Testing Usability of an Active Server Page-based Geospatial Search Engine Tool

Kit Na Goh, Idaho State University, GIS Training and Research Center, 921 S. 8th Ave., Stop 8104, Pocatello, Idaho 83209-8104

ABSTRACT

Using Active Server Page technology, an intelligent web interface was created having the capability to 1) query spatial data against various keywords or search phrases and 2) allow users to view the metadata without first downloading the entire dataset. The web interface was built to facilitate rapid data discovery and download. The efficacy of the interface was tested by comparing it with 1) Idaho State University's GIS Center existing search engine, 2) the search capabilities found on USGS's Earth Resource Observation and Science data center website, and 3) the search capabilities of Inside Idaho's data center website. The efficiency of each interface was evaluated using a survey-based approach with both quantitative and qualitative metrics. The goal of this paper was to assess the newly developed web interface to facilitate its effective use and evaluate its efficiency. This paper includes a description of the applicable research, the web application development, and a summary of the survey results.

KEYWORDS: ASP, metadata, GIS

INTRODUCTION

The GIS Training and Research Center (GIS TReC) currently offers two search options to their web clients: a manual search and a simple search powered by Google. This approach has limitations which limit geospatial data discovery and delivery needs. Because of the size of geospatial data, most of the nearly 30,000 geo-spatial datasets stored at the GIS TReC's spatial library have been bundled and compressed in ZIP format. This unfortunately, hides much of the data from clients and even server-based indexing services are not sufficiently robust to reveal files stored within Zip files. To maintain the advantages of file bundling and compression, yet allow clients the ability to easily locate and retrieve data, an intelligent web interface was required enabling clients to enter search criteria and preview the metadata (all done within the database) without having to first download the dataset or try to memorize the server path to manually retrieve the required dataset.

Search engine tools are becoming common and most search through the meta-databases against meta descriptions of their geospatial data range.

> "A natural approach is to add advanced features to search engines that allow users to express constraints or preferences in an intuitive manner, resulting in the desired information to be returned among the first results. In fact, search engines have added a variety of such features, often under a special advanced search interface, though mostly limited to fairly simple conditions on domain, link structure, or last modifications date" (Markowetz et al, 2005).

The method of data retrieval will contain features that allow clients to enter keywords or search phrases and also permit the clients to preview the metadata in XML stylesheet before downloading the files. This will help the clients to locate the correct dataset. This feature will add more capability than typical geospatial data search engine (Figure 1).
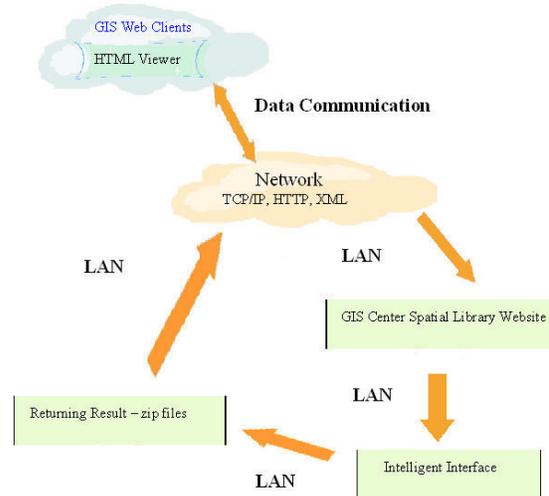

Figure 1. Schematic of new data discovery concept.

MATERIALS AND METHODS

Development of Intelligent web interface

The two technologies used to develop the intelligent web interface were Active Server Pages (ASP) and ActiveX Data Objects (ADO). "Active Server Pages (ASP) can intelligently interact with the user at the front end and with servers and datastores at the back end. ActiveX Data Objects (ADO) allows easy, powerful and robust control of technologies that access datastores" (Kauffman et al, 1999). ASP and ADO work together by reflecting current state of the database on the website without modifying scripting codes (Kauffman et al, 1999).

 In addition, simple JavaScript codes were placed within the HTML and ASP files to capture and pass the input elements between the various WebPages. Compared to the common gateway interface (CGI), JavaScript can be inserted between codes and called within the webpage; However, CGI differs in that its process is started when the Web server receives a CGI-enabled URL request but there is no sequence of interaction among the browser and server (Thiemann, 2002). There are other server-scripting languages that are already existed, such as, PHP, CGI, Perl and etc. However, I chose to use ASP with JavaScript embedded because it is commonly-used scripting language and most browsers support ASP codes.

