User Interfaces for Information Retrieval on the WWW

Gary Marchionini
University of North Carolina at Chapel Hill, USA

INFORUM 2005: 11th Conference on Professional Information Resources
Prague, May 24-26, 2005

Information seeking has become increasingly interactive as tools and services on the WWW have evolved. Thus, there is more to searching than typing in a query and waiting for the search engine to display a set of possible webpages. In this paper, I argue that the only way to achieve substantial advances in search and browse capabilities is to combine research and development in human-computer interaction with research and development in information retrieval to create highly interactive systems that engage the user in defining their needs iteratively and going beyond retrieval to understanding the corpus and the retrieved information. This human-computer information retrieval (HCIR) perspective is the basis for various designs and will be illustrated with examples from the Open Video Project, a digital video library for the education and research communities (www.open-video.org), and the Govstat Project, an effort to design interfaces, information architectures, and online help for non-specialists looking for statistical information from government websites (www.ils.unc.edu/govstat). Our experience demonstrates how good interface design and usability testing leads to improved information services.

Traditionally, information retrieval has been approached as a problem of matching queries to documents. Documents are represented as sets of words and collections as large term-document matrices or inverted indexes. Queries are then similarly represented and a similarity function applied. This approach to information retrieval has been advanced in the WWW mainly by adding new kinds of representations such as the links between webpages and similarity metrics based on these links. However, the user interfaces have remained basically the same—either people type a query made up of a few words or they make selections from hierarchical menu structures. Marchionini (1995) calls the former 'analytical' search and the latter query style one form of 'browsing.' Increasingly, system designers are looking for new ways to improve user interfaces for WWW search services to get beyond the query and ranked list of results displays typical in search engines. This new paradigm in HCIR design is based on bringing the human information seeker more actively into the retrieval process. In this new HCIR paradigm information retrieval becomes human-centered, engaging an active human with information ***needs***, information ***skills***, powerful IR ***resources***, and situated in global and local connected ***communities***, all of which ***evolve*** over time. I term this close coupling of people and information 'syminforosis' to mean that people are continuously and inextricably engaged with meaningful information as part of their day-to-day lives.

WWW trends toward this perspective are perceptible in search engines such as Clusty, which presents results as clusters of results rather than a single list. The idea is to get people closer to the information they need by giving them more control over the results partitions. This implies that people must take increasing responsibility and control for their activities. The key challenges this trend poses are in linking the conceptual interface to the system backend (e.g., alternative representations and control mechanisms, metadata generation); raising user literacy and involvement without insulting or annoying; and moving beyond retrieval to understanding. I view these as primarily user interface challenges.

