked with an *id* attribute to make it easier to access later on, and further, a *<script>* element is also introduced that binds the MEDDLE JavaScript component into the page being served up.

The JavaScript component adds an *OnDocumentReady()* handler. When the Document Object Model (DOM) is formed in the user's browser, this handler is triggered. It checks the DOM's URL with a regular expression to determine if it is a DL domain that is one that it will makes changes to (i.e., meddle with!) The procedure for changing the different DLs is modularized. For each DL, an *Adjustment()* method is provided which gets called upon a match being

[1]https://en.wikipedia.org/wiki/CGIProxy

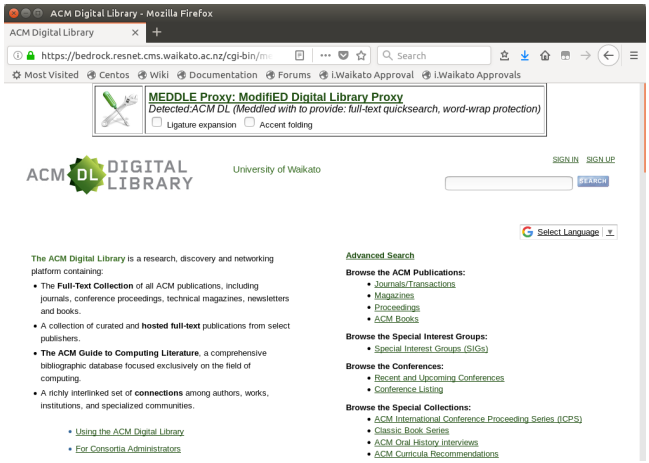

Figure 2: Example use of Meddle with the ACM Digital Library

made. Within this method, three key steps are typically taken: customization of the Infobox to include elements relevant to the given DL, changes to DOM elements in the page that has been retrieved from the DL, and overriding what to do when the Submit/Search button is pressed.

To take the ACM DL as an example, the Infobox is changed to include options for making adjustments concerning queries with accents and ligatures in them (see Figure 2). In the case where the page is the Advanced Search page, then DOM manipulation is undertaken to avoid the page defaulting to "any fields" a term that is ambiguous in the interface as it lists "full text" as one of the fields under "any fields" however our experiments with the interface has determined it is not actually included when such a search is initiated. In the case of the home page, its quick search box is changed to be a full-text query when the Search button is pressed.

CONCLUSION

The MEDDLE implementation is available, open source, through:

`http://trac.greenstone.org/browser/other-projects/meddle/trunk`

The code is intended to be exploratory. We invite others to experiment with using the technique.

REFERENCES

[1] David Bainbridge, Sally Jo Cunningham, Annika Hinze, and J. Stephen Downie. 2017. Writers of the Lost Paper: A Case Study on Barriers to (Re-) Finding Publications. In *ICADL*, Songphan Choemprayong (Ed.). Springer International Publishing, Bangkok, Thailand, 212–224.

[2] Suzanne Chapman, Shevon Desai, Kat Hagedorn, Ken Varnum, Sonali Mishra, and Julie Piacentine. 2013. Manually classifying user search queries on an academic library web site. *Journal of Web Librarianship* 7, 4 (2013), 401–421.

[3] Michael Khoo and Catherine Hall. 2012. What Would 'Google' Do? Users' Mental Models of a Digital Library Search Engine. In *Theory and Practice of Digital Libraries*, Panayiotis Zaphiris, George Buchanan, Edie Rasmussen, and Fernando Loizides (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 1–12.