Three HyperText Markup Language (HTML) and two ASP pages (SearchResult.asp and XML.asp) were embedded with JavaScript codes and used to develop the Geospatial Search Engine tool (Figure 2). The HTML files (Result.htm, KeywordSearch.htm, and Other.htm) were designed to display a simple graphical user interface that allows clients to enter keywords or search phrases subsequently locate the matching dataset(s).
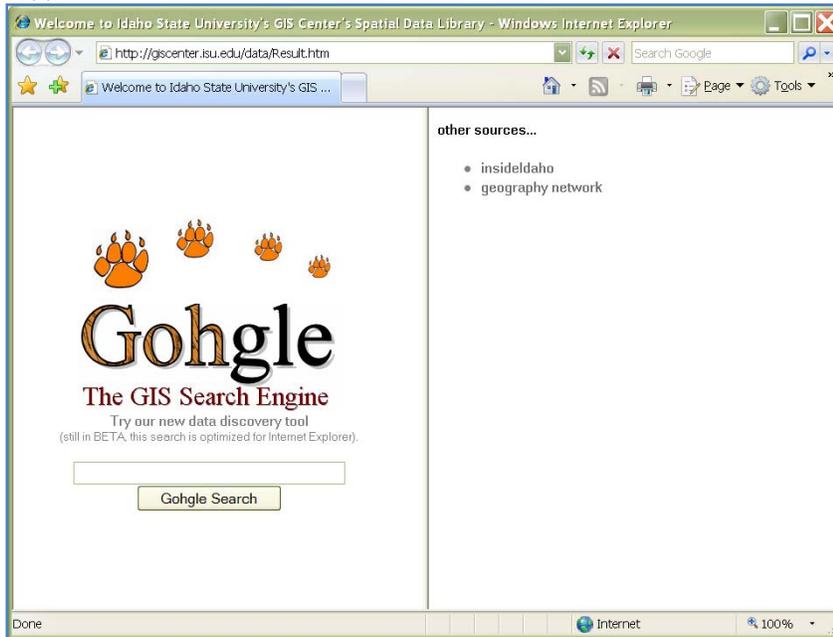

Figure 2. The simple interface of the geospatial search tool

When the clients enter a keyword or search phrase inside the input textbox and click the "Goghle Search" button, JavaScript code is called to capture the input element. If the input element contains an empty string, an alert is shown requesting proper input from the clients. If the input element is not null, the input element parameter (txtKeyword) is passed to *SearchResult.asp* page (Figure 3).

```
function submitkeywords()
{
        var urlToOpen = "";
        var KeywordValue = "";
        KeywordValue = document.getElementById("txtKeyword").value.toLowerCase();
        if (KeywordValue=="")
          {alert("Please enter a keyword or phrase to search the data!")
            }
        else
            {
        parent.right.location.reload("Other.htm");
        urlToOpen = "searchresult.asp?txtKeyword=" + KeywordValue;
        window.open(urlToOpen, "_self");
        document.getElementById("txtKeyword").focus();
            }
}//end of submitKeywords()
</script>
```

Figure 3. Javascript captures the input element and passes the element to SearchResult.asp page

Next, the ASP will connect to the database (using ADO) and retrieve the description (Abstract, Purpose, Data Type and URL link) of the file(s) that match the input element provided by the client through built-on-the-fly Structural Query Language (SQL) (Figure 4) expressions. The result is shown on the left areas of the webpage (Figure 5).

```
dim keyword
keyword=Replace(Request.queryString("txtKeyword"),"'", "''")
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open(server.mappath("Database/MetadataSpatialGDB.mdb"))
set rs=Server.CreateObject("ADODB.Recordset")
sql="SELECT distinct IDGeneral.Abstract,
IDGeneral.ID_General_RecordID,IDGeneral.Purpose,
SupCitation.OnlineLinkages, DOGeneral.DirectSpatialRef "
sql=sql & "FROM IDGeneral, SupCitation, DOGeneral,
LinkIDGeneralIDKeywords, ThesaurusKeyword "
sql=sql & "WHERE ((LCASE(IDGeneral.Abstract) LIKE '%" & keyword & "%')"
sql=sql & " OR (LCASE(ThesaurusKeyword.keyword) LIKE '%" & keyword &
"%')"
sql=sql & " OR (LCASE(SupCitation.OnlineLinkages) LIKE '%" & keyword &
"%'))"
sql=sql & " AND
ThesaurusKeyword.KeywordID=LinkIDGeneralIDKeywords.KeywordID AND
IDGeneral.DataOrganizationID=DOGeneral.DataOrganizationID "
sql=sql & "AND
LinkIDGeneralIDKeywords.ID_General_RecordID=IDGeneral.ID_General_Record
ID AND IDGeneral.CitationID=SupCitation.CitationID"
rs.Open sql, conn
```

