Enhanced Web Publishing: Towards integration of search and browsing

Natasa Milic-Frayling
Ralph Sommerer

Microsoft Research
7 JJ Thomson Avenue
Cambridge CB3 0FB, UK

{natasamf, som}@microsoft.com

ABSTRACT

Information access on the Web typically involves combination of search and browsing. The users may consult a centralized Web search service or browse through on-line Web directories to identify the URL of a relevant page. As they access the page they may engage in further navigation through Web sites by following hyperlinks. During both search and browsing only a limited site context is provided.

The URL of a page reflects the hierarchical file organization of the Web site content. Navigation elements on the page, such as menus, expose the high level navigation structure of the site. A sitemap, implemented as a separate Web page, provides a high level overview of the site content but without explicit reference to the currently viewed page or indication of the navigation structure. In all these cases, large parts of the site content and structure remain unrepresented, constraining our ability to provide context for search results within Web structure.

Similarly, on small devices such as Personal Digital Assistants (PDAs), Web pages of complex layout structure require extensive horizontal and vertical scrolling. That leads to user's disorientation during page viewing. Furthermore, in the context of search, there is no facility to aid systematic navigation through the page and identify relevant structural elements of the page.

This paper promotes the idea of generating site and page structure information at the authoring or publishing time and providing that metadata to applications and services upon request. We describe the MIDAS framework (Meta-Information Delivery and Annotation Services) for creating, distributing, and utilizing rich metadata of Web contents. We also present two client applications that utilize this metadata: SiteExplorer for the use of site structure and SearchMobil for the use of page structure, both enhanced by local search facilities.

1. INTRODUCTION

Information access on the Web is facilitated by various services, such as search engines and directories that provide the user with the URI (Universal Resource Identifier)[1] of relevant Web content. After accessing the page, the user typically engages in further exploration by following hyperlinks on the page. When performing these activities using standard browsers the user is presented with very limited information about the context of the viewed page. The page URI reveals a part of the directory structure of the site that the page belongs to. Menus and individual links on the page partially reveal the navigational structure of the site. However, from the limited information about these structures, it is often unclear how the page fits within either of them.

One way to explore parts of the site that are not directly linked to a page is by using local search capability on the site, if one is available. Otherwise, the user is strictly constrained to the site content accessible from a given page. In well designed sites, almost each page includes a menu that points to 'top-level' pages on the site. Following these the user can explore the site in the browsing mode. This can be time consuming and without guarantee that

[1] URI is also often referred to as URL (Universal Resource Locator).

the desired parts of the site would be covered. On the other hand, while search is more efficient for zooming onto potentially relevant pages, it is more or less separated from the site structure and the user's browsing experience. This prevents the user from judging relevance of search results by considering the page context. It also, does not connection with the conceptual model of the site that the user has built through navigation. Our goal is to remove this separation through providing users with representation of the site structure and content that could be annotated with both search results and the user's navigation history.

A similar need for explicit information about the structure is experienced when viewing Web pages on devices with small displays, such as Personal Digital Assistants (PDAs). This time, however, the structure refers to the layout of a single Web page, more precisely, the organization of logical elements of the page content. Most HTML pages are designed on the assumption that they will be viewed from a standard desktop screen. Those that involve complex layout cannot be automatically re-flown to fit on small size displays based on the current publishing format. Thus, the viewing of pages involves extensive horizontal and vertical scrolling which very quickly leads to the user's disorientation. On the other hand, Web pages are often composed from a number of logical units that could be viewed independently from the rest of the page content. For some tasks, such as search, this is particularly true. Access to content elements that contain relevant information, rather than the whole page, is the main objective in that scenario. However, the reference to the whole context is important. Thus, having a representation of the page structure that allows direct access to individual components, with annotations from search and browsing, is expected to be beneficial.

These two scenarios illustrate the need for generating and distributing information about the structure of information resources, such as sites and individual pages, alongside with the resource content. With this in mind we implemented a framework MIDAS (Meta-Information Delivery and Annotation Services) that facilitates generation and flexible distribution of various types of Web content metadata (Section 2). We demonstrate how site and page structure metadata can be exploited to provide a coherent search and navigation experience. This takes us closer to achieving a tighter integration of search and navigation in the Web environment, ensuring that the structure and the content representations are available. In Sections 3 and 4 we describe in more details our prototypes, the Site Explorer for the desktop and the SearchMobil for PDAs. They illustrate how the existing MIDAS framework makes the transition between navigation and search efficient and thus closely connect the two activities.

2. ENHANCED WEB PUBLISHING

Web is a general information resource that is to some degree customized for specific use by search engines and topic directories. However, further customization, sensitive to the user's context or task, has to happen 'on the fly'. This, in turn, prohibits use of any processes that require significant time and resources. For example, even a simple feature like document highlighting that correctly disambiguates multiple uses of a word may require deeper syntactic analyses and knowledge resources. Such is a problem of identifying references to proper names, e.g., 'Jasmine Baker' in the text that uses the same word surface form to refer to a common entity, e.g., 'jasmine tea'. Our observation is that some of these resource intensive processes are generic enough to be performed at the authoring or publishing time and made available to Web client applications or services as needed.