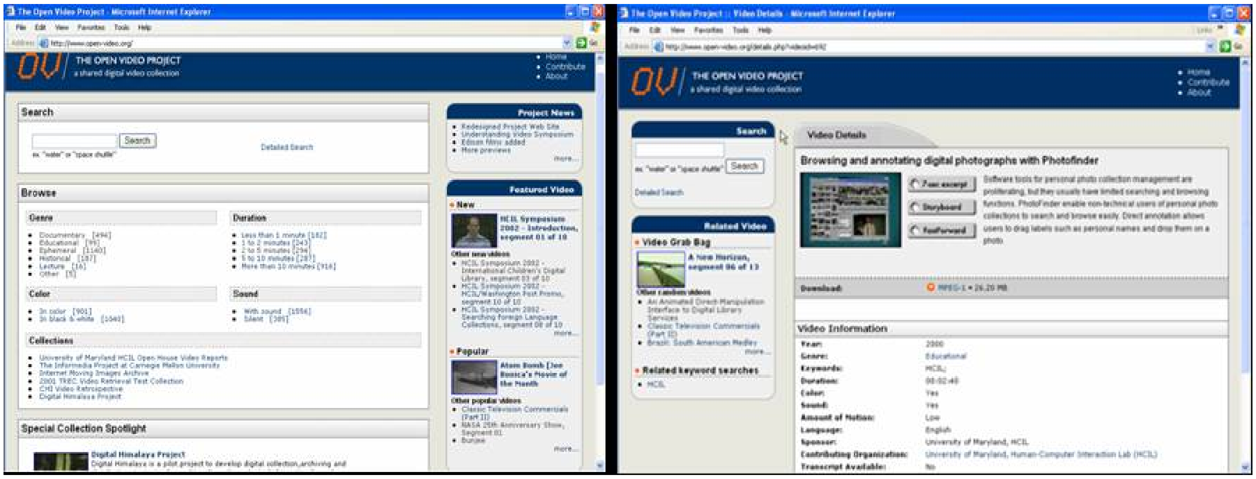
HCI Principles for HCIR

Human-computer interaction researchers have adopted Shneiderman's direct manipulation paradigm (Shneiderman, 1983) for many kinds of graphical user interfaces. Direct manipulation provides immediate feedback for physical actions and supports easily reversible actions that promote exploration without penalty. For information retrieval applications, Sheiderman and his colleagues developed dynamic query interfaces that allow people to pose queries by simply moving the mouse and immediately seeing results (Shneiderman, 1994; Ahlberg & Shneiderman, 1994). Marchionini and his colleagues have adopted these principles in the agileviews design framework (Marchionini et al., 2000). Agileviews aims to give people alternative views of information spaces and easy-to-use control mechanisms for shifting the focus of these views. Overviews display the entire collection of a partition of the collection; previews display an abstract or surrogate for a specific information object; reviews display histories of a collection or object; and shared views display the collection or object from the perspective of another user or a user community. At any given time, one of the views is in focus and the others are in the periphery (peripheral views). The control mechanisms for shifting among these views are easy to apply and reverse and in most of our work to date have involved mouse over, mouse down, mouse drag (e.g., on a slider), and mouse click actions as the control mechanisms for shifting between these views. The agileviews design framework has been applied in several projects, two of which are described below.

The Open Video Digital Library

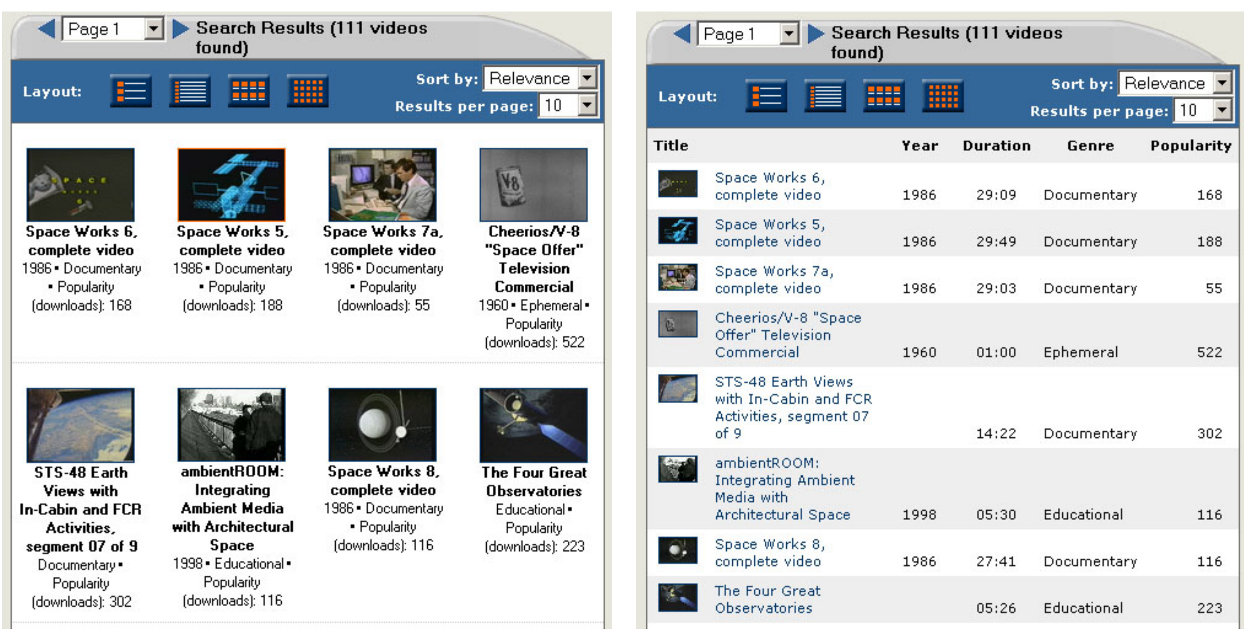
Open Video (www.open-video.org) is an open access digital library of digital video for education and research (see Geisler et al., 2001 and Marchionini & Geisler, 2002 for overviews of the system and design principles). The system used open source tools (MySQL database for metadata, Apache server, PHP middleware) and has become an important resource for educators and researchers around the world. There currently are more than 2500 video segments available as MPEG1 files with many segments also having MPEG-2, MPEG-4, and QuickTime versions. For each video, we provide multiple visual surrogates in addition to the standard textual bibliographic record (storyboard, fast forward, seven-second excerpt). The digital library received more than 15,000 unique visitors each month. Figures 1 and 2 illustrate how the agileviews design framework has been applied. The system homepage (Figure 1 left) provides different views of the entire database in different partitions. For example we can see the database partitioned by genre, duration, and by special collections. Additionally, on the right, there are two shared views in the periphery that show new videos added recently and videos that