Figure 4. The ADO connection is made to retrieve the description of files that match the input element.

When the client clicks a "Details and Download" link, the ID number and the URL link of the selected file are passed to the *XML.asp* page as these two unique attributes are used to identify individual metadata records stored within the database.   A preview of the metadatae that selected by the clients is then generated on the right side of the webpage using JavaScript (Figure 6) and SQL (Figure 7). If the client then decides to download the file, they would click the "Download Data" button above the metadata preview (Figure 5).
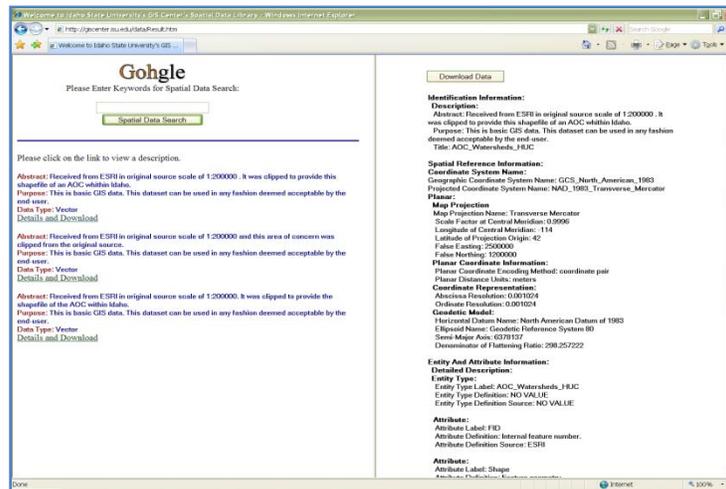

Figure 1. Search Results with XML previews and download available

```
<script language="javascript">
function submitkeywords()
{
        var idx = document.URL.indexOf('&');
        var link = document.URL.substring(idx+8,document.URL.length);
        var urlToOpen = "";
        urlToOpen = link;
        window.open(urlToOpen, "_self");
}//end of submitKeywords()
</script>
```

Figure 6. Algorithm used to extract the ID number of the selected file when the user clicks the "Download Data" button.

```
<!--Print ARC coordinate system information -->
<%sub ARCSS()
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open(server.mappath("Database/MetadataSpatialGDB.mdb"))
set rs=Server.CreateObject("ADODB.Recordset")
rs.open rs5, conn
If Not rs.EOF Then%>
  <%if rs.Fields("ZoneIdentifier") <> 0 Then%>
     <%response.write("   <b>ARC Coordinate System: " & "</b><br>")%>
     <%response.write("    ARC System Zone Identifier: " & rs.Fields("ZoneIdentifier") & "<br>")%>
  <%end if%>
<%end if%>
<%rs.close
set rs=Nothing
conn.close%>
<%end sub%>
```

Figure 7. Example of displaying the information of the selected file for metadata preview.

Once the database and web interface were developed and deployed an assessment was made to determine its efficiency. This determination was made using a survey-based approach.

Survey Analysis
Questionnaires were created which helped determine whether the new database-driven web interface is better (i.e., easier to use and more efficient) than 1) the GIS TReC existing search tool (Browse folders), 2) USGS' Earth Resource Observation and Science (EROS) data center website (http://edc.usgs.gov/) and 3) the Inside Idaho data center website (http://insideidaho.org/asp/geodata.asp). The EROS and Inside Idaho data center search engines were chosen because 1) both are well known and popular data clearinghouse-type sites within the GIS field and 2) both data centers store some of the same data as ISU's GIS TReC. The efficiency of these interfaces was quantitatively evaluated based on the speed each could find and retrieve data and qualitatively evaluated based upon the respondents' ratings for ease of use, user-friendliness, clear navigation and web interface design.

