appeared in *Proc. of the ACM Conf. on Electronic Commerce*, ACM Press, pg. 41–7, Nov. 1999

Opportunistic Exploration of Large Consumer Product Spaces

Doug Bryan and Anatole Gershman
Accenture, Center for Strategic Technology Research (CSTaR®)
3773 Willow Road
Northbrook, IL 60062 USA
1-847-714-3015
Douglas.Bryan@accenture.com

ABSTRACT

The advent of the Web has brought an unprecedented amount of information together with a large, diverse set of users. Online users are performing a wider variety of tasks than ever before. For example, not only is the Web being used to search conventional databases like Lexis/Nexus, it is also being used to broker Beanie Babies®. Today's common information seeking metaphors (i.e., keyword search and hypertext) cannot be expected to support all these new tasks well.

We characterize a new user behavior called *opportunistic exploration*. We show how it is significantly different than both browsing and searching. A novel visual metaphor for opportunistic exploration, an aquarium, is presented. In an aquarium users may explore a large corpus at any level of granularity. The aquarium's implementation is discussed and demonstrated on a collection of 12,000 consumer products. The implementation automatically controls granularity based on the history of operations performed by a user.

Keywords

Information visualization, information retrieval, visual navigation, visual metaphor, browsing, searching, retail eCommerce, online shopping

1. INTRODUCTION

You have to buy a wedding present for cousin Edith, again. You're not sure what to get; you don't know her too well. You want something classy, to reflect your good taste; something a bit unusual but not too unusual. Maybe red, Czech crystal, or a ceramic picture frame. You will know it when you see it. While you are shopping for cousin Edith, you also buy a shirt and an audio CD for yourself, though you didn't plan to. You began the shopping trip with an ill-defined goal: something classy and unusual. Other interests (shirts and music) arose during the course of the trip.

Compare this behavior with what is currently supported by today's online stores. Keyword search cannot be used to find "classy" and "unusual" items. Even if items were tagged with these words, the items probably won't be what *you* consider classy and unusual.

Today's online stores are well suited for finding information that can be specified in a common vocabulary (e.g., boy's Schwinn bicycle). That is, they are good for finding items if you know what items you are looking for. This is not shopping; this is information retrieval and order entry. Nearly 50% of Americans consider shopping a recreation, not a task [9]. In reality, shoppers often do not know what they are looking for.

People have multiple, overlapping interests. When they go shopping, one interest is often initially given a high importance. For example, an interest in Edith's wedding may prompt a trip to the mall. The primary goal may be a wedding present, yet many other interests are still present, ranging from short-term (e.g., need more shirts), to long-term (e.g., tastes in music), to demographic (e.g., feed and clothe children). To exploit this fact, retailers arrange shelves and store layout so that shoppers are exposed to many interesting products. Over a century ago the Chicago retailer Marshall Fields recognized this when he said that he wanted to sell people things that they didn't know existed ten minutes earlier. He was appealing to their multiple, ill-defined interests, rather than their immediate goals.

We call the type of behavior exemplified by shoppers *opportunistic exploration*. The goal of our research is to develop new metaphors which support opportunistic exploration online. In this paper we use retail shopping as an example domain. In the next section we characterize opportunistic exploration. Subsequent sections present a novel interface metaphor which supports opportunistic exploration, and the underlying algorithms which govern our implementation of the metaphor. In the discussion section we differentiate opportunistic exploration from the two best known online behaviors: browsing and searching.

2. OPPORTUNISTIC EXPLORATION

The main characteristics of opportunistic exploration are:

- Users have multiple, overlapping interests.
- Users view many diverse items but examine few in detail.
- Exposure to items affects interests. A latent interests may be activated when users are exposed to items which appeal to that interest. Similarly, an active interest may be subdued if users are not soon exposed to relevant items.

Figure 1: A still shot of an aquarium of consumer products.

- Interests may change suddenly due to exposure or whim.
- Navigation must be simple. The casual nature of opportunistic exploration requires it to demand little effort on the part of users.

Overall, to be sustained or repeated, opportunistic exploration must be informative and enjoyable. As a recent study of online stores put it, diligence is not a virtue retailers should expect from their customers [11]. If it isn't interesting and enjoyable, customers will leave.

A day at an amusement park is an example of this behavior. During the course of the day a visitor would like something interesting to eat, have fun, get scared, and maybe buy a funny hat. There is no specific goal, just general interests on the part of the visitor, and opportunities presented by the park.

3. THE AQUARIUM METAPHOR

In this section we describe our new visual metaphor for opportunistic exploration.

Imagine a dozen interesting products floating all around you. The products move slowly, almost randomly, like fish in an aquarium. Occasionally some products leave and new ones appear. Users may passively watch the aquarium change, or they may interact with it. (See Figure 1.)

Users interact with a product by performing a positive or a negative operation on it. A positive operation changes the aquarium to contain more items like the one selected (i.e., more like this). The change is gradual, so as not to disorient users. New products come to the user as the user watches. A negative operation on a product results in less like that selected (i.e., less like this).

Users may also interact with the aquarium as a whole. A positive operation on the aquarium changes it to contain different products which are similar to those it currently contains (i.e., more like these). A negative operation changes it to different products which are unlike the current ones (i.e., less like these). If no operation is performed for a period of time, then the aquarium gradually changes by itself to show a diversity of products.

With this metaphor, there is no complex information structure for users to understand. Users need only be concerned with the small set of products currently on display. Cognitive overhead is very low. Products come to the user, rather than the user going to the products. Products and categories do not have a "location," thus users never ask questions like, "Are boy's mitts in sporting goods or toys?", or "Is the toy department to my left or my right?"

3.1 Governing Parameters

A small set of parameters govern the aquarium. *Change period* and *add rate* determine how fast the set of products in the aquarium changes. Change period is the maximum amount of time between operations. If a user does not perform an operation within this period, then the aquarium changes automatically. The add rate is the time between the introduction of new products. For example, if a user operation results in the introduction of two new products, then add rate determines how quickly one appears after the other.

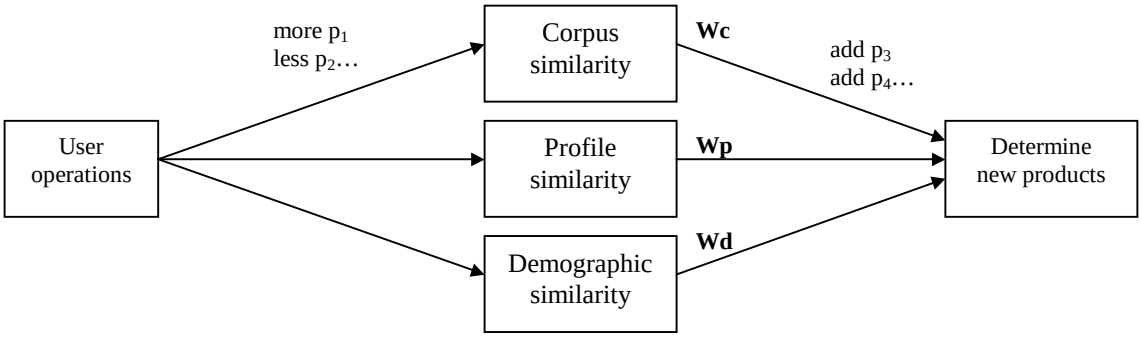

Figure 2: Competing, weighted relevance measures.

Each product in the aquarium simultaneously represents an instance (i.e., the product) and multiple categories. A Ken Griffy Jr. baseball mitt represents baseball, team sports, ball sports, and sporting goods. A 5"x7" crystal picture frame represents crystal, pictures, and frames.

The *breadth* parameter governs the meaning of "more" and "less" in user operations. Large breadth causes the aquarium to look for large categories (i.e., categories containing many products), while small breadth causes it to look for small categories. For example, a positive operation on a baseball mitt while breadth is high results in more sporting goods, while the same operation while breadth is low results in more baseball mitts. We have developed an algorithm for automatically regulating breadth based on a user's operation history. It is presented in the next section.

The core problem in governing the aquarium is product similarity. What does it mean for two products to be similar? In our metaphor, similarity is governed by three, competing measures:

1. Corpus similarity measures similarity between the descriptions of products. The set of products available to the aquarium defines a corpus. The corpus contains text descriptions and photographs of products. Corpus similarity applies conventional information retrieval techniques [10] to product descriptions to determine similarity. (See the implementation section for details.) Corpus similarity is the same for all users, but varies by product collection.
2. Profile similarity measures affinity between products and users. Buying history and past operations can be used to develop a profile of each user. Profiles estimate users' interests and the relative interests between products. For example, if I buy flowers once a year and audio CDs once a month, then I have a higher interest in CDs than in flowers. Profile similarity varies per user.
3. Demographic similarity measures affinity between products and demographic groups. Customer demographic databases estimate a user's interests based on simple statistics about the user (e.g., age, sex, marital status, address). The databases also estimate relative interests between products. For example, 60% of the male consumers interested in diapers are also interested in beer, but not vice versa. Demographic similarity is the same for large groups of users, but varies between demographic groups.

Each of these similarity measures is embodied in a separate product affinity engine and each engine is given a weight. Conceptually, the aquarium contains an array of product affinity engines, each competing for screen space. (See Figure 2.) The weights, Wc, Wp, and Wd, determine how much space each receives.

3.2 Support for Opportunistic Exploration

The aquarium metaphor is simple and yet powerful enough for opportunistic exploration of large consumer product spaces. The metaphor supports multiple, overlapping interests by considering the immediate interests based on the last few products selected (corpus), long-term interests (profile) and demographic interests. The absence of a prominent information structure and the low cognitive overhead of recognizing photographs, allow users to scan many products in a short period of time.

4. IMPLEMENTATION

This section presents our implementation of corpus similarity and the operations that users may perform. The main goals of our implementation are to:

1. Perform well on standard Wintel machines using off-the-shelf graphics hardware.
2. Change breadth quickly enough so that users may get to products of interest before their interest wanes.
3. Display a diverse collection of products so that users may quickly change interests.

These last two goals conflict. If we allow the user to move to a small collection too rapidly, then we lose the opportunity to display diverse products (e.g., cross-sell). If we display diverse products for too long, then the user cannot quickly reach a small collection of interest. Our approach is to automatically calculate breadth based on user operations, and to change breadth gradually. Initially breadth is very high. As a user makes choices, breadth may decrease and allow the user to focus on certain

$$\text{weight}(w) = (-1 + 2 * \text{breadth}) * \text{DF}(w) + (1 - \text{breadth})$$

products. This gives us time to display diverse products, while allowing the user to get where they want to go. If a user makes erratic choices, then breadth is increased.

Given breadth, changes in the aquarium become an information retrieval (IR) problem. The standard IR problem is, given a query q, to determine the set of documents which match q: $\{d \mid match(q,d) > t\}$, where t is some threshold.

Figure 3: An example product whose keyword set is: {clear, baby, crystal, picture, frame, frost}.

When queries and documents share a common representation (as in the vector model of IR [16]), then *match*(*q*,*d*) reduces to a similarity metric *sim*(*q*,*d*). Our implementation uses the vector model, and a similarity function based on dynamic keyword weights.

The free text of readily available product descriptions are reduced to sets of keywords (e.g., Figure 3). We use sets, rather than term frequency, because we found that the writing styles of retailers varies greatly. This caused frequencies to weight terms by retailer, rather than the nature of the product being described. We assume that users are more interested in the product than the retailer, so we use sets to remove the retailer bias.[1]

A thesaurus is used to augment and normalize keyword sets. We found that part of the thesaurus must be built manually. Terms used by retail marketeers simply are not available in commercial thesauri, and product descriptions are too short to allow automatic generation of a thesaurus. For example, in the jargon of retail apparel, a sneaker is a shoe and a skort is both a skirt and a pair of shorts. The product collection we are currently using contains 12,000 products from four vendors; 3,000 keywords describe the products. The keyword sets contain an average of nine keywords, with a standard deviation of 2.5.

4.1 Term Weights

The similarity function we use is the inverse of the absolute value of the difference in the sums of keyword weights:

$$sim(p,q) = \frac{1}{\left| \sum_{w \in p} weight(w) - \sum_{w \in q} weight(w) \right|}$$

Special considerations are given to zero sums and zero differences. Although the IR community has developed many similarity functions over the years, the main reason we use this one is because of its high speed.

The weight given to a keyword at run-time is a function of document frequency (DF) and breadth. The document frequency of a word is the number of products in a corpus that are described by the word. When breadth is high, weight is proportional to DF so that words with high DF, namely, large product categories, have high relevance. When breadth is low, weight is inversely proportional to DF so that small categories become most relevant. (See Figure 4.) The aquarium creates smooth changes in weights as breadth varies using the following function:[2]

DF is static per corpus. The problem of automatically supporting navigation through a product space then becomes determinining breadth. Our implementation of breadth is conceptually based on a 2.5D metaphor of a corpus [20]. In this metaphor, keywords are arranged on a 2D plane such that the distance between two words is proportional to the frequency that the words appear together in the same documents. That is, related words are close to each other. Altitude is then added to the 2D map such that altitude is proportional to DF. Large categories, denoted by frequent keywords, appear as peak and small categories, denoted by infrequent words, as valleys.

4.2 Automatic Determination of Breadth Based on User Moves

To determine breadth, we examine a user's past few moves. We take the products selected in positive operations, plot them on our 2.5D map, and examine the user's path. Breadth is then:

- inversely proportional to the degree to which the user is moving in a consistent direction,
- proportional to speed (i.e., 2D distance per move), and
- proportional to altitude.

These three factors are weighted to determine breadth:

$$\text{breadth} = w_1 * 1/\text{direction} + w_2 * \text{speed} + w_3 * \text{altitude}$$

We estimate direction as the number of words that recent moves have in common, speed as the number of words in the symmetric difference between consecutive moves, and altitude as the DF of common words. Let m_i be the keyword set of the product selected

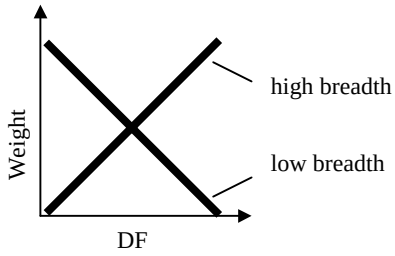

Figure 4: Keyword weight, document frequency and breadth.

[1] This assumption is weakened as companies begin to organize themselves around consumer activities, rather than products. For example, if you are interested in home gardening, then Smith & Hawken probably has many products of interest. See [6].

[2] Breadth and DF are normalized to be in [0,1].

in the i-th move, and *n* be the number of past moves to examine:[3]

$$common_i = \bigcap_{j=i-n}^{i} m_j$$

$$direction_i = |common_i|$$

$$speed_i = \sum_{j=i-n+1}^{i} |(m_j \cup m_{j-1}) - (m_j \cap m_{j-1})|$$

$$altitude_i = \sum_{w \in common_i} DF(w)$$

We have found that altitude is the best indicator of breadth, and thus weight it twice as much as direction and speed. Also, we have found that examining only three past moves (n=3) is sufficient. Large values of *n* do not allow for rapid changes in users' interests.

4.3 User Operations

The implementation of the aquarium includes the usual navigation operations: back (undo), forward (redo) and home. A keyword entry form is also provided to allow users to directly search for a given category. The main user operations are:

- more(p). Show products like p.
- much-more(p). Show products much more like p.
- less(p). Show products less like p.
- much-less(p). Show products much less like p.
- mix. Show different but similar products.

More(p), the positive user operation, is the only one that affects breadth. A user's "moves" are a sequence of more(p) operations. Much-more(p) is implemented simply as two consecutive more(p) operations.

The implementation of more(p) has two steps:

1. Find products similar to p, and
2. Display a diverse collection of these products.

The first step uses the keywords of p, and the weight and similarity functions defined earlier. The second step uses the similarity function to measure diversity. A maximum similarity threshold (*t*) is calculated. The threshold is inversely proportional to breadth. The product most similar to p is displayed. Then, the product q with the highest similarity to p such that the similarity between q and each product displayed does not exceed *t*, is displayed. This last step is repeated until the display is full.

Less(p) is implemented much like relevance feedback in information retrieval systems. The weights of the keywords of p are decreased a fixed amount and the set of products to display is recalculated. Much-less(p) is simply two consecutive less(p) operations.

The mix operation may be invoked by the user, and is automatically invoked if no operation is performed for a certain amount of time. This operation decrements the maximum similarity threshold, and then performs more(p) where p is the operand of the most recent positive operation.

Finally, users may directly manipulate breadth and keyword weights. The user interface includes a slider depicting breadth. The user need not change the slider, since breadth is automatically governed by past moves. However, users may directly set breadth using the slider.

The five keywords with the highest weights are arranged along the side of the screen. A word's distance from the bottom of the screen is proportional to its weight. Users may drag words up or down to directly set their weight.

5. DISCUSSION

When we began this work, we considered whether opportunistic exploration is more like browsing or searching. We concluded that it is fundamentally different than both, and thus worth investigating.

Searching is characterized by careful examination of items in pursuit of a goal. Most all online stores support search. Users must enter keywords which describe their goal, and have the system do the searching for them. Also, most online stores organize their products in a fixed hierarchy. This allows users to manually search for items by examining the categories and products in the hierarchy.

Browsing is the other common online behavior. When browsing, users move leisurely along a predefined path. For example, browsing a bookstore is to casually moves through the aisles, examining some small subset of the items available. Similarly, browsing the Web means following predefined hyperlink path, without carefully reading every page.

Both browsing and searching can be disorienting when navigating large product spaces. Searching is disorienting because users are never afforded a view of the space; rather, they jump from subset to subset via a search engine. Browsing, and specifically hypertext browsing, can be disorienting because of the sudden changes between pages [2]. Most online spaces, including stores, support both searching and browsing. However opportunistic exploration is not well-supported by either. Searching does not expose users to enough items, while browsing confines users to predefined paths. Lastly, both keyword search and hypertext browsing have a high cognitive overhead due to their textual nature.

Customer exposure to a variety of products is fundamental to retail. As commerce moves from physical retail spaces to virtual spaces, the goals of consumers and retailers largely remain the same while the constraints on approaches to satisfying these goals change drastically. Two major constraints on physical retail are shelf space and the cost of changing a store's layout. Both of these constraints are greatly relaxed in virtual stores.

Relaxing space constraints allows online stores to change their layout every day, for every customer, or even every few seconds. Shelf space is no longer constrained by walking speed. While large physical stores can effectively display about 50,000 items, we expect online stores to soon display millions of items from all over the world.[4]

[3] For clarity, the normalization of these values is not shown.

[4] Shopping.com and Netmarket.com already have a million items online.

Relaxing time constraints allows online stores to be used in ways that physical stores never were. A 3-minute virtual shopping trip, to see what is new from your favorite vendors, becomes possible. Online stores may also be browsed passively, while shoppers are engaged in other activities, much like one "watches" TV while reading a newspaper.

5.1 Related Work

For nearly 40 years researchers have recognized the Gestalt powers of users, and that many information seeking tasks will always be ill-defined [7]. Opportunistic exploration of large, online spaces can be viewed as an ill-defined information retrieval problem or as an information visualization [1] problem which relies on the Gestalt abilities of users. Historically information retrieval has focused on producing the few closest matches to a given query. That is, none of the underlying information structure is exposed for users to explore, and users have access to a very small subset of objects at a time. In effect, a keyword search reduces millions of documents to the few most relevant documents in seconds.

Information visualization, on the other hand, has focused on exposing large information structures that users can navigate in intuitive ways. Users have easy access to all objects in a collection. Information visualization abstracts information to the point where users can find patterns and get around on their own. We view our work as the best of both approaches, applied to a new problem. We use the robustness and scalability of statistics-based information retrieval to aid users in navigation, and the accessibility of information visualization to allow uses to navigate.

Rabbit [19] was an early, intelligent database assistant that aided users in formulating queries. As with our system, Rabbit assumed that users lacked expert knowledge of the corpus being used, and that users were performing ill-defined tasks. Users interactively constructed descriptions of a target instance by criticizing successive exemplars. While the goals and approach of Rabbit and our work are very similar, the implementations differ greatly. Rabbit instances and queries were general attribute-value pairs, while in our work instances are described by keyword sets (or sparse boolean vectors). Also our system automatically determines which attributes (keywords) are important whereas in Rabbit the user explicitly adds and removes attributes.

More recently there has been a good deal of work on systems for querying databases of digital images. Much of the work has been based on very low-level attributes of images, such as size and the mean brightness of pixels. The basic assumption here is that if, for example, a user is looking for an image of a sunset, then many sunset images will have similar attributes (i.e., the cluster hypothesis of IR). The work in this area that is most similar to ours is PicHunter [3]. PicHunter presents users with four images, the user selects zero or more of them, and then clicks "go" to get the next batch of images. The main differences between our aquarium and PicHunter are, (1) to measure similarity between images, we use text keywords that were automatically extracted from image descriptions, rather than low-level image attributes, (2) the basic user commands of PicHunter are slightly more complicated than those of the aquarium, and (3) our interface is not designed for searching; we assume that user tasks may remain ill-defined indefinitely.

Multi-dimensional scaling has recently been used to navigate small product spaces [18]. One difference between this work and ours is that they use a small set of dimensions crafted by a domain expert (e.g., design properties of in-line skates), whereas we use many dimensions (i.e., 3000 keywords) automatically extracted from text descriptions. