BALANCING HUMAN AND SYSTEM VISUALIZATION DURING DOCUMENT TRIAGE

A Dissertation

by

SOON IL BAE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2008

Major Subject: Computer Science

BALANCING HUMAN AND SYSTEM VISUALIZATION DURING DOCUMENT TRIAGE

A Dissertation

by

SOON IL BAE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Frank M. Shipman, III
Committee Members,	Richard K. Furuta
	John Keyser
	Ergun Akleman
Head of Department,	Valerie E. Taylor

December 2008

Major Subject: Computer Science

ABSTRACT

Balancing Human and System Visualization during Document Triage. (December 2008)
Soon Il Bae, B.S., Yonsei University;
M.S., Yonsei University
Chair of Advisory Committee: Dr. Frank M. Shipman, III

People must frequently sort through and identify relevant materials from a large set of documents. Document triage is a specific form of information collecting where people quickly evaluate a large set of documents from the Internet by reading (or skimming) documents and organizing them into a personal information collection. During triage people can re-read documents, progressively refine their organization, and share results with others. People usually perform triage using multiple applications in concert: a search engine interface presents lists of potentially relevant documents; a document reader displays their content; and a text editor or a more specialized application records notes and assessments. However, people often become disoriented while switching between these subtasks in document triage. This can hinder the interaction between the subtasks and can distract people from focusing on documents of interest. To support document triage, we have developed an environment that infers users' interests based on their interactions with multiple applications and on an analysis of the characteristics and content of the documents they are interacting with. The inferred user interest is used to relieve disorientation by generating visualizations in multiple applications that help people find documents of interest as well as interesting sections within documents.

DEDICATION

To my father

ACKNOWLEDGEMENTS

Returning back to academia to start a Ph.D. at Texas A&M after working in industry for 10 years was a big challenge for me. However, thankfully, I had many people who encouraged me.

First I want to thank my family. My wife always supported me when I was struggling with my research problems. My two sons, Uihyun and Junghyun, had patience with my study even when it kept being delayed. They pleased me. Even when we were separated because of my wife's return to her job in Korea, we encouraged and cared for each other. This allowed me to finish my dissertation. My parents always supported me.

Next, I want to recognize my committee. First I want to thank my advisor, Dr. Frank M. Shipman III. He has been helpful and patient even when my progress was not noticeable. Without his feedback, advice, and mentoring, I would not have finished my dissertation. I also want to thank Dr. Richard Furuta, Dr. John Keyser, and Dr. Ergun Akleman for their time and feedback.

My friends and colleagues in the Center for the Study of Digital Libraries helped me in many ways. In particular, I want to thank Michael and Dohyoung. The time at Texas A&M was not only for study but also for opportunities to learn many things and also to make many good friends.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

1 INTRODUCTION

The Internet has become an important information source, and searching for information on the Internet is now commonplace. Many searches return thousands, if not millions, of matching documents. This dissertation is focused on document triage, a specific form of information collecting, reading (or skimming), and organizing, where people quickly evaluate a large set of documents from the Internet and organize them into a personal information collection. They can re-read documents, progressively refine the organization, and share results with others. In particular, with unfamiliar topics, people learn about the topic as they read documents and organize them. During this process, their knowledge forms incrementally, as their initial understanding of the topic changes and becomes more refined over time. This dissertation explores the potential and challenges for human-authored and system-generated visualizations to help people quickly recognize relevant documents and efficiently organize them.

Visualization has the potential to help people be more effective and efficient during document triage, and to improve the resulting organization of documents. One example of the potential for visualization is that document triage involves quickly switching attention between reading and organizing activities and between documents. These transitions in attention result in people forgetting which documents they have already looked at and feeling like they must take a second look at all documents to ensure proper categorization. This revisiting of even irrelevant documents results in less

---

This dissertation follows the style of the *ACM Transactions on Information Systems*.

attention on the valuable documents found. Visualization can be used to indicate prior visits and inferred importance or similarity to important documents.

However, a prior study of document triage in VKB shows that system-generated visualization can conflict with user-authored visual expression [Shipman, Hsieh et al. 2004]. This conflict discourages users from expressing their ideas visually. Therefore, it is essential to balance the system-generated visualizations and the user-authored visual expression so that the conflict may be minimized while retaining the benefit of the system visualization. The visualization that this dissertation considers includes separate visual features for system and user visualization. The system-generated visualization combines indications of a user's prior activity with user interests heuristically inferred from user behavior.

This dissertation is fundamentally based on the Visual Knowledge Builder (VKB), a spatial hypertext tool providing an integrated environment for searching and organizing documents from the Internet. A *document object* in VKB can refer to a Web page (or other document) and opens up the Web page (or other document) when users select "open URL" in a dialog. VKB provides a hierarchical workspace by introducing a *collection* that is a container of document objects or other collections. Users express their opinions about objects by organizing them in collections and changing visual attributes, such as color or border width. The design, implementation, and evaluation of the mechanism for human/system visualization in document triage are within the context of VKB.

The design and development of human-authored and system-generated visualization of this dissertation was inspired by the prior studies of document triage [Marshall and Shipman 1997; Shipman, Hsieh et al. 2004; Bae, Badi et al. 2005] . Section 2 describes the problems observed from the prior studies. Section 3 introduces related work in three parts: systems for human visual expression, visualization of search results and Web-based documents, and visualization of user interests in a document or document collection. Section 4 introduces quantitative and qualitative analysis of user behavior from a prior study of document triage [Bae, Marshall et al. 2006].

Section 5 presents high level approaches to deal with the observed problems, which are primarily based on the implications from the prior studies. The system is composed of three subsystems: the new Document Object in VKB, a Firefox extension to enhance reading called WebAnnotate, and the Interest Profile Manager (IPM). Section 6, 7, and 8 introduce the design and development of each subsystem.

Section 9 describes an evaluation comparing alternative visualizations, including visualizations based on the approach described in this dissertation and the visualization from the previous version of VKB. Finally, the conclusions and future work are presented in section 10.

2 PROBLEM

Document triage can be divided into three intertwined activities, searching, reading, and organizing: people quickly evaluate documents on Web search results, selecting documents to read, perform short forms of reading, and organize the information. People re-read the documents, and progressively refine the organization.

2.1 Lack of Support for Interaction between Searching, Reading and Organizing

Switching between the subtasks can cause people to lose their place and to have trouble remembering and utilizing what they have already learned [Neerincx, Lindenberg et al. 2001; Monsell 2003]. This problem indicates the potential value for systems to prior activity to proactively support later activity in document triage tasks.

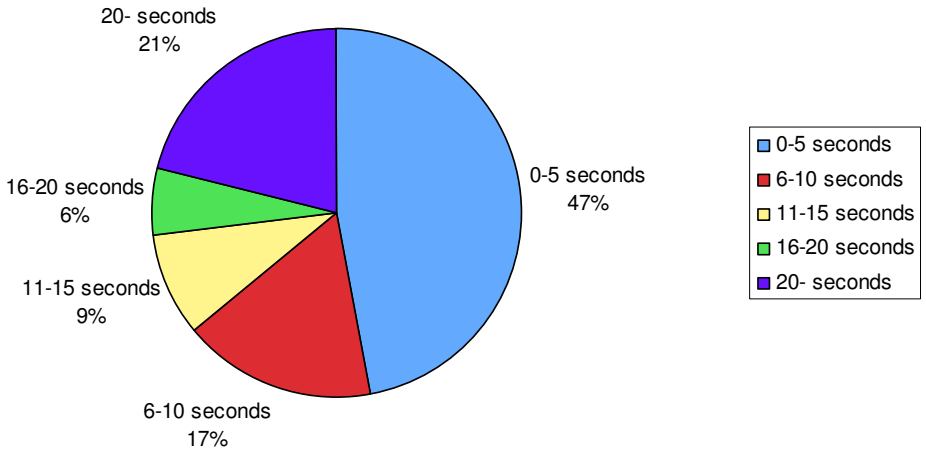


Figure 1. Reading time spent in document triage

Prior studies of document triage show that people do quick forms of reading, such as skimming or reading short portions of a longer document, while trying to quickly

determine the disposition of relevant documents rather than reading from the beginning to the end of documents [Marshall and Shipman 1997; Shipman, Hsieh et al. 2004; Bae, Badi et al. 2005]. Figure 1 shows the distribution of how long subjects read a document once they visit it in an earlier study of document triage [Bae, Badi et al. 2005]. It shows that 47% of reading occurrences lasted less than 5 seconds, while reading of more than 10 seconds occurred 36% of the time. In other words, when subjects visited a document, 47% of the time they would switch to another document or application within 5 seconds. Clearly, short forms of reading (e.g. glancing, skimming) are part of common practice for the task in the study.

Without any prior knowledge about the given topic, subjects process through documents incrementally, initially needing a significant amount of effort to get a sense of the given topic and documents. People predict the relevance of documents using metadata, such as the page title, URL and summary provided by search engines: in the same prior study, 14 out of 19 subjects mentioned that they did pay attention to the metadata, the page title, URL and summary, before they read documents [Bae, Badi et al. 2005]. This affects subsequent reading and organizing. Sometimes, people delete document objects – surrogates that represent documents such as icons in a file folder – and organize them based on the metadata without ever viewing or reading the corresponding documents. However, people normally build up semantic categories in an incremental manner as they read parts of documents, and organize subsequent document objects encountered based on previously-defined categories. Thus, organization affects

the subsequent searching for and reading of documents. In this way, searching, reading and organizing are closely intertwined with each other in document triage.

In the prior study of document triage [Bae, Badi et al. 2005], 13 out of 17 subjects mentioned that they read the page title and URL to remember the previously read documents during organizing. However, prior studies of document triage [Shipman, Hsieh et al. 2004; Bae, Badi et al. 2005] also show that subjects are often confused about: (1) whether or not they have previously read a document referred to by a document object, and (2) which document object in VKB matches the document currently opened in a Web browser. In addition, some people have difficulty remembering or locating previously read document objects in VKB. This indicates that various forms of visual breadcrumbs, cues to help people remember or recognize/remember their prior activity, are crucial in a system-generated visualization supporting triage activity.

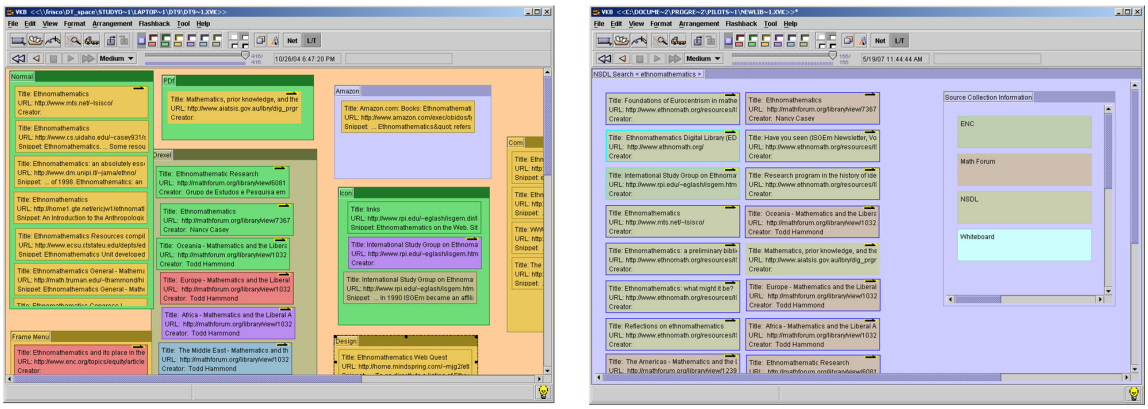
The prior study [Bae, Badi et al. 2005] also shows that frequent switching between the three document triage subtasks occasionally causes people to forget planned activities and more generally breaks people's cognitive flow. This distracts subjects and hinders the intermediate knowledge acquired during document triage from being fully utilized in later phases of document triage. Visualizations based on patterns of user behavior during tasks may potentially be used to improve the interaction between document triage subtasks and subsequently improve cognitive flow.

Thus, this dissertation has designed, developed, and evaluated a visualization that combines breadcrumbs indicating prior activity with visualization aimed at predicting

desired future activity. Instantiation of such a visualization is made more difficult by the potential for interference with user expression.

2.2 Conflict between User-Authored and System-Generated Visualization

User-authored visual expression is an important part of the sense-making process in document triage practice. Subjects in prior studies of document triage [Marshall and Shipman 1997; Shipman, Hsieh et al. 2004; Bae, Badi et al. 2005] took advantage of the visual expression provided to them by building hierarchic collections of document objects and made visual annotations by changing visual attributes (e.g. background and border colors, border width, size) of document objects and making comments. While all using the same visual attributes, subjects created different patterns of organization according to their individual preferences or styles (Figure 2.a).



(a) (b)

Figure 2. User-authored visualization (a) and system-generated visualization (b)

This user-authored visual expression has to share the same visual workspace with system-generated visualization. The conflict between the two different types of

visualization is inherently unavoidable – any visual attribute modified by the author or by the system is either unavailable to the other (author or system) or becomes a point of potential contention or miscommunication. A prior study of document triage including an initial system-generated visualization based on metadata is an example of such a conflict (Figure 2.b showing the source of information using different colors). In this case, users had to override the system-generated visualization in order to use document object color as part of their own interpretation. The results showed that users were less willing to express themselves using color, feeling that they would lose information by changing the system-provided color [Shipman, Hsieh et al. 2004]. Therefore, regardless of the value of the system-generated visualization, it has the potential to worsen the overall performance of document triage due to its impact on user expression.

2.3 Summary of Problems and Goals

For a visualization system to support document triage, the visualization needs to enable a rapid assessment of progress on the current task generally and the status of individual documents specifically and to provide system-generated visualizations increasing the time users spend with high-value documents without hindering user expression.

3 RELATED WORK

The research into balancing human and system visualization during document triage has three areas of related work: systems for human visual expression, visualization of search results and Web-based documents, and visualizations based on inferred user interests.

3.1 Systems for Human Visual Expression

Malone's analysis of the physical offices of ten people describes how the cognitive difficulty of categorizing information impacted leads people to often prefer spatial classifications [Malone 1983]. In this study, uncertainty about future information use caused people to use less explicit and less permanent groupings of information, such as piles on tables and desks, rather than classifications requiring a long-term commitment to structure, such as labeled files in a file cabinet. Many computer interfaces have been developed with the goal of mimicking this form of spatial expression. Indeed one example is the desktop metaphor, which has been the de facto user interface for desktop computers since the 1980s [Ravasio, Sch et al. 2004]. People organize icons representing files and applications in 2D desktops in MS Windows or Mac OS, and can further classify icons by adding distinct background colors to the icon label (Mac OS).

The potential for spatial expression of formal relations and associations was explored by diSessa and Abelson [1986]. Their system, Boxer, supported programming by non-experts by providing a reconstructible medium that could be personalized to individual needs. The system design recognized that spatial relations are considerably

expressive and semantically meaningful with computational objects represented as boxes containing text, graphics, or other boxes in a two-dimensional space. The Boxer system represented the programming hierarchy (e.g. the hierarchic decomposition of the application being programmed) as boxes within boxes. Users could expand a box to full screen and enter the sub-environment of the box.

TopicShop [Amento, Hill et al. 1999; Amento, Terveen et al. 2000] is an integrated interface for information analysis on the Web focused on finding relevant information, evaluating the quality of information in the collection, and organizing information in the collection. *Topicshop* employs thumbnails augmented by textual annotations as memory aids and *site profiles*, containing the page title, an estimate of the page count, the number of in and out links, and a roster of audio, movie and image files in order to help people evaluate Web sites.

Data Mountain [Robertson, Czerwinski et al. 1998; Czerwinski, Dumais et al. 1999; Czerwinski, van Dantzich et al. 1999] is a 3D (actually 2½D) document management system designed to take advantage of human 3D spatial memory as an alternative to the current bookmark mechanisms of Web browsers. Users can position thumbnail images representing Web pages at any place on an inclined plane: the layout on the 3D workspace is very personal, but may imply meaning for users who created it. They claim that people built a very accurate mental map regarding the categories even with no mechanism for labeling thumbnail images. Czerwinski et al. explored the utility of retrieval cues of the Data Mountain in retrieving previously stored Web pages [Czerwinski, van Dantzich et al. 1999]. They observed that removing thumbnail images

initially slowed down retrieval times and increased failed and incorrect retrievals, but the difference went away soon as subjects retrieved Web pages. Among the four different retrieval cues, thumbnail image, mouse-over text, spatial location memory, and audio feedback, subjects answered on average thumbnail image was most helpful.

Spatial hypertext systems, such as VIKI [Marshall and Shipman 1995] and VKB [Shipman, Hsieh et al. 2001], use a similar hierarchy of spaces for expressing relationships between information objects. These systems are workspaces for collecting information where the user has control over a number of independent visual attributes (e.g. the border width, background color, border color, and font type, size and color) for expressing characteristics of information objects and associations between information objects. Over time, as the incremental knowledge building process proceeds, the meaning of particular visual features become more well defined. This results in the incremental development of visual languages for collection building, metadata assignment, and knowledge representation [Shipman, Hsieh et al. 2004].

Marshall et al. investigated how people searched and organized information in paper-based triage and in triage conducted using the spatial hypertext system VIKI [Marshall and Shipman 1997]. The study showed that people naturally used visual attributes, spatial layout, hierarchic categorization, and annotations for organizing information in triage. Shipman et al. explored how people performed document triage with Web materials [Shipman, Hsieh et al. 2004]. They compared user behavior with two different software configurations: (1) a Web browser and VKB; (2) a Web browser and text editor such as Microsoft Word and Notepad. The results showed that people felt

better able to express their ideas using VKB. It also indicated that a system-generated visualization impedes human visual expression, since people worried about losing information provided by the existing visualization.

A diary study of reading demonstrated that more than a quarter of the writing that happened during work-related reading comes under the form of annotation [Adler, Gujar et al. 1998]. Prior studies showed that various forms of annotation have their own functions: e.g. improving understanding of documents, helping memory, interpreting on the document, reflecting readers' interests, and organizing reading for later review [Marshall 1997; Price, Golovchinsky et al. 1998; Schilit, Golovchinsky et al. 1998]. Wolfe mentioned annotation could be used as a bridge between reading and writing [Wolfe 2000].

XLibris is a reading device that allows users to make free-form ink annotations in paper-like interface [Price, Golovchinsky et al. 1998]. In particular, XLibris suggests related documents and further reading list by interpreting the annotations on the current document. Marshall suggested design implications for annotation tools based on the study of annotations in university textbooks [Marshall 1997].

3.2 Visualization of Search Results and Web-based Documents

A number of approaches to the system-generated visualization of search results and the presentation of Web-based documents have been explored. These can be divided into the visualization of a set of documents and the visualization of a single Web-based document.

For working with a set of documents, Card et al. introduced a tightly integrated environment composed of the *WebBook* and the *Web Forager* [Card, Robertson et al. 1996]. The *WebBook* is a 3D interactive book of individual Web pages and allows users to read and to organize them. Users can access each Web page in a sequential way, such as page-turning and buttons, or by a *focus+context* interface called *Document Lens*. In contrast, the *Web Forager* allows users to search for Web documents and to organize multiple entities of the *WebBook* in three hierarchical levels for searching and organizing. Dumais et al. evaluates Category and List interfaces of Web search results with different conditions [Dumais, Cutrell et al. 2001]: primarily, Category interfaces organize Web search results into hierarchical categories, while List interfaces show Web search results by rank. They shows that all Category interfaces were more effective in the given search tasks than all List interfaces regardless of the different conditions. In addition, they suggest summaries of Web pages and the category name as meaningful context for Web searching. Paek et al. address the problem of the limited screen real estate in displaying Web search results by introducing a dynamic layout technique called WaveLens [Paek, Dumais et al. 2004]. The WaveLens technique employs a fisheye lens to compact many search results without scrolling and adapts the layout of the search results to user interaction.

To visualize a single Web-based document, Wynblatt et al. introduce a *Web caricature*, a visual representation of a Web page, which highlights the key points of a Web page to help a user quickly figure out the disposition of a Web page [Wynblatt and Benson 1998]. They employ a *document feature vector* of four categories: basic

properties, measurements of link density, measurements of media content, and a representative image. They suggest the presentation of *Web caricatures* as an alternative to the current text-based Web search results. Similarly, Woodruff et al. propose an *enhanced thumbnail* as a Web search result. The enhanced thumbnail is the result of image processing [Woodruff, Faulring et al. 2001]: changing the fonts to make the text more readable and highlighting callouts (enlarged text overlays on top of the original thumbnails) for keywords. They claim that text-based search interfaces, plain thumbnail search interfaces, and enhanced thumbnail search interfaces show different performance for different categories of search queries and tasks. Dziadoz and Chandrasekar evaluate how users investigate Web search results and estimate which items will lead to the wanted information with three different visualizations [Dziadosz and Chandrasekar 2002]: text-only, thumbnail preview only, the combination of text and thumbnail visualizations. The user study shows that the combination of text and thumbnail reduce errors in predicting the relevance of a document. They claim that thumbnails alone might hinder more than help users search Web pages that they have not previously visited.

Many prior studies using thumbnail images of documents are focused on computer-generated visualization, where users cannot change the visual attributes of document objects [Robertson, Czerwinski et al. 1998; Wynblatt and Benson 1998; Czerwinski, Dumais et al. 1999; Czerwinski, van Dantzich et al. 1999; Woodruff, Faulring et al. 2001]. However, user-authored visual interpretation is a crucial part of document triage.

3.3 Visualization of User Interests in a Document or Document Collection

A third form of visualization related to document triage is system-generated visualization based on users' explicitly defined or heuristically inferred interests. The Reader's Helper [Graham 1999] is a personalized document reading environment for helping users quickly evaluate documents and locate the relevant portions of a document based on user profile including information about the users. It estimates the relevancy of document and visualizes it on the document by highlighting, bolding, and underlining phases. The visualization is dynamically displayed on the Thumbar, a thumbnail image of the document with scrolling functionality: for example, the Thumbar demonstrates relevant phases by red lines so that users can quickly navigate the relevant portions of the document by dragging the lens of the Thumbar. The Galaxy of News system [Rennison 1994] provides 3D visualization of relationships between a large set of news articles, which covers the abstraction of the entire articles and automatically reorganize the visualization to the focus of interest through user interaction. The system parses the content of news articles and constructs relationships between them. The relationships are represented as visual clustering that is automatically reconstructed as users zoom or pan the information space.

An example is found in *XLibris* [Price, Golovchinsky et al. 1998; Price, Schilit et al. 1998; Schilit, Golovchinsky et al. 1998; Schilit, Price et al. 1998], a system that allows users to make free-form annotations directly on a document in a pen-based computer while they are reading. To recognize user interests, Shipman et al. designed the mark parser to categorize and to rank users' annotations in order to identify high-

emphasis passages of a document [Shipman, Price et al. 2003]. High-value passages are visualized in a thumbnail overview using colors and icons based on their rank. Different types and numbers of annotations on a document result in different visualizations of the document. *XLibris* integrates a number of the features of document triage emphasized in this dissertation: reading, annotation, and inference of user interest.

Data Mountain visualizes the similarity of Web pages when users select Web pages on 3D workspace [Robertson, Czerwinski et al. 1998; Czerwinski, Dumais et al. 1999; Czerwinski, van Dantzich et al. 1999]: it suggests Web pages similar to the currently select Web page by using a green outline around a thumbnail image on the workspace. *Data Mountain* employs similarity metrics called *Implicit Queries* for estimating the similarity. There are two types of similarity metrics included: co-occurrence similarity and content-based similarity. Co-occurrence similarity is derived from previous users' classifications of Web pages, while content-based similarity is based on the term vector model from information retrieval. The user study shows that users using Implicit Queries created somewhat more categories and Web page retrieval time tended to be faster. *XLibris* and *Data Mountain* exemplify how two different types of user activity can be used to infer user interests. *XLibris*' query-mediated links are based on annotations that are created while the user is reading a single document. The *Implicit Queries* in *Data Mountain* are based on the text content and the organizing activity concerning multiple documents. As is indicated by these examples, both user reading and organizing behavior can be used to infer user interests.

Kaasten and Greenberg introduce the integrated Back, Bookmarks, and History facility in a Web browser, which represents a Web page as a thumbnail image to help users recognize the previously visited Web pages [Kaasten and Greenberg 2001]. As users visit Web pages, the system creates small thumbnails of the pages into a list so that users can scan for locating Web pages later. The system employs system-generated visualization and user-authored visualization for page marking: the system visualizes the importance of Web pages by a vertical band whose height and color implies visit frequency, while users can express the interests on the Web pages by dog-earing thumbnails.

4 USER BEHAVIOR DURING TRIAGE

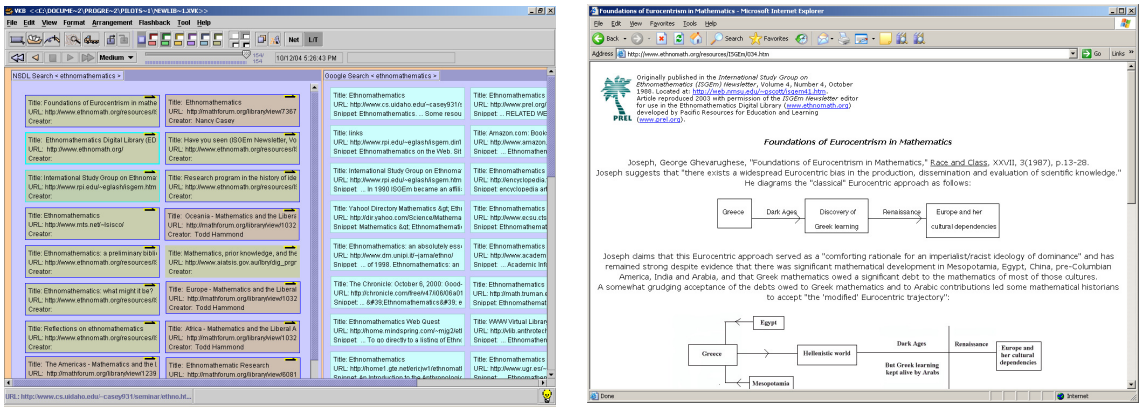
To better understand the work practices of users engaged in triage tasks, we analyzed user activity and results from a prior study. The analysis focused on (1) what attributes tend to be shared by documents that users find valuable, (2) whether user activity correlated with their assessment of document value, and (3) what strategies/processes did users follow while performing the triage task. The results of (1) and (2) inform the design of algorithms for identifying documents of likely value while the results of (3) inform the design of the overall environment supporting triage.

4.1 User Study

This analysis of user behavior is based on the prior study of document triage at Texas A&M University in 2004 [Bae, Badi et al. 2005]. 24 subjects (19 males) were recruited within the Computer Science department. 23 out of 24 of the subjects had more than 5 year experience in computers: all of the subjects were heavy computer users. Subjects organized material regarding ethnomathematics in the role of a reference librarian who supported a high school teacher preparing a class. Subjects started with 40 documents (Web pages) about ethnomathematics: 20 documents responded by the National Science Digital Library (NSDL) and 20 documents responded by Google.

Figure 3 displays the initial view of the documents in VKB (left) and a document opened for reading in a Web browser (right). Each document was represented in the overview of VKB by a *document object* that linked to the document. The document object displays metadata of the document: the document's title, URL, creator (only for the NSDL results), and a document summary (only for the Google results). Documents

are opened in a Web browser (Internet Explorer–IE) by double-clicking on the document object. VKB allows users to search and organize documents using a hierarchy of two-dimensional workspaces, *collections*, including document objects or other collections. Users express their ideas about documents by organizing the document objects in the workspaces and by changing visual features such as color or border width.



(a) (b)

Figure 3. A VKB space for searching and organizing (left) and Internet Explorer for reading (right)

After the triage task, subjects were asked to fill in a questionnaire about the task and to choose the five most and least useful documents. Short interviews were conducted for clarifying or elaborating some answers on the questionnaire. In addition, reading and organizing related user activities were recorded: e.g. scrolling and click events in IE and changing visual attributes in VKB.

4.2 Document Attributes and User Interest

In the interview after the completion of the triage task, subjects were asked how they selected the five most and least useful documents. Many subjects answered that they evaluated documents by the content. Some subjects mentioned that they did not like documents having just links to other documents without any substantial content. In addition, two subjects said that document length was one of points when they evaluated documents. Some subjects mentioned that they did not pay attention to a document at the Amazon site in the task, because they thought commercial Web pages were most likely not relevant to the given task. In addition, one subject said that he did not like scanned PDF documents. In summary, the interviews demonstrated that some attributes related to document style/genre affected user interest, although in a somewhat idiosyncratic manner.

Figure 4 shows the six most useful and least useful documents across all subjects, selected based on averaged document score. Documents of the left side in Figure 4, the most useful documents, mostly demonstrate substantial content and a meaningful layout, while documents of the right side in Figure 4, the least useful documents, have much less content being mostly links to other documents. This result matches comments from some of the subjects in their interviews about how they evaluated documents.

In addition, a correlation analysis of document attributes and users' value assessments shows that document length (the number of words) was positively related to user interest (Pearson coefficient=0.397 and p=0.015). [Bae, Badi et al. 2005]. Given

these results, it appears that some document attributes related to document style/genre can be used for inferring likely user interest.

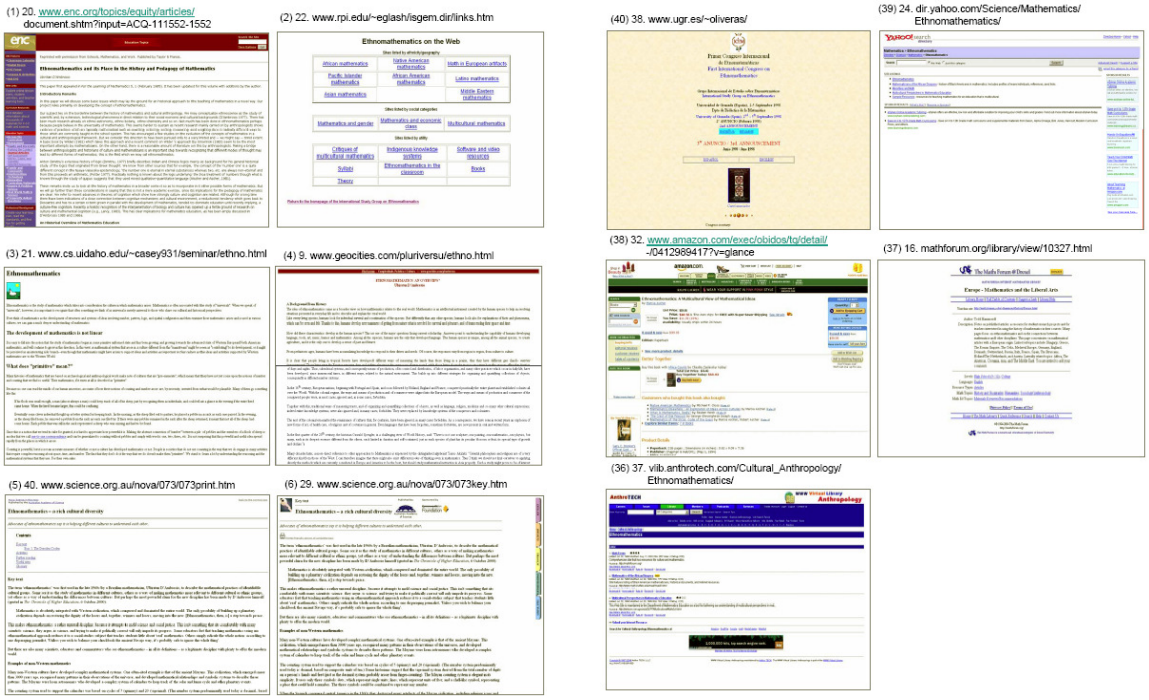


Figure 4. Most useful documents (left) and least useful documents (right)

4.3 User Behavior and User Interest

User activity in the various applications involved in document triage is another potential source of data for inferring document value [Badi, Bae et al. 2006]. Analysis of the relationship between user activities and user interest is based on the questionnaire responses, users' document ratings (explicitly expressed user interest), interviews, and system logs from the study.

As an example of one form of user behavior that indicates document value, Figure 5 illustrates the relationship between the final location of document objects after

triage and the 24 subjects' stated assessment of the documents [Badi, Bae et al. 2006]. Figure 5a indicates the post-triage distribution of the subjects' most useful documents (top and left), while Figure 5b indicates the post-triage distribution of the subjects' least useful documents (more widely scattered, avoiding prominent top and left positions). Users were more likely to place valuable document in the top-left area of the workspace for ease of access while pushing less valuable document further to the fringes of the workspace. Thus, document placement by users could be used by algorithms trying to infer document value.

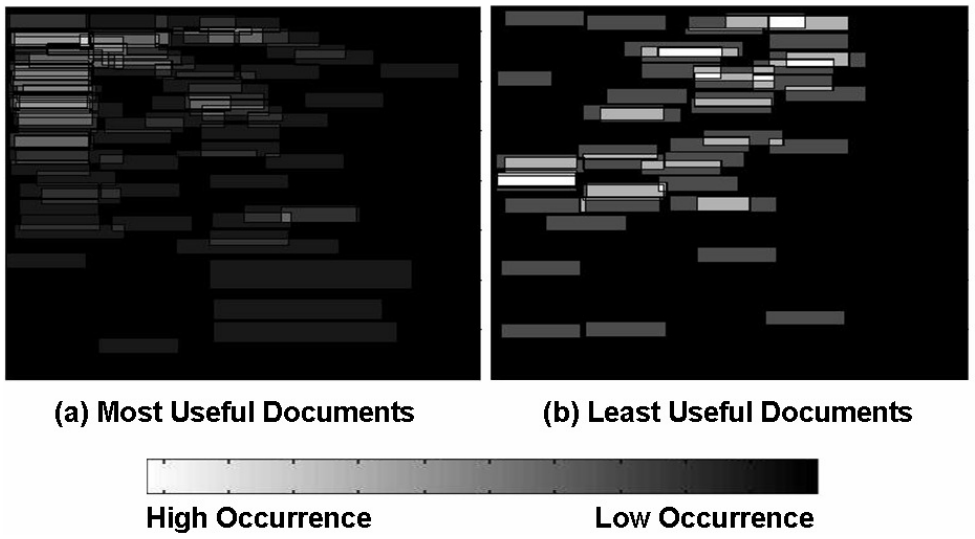


Figure 5. The post-triage locations of document objects judged to be the most and least useful

A variety of forms of user activity in the organizational workspace (VKB) and in the reading interface (IE) was logged and then correlated to document value. The results of the analysis, shown in Table I, show that a large number of user activities were

significantly correlated to user ratings on documents [Badi, Bae et al. 2006]. Scrolling activity in the reading interface and changes to the document objects' visual properties or location in the workspace had the strongest correlations. The only indicator of a document's lack of value was the deletion of its document object from the workspace. While data across all subjects showed many correlated actions, there was no single user activity (or document attribute) that was dominant in the correlation analysis or was consistent across individuals.

Table I. Results from correlation analysis

Parameter	Pearson Coefficient	P value
Number of characters	0.431	0.010
Time spent in a document	0.527	0.001
Number of scrolls	0.630	<0.001
Scroll offset	0.641	<0.001
Number of scrolling direction changes	0.589	<0.001
Total number of scroll groups	0.590	<0.001
Number of document accesses	0.476	0.004
Number of object moves	0.711	<0.001
Number of object resizes	0.622	<0.001
Number of object deletions	-0.495	0.003
Number of background color changes	0.597	<0.001
Number of border color changes	0.628	<0.001
Number of border width changes	0.525	0.001

Based on this initial analysis, we developed four preliminary user models based on document attributes, aggregated user activity, and average document score [Badi, Bae et al. 2006]. The user models were derived based on the overall activity of the subject

population and average document value; they were not designed to match an individual user's working style.

Three of the models were mathematically derived to best account for the document value assessments [Badi, Bae et al. 2006]. All included document attributes in their prediction but included different activity data. In particular, the three models were used to compare the value of activity data coming from the reading application, the organizing application, and the combination of the reading and organizing applications. Data used came from the system logs and was composed of three groups: (1) document attributes including document length (e.g. number of pages, number of characters and number of words), number of links or images in a document, and file size; (2) document reading activity including time spent on a document, number of mouse clicks, number of text selections, characteristics of the scrolling behavior, and number of document visits; (3) document organizing activity including creating collections (categories) and changing spatial or visual features of collections or document objects.

Evaluation of the models showed that R^2 and the adjusted R^2 of the combined-activity model (0.708 and 0.669) were significantly better than the one of the reading-activity model (0.477 and 0.444), and marginally better than the one of the organizing-activity model (0.636 and 0.613). R^2 implies how much of the variability in the outcome of the models is accounted for by the predictors, while the adjusted R^2 shows how the model generalizes. The model with user activity data from both the reading and organizing applications better modeled users' document value assessments, as it must given it has all the information available to the other models combined. This result does

show that user activity in the organizing application seems to be a better predictor than activity in the reading application.

While the three user models compared above were based on statistical analysis, the qualitative model was built from the data used in the combined-activity model and the data acquired from the interviews with subjects and from analyzing videotapes of user activity. Since user and document collection idiosyncrasies could be skewing the mathematically-derived models, the qualitative model was designed to generalize to different users and a different document collection.

Table II. Residue comparison of models

Model	Average Residue	Standard Deviation
Reading Activity Model	0.258	0.192
Organizing Activity Model	0.216	0.146
Combined Activity Model	0.175	0.138
Qualitative Model	0.197	0.134

A second user study was run to assess the user models derived from the data in the prior study [Badi, Bae et al. 2006]. This study included 16 subjects performing the same task with the same collection of documents as in the prior study. One difference was that these subjects were asked to assess each document's value individually, resulting in more accurate data. User assessments were averaged and mapped to a 2 point scale, where 0 represents no value and 2 represents high value. To investigate the accuracy of each model's prediction of user interest, we calculated the residue for each document. The residue is the absolute value of the difference between the explicit user

rating and a model's predicted rating. A perfectly predictive model would have an average residue of zero. The results are averaged across all documents for each model. Table II shows the overall prediction accuracy for each model.

Both models relying on a combination of reading and organizing activity have substantially lower residue values than either of the models based solely on either reading or organizing activity while the model built on organizing activity outperformed the model built on reading activity. An ANOVA analysis of the data indicates a statistically significant ($p = .025$) difference between the combined model and the reading model. A more detailed analysis of the models is available in [Badi, Bae et al. 2006].

The analysis shows that considering reading and organizing separately limits the potential prediction capability of user models: merging user behavior from different applications can improve the capability of user models. The results presented take aggregate user activity and average document value assessments. To be practical in real document triage tasks, the models need to predict an individual's interest during a task – meaning that there would be much less activity data on which to build a model.

4.4 Patterns of Reading and Organizing

Our analysis of users' work practices during triage divided their activity into fifths and then compared reading and organizing activity at different points during the document triage task [Bae, Marshall et al. 2006]. Since the subjects were given no firm constraint as to how much time they could spend on the document triage task, the

subjects' total time varied. The total experiment time has been normalized and partitioned into fifths: user activity was examined during the five segments.

User events collected in the system logs showed that there were significant differences in when and how much different subjects read during the course of the task. Instead of finding one or two canonical patterns of reading and organizing activity, the analysis showed variations on a few common patterns, with a few outlying individuals exhibiting additional patterns of activity.

4.4.1 Patterns of Reading-related Activity

The videos and user event logs identified some rough patterns of user activity that transcend the substantial number of exceptional cases. The following is a summary of these patterns according to our normalized breakdown of the document triage period.

Time periods 1 and 2 (0-40%): During this initial 40% of the triage period, subjects tended to perform any relatively deep reading, to spend a greater proportion of their time reading (especially the most useful documents), and to visit a greater number of the most useful documents. The subjects' judgment of document utility significantly influenced the time they spent reading them, the number of visits, and the reading time per visit. Subjects may have the most interest in understanding the topic or the collection's scope during these early phases.

Time period 3 (41-60%): Although there is no significant change in the time subjects spend reading and the number of times they visit documents during the third time period, subjects are spending less time reading *per visit*. This trend toward briefer time with individual documents indicates a change in how subjects are reading. Thus, the

middle time period might be characterized as a shift from the early "reading-in" style to the scanning and reminding types of reading that are associated with the later interpretive or organizational periods.

Time period 4 (61-80%): Although reading time remained stable through this period, the number of visits and the number of scrolls increased. This may imply that the form of reading has changed to some extent. Reading is probably no longer directed toward seeking new information or developing a categorization strategy, but rather toward revisiting and re-reading to confirm and refine the resulting structure or to populate portions of the structure with other documents.

Time period 5 (81-100%): Evidence from the event logs suggests that there was significantly less reading activity during the final period of the triage task: less time spent reading, fewer document visits, and fewer user events in general. Subjects focused on organizing during this period. Any reading manifested itself in the form of short visits to documents (re-reading or scanning) to complete the task. Videos reveal that some subjects quickly checked documents to make sure they were categorized correctly or to confirm their ideas.

4.4.2 Patterns of Organizing-related Activity

Subjects created categories throughout the triage process, organizing the information in part as they encountered it. The earlier phases of document triage – periods when the subjects tended to read more material more deeply and certainly to encounter more new information and notice more new topics – involved significantly more category creation. Other types of organizing-related activities, such as moving and

resizing objects or changing their background colors – interactions usually associated with expressing more implicit characteristics of the documents – show different temporal patterns. This illustrates how reading and organizing patterns may be intertwined: category creation may be related to the relatively focused reading that takes place in the early stage of triage, and organization refinement may be related to the relatively faster form of reading that takes place later in triage.

4.4.3 Overall Patterns

Overall, the analysis of document attributes and user activity provides insight into the design of an integrated set of tools to support document triage [Bae, Marshall et al. 2006]. It documents the value of basing models of user interest on user activity in both the organizing and reading application while making clear the difficulty of inferences based on a relatively small quantity of user interaction for such an idiosyncratic activity. Furthermore, it uncovers a triage process that moves from focused reading and category creation to faster forms of reading and category refinement. Based on this analysis, the next section presents an architecture for the document triage environment.

5 SUPPORTING DOCUMENT TRIAGE

The studies described in Section 4 explored ways to support document triage based on estimated user interest [Badi, Bae et al. 2006]. We defined four steps for proactive support of document triage: recognizing user interest and document value, representing user interest, recognizing documents of interest, and visualizing interest information. This dissertation has implemented the four steps with a focus on the visualizing interest information step.

5.1 Recognizing User Interest and Document Value

Systems can infer user interest and document value in an explicit or implicit way. Systems can ask users for explicit ratings on given documents, which is easy to implement and fairly precise. However, this approach can distract normal patterns of triage tasks and increases overall cognitive load for users. As a result, applications that require such ratings or include a modal dialog that requires the user to close are often not used and applications that make such ratings optional get very few ratings. Alternatively, systems can watch user activity for indicators of interest such as time spent reading a document, scrolling, and mouse events, and user annotations.