The questionnaires were given to 35 Geographic Information Science (GIS) students who were currently taking GIS- related classes at Idaho State University. Their self-identified knowledge levels ranged from beginners (14 students), intermediates (13 students) and professionals (9 students). The different levels of knowledge groups were used to allow better understanding of the usefulness of each search engine tools. The responses from the questionnaires were collected and entered into Microsoft Excel to evaluate the effectiveness of the tools.

According to survey results, 74% of those surveyed believed that the new interface (Gohgle) is the easiest tool to locate data (Figure 8) and 91% of the respondents were able to find their required data using the Gohgle search engine tool. It is noted that 64% of beginners thought Gohgle was equally as easy to find data as the Inside Idaho search tool (Figure 9).   In addition, 40% of the respondents agreed that the overall ease of use of Gohgle was good.
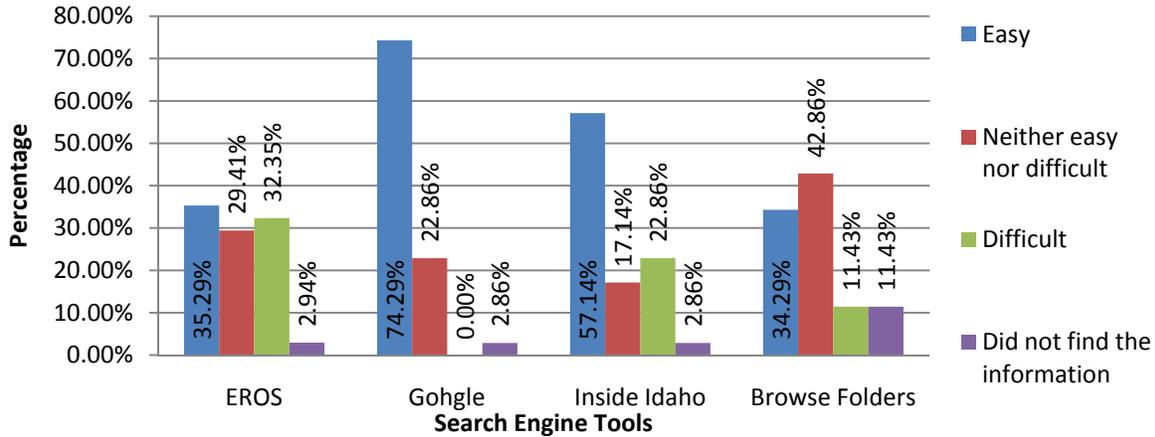
Figure 8. Difficulty level for all respondents to locate data
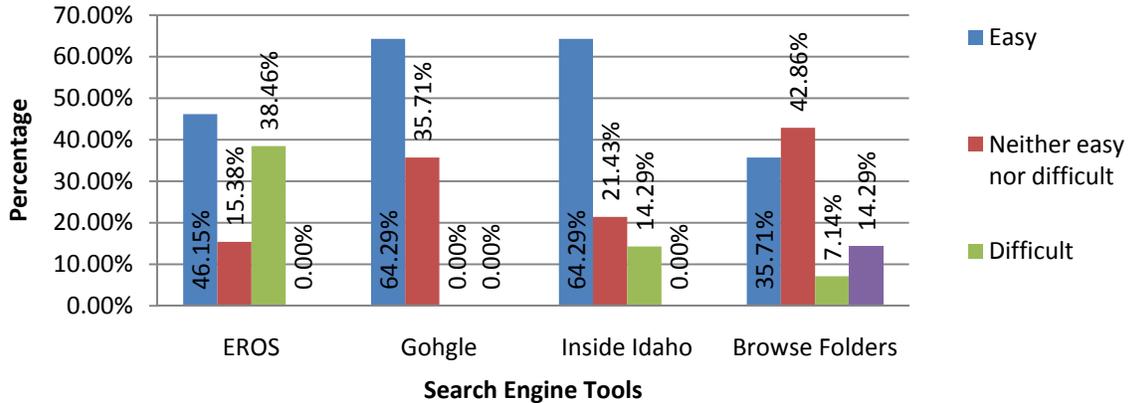
Figure 9. Difficulty level for beginner users to locate data

The efficacy of the Gohgle interface was the highest compared to the other search tools. Nearly 82% of respondents were able to locate and retrieve data in less than 5 minutes with the longest search and retrieve session lasting 10 minutes.   The average time for Gohgle user's to search, return and download data was 7.8 minutes, while GIS TReC folder browsing required 9.4 minutes to complete the same process. In comparison, 11.4 minutes was required for Inside Idaho search tool users and 13.4 minutes was required for users of the USGS EROS data center search tool (Table 1).