We thus implemented a prototype framework MIDAS that illustrates how various levels of text analysis can be captured and distributed using adopted standards. Our work can be contrasted with a far more ambitious Semantic Web initiative ([3],[4],[33]). Semantic Web aims at embedding knowledge into Web content and enabling seamless use of that information. We are focusing on simple metadata that can be captured at minimal cost and potentially bring significant improvement to the user experience when accessing information on the Web.

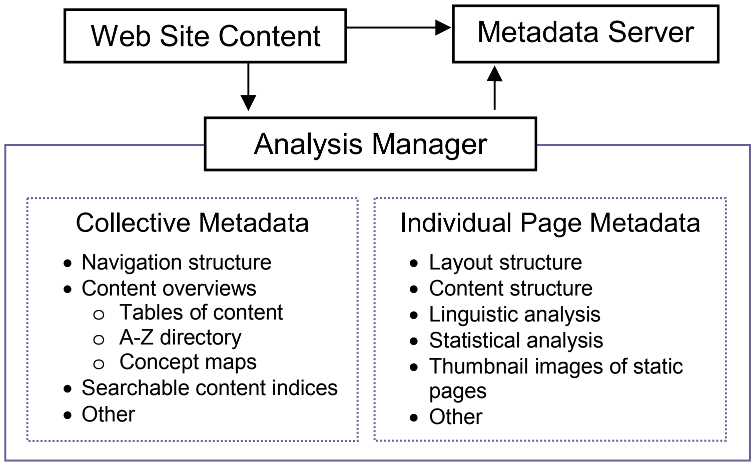

Figure 1. MIDAS Web Site Analysis Services

2.1 MIDAS Framework

Framework includes services for generating metadata within the Web site hosting server and for incorporating metadata from authoring applications. It also incorporates mechanisms for analyzing, extracting, and packaging metadata for distribution and consumption by services and applications. In the current implementation MIDAS coexists with the Web server and consists of two main components: the Analysis Manager which incorporates text analyses modules and Metadata Server that stores the metadata, processes metadata requests, and distributes the metadata to a client or a service (Figure 1). Depending on the usage scenarios, applications may require various subsets of the metadata. Thus we differentiate between the *core metadata* that is persisted alongside the content and *on-demand views* that are generated in response to the metadata requests.

For example, core metadata includes full structure of the site. However, if the application needs to present only a subset of the site, the metadata server will package and send only the relevant parts of the site structure. To facilitate flexible metadata views, we use the SQL server which provides the storage and analyses of the metadata. In conjunction with the standard Web server that delivers HTML documents, this provides a flexible and extendible framework for both data and metadata delivery. The sophistication level of the service is essentially determined by the metadata included in the database and the server's power to generate dynamic views upon request.

Communication between the client and the MIDAS service is achieved by function calls using SOAP protocol[2]. Metadata is then transported back to the client using MIDAS XML format. Depending on the scenario, XML is imbedded into the HTML content or delivered separately. The XML format that we adopted for representing structure of a Web site or a document is essentially an extension of the RSS standard[3]. Similarly, a simple transformation can render appropriate parts of MIDAS' XML format into RDF (Resource Description Framework) syntax ([15]).Our future work will include evaluation of these standards to see which can be effective in implementing MIDAS metadata.

---

[2] http://www.w3.org/TR/SOAP/

[3] RSS, standing for 'RDF site summary' according to the RSS 1.0 specification [http://web.resource.org/rss/1.0/spec] or 'really simple syndication' according to the RSS 2.0 description [http://backend.userland.com/rss], was designed initially for describing 'channels' of information—groups of related resources (typically HTML pages) with some associated timing metadata. It has been extended and repurposed several times, and can now easily be applied to any grouping of resources.

Linguistic analysis is performed using MS NLPWin software ([13]) that provides deep syntactic and limited semantic analysis of the text content. It is available for a number of languages but for now we incorporated only NLPWin for English language. The analysis includes syntactic analysis of sentences and full logical form with functional elements, subject-object-predicate triples, and clause formation. As a post process of the NLPWin analysis we also provide a partial anaphora resolution across sentences.

This variety of text analyses enables easy implementation of various features. For example, the NLPWin analysis can be used to implement a question-answering facility within a given site or a specified set of pages. On the other hand, the standard full-text search function of the SQL server can provide efficient, statistics based information search. Combined with the page analysis (section 2.1.2), we can build indices and perform search at the level of logical units and thus provide a fine grain feedback on the page relevance to the user's query.