Figure 1. Database overview (left) and Preview page (right)



have been downloaded many times. Figure 2 shows two alternative views of results. On the left, the emphasis is on visual elements and on the right a compact list with a more typical results layout. Any of these views can be changed by clicking the layout icon and can be sorted by year, duration, popularity, title, or relevance (the default); and the user can control how many results to display per page. These examples illustrate how people are given different views and can take active control over how they slice and dice (partition) the database and view those partitions or individual records. Over a four year period we conducted several user studies focused on various aspects of these designs in order to ground our design decisions in empirical data. For example, we did extensive testing of the fast forward speeds before settling on 64-fold speed ups for Open Video (Wildemuth et al, 2003).

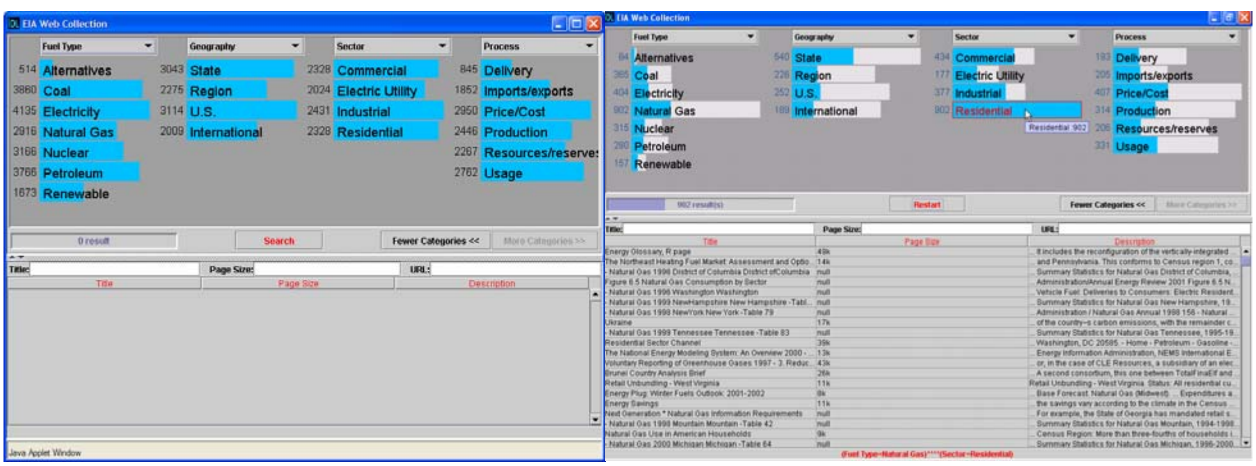
Figure 2. Alternative Overviews of results sets.