Section 4 showed that some of the user events and document attributes were correlated to user interest [Bae, Badi et al. 2005]. Our first attempt to infer user interest resulted in four user models for estimating user interest based on user activity and document attributes. As described, the models were effective in estimating averaged user interest and document value when given a whole community's actions with the documents, but were less effective at recognizing interest and document value for

individuals engaged in a single triage activity. Different people have different ways of working through documents during triage and the models were identifying alternate methods of expression. By itself, this is good but the models become conservative in estimating interest since the models include plenty of contradictory evidence (e.g. one person leaves unwanted objects where they start while another moves them into a trash pile or collection). This means that there is not enough user activity data available to compare between documents (to estimate interest versus disinterest) until the task is nearing completion. Thus, while the models could be used in a collaborative setting to infer group interest and could be used for a long-term triage task (e.g. an analyst whose job is to report on particular topics), they seemed too indirect for an individual with triaging material for a new domain/task.

A second problem is that the location of a set of documents of interest requires the system to cluster the documents or their contents to identify interests since treating the set as a single interest results in a generic Second, once documents of interest are inferred, the question of how to represent those interests and intuitively visualize the existence of information related to those interests.

To make inferences more rapidly, we decided to make use of the fact that annotations reflect readers' interest on documents relatively unambiguously [Price, Golovchinsky et al. 1998]. In a reading interface, people highlight or comment on text they find central to their interpretation of the text. This has been the basis for suggestion techniques found in some electronic book reader applications. For example, XLibris estimated user interest based on user annotations created while users were reading. By

estimating user interest, XLibris suggested related documents to users by providing related links and further reading list [Price, Golovchinsky et al. 1998; Price, Schilit et al. 1998; Schilit, Golovchinsky et al. 1998; Schilit, Price et al. 1998; Shipman, Price et al. 2003]. The identification of user interest in the remainder of this dissertation is based on annotations in the reading application and annotations in the organizing application. In the future, this can be switched to other algorithms.

5.2 Representing User Interest

Document triage is composed of the intertwined tasks of searching, reading, and organizing. Thus, multiple software applications are usually involved for each of the document triage tasks. User activities primarily rely on the applications, and each application is likely to have its own techniques for recognizing user interest. The results of the study presented in Section 4 show that combining the interest information collected from the multiple applications involved in triage can improve the prediction capability of user models.

User interest can be represented as a set of user actions and attributes that are specific to application types. For sharing the interest information across multiple applications, a common user interest representation is crucial. We previously proposed a representation for expressing and sharing interest related information from various types of software applications that records user actions with a term vector of the content being effected by the action [Badi, Bae et al. 2006]. The approach in the remainder of this dissertation refines and extends the representation. In particular, we currently use user annotations in organizing (changing color) and in reading (highlighting) among the

collected interest related information [Bae, Hsieh et al. 2008]. The annotations that share certain attributes, such as color and application, are grouped together and represented as an interest. Thus, all document objects colored red in the organizing application are grouped together in one interest. Similarly, document objects colored yellow are grouped together as an interest while yellow highlights in the reading interface are grouped together in a different interest.

5.3 Recognizing Documents of Interest

Once the system has representations of user interests, it can be used for recognizing related documents in a document/content organizing application or related paragraphs within a Web page [Badi, Bae et al. 2006; Bae, Hsieh et al. 2008]. Different algorithms can be also used for recognizing related documents or paragraphs.

For the system presented in this dissertation, whenever a document object is created in the workspace of VKB, its full text content of the corresponding document is also collected and a term vector is created [Bae, Hsieh et al. 2008]. Similarly, when a document is opened in the reading interface (Web browser), the text of the document's paragraphs is collected, and a term vector is created for each paragraph. As users make annotations while organizing or reading, the term vectors representing user interests are compared with the term vectors for the documents in the workspace and the paragraphs in the reading interface using the standard cosine-similarity metric. The estimated similarity is shared across all software applications so that each application can visualize and suggest the related documents or paragraphs.

5.4 Visualizing Interest Information

Once the interests have been inferred and represented and related documents or document segments have been identified, the individual applications in the triage environment provide visual cues indicating the potential relationship [Badi, Bae et al. 2006; Bae, Hsieh et al. 2008].

There could be many visualization techniques for suggesting inferred user interest by changing visual attributes of document objects in VKB – the documents could be moved to be nearer to related documents [Buchanan, Blandford et al. 2004], links could be drawn in the workspace [Czerwinski, Dumais et al. 1999], or, as is the case in the current design, visual properties of the objects could be modified [Bae, Hsieh et al. 2008]. Moving document objects is likely to create disorientation and exacerbate user issues surrounding keeping track of their progress. Drawing links requires that there are not too many and there is blank space in the workspace to present them. Thus, modifying visual properties was chosen even though prior studies showed that user expression was inhibited by the system's automatic assignment of visual attributes [Shipman, Hsieh et al. 2004].

5.4.1 Layered Visualization

The conflict between the user-authored and system-generated visualizations primarily came from sharing the same space in VKB. For avoiding the conflict the visualization proposed in this dissertation employs two layers for each document object, the user layer and the system layer. The user layer is for end-user visual expression occurs, where users change visual attributes. The system layer is for system-generated

visualization, where interest (or other system generated) information can be visualized without overwriting the user-authored visual expression. The system layer and the user layer are partially overlapped and the z-order of the two layers can be switched. By dividing VKB's document object into two independent layers, this visualization removes the conflict between the two.

5.4.2 Visualization of Search Result for Quick Evaluation

In our earlier studies of how people evaluate documents during document triage [Shipman, Price et al. 2003; Bae, Badi et al. 2005; Badi, Bae et al. 2006; Bae, Marshall et al. 2006], people used a variety of cues to evaluate documents before and during reading documents. For instance, metadata such as page tile, page URL, and page snippet helped determining whether or not to read documents in search results of VKB workspace. In the study presented in Section 4, participants evaluated 19% of the given documents based on metadata alone [Bae, Badi et al. 2005].

In addition to metadata, the visual layout of a document can provide cues about document genre and style that are helpful as people evaluate documents. Wynblatt et al. designed a visualization of search results, the Web Page Caricature, which enables people to recognize document genre and styles. Woodruff et al. compared the performance of three types of Web search results, metadata-only, thumbnail-only, and thumbnail enhanced with metadata, and found that categories of search queries and tasks affected the performance of each type of Web search result.

This is congruent with comments we received from participants in the study presented in Section 4. In the interview after completion of the task, subjects were asked

what kind of extra information in the document object would be useful for a better understanding of the document content before reading it. Only one out of fifteen subjects answered that the metadata, provided in the study, was sufficient. Six out of fifteen subjects answered that thumbnail image of a document would be useful [Bae, Badi et al. 2005].

Supporting quick evaluation of documents in search results is crucial in document triage. However, the metadata-oriented visualizations in the previous versions of VKB were limited with regards to indicating document genre and style. The document object visualization of this dissertation employs thumbnail images of documents enhanced with metadata in search results to better support the quick evaluation of documents.

5.4.3 Visual Cues for Helping Remember Prior Activity

Quick switching between searching, reading, and organizing during document triage was in common among all subjects in the prior study [Bae, Badi et al. 2005]. Quick task switching can interrupt users' cognitive flow and hinder the smooth interaction between searching, reading, and organizing. In the interview after the task in the study, 13 out of 17 participants answered that they looked at metadata for identifying the previously read documents. However, it was observed that some participants still had difficulty remembering or recognizing their prior activity.

Prior research has shown that document thumbnails help people remember previously read documents in searching [Czerwinski, van Dantzich et al. 1999; Woodruff, Faulring et al. 2001; Dziadosz and Chandrasekar 2002]. Consequently, the

prior studies showed that thumbnail of a document plays a role not only for visualizing document genre and style but also for helping people remember or identifying previously read documents.

5.5 Architecture

The triage of Web documents requires applications for three subtasks. These can be thought of as three applications: (1) an overview application for dealing with Web search results; (2) an application for reading or skimming Web documents; and (3) an application for organizing the documents.

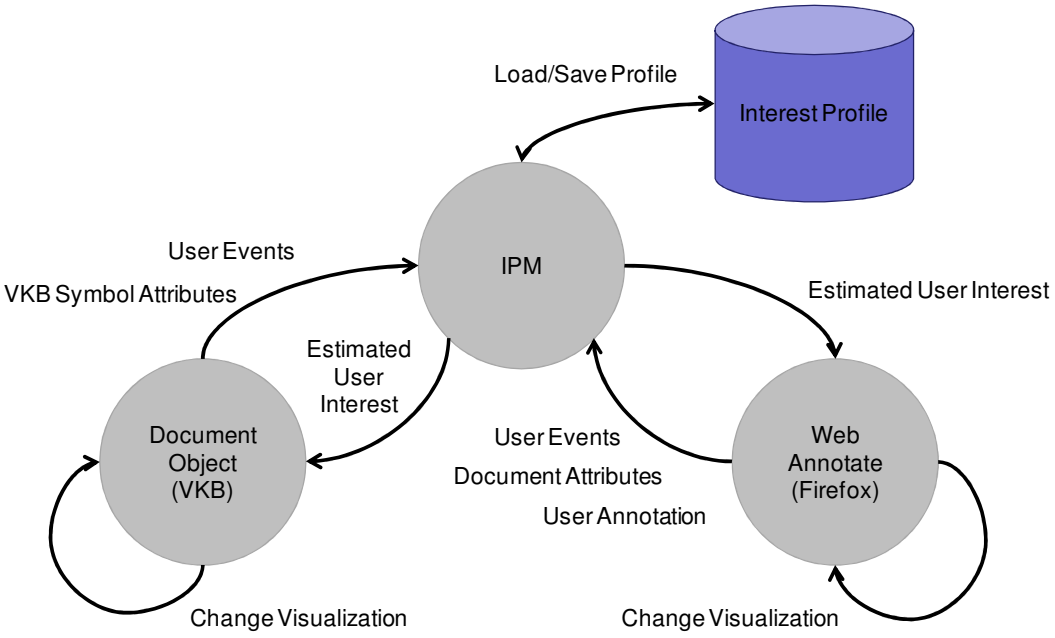


Figure 6. Overall architecture

Figure 6 shows the overall architecture consisting of three applications: VKB, the WebAnnotate (plug-in for Firefox), and the Interest Profile Manager (IPM). VKB provides both the overview of Web search results and workspaces for organizing

documents. The Web browser (Firefox) augmented by WebAnnotate is used for reading Web documents. WebAnnotate allows users to make persistent annotations on Web documents and suggests potentially useful paragraphs within a Web document based on user annotations in VKB and Web documents. The Interest Profile Manager (IPM) is a personal profiling application, which collects interest related information across all applications, infers user interest using the collected interest related information, and shares the inferred user interest with all applications.

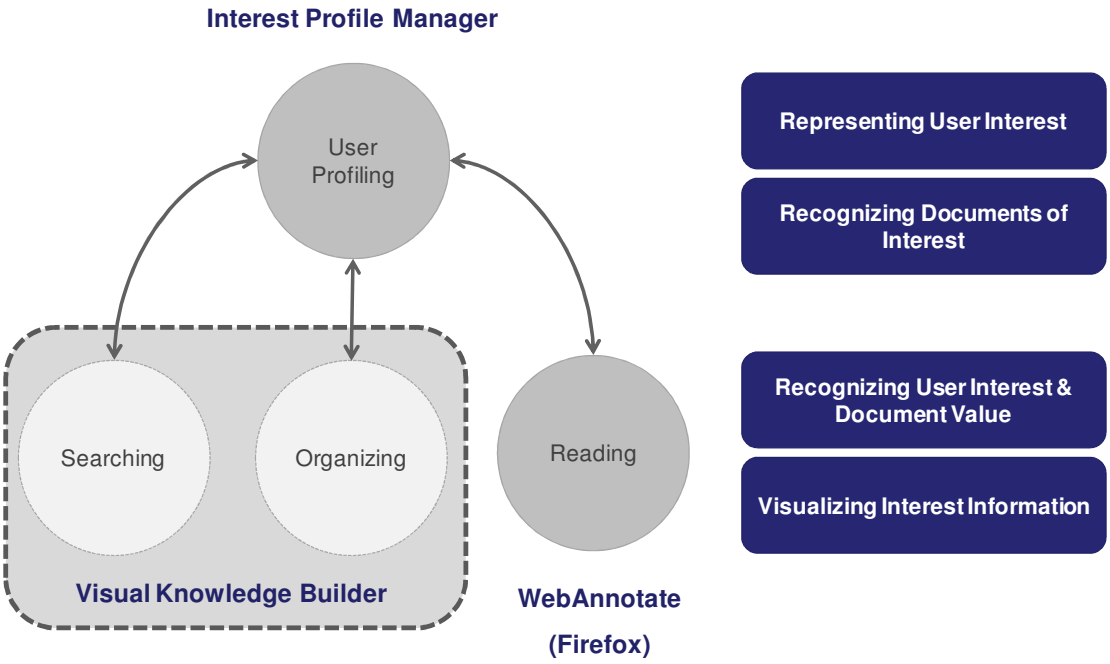


Figure 7. Applications and four steps for proactive support of document triage

Figure 7 also shows relationships between the applications and the four steps for supporting document triage. The first step, recognizing user interest and document value, and the forth step, visualizing interest information primarily occur in VKB and

WebAnnotate. The second step, representing user interest, and the third step, recognizing documents of interest primarily occur in the IPM. The IPM can interact with any application that supports the defined message format.