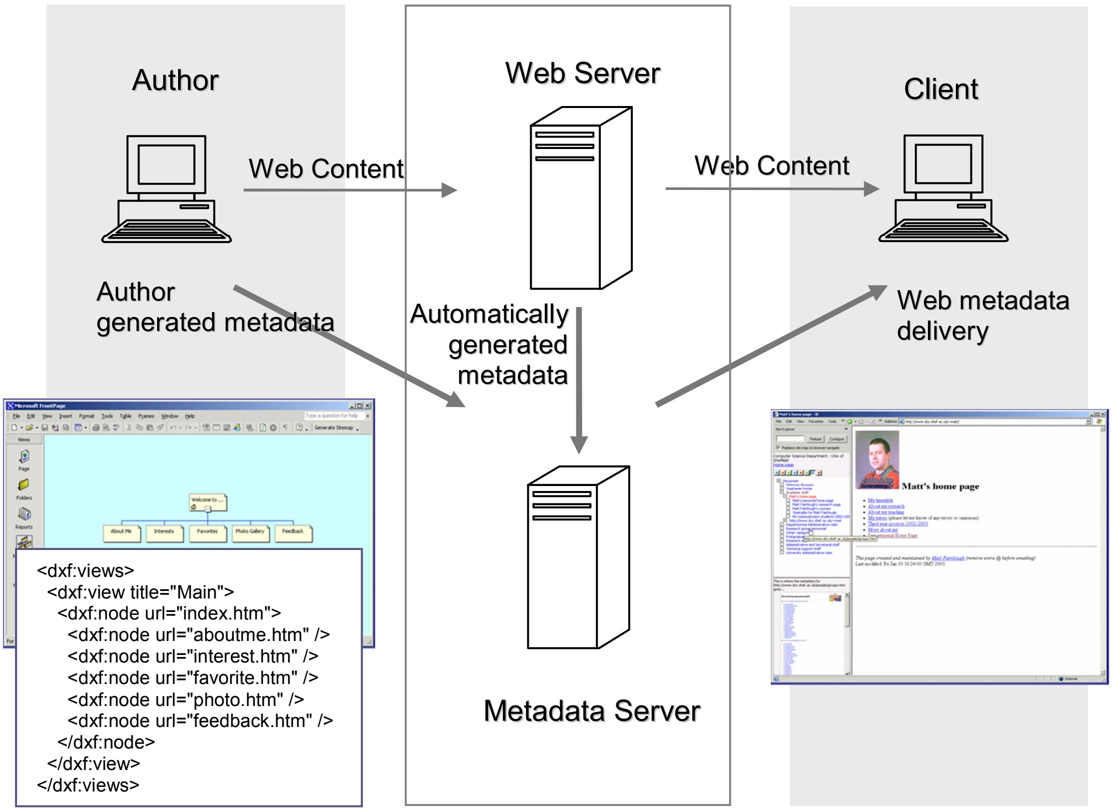

Figure 2. MIDAS Framework. Metadata created during authoring, with or without author's assistance, and metadata created during publishing are stored and served by the Metadata Server upon the request by a client application. On the left are an example of a template for site structure and the corresponding XML encoding of the structure. On the right, a client application presents the site structure, in the left pane.

2.1.1 Site Structure Analysis

For the purpose of this discussion, 'site' is a set of pages designated as a 'site' by the author or the site administrator. To a given set of pages, one can associate a number of different structures or views on the data, using hyperlinks. That is the essence of hypertext. However, the current Web publishing format, HTML, does not preserve any explicit structure of a set of pages. Even when the author explicitly specifies the structure of a site during the authoring process, this information is not stored in the HTML code of individual pages or anywhere else. Thus, the intended organizational is not preserved.

Viewed as a graph of links and pages, the site can be traversed in many different ways. Thus, inferring the structure from the links can be rather imprecise, if recovering the author's intended structure is the objective. A number of methods for inferring the structure have been explored in the context of hypertext research ([9],[26],[27]). While optimization of that process is of interest, it is not the primary focus of our work. In fact, while our objective is to show that site structure information is useful for facilitating information access, we highly recommend that the structure be preserved at the authoring time rather than inferred subsequently. In the current prototype we implemented a simple breadth-first traversal of site pages ([8]), avoiding cyclical subgraphs. The resulting structure can be used as the starting point for further manual editing, such as adding page titles where missing, or grouping links based on the page layout structure.

2.1.2 Page Structure Analysis

HTML does not explicitly describe conventional layout features such as multiple columns, sidebars, etc., commonly used in web page designs. Web site authors usually resort to HTML tables with fixed column widths and small blank images to achieve a desired content organization. On the other hand, a typical Web page is structured to contain a number of logical units of information, such as menus, tables, sections, forms, etc. Knowing more about these structural elements would be beneficial in many instances.

In MIDAS metadata we include the analysis of the page structure that is based on SmartView technology ([20],[21]). Page structure is represented in the form of a hierarchy of sub-elements, called logical sections, which are discovered automatically by recursively traversing the corresponding HTML document object model. The algorithm considers the arrangement and the size of tables, cells within tables, and forms. Applying few simple heuristics about the relative width and height of these elements, the algorithm determines whether a table or a cell is to be designated as a "logical section" or whether processing is to be continued recursively.

Figure 3. SmartView analysis of the page layout is indicated by green boarders around the regions of the logical sections.

At the moment no semantics is associated with the page decomposition, although some heuristics for discovering menus has been implemented. For each logical section of a page we further provide the structure in terms of paragraphs. Paragraphs, in turn, refer to the list of sentences, while each sentence is decomposed into the main and subordinate clauses, as appropriate.

With each static page we also associate a thumbnail image with the corresponding partition into logical sections.

3. CONNECTING SEARCH AND NAVIGATION

MIDAS framework enables us to explore the ways in which site structure and searchable indices can be used to provide effective access to information on the Web. In the following two sections we describe two prototypes: SiteExplorer, an extension of the desktop browser, and SearchMobil, an application for search support on PDAs.