To evaluate the qualitative aspects of the search tool (i.e., user-friendliness, clear navigation, self-explanatory use, and the "look and feel" of Gohgle) the survey responses were studied. For self-explanatory aspect, the users will know what they need to do with the interface without further instructions. Clear navigation means the interface is easy to browse and search for data. In addition, user-friendliness explains if the interface is common looking and whether the users like the design of the interface. The analysis suggests that the majority of respondents felt the Gohgle interface provided fairly

clear navigation (23% of the respondents rated average, 51% respondents rated above average and 23% rated good navigation). Nearly 83% of respondents were satisfied with the self-explanatory nature of Gohgle and 86% of respondents liked the look and feel of Gohgle. This suggests that the Gohgle web interface is user-friendly to the GIS TReC's clients.

Table 1. The length of time for each tool to search, retrieve and download data

	EROS	Gohgle	Inside Idaho	Browse Folders
<5 minutes	55.88%	81.82%	64.71%	62.50%
5-10 minutes	20.59%	18.18%	20.59%	34.38%
10-20 minutes	17.65%	0.00%	5.88%	0.00%
>20 minutes	5.88%	0.00%	8.82%	3.13%
Mean time	13.4min	7.8min	11.4min	9.4min

While survey results illustrate that Gohgle had the highest satisfaction among beginners (57%), intermediate users (85%), and professional users (88%), 50% of beginner users preferred the Inside Idaho search tool, but overall (Figure 10). In comparison, both intermediate (69%) and professional (75%) users preferred using the Gohgle search tool (Figure 11).
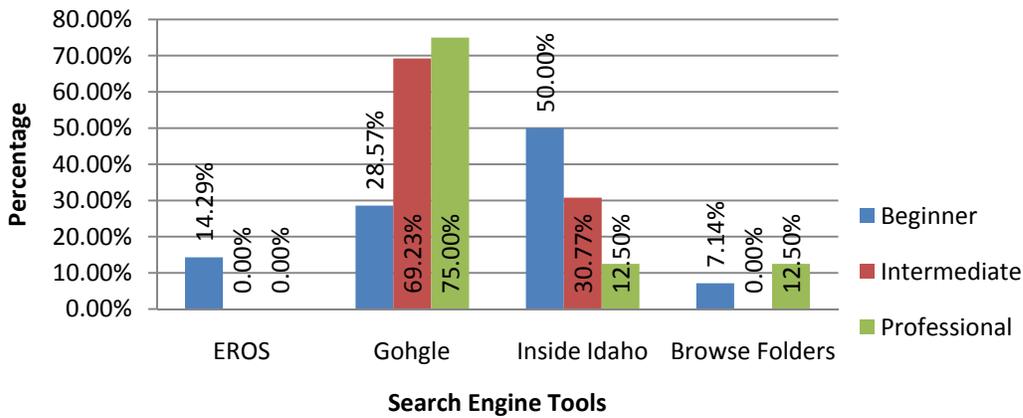
Figure 10. Respondent's search tool preferences organized by user category.
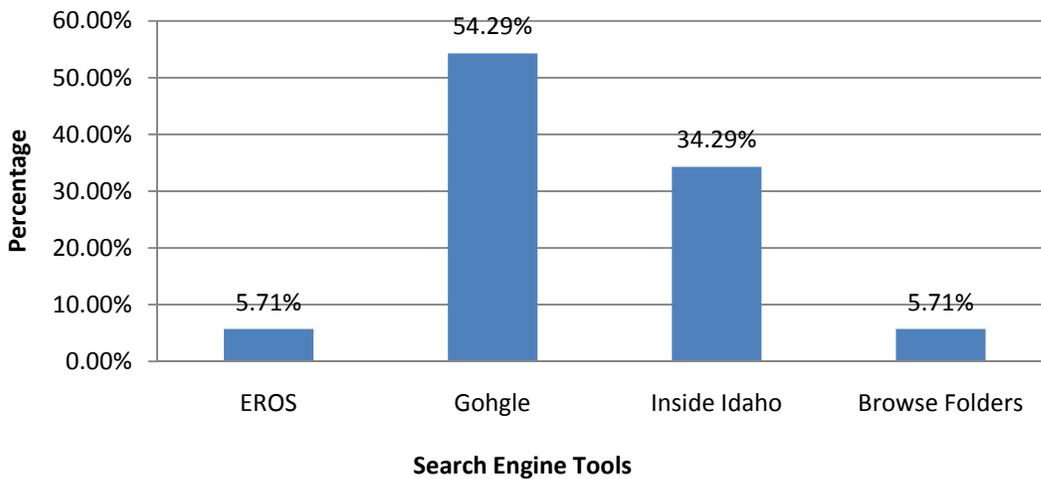
Figure 11. Respondent's search tool preferences.