5.6 Common Communication Interface

VKB and WebAnnotate are the only applications that currently interact with the IPM. However, the IPM has generic communication interface by which any application can cooperate with the IPM. Figure 8 shows a XML format message sent from WebAnnotate to the IPM.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <Request operation="29" port="51294">
  - <InterestProfile>
    - <Object>
        <Id>http://en.wikipedia.org/wiki/Antimatter_rocket</Id>
      - <AttributeVector type="3" instanceId="1208316379216">
        - <Attribute>
            <Name>link</Name>
            <Action>0</Action>
            <Value>99</Value>
            <TimeStamp>1208316379216</TimeStamp>
          </Attribute>
        - <Attribute>
            <Name>image</Name>
            <Action>0</Action>
            <Value>3</Value>
            <TimeStamp>1208316379216</TimeStamp>
          </Attribute>
        </AttributeVector>
      </Object>
    </InterestProfile>
</Request>
```

Figure 8. An example of message from WebAnnotate to IPM

In Figure 8, WebAnnotate informs the IPM of two document attributes, the number of links and images in the document. The “Request operation” value “29” indicates that

this message is about document attributes. The “Object” portion identifies the document as being found at “en.wikipedia.org/wiki/Antimatter_rocket” and the two document attributes being provided, the number of links (99) and the number of images (3). This message format is also used for saving the interest profile. Attributes are used to represent user activity with documents or document segments as well. In this case, the content of the attribute element depends on the features in the application and their use. Any application can send and receive interest information with the IPM as long as the application uses the message format as a communication interface with the IPM.

Section 6 describes the additions to VKB to support interest modeling and presentation while Section 7 describes WebAnnotate’s design and capabilities. The Interest Profile Manager is presented in more detail in Section 8.

6 VISUALIZATION OF DOCUMENT OBJECT

The Visual Knowledge Builder (VKB) is a spatial hypertext tool providing an integrated environment for searching and organizing information from the Internet during document triage [Shipman, Hsieh et al. 2001]. There are currently three types of visual symbols in VKB: the classic object, the collection, and the document object. A classic object contains basic textual information, and the collection is a container object for symbols in VKB including other collections, providing a hierarchical workspace. Users can organize these symbols by arranging them, putting them into collections, and changing visual attributes, such as color or border width. The document object refers to a document such as a Web page which can be opened by double-clicking or through the popup menu of the document object as shown in Figure 9.

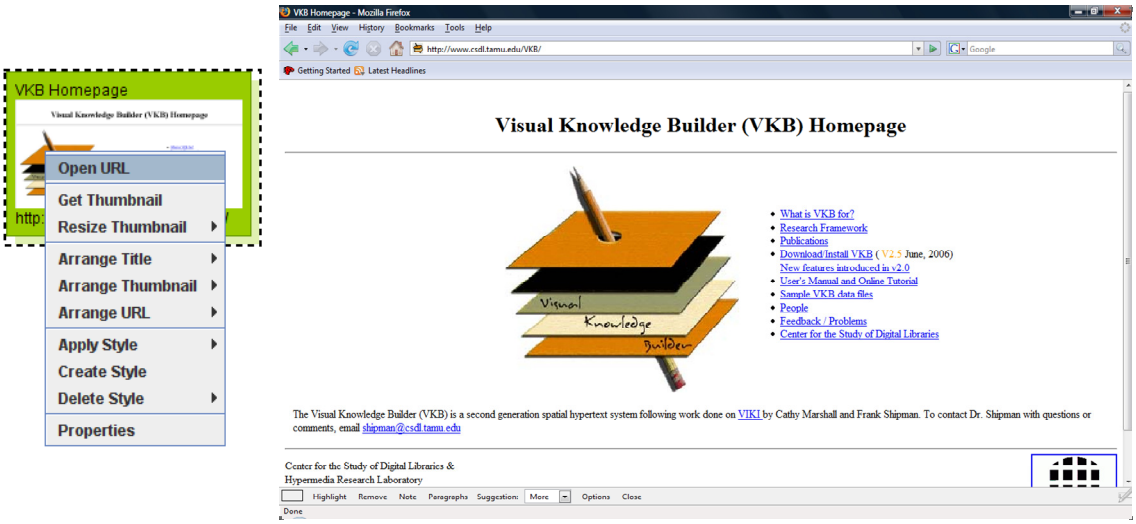


Figure 9. Opening Web page from the document object

The document object is composed of two independent layers, the user layer and the system layer as shown in Figure 10 [Bae, Hsieh et al. 2008]. The two layers are partially overlapped, and users can change the z-order of the layers by the popup menu. Users can change visual attributes of the user layer without overwriting the system-generated visualization in the system layer. In addition, the system can visualize the estimated user interest without invalidating user-authored visual expression in the user layer. The system layer and the user layer can be switched within the document object by users.

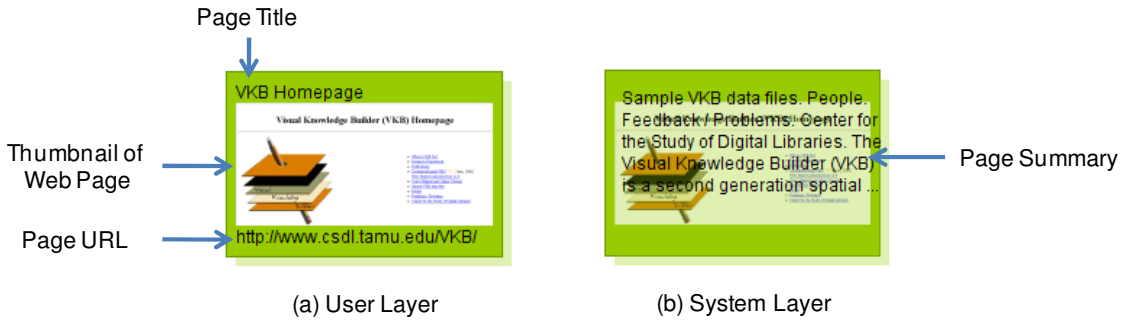


Figure 10. Two layers of the document object

6.1 User Layer

The user layer is a main component of the document object, where users express their interpretation of the document's relation to their current activity by changing visual attributes of the document object [Bae, Hsieh et al. 2008]. The default components within the user layer (Figure 10.a) were determined by observation and feedback from the earlier studies. However, prior studies showed that no one visualization meets the

individual preferences or characteristics desired for different types of tasks. VKB deals with this issue by providing users the ability to change or create styles for the user layer.

6.1.1 Formation of User Layer

The prior document triage showed that subjects utilized metadata to evaluate documents before reading the document and also to identify previously read documents[Bae, Marshall et al. 2006]. More specifically, most subjects used the page title and URL, but the value of the page snippet depended on individual styles. Also, subjects tended to look at the page snippet not for identifying documents after reading but for evaluating documents before reading. This implies that page title and URL was rather commonly useful for searching and organizing at the same time, while the page snippet was more useful for the initial assessment during searching. The document object is the visual object for both searching and organizing in document triage. In terms of information density of visualization, the page title and URL is included in the default user layer so that they can be seen in searching and organizing. In contrast, the page snippet is presented in the system layer so that users can see choose to view it when desired. Users can switch layers in the document object by popup menu.

Prior studies showed the usability and value of providing thumbnails of Web pages for evaluating documents before reading them and identifying documents after reading them [Czerwinski, Dumais et al. 1999; Woodruff, Faulring et al. 2001; Dziadosz and Chandrasekar 2002; Bae, Badi et al. 2005; Bae, Marshall et al. 2006]. This implies that thumbnail of Web page can be useful in both searching and organizing. For that reason, the thumbnail of the Web document is included in the default user layer. Figure

10.a shows a basic style of the user layer, which is composed of three components, page title, page URL, and thumbnail of Web page.

However, Figure 10.a is just one of the three system-provided styles as shown in Figure 11. Users can select one of the three styles according to their preferences or tasks or create their own styles.

Figure 11. Three system-provided styles of the user layer

6.1.2 Human-authored Visualization in User Layer

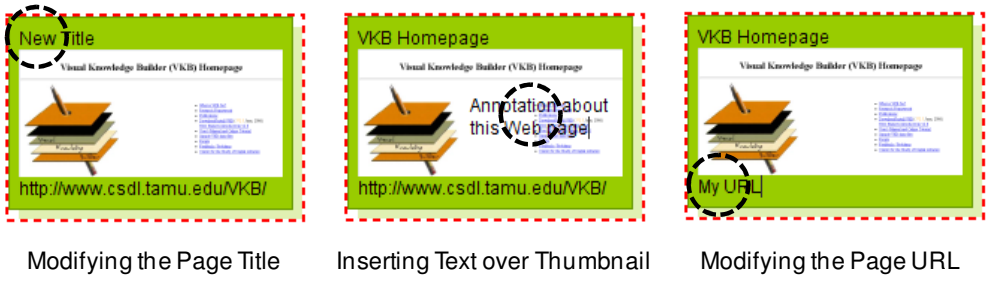


Figure 12. Textual annotation on the document object

The user layer provides a space where users can freely express their opinions without any conflict with system-generated visualization and loss of any system-provided information. Users can change the visual attributes such as background color, border color, border width, font name, font color, font size, and font properties (italic or

bold). In addition, users can resize the document object or move it over workspaces of VKB.

The prior study showed that some subjects changed text information provided by VKB within a symbol for making it more meaningful (more descriptive or a better differentiator) for their particular activity [Bae, Marshall et al. 2006]. Some subjects in the prior study created textual comments, representing their understanding of the documents, in their organization [Bae, Marshall et al. 2006]. The user layer allows users to make textual annotation as shown in Figure 12. Users can modify system-provided page title and URL and make textual comment on thumbnail of Web page.

Figure 13. Resizing thumbnail of Web page

User can also resize the thumbnail of the Web document using popup menu as shown in Figure 13. Users can adjust the size of the thumbnail according to their preferences or tasks. There are three options in the popup menu: *Fit Thumbnail Width*, *Fit Thumbnail Height*, and *Set As Default Thumbnail Size*.

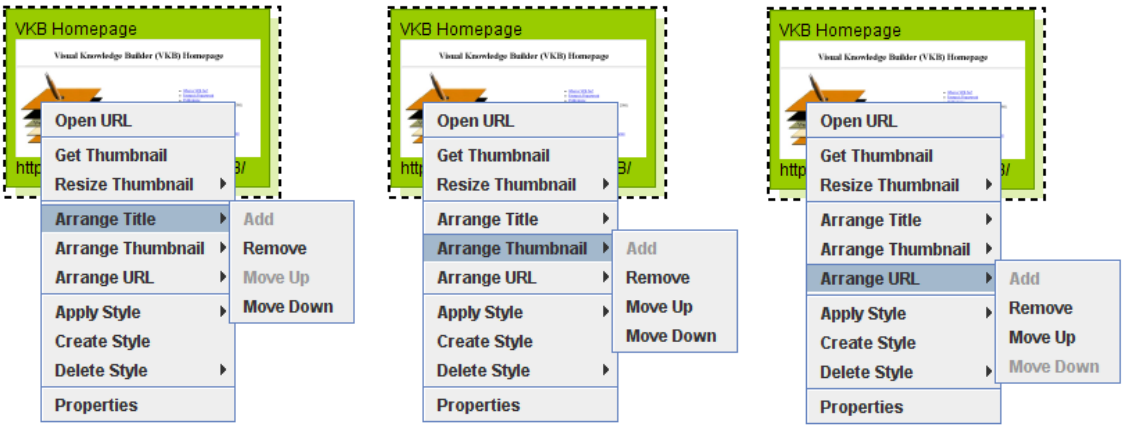


Figure 14. Changing the layout of components within the user layer

In addition, users can change the layout of three components (page title, page URL, and thumbnail image of Web page) within the user layer using popup menu of the document object as shown in Figure 14. Users can add, remove, and move the components within the user layer using popup menu of the document object.

Figure 15. Different layout of components within the user layer

Figure 15 shows various layouts of components with the user layer changed from the system-provided style.

6.1.3 Style of User Layer

The style of the user layer includes some of the visual attributes mentioned in the previous section including background color, border color, border width, font name, font

color, font size, and font properties (italic or bold), thumbnail size, and the layout of components within the user layer. As the prior study showed that visualization of search results affected the performance of given tasks, the style of the user layer can affect the overall performance of document triage. However, it is not likely to design a set of styles that covers all the different individual styles and the various types of tasks one might perform. Therefore, VKB allows users to create their own styles for the user layer and reuse them later.

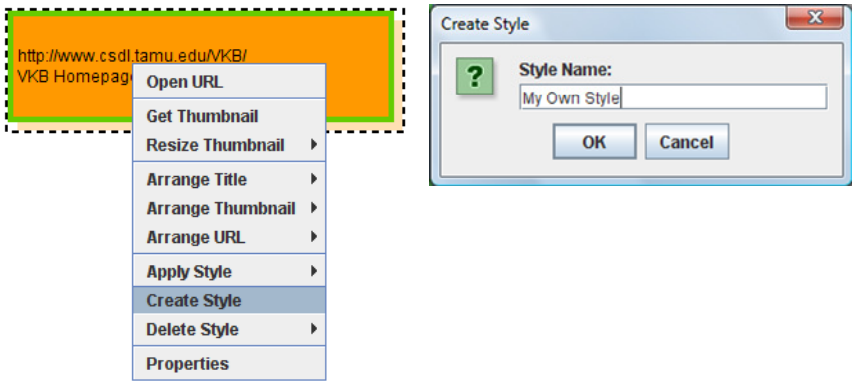


Figure 16. Creating new style of the user layer

Figure 16 shows how to create new style from an existing document object. The style is named as “My Own Style” in Figure 16 (right). The created style can be reused for changing visual attributes of other document objects as shown in Figure 17.

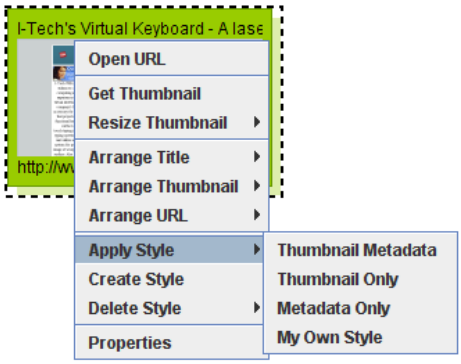


Figure 17. Changing visual attributes of the user layer by the created style

6.2 System Layer

The system layer is the space where the system can visualize user interest without affecting user-authored visual expression in the user layer. Users cannot directly change the visual attributes of the system layer. The system-generated visualization is for helping users locate interesting documents. The use of a mostly hidden and translucent system layer makes the system-generated visualization less obtrusive, which is particularly valuable when its estimation of user interests and preferred styles is not acceptable to users. By being out of view, the system layer is expected to be less annoying to users, giving the system more time to learn user interests and preferences before users overwrite the visualization or turn it off completely.

There are two ways that the system layer expresses user interest: background color and transparency. The visualization in the system layer is independent from user interest estimation algorithms, which implies that various types of user interest estimation modules can be used in the IPM. With the currently employed estimation algorithm, background color indicates which documents are similar to other documents,

and transparency indicates the similarity level. However, the meaning of background color and transparency in the system layer can be changed according to the employed estimation algorithm.

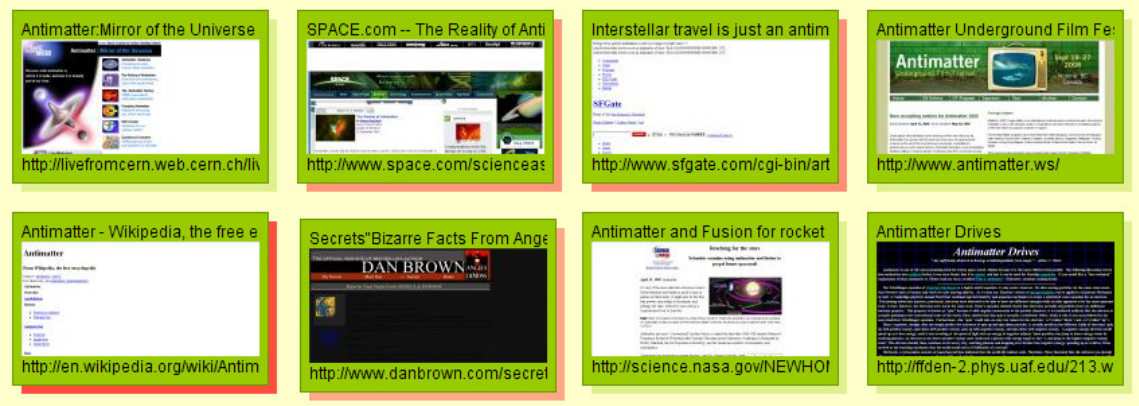


Figure 18. System-generated visualization in the system layer

As users make annotations in the triage applications, user interest (similarity between user annotation and documents) is estimated. The system layer changes its background color and transparency in real time based on the estimation. Figure 18 shows visualization of potentially related documents in the system layer. This visualization could be generated based on user annotation while either organizing or reading [Bae, Hsieh et al. 2008]. The system suggests four documents by red translucent background color of the system layer. Transparency of the system layer indicates the confidence level of the suggestion: the more transparent background of the system layer is the less confident the suggestion is. In this case the system is most confident about the document in the lower left while having a lower degree of confidence about the other three suggestions.

6.3 Creation of Document Objects

Within the context of document triage, document objects are usually created as a result of Web search although there are other ways to add documents to the workspace [Bae, Hsieh et al. 2008]. VKB currently includes an internal interface to the Yahoo search engine. Using the Web search dialog (Figure 19), keywords are sent from users to Yahoo search engine. Once search results have received from Yahoo search engine, VKB visualizes the search results using document objects as shown in Figure 20.

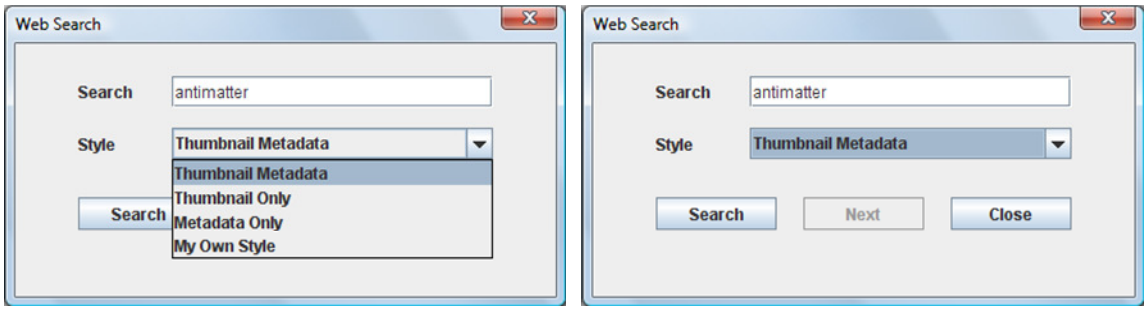


Figure 19. Web search dialog

When users perform Web search, they can choose the document object style for the results using the Web search dialog as shown in Figure 19. Users can choose system-provided styles such as “Thumbnail Metadata”, “Thumbnail Only”, or “Metadata Only”. Or users can choose their own style, “My Own Style” in Figure 19.

The two figures in Figure 20 are the results of the same query with different styles for the results. In Figure 20 (upper), users see both the metadata and the thumbnail image of Web page at the same time. However, users can use another style such as

"Metadata Only", as shown Figure 20 (lower) if they find that the style is more appropriate for their activity.

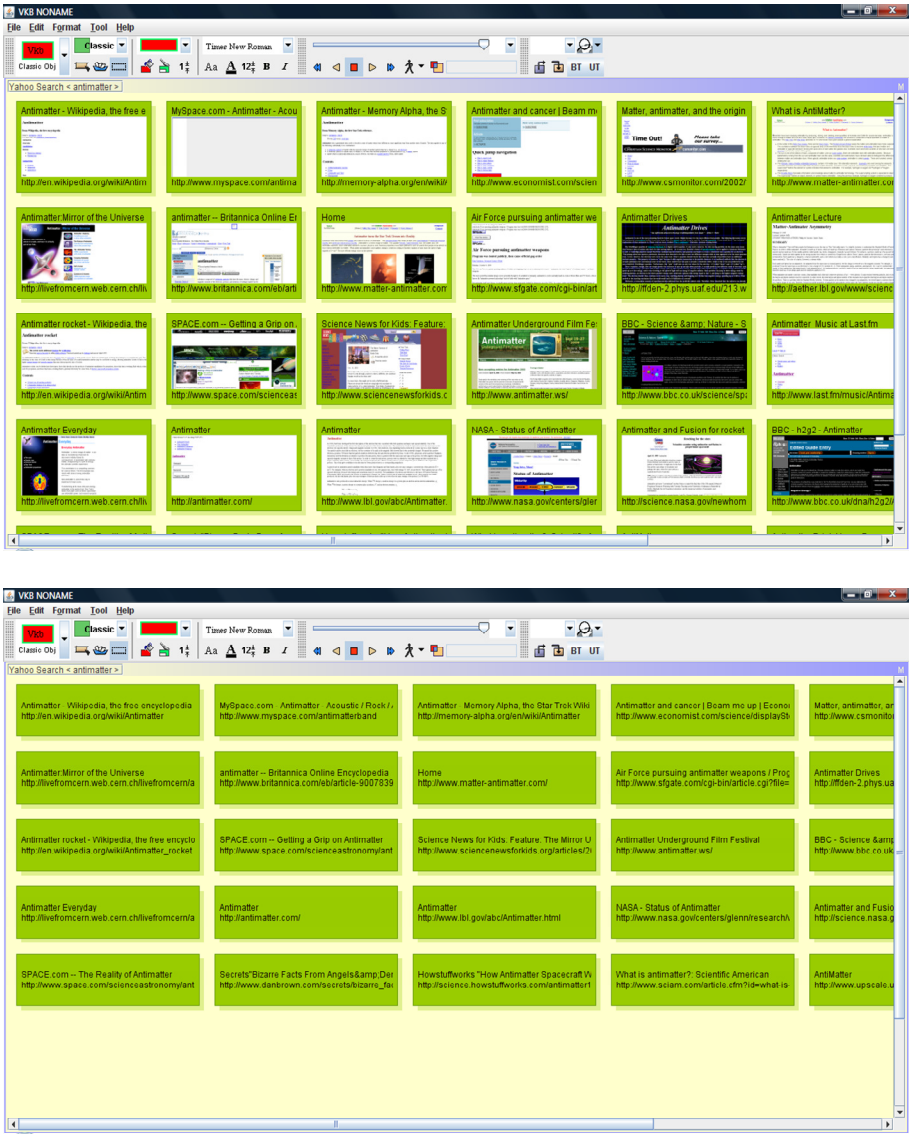


Figure 20. Examples of Web search results with different styles: *Thumbnail Metadata* (upper) and *Metadata Only* (lower)

In addition to Web search, a document object can be added to the workspace by a drag-and-drop operation from the Web browser. In addition, blank document objects can be created manually with the user then specifying the document's URL.

6.4 Visualization of User Interest

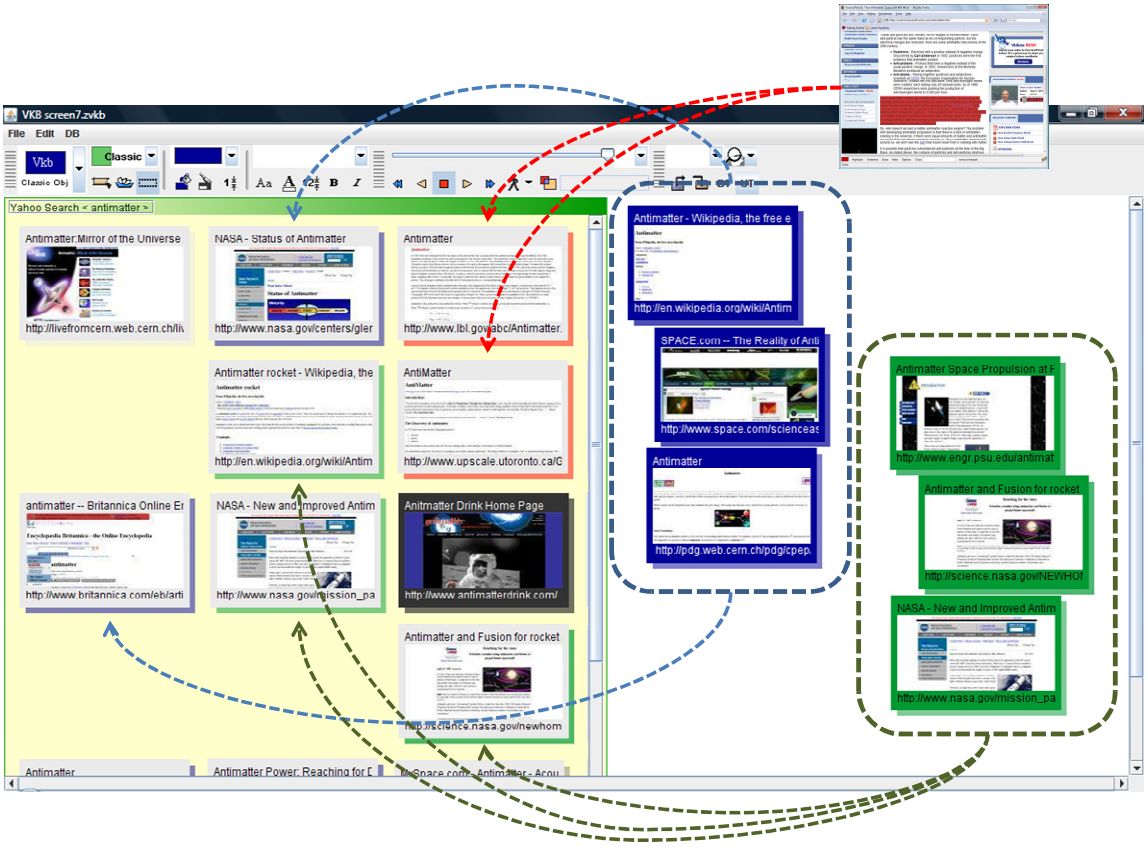


Figure 21. Visualization of user interest in VKB

The IPM collects various types of interest-related information from VKB and WebAnnotate as users search, read, or organize documents. However, the current user interest estimation module employed in the IPM utilizes the information about user annotation in reading or organizing. Whenever users changes the background color in

VKB or highlights a passage in WebAnnotate, the user interest estimation module calculates the similarity between the effected document or document segment and the full text of the documents in VKB [Bae, Hsieh et al. 2008]. Based on the estimated similarity, VKB suggests potentially related documents as shown in Figure 21.

In Figure 21, a user changed background color of three document objects in lower right of workspace to green and also changed background color of other three document objects in upper right of workspace to blue. Based on these actions, VKB changed the system layer's background color of three document objects in left side of workspace to green with different transparency and also changed the system layer's background color of two document objects in left side of workspace to blue. There are two document objects of which system layer's background color is red, which is generated as the result of user annotation in WebAnnotate as shown in the thumbnail image of the Web page in the upper right of Figure 21. In addition, the user changed the background color of a document object around center of Figure 21 to black, but this did not generate any further visualization. The document object in upper left of Figure 21 is not considered related to any of the user's current interests.

The system-layer visualizations are not final, but are changed as users make more annotations while organizing or reading. As more user annotations are collected in the IPM, it is expected that more reliable estimations of user interest become available.

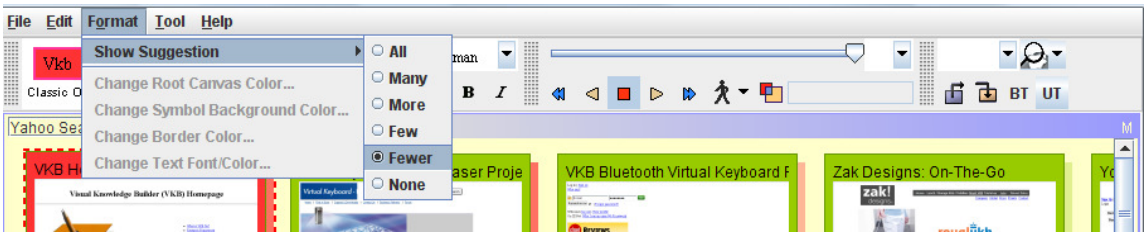


Figure 22. Adjusting the number of suggestions in VKB