3.1 Web Site Search and Navigation: SiteExplorer

On the Web, search and navigation are currently supported as two independent functions. Search is available on the global level, through centralized Web search services such as Google, MSN Search, NorthernLight, and similar. Individual sites also often include a local search facility. Response to the user query is typically a ranked list of document titles with a URL pointing to the location of the page on the Web. The same approach is taken when presenting results of local search.

Navigation, on the other hand, is facilitated by careful design and hyper-linking of pages. Extensive research has been done in the area of Web site design and usability to provide guidance to Web designers ([29],[29]). Using links to navigate through the information space does suffer from the 'tunnel vision' unless a conscious effort is made to provide sufficient context. This is typically attempted through careful positioning of menus on individual pages. Furthermore, authors also include site content overviews in the form of sitemaps and A-Z listings. Sitemaps are typically HTML pages with static content, showing top layers of the site organization, more or less related to the navigational structure. A-Z listings, on the other hand, provide an exhaustive list of site's in alphabetic order. This enables fast look-up but provides no reference to the navigational structure.

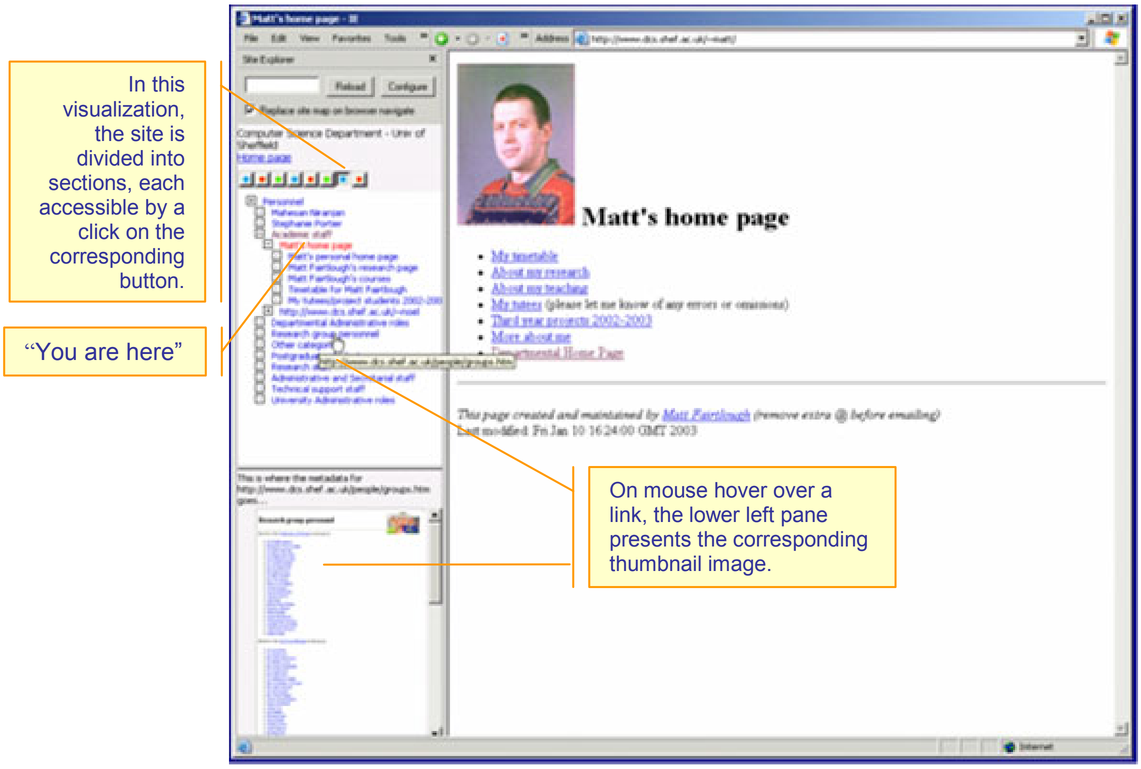

Figure 4. SiteExplorer user interface: the left pane includes display of the site structure, fully customizable by the author. In this example, a site consists of eight sections, represented by buttons with varying colors. The currently viewed page is in the section with 'selected' button. The display shows the hierarchical structure of that section, indicating the viewed page in red. On mouse hover over a link, the thumbnail image of the corresponding target page is shown in the bottom part of the left pane.

User research of standard Web sitemaps ([31]) has established the importance of this information access tool. However, it has been shown that, in the current format, sitemaps are not frequently used. The usability tests and resulting recommendations point to basic inadequacies: Web sitemaps are static and insensitive to the context; they have limited coverage, and are often not easy to find. Even the basic function that a map is expected to provide, the 'you are here' pointer, is not achieved since the Web sitemap is a Web page itself, imbedded in the site and accessed as any other page.

We should point out that standard browsers provide navigation support in the form of Back and Forward buttons, addressing the issue of capturing and exposing the local navigation history. Their aim is not to provide orientation within the site structure but to facilitate traversal of the user's navigation path that may cross multiple sites.

Setting our objective at providing an enhancement to the current Web information access by connecting search and navigation, we implemented an extension of the browser, SiteExplorer (Figure 4). In the design, we incorporated results and recommendations of the user research ([29],[31],[32]) and relied upon the MIDAS framework to create and expose the site navigation structure and provide the search facility.