RESULTS AND DISCUSSION
The survey data indicates that the majority of respondents agree that Gohgle is efficient, user-friendly and easy to use. It also suggests that the respondents were highly satisfied with Gohgle's functionalities; however, some results are different than expected, in that the simple Gohgle interface was not preferred by the beginner user group as this web interface was created to specifically support novice users and students. The tool was developed to be informative and easy to navigate for all the clients. Nevertheless, the analysis (Figure 10) suggested that the beginners preferred using Inside Idaho search tool than Gohgle search tool. In contrast, both intermediate and professional users favored Gohgle while more specialized - EROS data center search interface was rated the least favorite search tool among the tools tested.

Based upon the written comments from the respondents (4 students), they tended to like the default data description information and metadata preview of Gohgle as this saved search time. The Gohgle design was considered very standardized in appearance and easy to follow. However, beginners felt that the descriptive information on the page was not clear enough and they were hoping for thumbnail images of the data. Both of these are good suggestion for further improvement of Gohgle. The descriptive information of the files were retrieved from the database (MetadataSpatialGDB.mdb) and displayed to the clients. The information could be improved by metadata librarian and re-import into the database. The thumbnails of the data were not provided since thumbnails were not created for the metadata and this could be added in the future. In addition, the beginners chose to favor to Inside Idaho, it could also be due to they were more familiar with Inside Idaho tool for their class assignment. However, Gohgle search engine tool is a new tool that they were not used to. People have more tendencies to like the tools that they are more familiar with.

For the Inside Idaho data center website, the respondents expressed that they used this website more often and were more familiar with its contents. It is also predicted that the students (from Idaho State University) were mostly given the class assignments based on the data inside the Idaho.

The USGS EROS data center website was the least favorite search engine tool overall because the respondents found the website to be "clunky" and not well documented. Users did not like the fact that they needed to open up a map window to figure out the layers, perform location searches, or browse a File Transfer Program (FTP) directory structure if a map name was known before hand. This method was difficult for both beginner and intermediate users that weren't familiar with the data. Similarly, these users indicated that the existing GIS TReC search tool (Browse with folders) and USGS EROS tool were alike since the existing GIS TReC website is using web-browsing enabled file folder structure. As a result, the USGS EROS data center and the existing GIS TReC search engine were the least favorite tools among the users.

The database-driven search engine tool (Gohgle) was built to reduce the time and energy required to locate and download geospatial data and better ensure the correct data is downloaded to the users. The survey comparing Gohgle with three other search tools helped to determine that Gohgle had reached its expected outcome. The responses from the survey also brought to light Gohgle's perceived weaknesses which provide an opportunity for future enhancements to the tool. The final results suggest the tool did perform its required tasks and has the potential to be improved in the future.

LITERATURE CITED

Kauffman, J., K. Spencer, and W. Thearon 1999. Beginning ASP Databases. New York, NY: Apress. Pp12.

Markowetz, A., Y. Y. Chen, S., Torsten, X. Long, and B. Seeger June, 2005. Design and Implementation of a Geographic Search Engine. Eighth International Workshop on the Web and Databases (WebDb 2005), Baltimore, Maryland, June 16-17, 2005.

Thiemann, P. 2002. WASH/CGI: Server-Side Web Scripting with Sessions and Typed, Compositional Forms. S.Krishnamurthi, C.R Ramakrishnan (Eds): PADL 2002, LNCS 2257, pp.192-208, 2002. Springer-Verlag Berlin Heidelberg.

[THIS PAGE LEFT BLANK INTENTIONALLY]

[THIS PAGE LEFT BLANK INTENTIONALLY]