The number of suggestions generated by VKB can be an issue. It would not be feasible to find the right number that is appropriate for every situation or individual preference. VKB provides a way for users to adjust the number of suggestions according to individual preferences or tasks in six levels (All, Many, More, Few, Fewer, and None) as shown in Figure 22.

6.5 Architecture

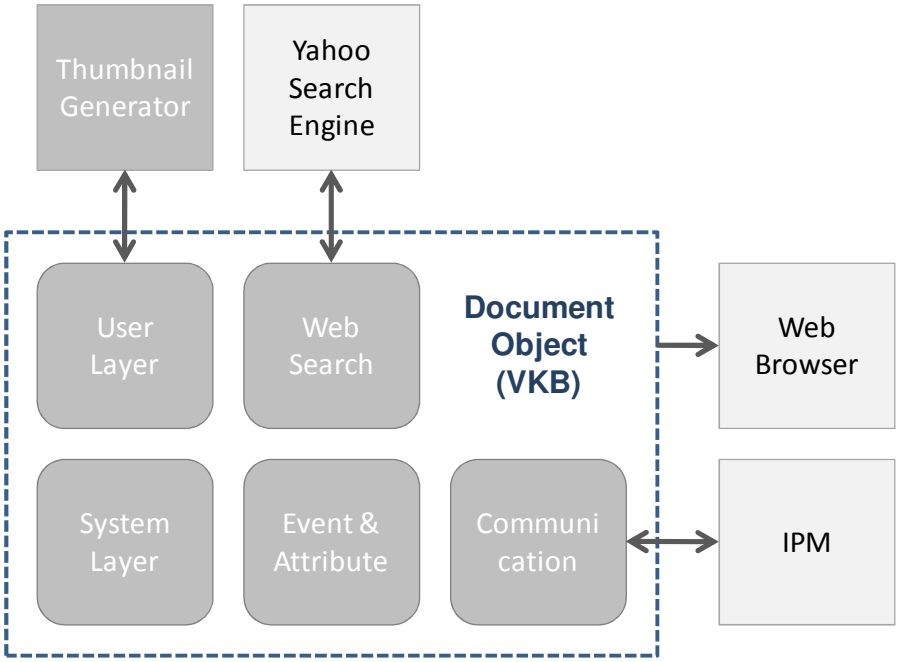


Figure 23. Architecture of document object

The document object is the new symbol in VKB that is designed to support document triage. As shown Figure 23, the document object functionality is divided into six software components: User Layer, System Layer, Web Search, Event & Attribute, Communication, and Thumbnail Generator. The Thumbnail Generator is not a component in VKB, but an independent application. However, the Thumbnail Generator is developed for the document object and is required by the document object.

The User Layer and the System Layer components are the visualization parts of the document object including supporting modules. These components respond to user operations such as changing visual attributes, moving, and deleting. The System Layer component also responds to the estimated user interest delivered from the IPM through the Event & Attribute component.

The Web Search component sends keywords from users to Yahoo search engine and generates search results using document objects based on the response from Yahoo search engine. The Event & Attribute component collects user events on document objects, and sends them to the IPM. In addition, the Event & Attribute component delivers the estimated user interest provided by the IPM to the System Layer component so that it can change the visual attributes. The Communication Module deals with communicating with the IPM. This architecture is shown in Figure 23.

7 WEBANNOTATE: ANNOTATION ON WEB-BASED DOCUMENT

WebAnnotate is a Mozilla Firefox add-on that allows users to annotate Web documents, communicates with the IPM, and suggests potentially useful information within a Web page [Bae, Hsieh et al. 2008]. There are various types of reading: reading for fun, reading for general knowledge, or reading for specific task [Shipman, Price et al. 2003]. People rarely annotate during fun reading; but annotation is more commonplace when reading for a particular task, such as the sense-making that occurs during triage. With WebAnnotate, users can make annotations, highlighting passages of text and adding notes on top of Web pages while reading them.

As users create, modify, or delete annotations, WebAnnotate sends the annotation information to the IPM. The IPM stores the information so that WebAnnotate can reuse the information when users revisit the Web page, and also estimates user interest based on the annotation information. When users open up a Web page, WebAnnotate requests previously generated annotations and estimated user interest information on the Web page from the IPM. It displays the annotations and the estimated user interest if they are available.

7.1 Annotation on a Web Page

There has been much research on annotation systems and practices, such as explorations of the utility of annotation in reading and the functions of different forms of annotation. WebAnnotate is specifically designed for helping read Web pages in a Web

browser. WebAnnotate allows users to make two simple forms of annotation, highlights and notes, directly on Web pages. This is conceptually similar to the annotations in Microsoft Word or Scrapbook [Gomita 2006] in terms of annotation functions. In terms of collecting annotation information and utilizing it, WebAnnotate is similar to Annozilla [Wilson 2006] and Annotea [Kahan and Koivunen 2001]. However, WebAnnotate tries to facilitate the interaction between reading and organizing activities of a single user over time, while Annozilla and Annotea are more focused on collaborative situation that requires sharing annotations among users.

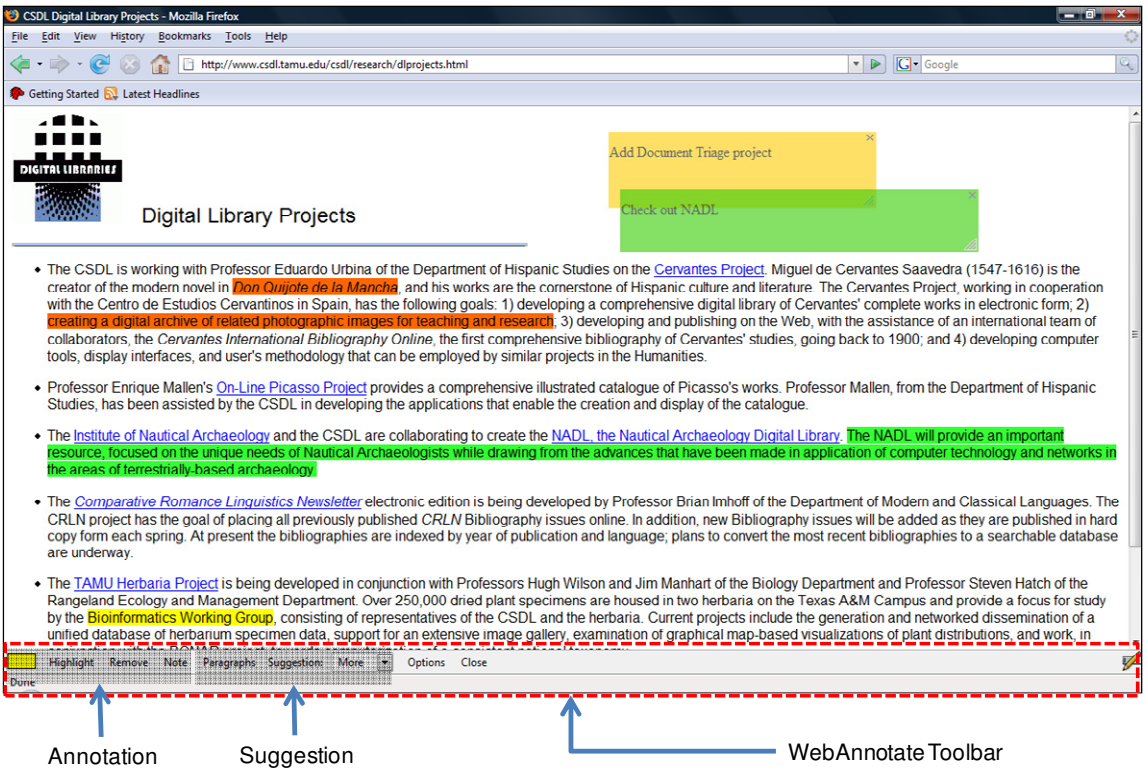


Figure 24. Annotation on a Web page using WebAnnotate toolbar

Figure 24 shows a screenshot of the WebAnnotate toolbar and a variety of annotations on a Web page. Users can use different colors for their highlights and notes and adjust visualization of the estimated user interest in Web pages using the WebAnnotate toolbar.

7.1.1 Highlights

WebAnnotate highlights allows user to emphasize text fragments within a Web page using various colors according to user preference and interpretation. The context menu of Firefox (Figure 25) is an alternative way for creating highlights on Web pages. To create a highlight, the user selects a color using the color selector in the WebAnnotate toolbar (Figure 24) and selects the text to highlight in the Web page before pressing the highlight button in the WebAnnotate toolbar (Figure 24) or the highlight menu item in the context menu (Figure 25).

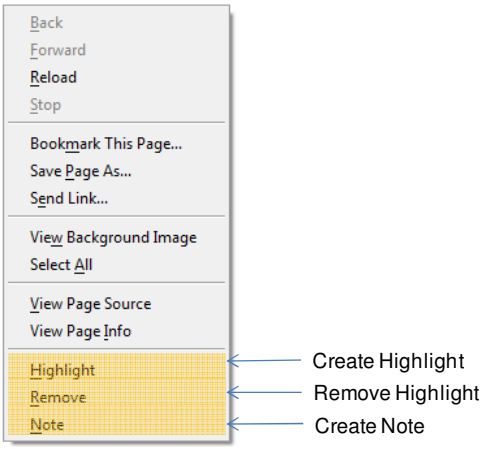


Figure 25. Context menu of WebAnnotate-installed Firefox

For removing a highlight, the user selects the text with the highlight to remove, and presses the remove button in the WebAnnotate toolbar (Figure 24) or the remove menu item in the context menu (Figure 25).

7.1.2 Notes

WebAnnotate notes allow users to add textual comments to a Web page using various colors, a bit like adding Post-it notes to a printed document. WebAnnotate notes are composed of four parts as shown in Figure 26: the header, the remove button, the resize button, and the body. There are two modes: the edit mode (Figure 26 – left) and the selection mode (Figure 26 – right). Users can move, resize, and delete the note when in the selection mode, while users can change the text of the note when in edit mode. Users switch from selection mode to edit mode by double clicking the body of the note. When users move or resize the note or click anywhere outside the body of the note, the mode automatically changes from edit mode to selection mode.

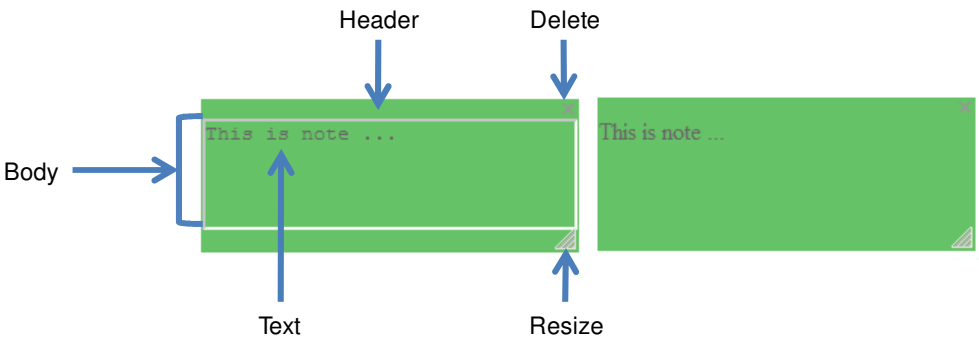


Figure 26. WebAnnotate node: the edit mode (left) and the selection mode (right)

To create notes, users select a color using the color selector in the WebAnnotate toolbar but they do not select text as they must do when creating a highlight. When users

press the note button in the WebAnnotate toolbar (Figure 24) or the note menu item in the context menu (Figure 25), a note is created in the center of the Firefox window in edit mode so that users can immediately add text to the note. In addition, users can move the note by dragging the header of the node or resize the note by dragging the note's resize button. Users can edit the note at any time. If the note is in selection mode, users need to double click the body area of the note to switch to edit mode. In addition, users can delete the note by pressing the delete button on the note.

7.2 Visualization of User Interest

Annotations made in WebAnnotate are stored in the IPM. The IPM uses these annotations, along with the information from VKB, when deciding on suggestions.

7.2.1 Suggesting Paragraphs in WebAnnotate

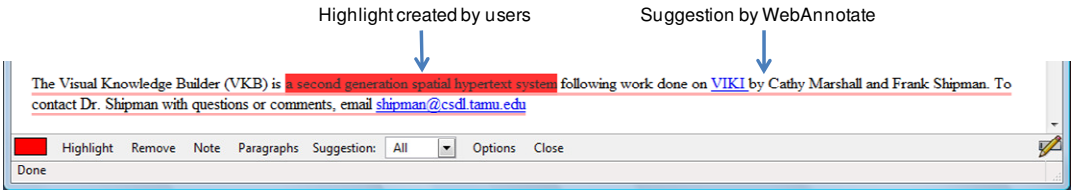


Figure 27. Suggestion of a paragraph by the WebAnnotate

Visualizations of user interest in VKB and WebAnnotate are based on the same interest-related information and the same estimation module in the IPM. However, the visualization in WebAnnotate is for helping find useful segments within a Web page, while the visualization in VKB is for helping find useful documents within workspaces of VKB. Like visualization in VKB, WebAnnotate's visualization in Web pages is

independent from the estimation algorithms in the IPM. WebAnnotate visualizes potentially related paragraphs (sometimes words) by using underlines as shown in Figure 27. The color of the underlines is determined by the color of the related annotations in the organizing or reading interface. Figure 27 shows how the suggestion underlines can be distinguished from highlights created by users.

7.2.2 Human-authored Visual Expression and System-Generated Visualization

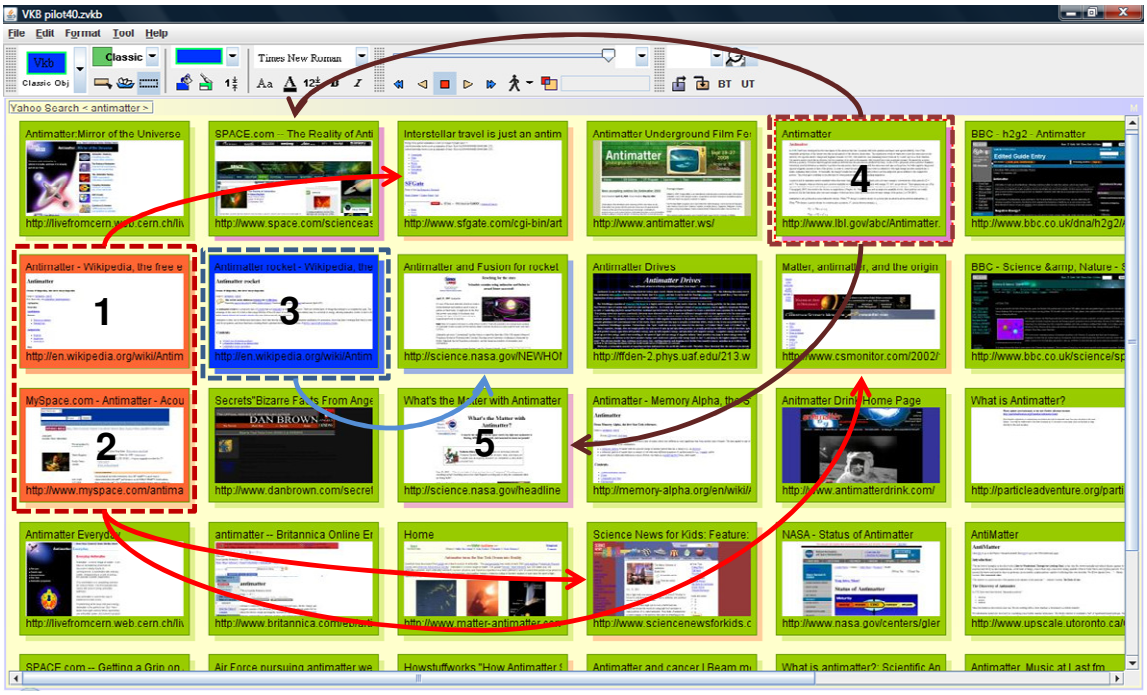


Figure 28. Human expression and system-generated visualization in VKB

During document triage, users express their understanding of the documents by changing visual attributes of the user layer of the document object in VKB and by creating highlights and notes in WebAnnotate. This expression is used to suggest related documents by changing visual attributes of the system layer of VKB document objects

and to suggest related paragraphs within a Web page by underlining them. Figures 28 and 29 show user expression and system suggestions in both applications.

In Figure 28, a user changes the background color of two document objects (1 and 2) to red and the background color of one object to blue (3). The IPM calculates the similarity between the full text content of the two red document objects and the other documents in workspaces of VKB. Based on the estimation, VKB changes the background color and transparency of the system layer for three document objects, suggesting them as potentially related documents. Similarly, one document object is assigned a blue system layer.

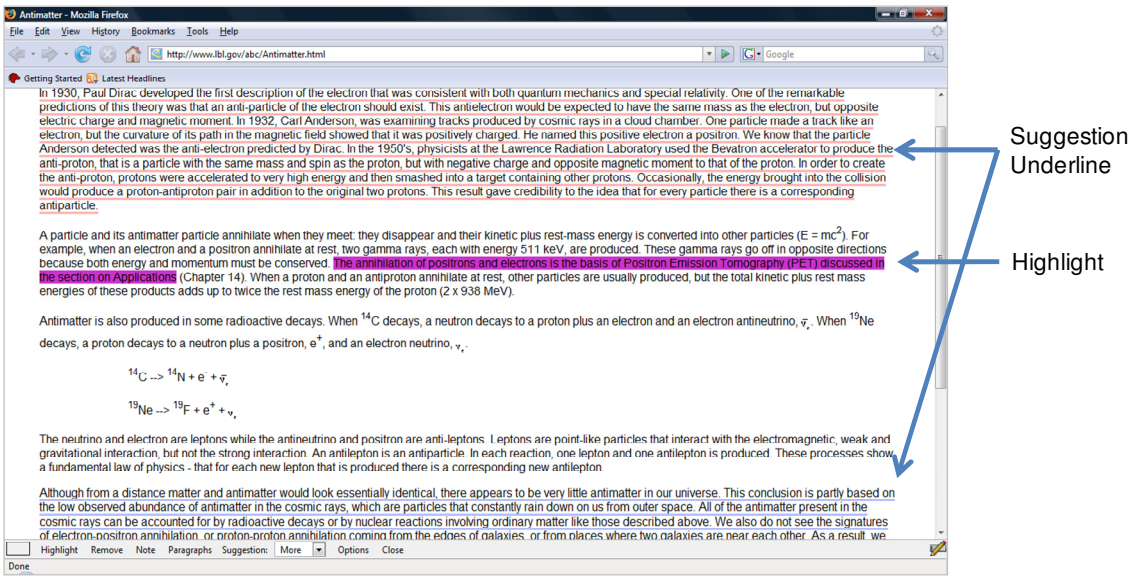


Figure 29. Human expression and system-generated visualization in WebAnnotate

When a Web page is opened in Web browser, the IPM compares the similarity between the full text content of the two red document objects (1 and 2) and paragraphs in the Web page and between the full text content of the blue document object and the

paragraphs in the Web page. Based on the estimation, WebAnnotate adds blue underlines to some paragraphs, suggesting them as potentially related to the content of the document colored blue by the user, e.g. the bottom paragraph in Figure 29.

Figure 29 shows that the Web page of document object 4 in Figure 28 opened in the Web browser. WebAnnotate asks the IPM if there are any suggestions or existing annotations for the document. Figure 29 shows one segment with a violet-colored highlight and two suggested segments, one related to the document colored blue in VKB and the other related to the segment with violet highlighting. Also, VKB has suggested two of the document objects in Figure 28 are related to the segment with violet highlighting.

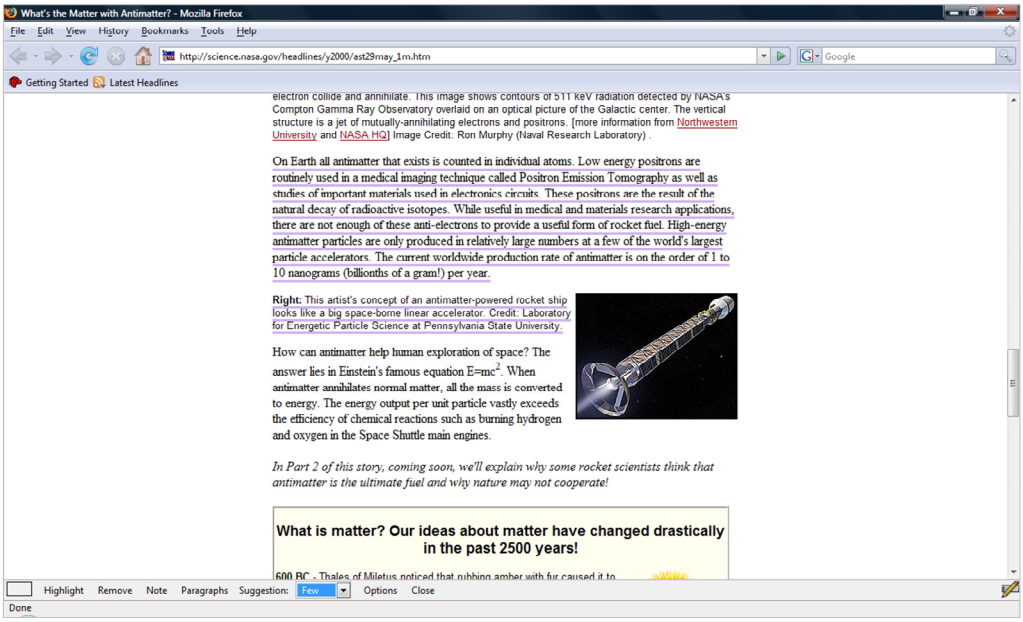


Figure 30. Suggestion in a Web page based on annotation in other Web page

Figure 30 shows the Web page of document object 5 in Figure 28 in the Web browser. When the Web page is opened, WebAnnotate asks the IPM if there are any prior annotations (highlights or notes) and whether there are any suggestions for the Web page. While there are no annotations, Figure 30 shows two paragraphs underlined in a violet color due to a relationship to the highlighting in Figure 29.

User-authored visual expression in organizing and reading is not overwritten by system-generated visualization in organizing and reading. In addition, user-authored visual expression in organizing and reading does not invalidate system-generated visualization in organizing and reading.

7.3 Representation of Annotation

User annotation created with WebAnnotate is sent to the IPM and used for estimating user interest. In addition, these annotations are saved by the IPM and sent to WebAnnotate when users revisit Web pages to regenerate the annotations. WebAnnotate uses XML formats for describing user-authored highlights and notes, which can be transferred between applications and also can be parsed for interpreting the user annotation.

WebAnnotate notes are described as shown in Table III. The Id is assigned by the creation time of notes to be unique with the text and style values. The Active and Changed fields are used to indicate when notes are modified or deleted. The Text and Style fields record the content and color of the note, respectively. The location of WebAnnotate notes is described by the Left, Top, Width, and Height fields, which imply x- and y- values of left-top most coordinate of the note, width, and height.

Table III. WebAnnotate note description

Values	Description
Id	Uniquely identifying note
Active	Showing whether or not note is currently deleted
Changed	Showing whether or not note has changed
Text	Text information on note
Style	Background color of note
Left	x value of left-most coordinate
Top	y value of top-most coordinate
Width	Width of note
Height	Height of note

Similarly, user-created highlights and system-generated underlines in WebAnnotate are described as shown in Table IV. The Id is assigned by the creation time. The Active field is used for dealing with deletion of highlights. The Text field contains the text affected by the highlight/underline and the Style indicates whether it is a highlight or underline and the color.

Table IV. WebAnnotate highlight description

Values	Description
Id	Uniquely identifying note
Active	Showing whether or not note is currently deleted
Representation	Representation of highlight (or suggestion)
Text	Highlighted text
Style	Style information of note such as background color

The Representation field indicates the location of highlighted text within a Web page. There are two alternative approaches for specifying highlighted text within a Web page: by offset in the text content and by the tree structure of Web page.

In the first approach, the offset of the highlighted text from the beginning of the text content is used. A Web page is composed of HTML tags, which forms a tree structure called Document Object Model (DOM) tree, and text content of Web page can be accessed by traversing DOM tree. Text content of Web page is usually spread over multiple nodes of DOM tree. For determining the offset of highlighted text within a Web page or accessing the highlighted text by the offset, WebAnnotate traverses the DOM tree of a Web page and examines the text value of the nodes. However, these operations are time consuming, especially for a long Web page with many highlights (or suggestions). Firefox does not currently provide a multi-threaded environment for browser add-ons, which means that long operations can be interrupted and dropped by following operations.

The second approach specifies the highlighted text by paths from the root node of the DOM tree to the nodes of highlighted text. In this way, WebAnnotate traverses only nodes along the paths from the root node of DOM tree to the start and end nodes of the highlighted text, which is more time efficient than the first approach. However, as users add highlights on a Web page or suggestions are generated, HTML tags are dynamically added for showing highlight or suggestion. This implies that the structure of the Web

page changes over time even though text content of the Web page does not change at all, and the second approach sometimes may not locate the desired content.

WebAnnotate and the IPM use an annotation representation to deal with these issues that combines the advantages of the two above approaches for specifying or accessing highlighted text. The representation first identifies a container node for the start and end nodes of the selected text. The container node is selected to be an ancestor node of the start and end nodes of the text segment that will not be changed by new highlights or suggestions.

The container node is usually a small fragment of the entire DOM tree for the Web page, and WebAnnotate can take the first approach within the container node without spending much time. Within the container node, WebAnnotate specifies the start and end node of highlighted text by using their offsets from the beginning of the text content of the container node. This approach works even when the structure of Web page changes because the text content of the container node does not change. WebAnnotate uses the second approach of specifying the path from the root mode to the container node. The result is more efficient than the first approach yet is stable with regards to low-level changes in the DOM tree.

Figure 31 shows an example of a Web-Annotate/IPM annotation Representation: aptr:/1/4/0/0/1/0/0/0 (0), /1/4/0/0/1/0/0/0 (132). It represents the path from the root node to the container node of start node of the highlighted text. Numbers in the path indicate the number of the child node at each point in the DOM tree. Only nodes of which type is

element and not created by highlight or suggestion are counted as a child node in the annotation representation. The path from the root node to the end container and offset within end container are encoded the same.

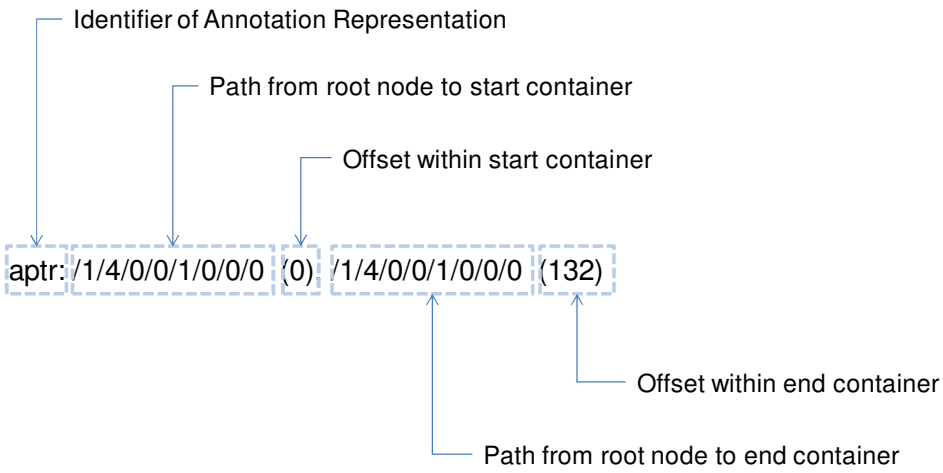


Figure 31. Example of annotation representation

7.4 Collecting User Events and Document Attributes

Table V. Interest related information collected by WebAnnotate

Group	Interest related information
User Events	Mouse clicking, mouse scrolling, focus in/out
Document Attributes	Number of characters, number of hypertext links, number of images, paragraph information

The earlier study of document triage showed that user activity in reading, such as reading time and mouse scrolling event, and document attributes, such as document length or the number of hypertext links, can be useful for inferring user interest [Badi,

Bae et al. 2006]. The user interest estimation module currently employed in the IPM uses only the changes to document object color in VKB and highlight and note annotations in WebAnnotate. However, the IPM can use different estimation modules or even multiple estimation modules. Therefore, WebAnnotate sends the IPM the user interest information as shown in Table V, because the information might be used by alternative estimation modules in the IPM.

7.5 Architecture

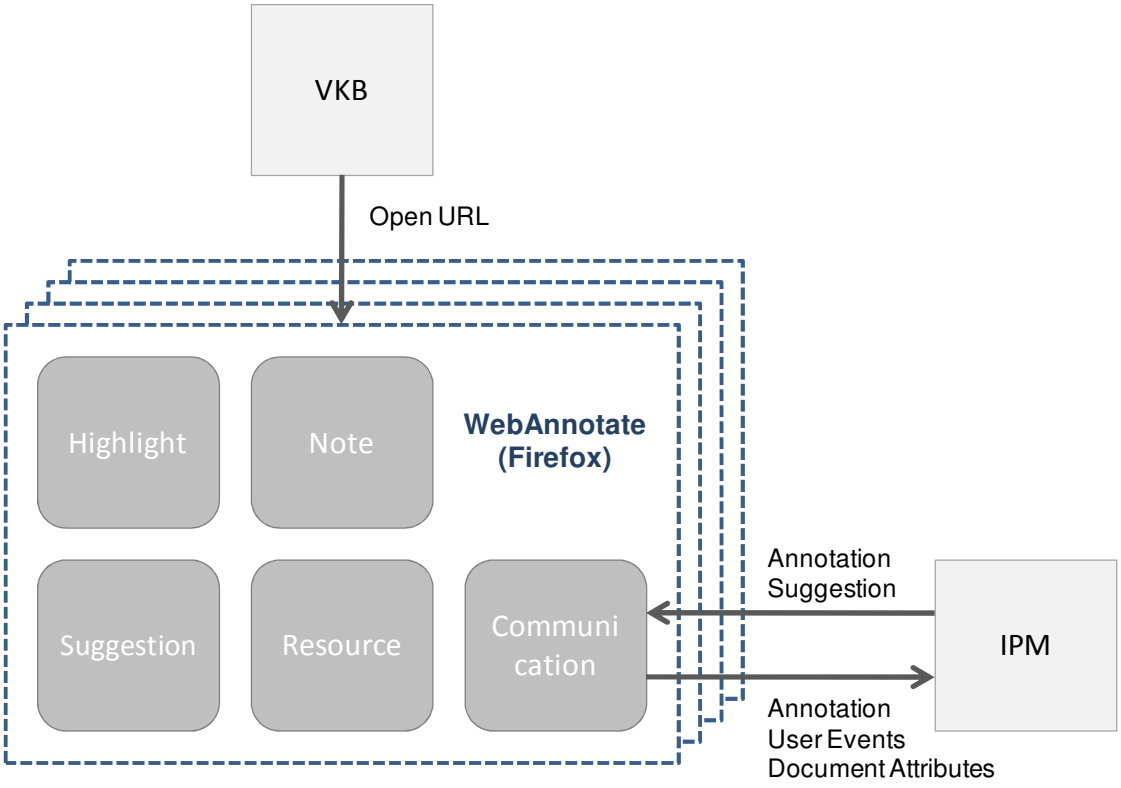


Figure 32. Architecture of WebAnnotate

WebAnnotate, as a Web browser add-on, can interact with any application following the defined XML format for communicating with other applications, but

cooperates with VKB and the IPM in document triage as shown in Figure 32. Web browsers (including WebAnnotate) are usually invoked from the document object of VKB. When a Web page is opened, WebAnnotate sends the document attributes of the Web page to the IPM, and receives previously generated annotations or suggestions on the Web page from the IPM. As users read Web pages and create annotations on them, WebAnnotate sends user events or user annotation to the IPM. The IPM sends updated estimations of user interest to WebAnnotate as they change.

WebAnnotate is composed of five modules as shown in Figure 32: Highlight, Note, Suggestion, Resource, and Communication. The Highlight and Note modules provide the functionality required for user annotations in Web pages such as creating, modifying, and deleting. When a Web page is opened, the Suggestion module parses the Web page into paragraphs and sends the paragraph information to the IPM. The IPM estimates the similarity between each paragraph and the interests represented in the IPM. Once WebAnnotate receives the estimation of interest for each paragraph from the IPM, the suggestion module generates underlines on the Web page.

The Resource module manages the information concerning annotations, document attributes, and user events. All the information in WebAnnotate is stored in the Resource module once it is generated. When users open a Web page, read the Web page, or create annotation on the Web page, the document attributes of the Web page, user events, and user annotations are all stored in the resource module. The information collected in the Resource module is usually bundled together and sent to the IPM when the Web browser loses focus, is exited, or changes URL. However, the Resource module

immediately sends new, edited, or deleted user annotations to the IPM for updating system-generated visualizations based on changes as quickly as possible. Similarly, when the IPM sends WebAnnotate previously generated annotations, the Resource module updates the information collection. The communication module establishes and closes TCP/IP connection with the IPM and also sends or receives data from and to the IPM.

8 INTEREST PROFILE MANAGER

8.1 Overview

The earlier study of document triage showed a correlation between user activity and user interest, which indicates the potential of user activity in estimating user interest [Badi, Bae et al. 2006]. The IPM plays a key role in inferring user interest during document triage. The IPM collects interest-related information from VKB and WebAnnotate in document triage into the interest profile and also estimates user interest and shares it among the applications. For estimating user interest, the IPM has a user interest estimation module, which computes the similarity of text content between documents, paragraphs, and highlighted text. However, the IPM is designed to be independent from any particular user interest estimation algorithm: different algorithms can be used instead of the current algorithm or even multiple algorithms can be used at the same time. The IPM has been extensively revised from earlier versions to support many features of the new versions of VKB and WebAnnotate.

8.2 Interest Profile

Applications in document triage can have their own methods for recognizing user interests, but the inferred interest should be shared among applications to be useful to the entire triage process. The interest profile is a collection of interest-related information and also provides a common representation of interest-related information. The interest profile uses XML for the common representation, where various kinds of interest-related information can be expressed within the information hierarchy as shown in Figure 33. In

Figure 33, the attribute is usually a basic unit expressing a primitive unit of interest-related information such as background color of the user layer of the document object in VKB or the number of hypertext links in documents. The attribute is composed of three parts, the attribute name, the attribute value, and the time stamp. The attribute vector is a set of attributes. For instance, document attributes of Web page or user events in VKB can be managed as an attribute vector. The attribute object is the highest level of interest-related information possibly composed of multiple attribute vectors. The IPM currently creates one attribute object for each document (or Web page).

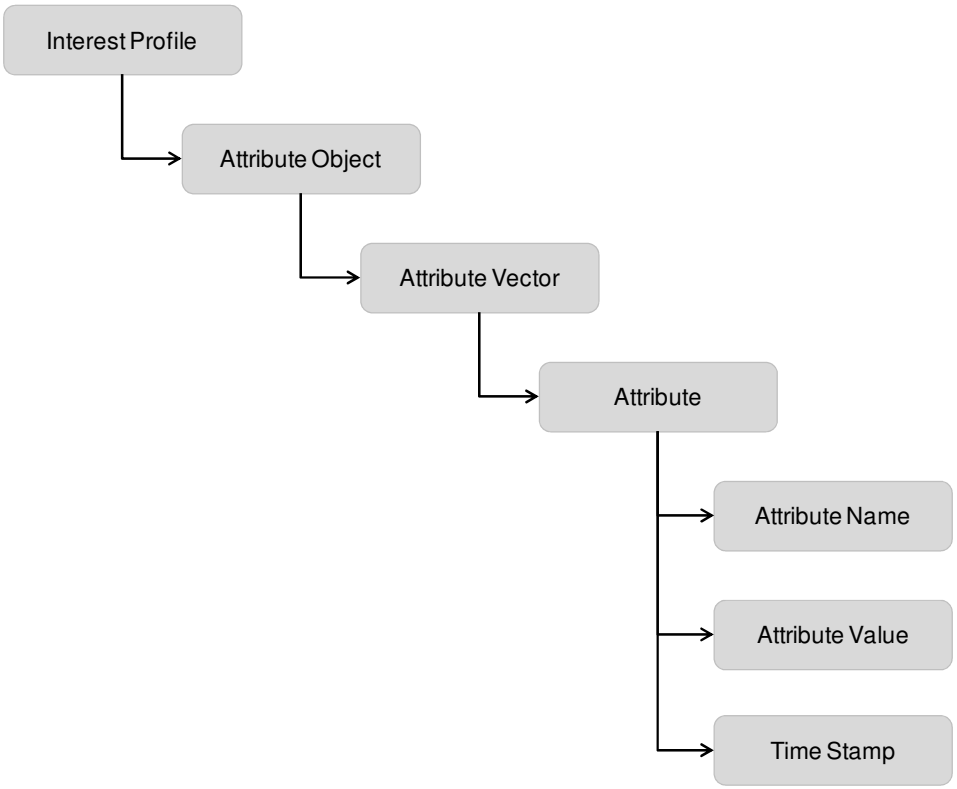


Figure 33. Information hierarchy of the interest profile

Based on the prior studies of document triage, the IPM currently collects interest-related information from VKB and WebAnnotate as shown in Table VI.

Table VI. Interest-related information in the interest profile

Group	Interest related information
User Events (Organizing)	Create document object, move document object, resize document object, change background color, change border color, change border width, change font color, change font, switch layer of document object, change layout of document object, change style, change title, change URL, change annotation
User Events (Reading)	Time spent, mouse click, mouse scrolling, text selection, highlight, note
Document Symbol Attributes (Organizing)	Location, size, background color, border color, border width, thumbnail size, title, URL, summary
Document Attributes (Reading)	Number of links, number of images, number of characters, paragraph information

New attributes can be easily added to the interest profile as different applications are added. The information profile is a XML file and loaded when the IPM starts.

8.3 Estimation of User Interest

8.3.1 Term Vector Model

A Term Vector model (or vector space model) is an algebraic model for representing text documents as vectors of terms [Salton, Wong et al. 1975]. Terms can be single words, keywords or longer phrases according to the application: single words

are used as terms in this dissertation. A document is represented as a vector of which each dimension corresponds to a distinct term. Relevancy of documents can be calculated by the cosine of the document vectors as follows:

$$\cos\theta = \frac{v1 \bullet v2}{\|v1\|\|v2\|}$$

where $v1$ and $v2$ are document vectors.

For each distinct term in the document vector, there are many different ways for calculating its value in the vector. This dissertation employs tf-idf (term frequency-inverse document frequency) weighting, which is often used in information retrieval area. The weight implies the importance of a word in a document within a document collection. The importance proportionally increases to the occurrence of a word in the document but is offset by the frequency of the word in the document collection. The tf-idf of the term t_i within the document d_j can be determined as follows:

$$tfidf_{i,j} = tf_{i,j} \bullet idf_i$$

where $tf_{i,j}$ is the term frequency in the document and idf_i is the inverse document frequency in the document collection.

The term frequency in the document basically implies how many times a term appears in the document. However, when a simple count is used, this can be biased towards longer documents that may have a higher term frequency regardless of the importance of the term in the document. Therefore, the term frequency is usually normalized as follows:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

where $n_{i,j}$ is the frequency of the term in document d_j and $\sum_k n_{i,j}$ is the frequency of all terms in document d_j.

The inverse document frequency is for measuring the general importance of the term and determined as follows:

$$idf_i = \log \frac{|D|}{|\{d_j : t_j \in d_j\}|}$$

where $|D|$ is the number of documents in the collection, and $|\{d_j : t_j \in d_j\}|$ is the number of documents that have the term t_j.

Therefore, terms with a high term frequency in the document and terms with a low document frequency can get a high weight in tf-idf. As a result, common terms (e.g. "the", "is", etc.) are reduced in their effect on computations in tf-idf weighting scheme.

8.3.2 Estimation of User Interest in IPM

The estimation of user interest in the IPM currently utilizes user annotation generated during organizing and reading [Bae, Hsieh et al. 2008]. As users change background color of the document object in VKB, the estimation module calculates the cosine similarity of term vectors between the full text content of the documents whose background color is the same as the background color of the document object, full text content of all the other documents in workspaces of VKB, and paragraphs of documents that are currently opened in Web browser. Similarly, as users highlight documents in the

Web browser, the estimation module calculates the similarity of term vectors between the highlighted text, the full text content of all the documents in workspaces of VKB, and paragraphs of documents that are currently opened in a Web browser. Therefore, the estimation of user interest changes as users express themselves via annotations, which can enable more reliable estimation over time.

The process above describes how the IPM estimates user interest on the documents already in VKB workspaces and documents already opened in Web browser when users make, edit, or delete annotations. When users add or delete document objects in VKB workspaces or open documents in Web browser, the estimation module calculates the similarity in the same way.

The IPM maintains two sets of term vectors. The first set of term vectors are for annotations (interests) and the second set is for the textual content of documents in VKB and paragraphs in Web browsers. An annotation term vector is created for each color used to annotate document objects in VKB and for each color of highlight annotation used in WebAnnotate. The term vector for a VKB-based interest is the sum of the full text term vectors for the documents assigned a particular color. Similarly, a WebAnnotate interest is the sum of the text segments annotated with a particular color.

There are four types of user activities that cause estimation of user interest to start: creating (or deleting) document objects in VKB, opening (or closing) documents in a Web browser, changing the color of a document object in VKB, and adding or modifying an annotation in WebAnnotate. Depending on the user activity, the corresponding term vectors are updated. When users make an annotation in VKB or

WebAnnotate, the term vector for the particular annotation is updated. Similarly, when users create or delete documents in workspaces of VKB or open documents in Web browser, the term vector for text content is updated. However, once the term vectors are updated, the calculation of the similarity is the same.

Figure 34 shows the relationship between the four types of user activities and the term vectors. The calculated similarity values are delivered to VKB and WebAnnotate so that they can update visualizations indicating potentially useful documents in VKB workspaces and paragraphs in the Web browser.

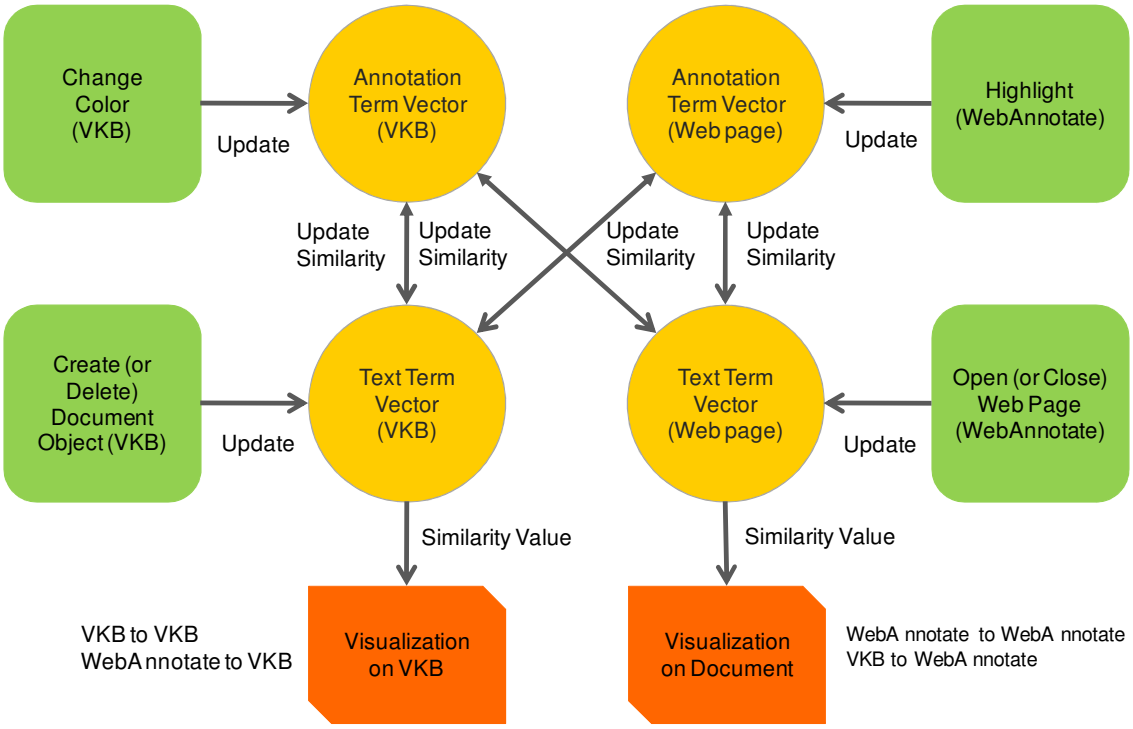


Figure 34. Calculation of the similarity with term vectors

8.4 System Architecture

The IPM is composed of 7 modules as shown in Figure 35. The Interest Profile is the information collection where the interest-related information is stored. The Communication module manages the communication with applications in document triage. The IPM defines the XML-based communication interface with which any application can interact with the IPM over TCP/IP. The Communication module receives requests or interest-related information from applications and also sends response to the requests or user interest information to applications. When the IPM receives events indicating the creation of new document objects, the Web Page Crawler collects the full text content of the document. The Text Processor tokenizes the collected text and selects nouns so that text can be used for estimation of user interest. When the IPM receives an event opening a document in Web browser and paragraph information, the Text Processor performs the same process, but also performs text segmentation on the collected paragraph for obtaining more meaningful fragmentation of text. As the various applications send or receive messages to and from the IPM, the Command Hander makes sure that information updates and requests are ordered for orderly processing and information access.

The Resource Manager loads and saves the interest-related information from and to the Interest Profile. In addition, the Resource Manager updates the interest-related information according to the events sent from applications.

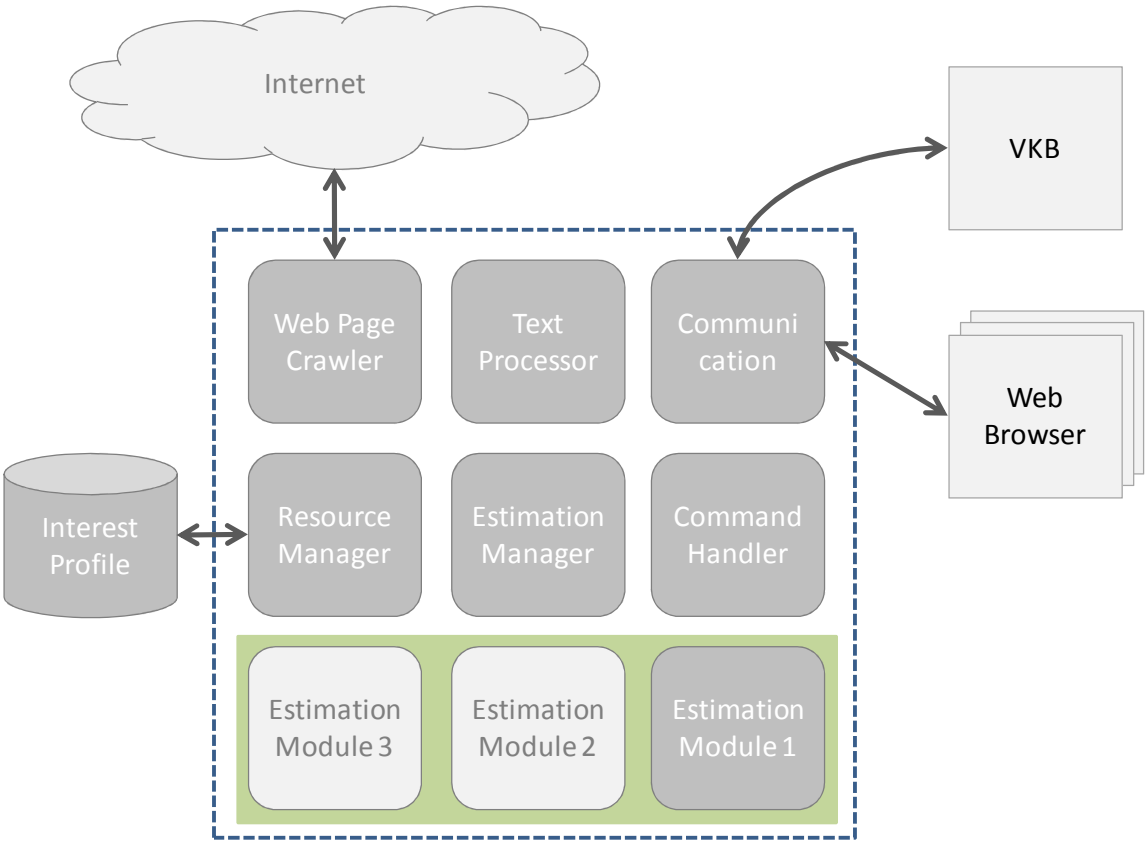


Figure 35. Architecture of IPM

The IPM includes two modules for estimating interest, the Estimation Manager and the Estimation Modules. The Estimation Manager provides a generic high level interface to the other modules within the IPM and also enables multiple estimation modules. There can be one or more Estimation Modules to estimate user interests using different algorithms, although the IPM currently employs a single estimation module.

9 EVALUATION

We have performed a user study to evaluate the effectiveness of the recommendations provided in VKB and the WebAnnotate-augmented Firefox. The evaluation focuses on whether the recommendations help participants find documents of interest, whether the visualizations help participants keep track of their progress, and whether the new concept of layers helps reduce the conflict between user-authored visualizations and system-generated visualizations that we identified in an earlier study.

9.1 Experimental Design

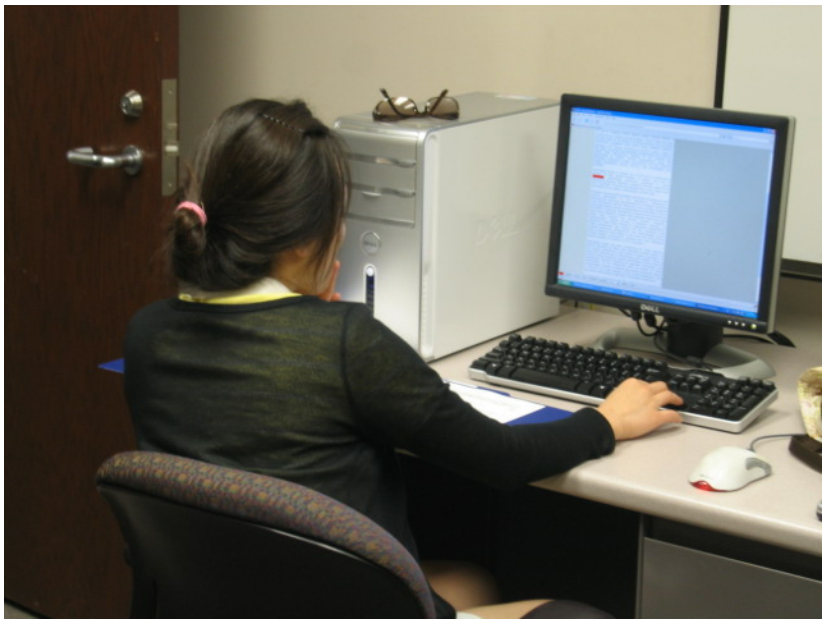


Figure 36. User study

Twenty undergraduate and graduate students were recruited via email. Sixty percent of the participants were from the Computer Science Department. Seventy percent were graduate students, and the other 30% were undergraduates. Participant ages

ranged from 18 to 39. The study was conducted in the Center for the Study of Digital Libraries at Texas A&M University. All participants use a computer regularly and are familiar with searching and browsing the Internet.

Participants were placed in the role of a research librarian who had to select and organize documents for a science teacher preparing a class on antimatter (Figure 36). They started with 40 documents that were returned from a Yahoo query and automatically placed in lists in VKB. None of the participants had prior experience with either VKB or WebAnnotate. The instructions suggested that the task would take about 45 minutes, but that they could continue working as long as they needed to. Participants were randomly divided into two groups with different software configurations. Participants in Group 1 were given a software configuration that including the enhanced visualization capabilities in VKB (the VKB 3 release) and the WebAnnotate plug-in to Firefox. Participants in Group 2 were given a software configuration without the enhanced visualization capabilities in VKB (the VKB 2 release) and Firefox without the WebAnnotate plug-in.

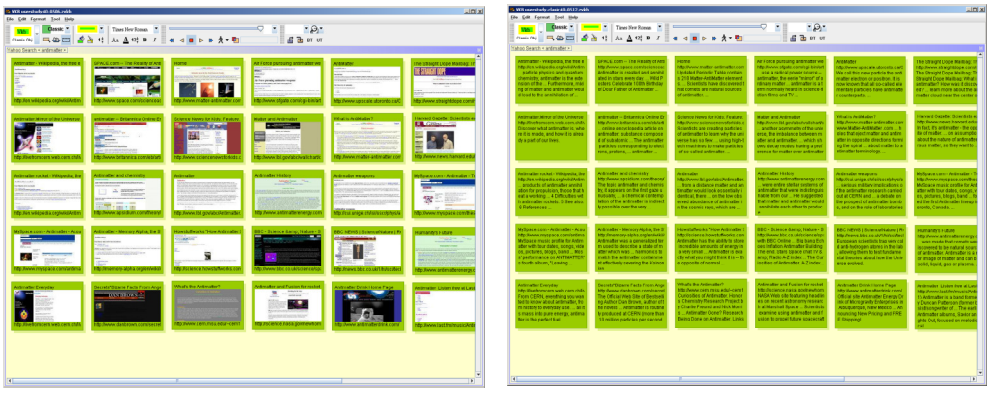


Figure 37. Initial document list: group 1 (left) and group 2 (right)

Figure 37 shows the initial document lists given to the participants in the two groups. Each of the two groups started with the same 40 documents returned from Yahoo; their x, y arrangement in the VKB space was also the same for the two groups. As is illustrated by Figure 37, Group 1 used document surrogates that included thumbnails; Group 2 used document surrogates that had text snippets in place of thumbnails. Group 1 also performed the triage task in an application environment that included the IPM and the visualizations it generated; Group 2 did not. The original metadata for each of the 40 documents was the same between the two groups.

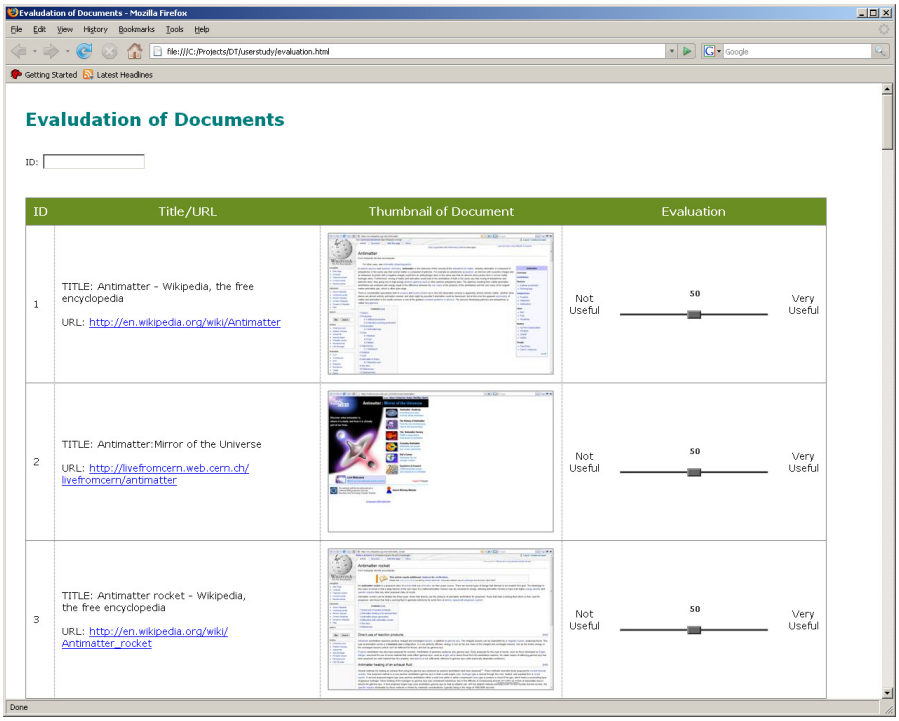


Figure 38. Web-based interface for evaluating documents

Participants were given a short training session as a prelude to performing the study task; at the conclusion of this session, they were able to ask questions about any

functionality that they did not understand or aspects of the task that they felt were unclear. At the conclusion of the triage task, participants were asked to assess the value of each of the 40 documents on a scale 0 to 100 given the interface shown in Figure 38. They were also asked to complete a short questionnaire so we could assess their attitudes to the enhanced applications and whether the applications helped them perform the task or distracted them in any way.

9.2 Final Organization

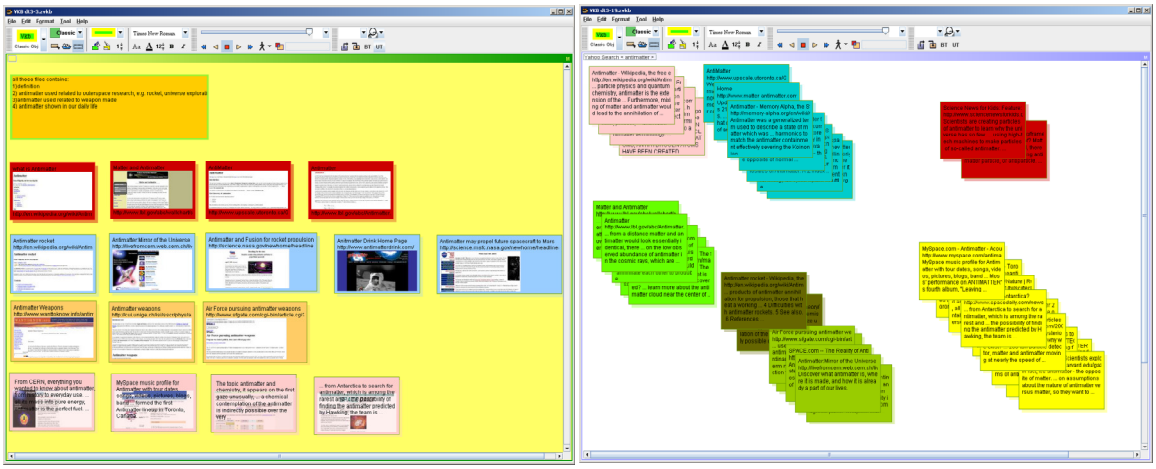


Figure 39. Screenshots of finished workspaces. The screenshot on the left is an example from group 1 and the screenshot on the right is an example from group 2

In both study conditions, participants were free to open the documents in Firefox to examine their contents; they were also given the same goal of organizing the document surrogates in a way that would be helpful for the teacher. The basic VKB functionality gave participants in both groups the ability to change visual attributes of document objects (background color, border color, border width, and size) as they read and interpreted them; they were also free to use links, add textual comments to their

workspaces, and create sub-collections as they saw fit. Thus, participants had many alternatives for expressing their understanding of the documents and their relation to the given task. Figure 39 shows one example of the finished workspace from each group.

As we have observed in previous studies, participants went about the triage task in significantly different ways regardless of their group. Some participants began by reading a few documents carefully to get a sense of the topic and the types of documents they were working with, and then proceeded to organize the remaining documents quickly. Some participants began by organizing documents, and then refined their initial organization as they read. Some participants arranged the documents in VKB's workspace using both space and color; others used space alone, stacking documents or creating sub-collections according to perceived categories; and a third set of participants organized documents using their visual attributes without changing their spatial layout.

9.3 Annotation on Document

Table VII. Number of highlights and notes of participants in Group 1

Participant ID	1	2	3	4	5	6	7	8	9	10
# of highlights	11	0	0	2	3	0	17	26	46	18
# of notes	0	32	24	0	0	0	0	9	0	25

The triage task could be performed without annotating documents (by working without WebAnnotate, participants in Group 2 were missing the ability to annotate); thus, annotations were neither required nor suggested. However, annotations were

perceived as one way of going about the task and of communicating the participant's intent to the teacher (the hypothetical consumer of the results).

Table VII summarizes the number of discrete highlights and notes that the participants in Group 1 (the Group using VKB 3 and WebAnnotate) created during the task. The type and quantity of annotations varied according to personal preferences and styles. Seven out of the ten participants in this condition were relatively active in either highlighting or adding notes (two participants, P8 and P10, did both); three participants were not. Figure 40 shows examples of annotations that participants created during the task.

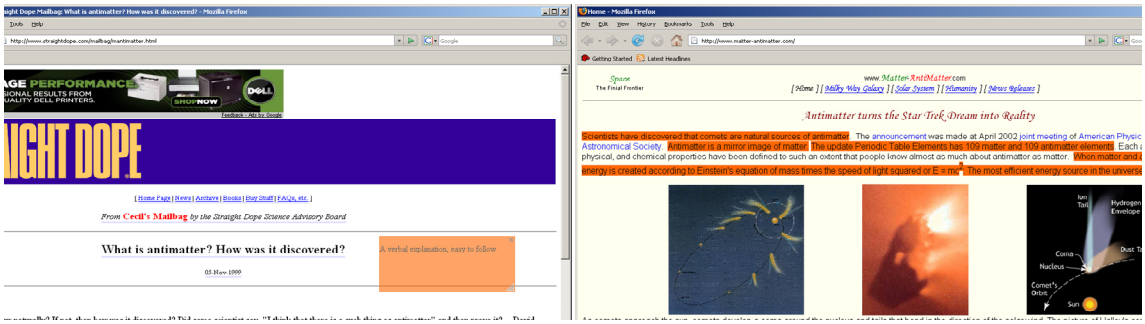


Figure 40. Examples of annotations on documents. The screenshot on the left shows a participant's note; the screenshot on the right shows a participant's highlight. Both the note and highlight are in orange

9.4 Activity Data and Analysis

While users were performing the task, user actions in VKB and the Web browser were logged. The log of document reading activity from the Web browser included time spent on a document, mouse clicks, scrolls, focus-in, focus-out, mouse over, mouse out, highlight, and note. Document organizing activity logged by VKB included changes to

spatial or visual attributes of objects and any object or collection creation. The data analysis is focused on evaluating whether the visualizations helped users perform the triage task from three perspectives: (1) keeping track of progress; (2) identifying documents of interest; and (3) reducing the conflict with user-authored visualizations.

9.4.1 Keeping Track of Progress*: Task Switching*

Our previous document triage study showed that users had difficulty keeping track of progress while they shifted their attention quickly between the triage-related applications or among multiple documents. VKB 3's new thumbnail-based document surrogates were designed to make the objects more recognizable (and potentially more useful) without opening them. To examine the effectiveness of this visualization in helping users keep track of their progress through the document list and shift from document to document, we have examined reading-related characteristics of the two study groups' interactions with the constituent applications.

Table VIII. Descriptive statistics – reading time

	N	Mean (second)	Std. Dev.
Group 1	918	17.76	41.863
Group 2	741	12.57	19.979

Table VIII shows the mean time that participants in the two groups kept documents open and in focus, which we call reading time. The average reading time of participants in Group 1 (17.76 seconds) is 41% greater than the average reading time of participants in Group 2 (12.57 seconds). Furthermore, the number of reading events of

participants in Group 1 (918) is greater than the number for participants in Group 2 (741). Thus it appears that participants in Group 1, the group using VKB 3 and WebAnnotate, are looking at more documents, and remaining engaged with them for a longer time, on average.

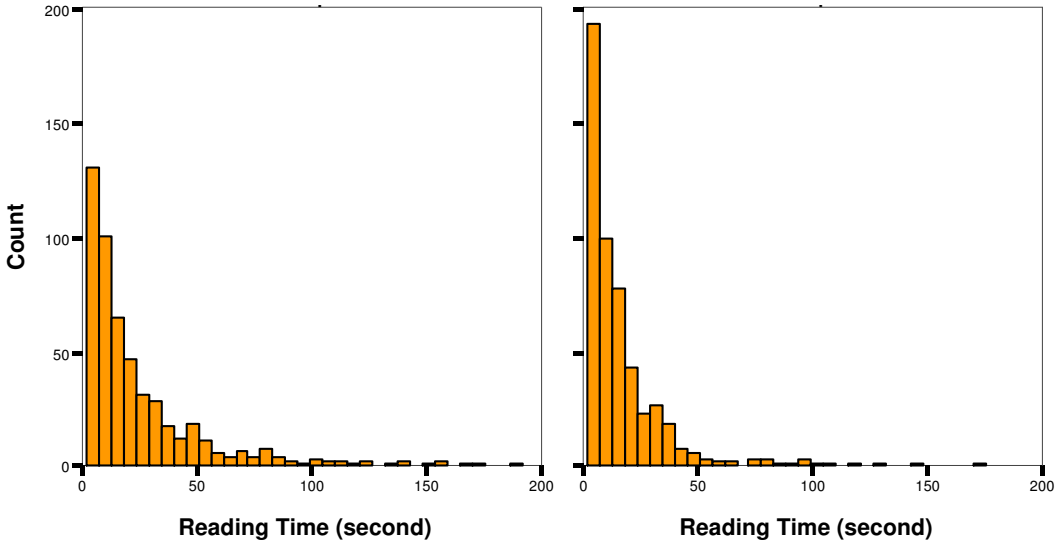


Figure 41. Reading time of group 1 (left) and group 2 (right)

Figure 41 shows the distribution of per-document reading times of participants in the two groups. The histogram shows that instances of relatively short reading intervals occurred more often in Group 2, while relatively long reading intervals were more common in Group 1. Using the Mann-Whitney test, we can see that participants in Group 1 do not differ significantly in their reading time (the time a page is displayed on the screen with mouse focus) from Group 2's per-document reading time (U= 321089, p = 0.050), but the difference is marginally substantial. It is notable that Group 2 participants had substantially more episodes in which they focused on a document very

briefly; it seems that VKB 3's facility for displaying document thumbnails helped alleviate the phenomenon of needing to put a document into focus just to remind oneself of what it is.

Table IX. Descriptive statistics – session reading time

	N	Mean (second)	Std. Dev.
Group 1	1525	10.69	34.057
Group 2	2148	4.34	14.214

Table IX compares the average session reading times for the two groups. For the purposes of this study, a session is defined by a continuous series of logged interactions that refer to the same document. That is, a user may read, scroll, annotate, scroll some more, and continue reading; this sequence of connected actions is considered to be a session. Session reading time is the sum of the separate reading times within in the session. Group 1's average session reading time (10.69 seconds) is almost 2.5 times (246%) greater than Group 2's (4.34 seconds). Group 1's total number of sessions (1525) is considerable fewer than Group 2's (2148) even though Group 1's total reading time (15,338,798 seconds) is almost double that of Group 2 (8,636,299 seconds). These results indicate that Group 1 looked at fewer documents, but when they actually engaged with a document, they spent far longer doing it.

Figure 42 shows the distribution of the participants' session reading times across the two groups. The histogram shows that Group 2's members had more instances of relatively short reading sessions than Group 1's members, and Group 1's members had

more instances of relatively long reading sessions. This means that participants in Group 1 tended to spend more time on a document before they switched to another document. Based on the Mann-Whitney test, participants in Group 1 significantly differ in session reading time from those in group 2 ($U=1391140$, $p < 0.0001$).

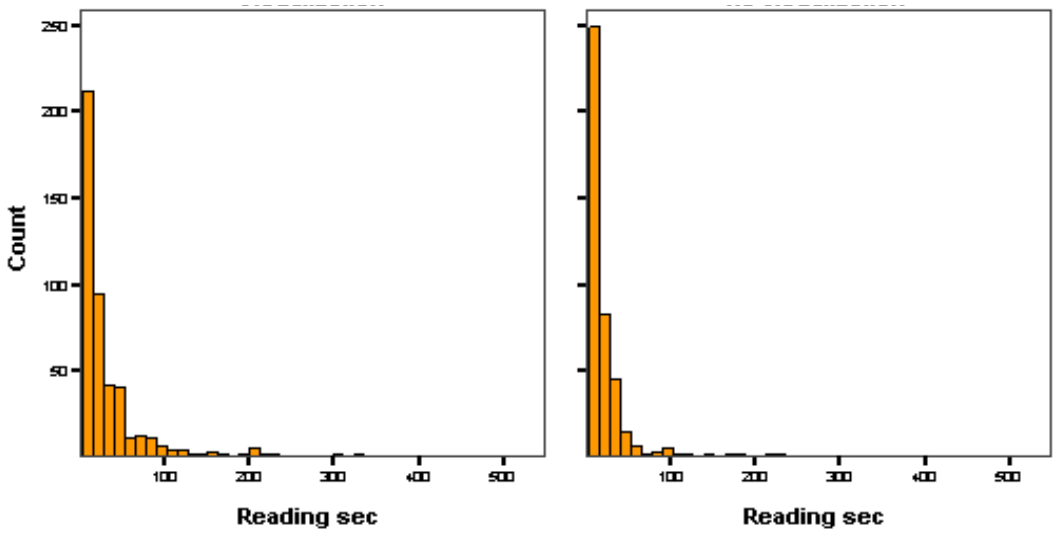


Figure 42. Session reading time of group 1 (left) and group 2 (right)

Table X shows the average number of times each document was opened from VKB (by double-clicking on the document's surrogate); this average was computed by dividing the total number of unique documents each participant opened during the triage task by the number of document-opening events. This average should reveal how many times documents were re-opened. This average for participants in Group 1 (1.25) is 14% lower than the average for Group 2 (1.46), even though Group 1's average reading time (25.55 seconds) is higher than Group 2's (14.40 seconds). This comparison between Groups 1 and 2 indicates that Group 1 re-opened fewer documents than Group 2 did, and

they spent more time reading the ones that they did open. Based on the Mann-Whitney test, participants in Group 1 significantly differ in how many documents they re-opened from participants in Group 2 (U = 55828, p = 0.010).

Table X. The average number of opening document events per document

	N	Mean	Std. Dev.
Group 1	340	1.25	0.988
Group 2	366	1.46	1.126

In summary, Tables VIII through X and Figures 41 and 42 demonstrate that Group 1 participants switched their attention between triage-related applications and among individual documents significantly less than Group 2 participants did. This improvement in maintaining focused attention implies that the new visualizations were effective in helping participants keep track of their progress through the documents during the task. They found it less necessary to re-open documents and they were able to spend longer reading individual documents.

9.4.2 Finding Documents of Interest: Time Spent on Documents

In our previous document triage study, users often spent a substantial amount of time examining documents that they decided later were not useful. This finding suggests that users may benefit from system assistance in locating interesting documents for the task. Did the new visualization help them find the documents they needed? If it did, the participants in Group 1 should have spent more time reading and manipulating the documents they later assessed as relevant than the participants in Group 2 did.

Table XI shows the relationship between reading time and each participant's evaluation of documents based on Spearman's correlation coefficient. All but one participant had a positive correlation, indicating they spent more time on documents they evaluated positively. The values for participants exhibiting a statistically significant correlation ($p < 0.05$) are shaded. For six out of ten Group 1 participants, the correlation is significant; the correlation is significant for only two out of ten Group 2 participants. The higher degree of correlation implies that Group 1 participants generally spent more time on relevant documents than Group 2 participants did; thus we may infer that the new visualizations in the enhanced applications were effective in helping participants find documents of interest.

Table XI. The correlation between reading time and the participant's assessment of document relevance for participants in the two groups

Group 1			Group 1		
Participant	Coefficient	Sig. (2-tailed)	Participant	Coefficient	Sig. (2-tailed)
1	0.429	0.018	11	0.277	0.093
2	0.397	0.014	12	0.111	0.565
3	0.356	0.087	13	0.210	0.205
4	0.409	0.011	14	- 0.148	0.376
5	0.576	0.008	15	0.367	0.024
6	0.206	0.214	16	0.633	< 0.0001
7	0.137	0.412	17	0.116	0.489
8	0.438	0.006	18	0.114	0.495
9	0.629	< 0.0001	19	0.101	0.547
10	0.170	0.309	20	0.240	0.147

9.4.3 Reducing the Conflict between User-authored Visualizations and System-generated Visualizations: Changing Background Colors of Document Objects

In our past research, users expressed their interpretation of documents during the triage task by changing the visual attributes of VKB's document surrogates. However, in an effort to bring certain documents to the users' attention, VKB also changed objects' visual attributes. This overloading of the objects' visual characteristics caused a conflict between user-authored visualizations and system-generated visualizations: many participants in the study were unwilling to overwrite the system's visualizations and became passive in changing objects' visual attributes to express their own interpretations. The visualizations introduced in the work we report here attempt to reduce this source of conflict; were they effective? We should get an indication of this strategy's effectiveness by comparing differences in participants' use of color across the two study groups.

Table XII. Descriptive statistics – changing background color events

	Mean Number of Color Change Events	Std. Dev.
Group 1	16.80	15.533
Group 2	11.30	13.723

Table XII shows the relative levels of color-changing activity between participants in the two groups (by color changing, we mean participants changing the background color of document surrogates in VKB's workspace, a capability offered by both versions of the application). The average number of color-changing events for

Group 1 participants is 48% higher than for Group 2 participants. This discrepancy implies that participants using the enhanced applications changed the background colors of the document surrogates slightly more frequently than participants using the baseline applications, although the difference is not significant based on the Mann-Whitney test ($U=0.250$, $p = 0.250$). This essential similarity in how the two groups (Group 1 with system-generated visualizations and Group 2 without) used color indicates that the enhanced applications successfully overcame the potential conflict we anticipated, and participants in Group 1 felt free to express their interpretations using background color; if the object layering approach had not been successful, we would have seen a diminished use of background color as an expressive medium by the participants in Group 1.

9.5 Questionnaire Results

After the completion of the triage task, the participants in Group 1 recorded their impressions of the utility of the enhanced applications for performing document triage; their responses were recorded using a Likert scale where a value of 1 indicated strong disagreement, and 5 indicated strong agreement. Because the questionnaire was developed to evaluate the new visualization capabilities in VKB 3 and WebAnnotate, participants in Group 2 were not included in this analysis.

9.5.1 Computed Visualization of within-Document User Interest

WebAnnotate recommends passages within a document based on estimations of user interest; user interest is assessed through an aggregated characterization of their interactions with the documents in all of the appropriate applications (as they organized

the documents or annotated them). Visual feedback that suggests potentially related information within a document is displayed as underlined passages. These suggestions change over time as the user continues to organize and annotate the remaining documents.

Eight out of ten participants answered that the within-document visualizations were helpful in finding new information of interest; two participants responded neutrally. Three out of ten participants answered that the within-document visualizations distracted them from their reading; one participant answered neutrally; and the remaining six participants said that the new within-document visualizations were not distracting. Thus we can conclude that most of the participants found the new within document visualization technique to be a helpful means of finding new information of interest within the document they were reading, but some of them found that the visualizations were also distracting.

9.5.2 Computed Visualization of User Interest in VKB

The enhanced version of VKB recommends documents within its workspace based on estimated user interest. Visual feedback is used to suggest potentially related documents, based on the participants' interactions with the applications as they organized and read the documents. The suggestions that VKB offers change over time according to user activity. Although VKB can be used as a search engine interface and general organizing tool, in this study, participants started with a set triage task given a specific set of documents; normally they would be formulating their own queries and would probably be working with a broader range of documents. Thus the questionnaire

focused on a fairly narrow range of finding and organizing tasks in VKB. Nine out of ten participants responded that the enhanced visualization capabilities in VKB were helpful in identifying documents of interest within the workspace. Only one participant answered neutrally. Two out of the ten participants in Group 1 answered that VKB's computed visualizations distracted them from their reading (1 neutral, 6 disagrees, and 1 strongly disagrees). All of the participants found that VKB's visualizations were helpful in organizing documents. Two out ten participants answered that the computed visualizations distracted them when they were trying to organize documents (2 neutral, 3 disagrees, and 3 strongly disagrees). In summary, most of the participants found that the new visualization capabilities in VKB were helpful in finding (identifying) documents of interest and in organizing documents even though a few participants found the computed visualizations to be distracting.

10 CONCLUSIONS AND FUTURE WORK

We have developed an architecture that supports triage across multiple applications. The central element of this architecture is the Interest Profile Manager, which receives information about user activity from the individual applications and broadcasts inferred user interests back to the applications. The IPM consolidates functionality necessary to characterize user interests, including the ability to collect, parse, and determine similarity among common forms of Web documents. In our example instantiation of this architecture, the IPM communicates with a Visual Knowledge Builder workspace and an annotation-enabled Web browser. The examples in Sections 6 and 7 show how the actions of users in either of these applications can generate assistive visualizations in both applications.

To extend this infrastructure with additional applications, the applications must be able to record and aggregate user activity and communicate it to the IPM and/or receive and use broadcasts from the IPM. Applications need not do both; it may make sense for an application that incorporates a non-interactive visualization technique to receive information about inferred user interests without sending any information to the IPM about user activity. Similarly, an application may be interactive, and may offer considerable insight into a user's interests, but it may not make sense to modify anything in that application accordingly; for example, if the user is writing a paper while she is performing triage, the topics that emerge in the paper may be a very effective source of interest profile data.

The classification of documents of into different user interests in the current IPM is based solely on explicit user expression in a single application. For example, documents or subdocuments that a user colors red will generate red visualizations for documents or subdocuments the IPM analyzes and finds to be similar. Other applications have identified classifications of documents by clustering the documents based on textual analysis or image processing [9]. Such capabilities may help determine when the user has multiple interests that are expressed using the same color or when the user has used different colors to express the same interest.

To make the IPM more readily extensible, the IPM needs to incorporate an abstract model that characterizes the expressive and presentational capabilities of applications. For example, such a model would specify that VKB allows users to assign colors to document surrogates to informally express their interest in a document, their understanding of what a document is about, or a general assessment of its worth to them. By contrast, WebAnnotate displays documents' contents; thus any user expression of interest or other interpretation conveyed through annotations happens at a sub-document level. Components of such an application model may include the granularity of the information presented, persistent forms of user expression, transient forms of user interaction, and visualization methods supported.

We have evaluated the effectiveness of the visualizations (including the enhanced presentation of document surrogates) and have found that they are successful in allowing people to do less switching among documents and applications; in promoting longer engagements with individual documents; and in recommending interesting new

documents and passages within documents based on what the user has indicated interest in already. We also found that users were relatively comfortable with the new capabilities and only a few of them found the computed visualizations to be distracting. Although the study was not designed to test the IPM architecture or basic capabilities of the applications, we found that they performed well during the study.

REFERENCES

Adler, A., Gujar, A., Harrison, B. L., O'Hara, K., and Sellen, A. 1998. A diary study of work-related reading: design implications for digital reading devices. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press/Addison-Wesley Publishing Co. Los Angeles, CA, 241-248.

Amento, B., Hill, W., Terveen, L., Hix, D., and Ju, P. 1999. An empirical evaluation of user interfaces for topic management of Web sites. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press. Pittsburgh, PA, 552-559.

Amento, B., Terveen, L., Hill, W., and Hix, D. 2000. TopicShop: enhanced support for evaluating and organizing collections of Web sites. *In Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, San Diego, CA, 201-209.

Badi, R., Bae, S., Michael, M. J., Meintanis, K., Anna, Z. et al. 2006. Recognizing user interest and document value from reading and organizing activities in document triage. *In Proceedings of the 11th International Conference on Intelligent User Interfaces*, ACM Press. Sydney, Australia, 218-225.

Bae, S., Badi, R., Meintanis, K., Moore, J. M., Zacchi, A. et al. 2005. Effects of display configurations on document triage. *In Proceedings of IFIP INTERACT Conference*, Rome, Italy, 130-143.

Bae, S., Hsieh, H., Kim, D., Marshall. C. C., Meintanis, K. et al. 2008. Supporting document triage via annotation-based visualizations. *In Proceedings of ASIS&T,* Columbus, OH.

Bae, S., Marshall, C. C., Meintanis, K., Zacchi, A., Hsieh, H. et al. 2006. Patterns of reading and organizing information in document triage. *In Proceedings of ASIS&T,* Austin, TX.

Buchanan, G., Blandford, A., Thimbleby, H., and Jones, M. 2004. Integrating information seeking and structuring: exploring the role of spatial hypertext in a digital library. *In Proceedings of the Fifteenth ACM Conference on Hypertext and Hypermedia*. Santa Cruz, CA, 225-234.

Card, S. K., Robertson, G. G. and York, W. 1996. The WebBook and the Web Forager: An information workspace for the World-Wide Web. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Vancouver, British Columbia, Canada, 111-117.

Czerwinski, M., Dumais, S., Robertson, G., Dziadosz, S., Tiernan, S. et al. 1999. Visualizing implicit queries for information management and retrieval. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Press. Pittsburgh, PA, 560-567.

Czerwinski, M., Dantzich, M., Robertson, G., and Hoffman, H. 1999. The contribution of thumbnail image, mouse-over text and spatial location memory to web page retrieval in 3D. *In Proceedings of IFIP INTERACT Conference,* Riccarton, Edinburgh, Scotland , 163-170.

diSessa, A. A. and H. Abelson 1986. Boxer: A reconstructible computational medium. *Commun. ACM* 29(9): 859-868.

Dumais, S., Cutrell, E., and Chen, H. 2001. Optimizing search by showing results in context. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press. Seattle, WA, 277-284.

Dziadosz, S. and Chandrasekar, R. 2002. Do thumbnail previews help users make better relevance decisions about web search results? *In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM Press. Tampere, Finland, 365-366.

Gomita. 2006. ScrapBook. from http://amb.vis.ne.jp/mozilla/scrapbook (accessed on 2006).

Graham, J. 1999. The reader's helper: A personalized document reading environment. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press. Pittsburgh, PA, 481-488.

Kaasten, S. and Greenberg, S. 2001. Integrating back, history and bookmarks in web browsers. *In CHI '01 Extended Abstracts on Human Factors in Computing Systems*, ACM Press. Seattle, WA, 379-380.

Kahan, J. and Koivunen, M. 2001. Annotea: An open RDF infrastructure for shared Web annotations. *In Proceedings of the 10th International Conference on World Wide Web*, Hong Kong, 623-632.

Malone, T. W. 1983. How do people organize their desks?: Implications for the design of office information systems. *ACM Trans. Inf. Syst.* 1(1): 99-112.

Marshall, C. C. 1997. Annotation: From paper books to the digital library. *In Proceedings of the Second ACM International Conference on Digital Libraries*, Philadelphia, PA, 131-140.

Marshall, C. C. and Shipman, F. M. 1995. Spatial hypertext: Designing for change. *Commun. ACM* 38(8): 88-97.

Marshall, C. C. and Shipman, F. M. 1997. Spatial hypertext and the practice of information triage. *In Proceedings of the Eighth ACM Conference on Hypertext*, ACM Press. Southampton, United Kingdom, 124-133.

Monsell, S. 2003. Task switching. *Trends in Cognitive Sciences.* 7(3): 134-140.

Neerincx, M. A., J. Lindenberg and S. Pemberton 2001. Support concepts for Web navigation: A cognitive engineering approach. *In Proceedings Tenth World Wide Web Conference*, Hong Kong, 119-128.

Paek, T., Dumais, S., and Logan, R. 2004. WaveLens: A new view onto Internet search results. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press. Vienna, Austria, 727-734.

Price, M. N., Golovchinsky, G., and Schilit, B. N. 1998. Linking by inking: Trailblazing in a paper-like hypertext. *In Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia*, ACM Press. Pittsburgh, PA, 30-39.

Price, M. N., Schilit, B. N., and Golovchinsky, G. 1998. XLibris: The active reading machine. *In CHI 98 Conference Summary on Human Factors in Computing Systems*, ACM Press. Los Angeles, CA, 22-23.

Ravasio, P., Sch, S. G., and Krueger, H. 2004. In pursuit of desktop evolution: User problems and practices with modern desktop systems. *ACM Trans. Comput.-Hum. Interact.* 11(2): 156-180.

Rennison, E. 1994. Galaxy of news: An approach to visualizing and understanding expansive news landscapes. *In Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology*, ACM Press. Marina del Rey, CA, 3-12.

Robertson, G., Czerwinski, M., Larson, K., Robbins, D. C, Thiel, D. et al. 1998. Data mountain: Using spatial memory for document management. *In Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, ACM Press. San Francisco, CA, 153-162.

Salton, G., Wong, A., and Yang, C. S. 1975. A vector space model for automatic indexing. *Communications of the ACM*. 18(11): 613-620.

Schilit, B. N., Golovchinsky, G., and Price, M. N. 1998. Beyond paper: Supporting active reading with free form digital ink annotations. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press/Addison-Wesley Publishing Co. Los Angeles, CA, 249-256.

Schilit, B. N., Price, M. N., and Golovchinsky, G. 1998. Digital library information appliances. *In Proceedings of the Third ACM Conference on Digital Libraries*, ACM Press. Pittsburgh, PA, 217-226.

Shipman, F., Price, M. N., Marshall, C. C., and Golovchinsky, G. 2003. Identifying useful passages in documents based on annotation patterns. *In Proceedings of the 2003 European Conference on Digital Libraries*, Trondheim, Norway, 101-112.

Shipman, F. M., Hsieh, H., Maloor, P., and Moore, J. M. 2001. The visual knowledge builder: A second generation spatial hypertext. *In Proceedings of the Twelfth ACM Conference on Hypertext and Hypermedia*, Århus, Denmark, 113-122.

Shipman, F. M., Hsieh, H., Moore, J. M., and Zacchi, A. 2004. Supporting personal collections across digital libraries in spatial hypertext. *In Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, ACM Press. Tuscon, AZ, 358-367.

Wilson, M. 2006. Annozilla (Annotea on Mozilla). from http://annozilla.mozdev.org (accessed on 2006).

Wolfe, J. L. 2000. Effects of annotations on student readers and writers. *In Proceedings of the Fifth ACM Conference on Digital Libraries*, San Antonio, TX, 19-26.

Woodruff, A., Faulring, A., Rosenholtz, R., Morrsion, J., and Pirolli, P. 2001. Using thumbnails to search the Web. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press. Seattle, WA, 198-205.

Wynblatt, M. and Benson, D. 1998. Web page caricatures: Multimedia summaries for WWW documents. *In Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, Austin, TX, 194–199.

VITA

Name:	Soon Il Bae
Address:	Department of Computer Science, Texas A&M University TAMU 3112 College Station, TX 77843-3112
Email Address:	soonil@cs.tamu.edu
Education:	B.S., Computer Science, Yonsei University, Korea, 1990 M.S., Computer Science, Yonsei University, Korea, 1992 Ph.D., Computer Science, Texas A&M University, 2008