3.1.1 SiteExplorer Features

SiteExplorer uses hierarchical organization of the site structure and local history of the user's navigation to facilitate browsing through a site. The idea of providing a Web site overview in the form of a hierarchical table of contents has been explored in the past, for example in the work by Nation ([28]). We are approaching this problem by recognizing that, in addition to a navigation hierarchy there might be multiple substructures of the graph that would be useful to users and desired by authors. Such are subsets of pages that possibly belong to various non-overlapping branches of the hierarchical organization but may form a coherent information resource for the site.

For example, for a university Web site, it might be useful to include access to the full list of staff members, although each department may already have a separate staff page. We refer to such groupings as *content views*. They may or may not be accessible through the site navigation elements. Most likely they would be available as a 'service' facilitated by a search or a browsing feature, such as A-Z listings, or similar. Page groupings may also be based on the navigation structure. We recognize that a site often consists of sub-sites which should in certain circumstances be treated as a unit and presented accordingly. Related to that, one of the content views could be a list of all the subsites accessible within a site.

Our current site visualization is obtained by editing of automatically inferred site structure and preparing graphics to be associated with top level nodes in the hierarchy. We rely on a two-tier display: the top level structure shows main areas in the site organization, typically following the menu structure of the home page. The detail view shows a detailed tree hierarchy associated with each element in the top level overview and exhibits the behaviour of a standard tree control in the browser. Thus, for a given page we can immediately view both the position of the page in the detailed hierarchy for a pre-defined surrounding area and the top level overview.

Alternative visualizations with different scope of page surrounding can be easily implemented. MIDAS metadata server can provide structure information at various levels of granularity. Furthermore, for each element in the sitemap, MIDAS can provide available page attributes: thumbnail overview of the page with logical structure indicators, list of most prominent phrases, and similar. The rendering of structure metadata is facilitated by a default style-sheet or a style-sheet specific to the site.

SiteExplorer can be used to support various information seeking scenarios. Conceptually, SiteExplorer sitemap is attached to a URL. The assumption is that for each URL there is a 'preferred' sitemap which should be displayed when user specifies the URL[4]. Thus, the site structure can be used without actually viewing the content of the pages.

[4] In the current version we make a simplifying assumption that for a URL there is a unique preferred sitemap.

For example, in relation to the Web search, for each URL in the result list the SiteExplorer can invoke the corresponding sitemap and show the context of the page within the corresponding site structure. Thus, the user can, by examining the structure and titles of surrounding pages, make further relevance assessment. One could argue that the associated site structure information could be best provided by the Web search service itself. Indeed, it may seem ideal to have both the structure and the content representation integrated at the index representation level. However, as long as Web crawling remains the chosen way of collecting information on the Web, this functionality would suffer from inaccuracies due to the dynamic nature of the Web content and structure. Thus, the site structure information would need to be provided directly from the hosting servers or the central search services would need to be notified of the site updates as they occur.

Furthermore, MIDAS' local search facility enables the user to focus on a particular site or part of a site. In contrast to standard site search facilities it marks titles of relevant pages within detailed hierarchical view of the site, providing clear visual feedback. This presents an alternative to standard ranked list of results. Similar approach has been taken in the past, for example, in designing a browser SuperBook ([11]) for exploiting hypertext content of a statistics textbook. The SuperBook has proven to be successful in supporting a variety of student tasks.

3.2 Page Viewing and Search on PDAs: SearchMobil

Both limited viewing area (see Figure 5) and slow access greatly influence the usage of mobile devices such as PDAs, thus limiting the spectrum of tasks that the user can efficiently accomplish. Various attempts have been made to address browsing and search related issues ([1],[2]) on such devices.

With respect to search, the user study by Jones et alt ([14]) measured the impact of small displays on the success of performing search tasks on the Web and points to a number of important issues. In particular, the users accessing the Web to answer a set of focused questions got twice as many questions answered correctly when using standard size screens as did the users who performed the same task on devices with small screens.

Furthermore, the analysis of navigation and search revealed that, while there was no significant difference in the number of hyperlinks selected and followed, small screen users resorted to search twice as many times as the large screen users. They followed shorter paths and returned to the search facility more frequently. Inspection of accessed pages also showed 50% non-overlap between the two groups which lead to the conclusion that, although small screen users did not make significantly many more page selections, those that they did make were inferior in quality in comparison to the full screen users. Among the recommendations of this study are design suggestions to reduce the scrolling, provide effective direct access method such as search, and give more focus to the less directed access methods, such as browsing.

Figure 5. A Web page with complex design, as seen on a Pocket PC.

On the other hand, we are also aware that the user's search strategy may involve a number of iterations as the topic of interest evolves with the user's exposure to data. Thus, similarly to our work on MSRead, an extension of the desktop browser [22], we put an emphasis on supporting refinement of search within a page or a designated set of pages and provide visual feedback through hit highlighting. In addition, we assume that PDAs are primarily used for focused information requests, rather than exploratory browsing. Thus, the user may desire to zoom quickly onto a detail within a page that satisfies the user's information need. This, in conjunction with the problem of discrepancy between page and display size, requires that the relevance assessment is provided at a finer level of granularity than the page itself.

Motivated by these observations we implemented SearchMobil, a prototype search environment for PDAs.

3.2.1 SearchMobil Features

SearchMobil relies upon SmartView technology for the analysis of layout and content and for creating the thumbnail representation of pages ([23],[24], Figure 6). It was originally implemented based on a proxy server which also served as a local search service. For each page requested by the user, the proxy would perform the page analyses and augment the local searchable index.

Here we present an implementation of SearchMobil that is supported by the MIDAS framework. Page layout analyses and thumbnail images of static pages are included in the MIDAS metadata server at the authoring or publishing time and can be served on demand (Section 2.1.2). For dynamic pages, the thumbnail representation is substituted by a page template which indicates the type of content in each logical section on the page. It is assumed that the content indexing happens at the time the content is authored and is therefore made available through MIDAS.

Figure 4. SmartView - overview of the page layout (left) and detailed view of a logical section on the page (right).

Upon the user's request, the browser first downloads the thumbnail image or a scaled-down template of a page, showing the page decomposition into logical units. Each element in the thumbnail partition can then be selected and loaded individually.

If the page was accessed in response to the search, the page structure is annotated to reflect the number of hits in individual logical units on the page ([24], Figure 7). This is possible because the browser captures the query when it is being submitted to the search engine. As the user selects a URL from the result list, the request is sent to the corresponding metadata server to obtain the thumbnail representation and hit annotations per elements of the page structure.

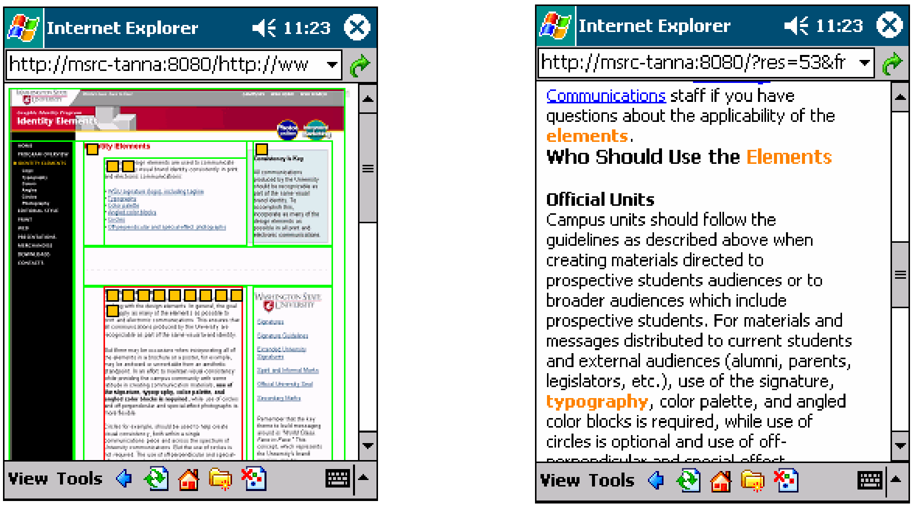

Figure 5. SearchMobil, showing one of the results from a search for "elements of typography". The image on the left shows the logical segments annotated with the number of hits, and the image on the right shows the selected segment with highlighted query terms.

SearchMobil also provides an interface for searching a set of pages simultaneously (Figure 8). This has, in fact, been designed for use with global Web search engines. For a given set of results, SearchMobil automatically generates a booklet of results by pulling the thumbnail representation and statistical and linguistic information from the corresponding metadata servers to create a searchable index on the PDA. With this information, cross document querying can be performed locally. Communication with the hosting server is then reduced to loading of the content of individual subsections of the pages. With the proper caching algorithm, the network traffic can be reduced while maintaining flexibility of the user's interaction with the page content.

The local search facility also complements browsing within the document structure, by providing the means for examining whether various elements on the page contain relevant information.

We performed a user study to investigate how useful SearchMobil is in various circumstances ([25]). It was assumed that the user's information need may be only partially expressed by the user's query and therefore the indication of search hits may not directly be useful. Thus, it was important to compare SearchMobil effectiveness with the standard user interface in various situations related to the co-occurrence of relevant information with query terms within logical units of the page. Furthermore, it was important to see whether the use of the thumbnail overview that presents the page structure gets in a way when the page cannot be decomposed into smaller logical units and provides no detailed location of the search hits. We measured the time required to perform the search tasks.

Investigation involved 9 users and showed that, in case when the highlights and the relevant information co-occurred in the section with most hits, thus with the red border, SearchMobil outperformed the standard IE browser. Situations where relevant information was in a section distinct from the one with most hits, i.e., without a red border, were very difficult to find. Those that we managed to identify confirmed our expectation that SearchMobil would perform worse since the users would be mislead by highlighting of the best region. Similarly, in situations where no decomposition of the page was done and, thus, no detailed hit highlighting was available, the standard interface performs better. Details of the study are presented in [25].

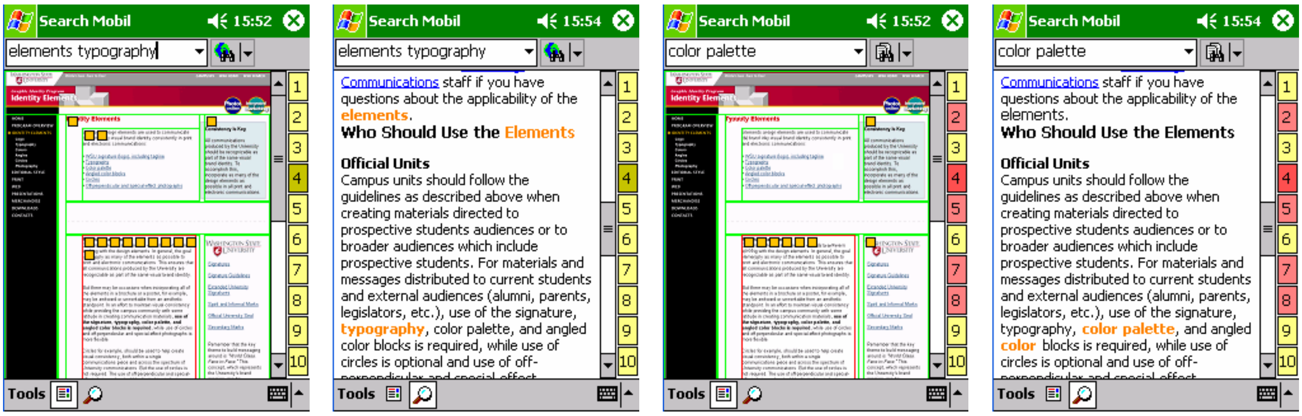

Figure 6. The SearchMobil interface shows the context of a particular document in a set of search results. First figure from the left shows the fourth document in the result list for the global search on "elements typography". It is followed by the view of a selected segment with highlighted query terms. The third image shows the local search over the original result set but for the query "colour palette". The tabs of relevant pages have changed colour and highlights show the new term hits. The same segment of the page is now shown with new query terms highlighted. Logical unit with most hits is designated with red border.

4. COMMENTARY

Merging of the two paradigms, searching and browsing, has been an area of active research in various research communities. From the search architecture point of view, the problem was defined as enabling or restricting search to a specific neighborhood of a page within the Web structure. For example, the WebGlimpse system ([18]), for searching over Web archives, starts with the assumption that data is partitioned into neighborhoods, specified at the indexing time. As the user browses around the archive, the search can be applied to explore the corresponding neighborhood. On the other hand, Letizia, a proactive Web browsing assistant ([16]), follows the user while browsing and indexes the neighboring pages; more precisely the pages that could be reached by proactive crawling. In this case, the neighborhood of a page is identified on the fly.

Furthermore, looking at the problem of providing context for search results, Chen at alt ([7]) has implemented the Cha-Cha system which to a certain degree connects search results and navigation structure. For every page the system creates in advance the shortest path from the root node to the page node. Each search result is therefore presented in the context of a path that is generated by combining the pre-defined shortest paths.

Current Web search engines, like Google (www.google.com) and WiseNut (www.wisenut.com), provide simple indication of the relationship between search results and the site structures by creating groups of result pages that belong to the same site.

Many interesting aspects of search and browsing have been explored within the hypertext community over the years. Among these is the question of the user's preference for search by browsing or, so called, 'analytic search' facilitated by a query. Depending on the task and the type of users, the studies showed mixed results ([6], [34]). Furthermore, in closed and stable hypertext systems, the content overview can be presented in the form of 'table of contents' which can be effectively used as a reference structure for showing search results and a starting point for browsing, as illustrated by the work on SuperBook ([11]).

In relation to the work that has been done so far, we are making a case for enhancing Web on-line publishing to include both content and metadata that characterizes the content structure. Instead of continuing to build impoverished models based on approximations of the resource structure we want to prepare a solid ground for integrating search and navigation based on rich structure information, both on the site and document level.

We argue that the best way to achieve this is by enhancing authoring tools or yet providing site management tools that can make the creation of metadata effortless. Client applications, such as SearchMobil and SiteExplorer, would provide authors with immediate benefit since their content would automatically gain a better exposure and higher level of utility from the user's perspective. We believe that this type of "push" and "pull" mechanism is the key for further advances in this area.

Over the past years, the Semantic Web initiative ([33]) has paved the road to adoption of metadata standards ([10],[15],[17],[19]), creation of metadata data resources, and development of metadata authoring tools ([12]). Our effort fits well into that context. It provides a scenario within which the usefulness of simple metadata brings immediate benefit and thus re-invigorates the idea of metadata on the wide scale, as opposed to serving a particular community, type of business, or specialized interest groups, as it has been the case so far.

5. REFERENCES

[1] Buyukkokten, O., Garcia-Molina, H., and Paepcke, A. Focused Web Searching with PDAs. In *Proceedings of the 9th World Wide Web Conference*, 2000.

[2] Buyukkokten, O., Garcia-Molina, H., and Paepcke, A. Seeing the whole in parts: text summarization for web browsing on handheld devices. In *Proceedings of the 10th World Wide Web Conference*, 2001.

[3] Berners-Lee, T. The semantics toolbox: Building semantics on top of XML, 1998. http://www.w3.org/DesignIssues/Toolbox.html

[4] Berners-Lee, T., Kerger, D., Andrea-Stein, L., Swick, R., and Weitzner, D. Proposal: Semantic Web Development, 2000. http://www.w3.org.2000/01/sw/developmentProposal.html.

[5] Botafogo, R. A., Rivlin, E. and Shneiderman, B. Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics. *ACM Transactions on Information Systems*, Vol. 10, No 2, April 1992, p.142-180.

[6] Campagnoni, F.R. and Ehrlich, K. Information Retrieval using a hypertext-based help system. *ACM Transactions on Information Systems*, Vol. 7, No. 3, p. 271-291, July 1989.

[7] Chen, M, Hearst, M., Hong, J. and Lin, J. Cha-Cha: A System for Organizing Intranet Search Results, in the *Proceedings of the 2nd USENIX Symposium on Internet Technologies and SYSTEMS* (USITS), Boulder, CO, October 1999.

[8] Cho, J, Garcia-Molina, H., and Page, L. Efficient crawling through URL ordering. In *Proceedings of the 7th WWW Conference*, Brisbane, Australia, April 1998.

[9] Conklin, J. Hypertext: an introduction and survey. IEEE Computer Society Press, Computer, Vol. 20, Issue 9, p.17-41, September 1987.

[10] Dublin Core Initiative and Metadata Element set. http://dublincore.org/.

[11] Egan, D.E., Remde, J.R., Landauer, Th.K., Lochbaum, C.C., and Gomez, L.M. Behavioral evaluation and analysis of a hypertext browser. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 205-210, May 1989.

[12] Handschuh, S. and Staab, S. Authoring and Annotation of Web Pages in CREAM. In *Proceedings of the 11th In. World Wide Web Conference* (WWW11), Honolulu, May 2002.

[13] Heidorn, G.E. (2000) Intelligent Writing Assistance. In Dale, R, H. Moisl, and H. Somers (Eds.) Handbook of Natural Language Processing (p.187-207) New York: Marcel Dekker.

[14] Jones, M., Marsden, G., Mohd-Nasir, N., and Boone, K. Improving web interaction on small displays. In *Proceedings of the 8th World Wide Web Conference*, 1999.

[15] Lassila, O. and Swick, R. Resource Description Framework (RDF) model and syntax specification, 1999, http://www.w3.org/TR/REC-rdf-syntax.

[16] Lieberman, H. Letizia: An agent that assists web browsing. *Proceedings of IJCAI*, 1995.

[17] Liechti, Ol, Sifer, M.J., Ichikawa, T. Structured graph format: XML metadata for describing Web site structure http://www.oasis-open.org/cover/sgfWWW7.html.

[18] Manber, U., Smith, M. and Gopal, B. WebGlimpse – Combining Browsing and Searching. Usenix Technical Conference, January 1997.

[19] Martin, P. and Eklund, P. Embedding knowledge in Web Documents. In *Proceedings of the 9th In. World Wide Web Conference* (WWW8), p.1403-1419, Toronto, May 1999.

[20] Milic-Frayling, N. and Sommerer, R. SmartView: Flexible Viewing of Web Page Contents. Poster presentation at the 11th World Wide Web Conference, 2002.

[21] Milic-Frayling, N. and Sommerer, R. SmartView: Enhanced Document Viewer for Mobile Devices. Microsoft Technical Report: MSR-TR-2002-114, November 2002

[22] Milic-Frayling, N., and Sommerer, R. MS Read: Context Sensitive Document Analysis in the WWW Environment, Microsoft Technical Report, MSR-TR-2001-63, 2001

[23] Milic-Frayling, N. and Sommerer, R. SearchMobil: Search support interface for PDAs, Workshop on Mobile Personal Information Retrieval, SIGIR 2002, Tampere, Finland, August, 2002

[24] Milic-Frayling, N., Sommerer, R., Rodden, K., and Blackwell, A. SearchMobil: Web Viewing and Search for Mobile Devices, Poster presentation at the 12the World Wide Web Conference, Budapest, May 2003

[25] Milic-Frayling, N., Sommerer, R., Rodden, K., and Blackwell, A. Effective Web Searching on Mobile Devices, HCI 2003, Bath, United Kingdom, September 2003

[26] Mukherjea, S. and Foley, J.D. Navigational View Builder: A Tool for Building Navigational Views of Information Spaces. In *ACM SIGCHI'94 Conference Companion*, p.289-290, Boston, April 1994.

[27] Mukherjea, S., Foley, J.D., and Hudson, S. Visualizing Complex Hypermedia Networks through Multiple Hierarchical Views. In *Proceedings of ACM CHI 1995*, Denver, Colorado, May 1995.

[28] Nation, A. Visualizing websites using a hierarchical table of contents browser: WebTOC. In *Proceedings of the Third Conference on Human Factors and the Web*, Denver, CO, 1997.

[29] Nielsen, J. *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000.

[30] Nielsen, J. and Tahir, M. *Homepage Usability: 50 Websites Deconstructed*. New Riders Publishing, Indianapolis, November 2001.

[31] Nielsen, J. et alt. *Site Map Usability Report*, Nielsen Norman Group Report, 2002.

[32] Rosenfield, L. and Morville, P. *Information Architecture for the World Wide Web*, O'Reilly, August 2002.

[33] S3C Semantic Web Activity: www.w3.org/2001/sw.

[34] Wildemuth, B. M., Friedman, Ch.P., and Downs, S.M. Hypertext versus Boolean access to biomedical information: a comparison of effectiveness, efficiency, and user preferences. *ACM Transactions on Computer-Human Interaction*, Vol. 5, Issue 2, June 1998.