The Relation Browser

Over a period of seven years, we have been developing a general purpose user interface that allows people to easily define and explore pairwise and n-ary relationships among different partitions of databases. The interface is called the Relation Browser (RB) and it has gone through several iterations of design, user testing, and application. It has mainly been applied to several US government statistical agency websites in the past three years. The main characteristics of RB are that it is a general purpose dynamic query interface for databases with a small number of facets (~10) and a small number of categories in each facet (~10); it provides overviews and previews as 'look aheads' so that people can get good ideas about what will happen before they follow a hyperlink; it couples interactive partitioning/exploration with string query; and it has the capability to leverage semi-automatic category generation and webpage classification techniques from machine learning to automate some of the facet and category analysis. See Zhang & Marchionini (2005) for a description and report of a user study that demonstrates some of the RB advantages over standard WWW query interfaces. Figure 3 shows two displays of RB at different points in an exploratory session with data from the Energy Information Administration website. The panel on the left shows the view of the entire 15000+ webpages in this website partitioned by fuel type, geography, energy sector, and energy process. As the user moves the mouse over any of the categories, the display immediately updates to show relationships. For example, if the user mouses over the coal category, the blue bars and numbers displaying how many webpages are pertinent in geography, sector, and process would immediately be updated. The right side of the figure shows the display when natural gas has been selected and the residential sector highlighted. The pertinent webpages are displayed in the results panel below (each result is a link to that page). The user can further explore by typing terms into the query boxes in the results display. Subsequent results are closely coupled to the browsing panel above that immediately updates as new results are displayed.

Figure 3. Relation Browser Screen Displays.



The RB is driven by several principles that instantiate the agileviews design framework, including:

- Look ahead without penalty
- Minimize scrolling and clicking
- Provide alternative ways to slice and dice
- Closely couple search, browse, and examine

- Foster continuous engagement—provide useful attractors
- Bring treasures to surface

Over the years, the RB has been applied to more than three dozen kinds of databases (see http://idl.ils.unc.edu/rave/examples.html for some of these examples). Although the RB can be applied to a variety of database problems, it has two main limitations that determine how it is applied. First, there is a scalability issue; all the metadata must be sent to the client side to obtain the dynamics. Secondly, metadata is crucial to RB applications and because it is expensive to obtain manually, we are working on automatically creating partitions by applying machine learning techniques.

These examples illustrate how user interface research and development are improving information retrieval in the WWW. However, there is a long way to go as increasing portions of the world's population use the WWW with various kinds of connectivity ranging from dial up to high-speed broadband and with platforms ranging from desktops to cell phones. Thus, universal access and more alternative interfaces are important considerations as we move toward effective human-computer information retrieval.

References

Ahlberg, C. and Shneiderman, B. (1994). Visual information seeking: Tight coupling of dynamic query filters with starfield displays. *Proc. of ACM CHI94*, 313-317.

Geisler, G., Marchionini, G., Nelson, M., Spinks, R., & Yang, M. (2001). Interface concepts for the Open Video Project. *Proceedings of the 2001 ASIST Annual Meeting, 38*, 58-75.

Marchionini, G. (1995). *Information Seeking in Electronic Environments.* Cambridge, England: Cambridge University Press.

Marchionini, G., & Geisler, G. (2002). The Open Video digital library. *D-Lib Magazine*, *8*(12). http://www.dlib.org/dlib/december02/marchionini/12marchionini.html.

Marchionini, G., Geisler, G., & Brunk, B. (2000). Agileviews: A Human-Centered Framework for Interfaces to Information Spaces. *Proceedings of the Annual Meeting of the American Society for Information Science (Chicago, Nov. 12-16, 2000)*, 271-280.

Shneiderman, B., (1983). Direct manipulation: A step beyond programming languages, *IEEE Computer 16*(8), 57-69.

Shneiderman, B. (1994). Dynamic queries for visual information seeking, *IEEE Software*, 11(6), 70-77.

Wildemuth, B. M., Marchionini, G., Yang, M., Geisler, G., Wilkens, T., Hughes, A., & Gruss, R. (2003). How fast is too fast? Evaluating fast forward surrogates for digital video. Paper presented at the *ACM/IEEE Joint Conference on Digital Libraries*, (Houston, May 2003). NY: ACM Press.

Zhang, J. & Marchionini, G. (2005). Evaluation and evolution of a browser and search interface: The Relation Browser. *Proceedings of Digital Government 2005: The sixth national conference on digital government research* (Atlanta, GA May 15-18, 2005), NY: ACM Press. 179-188.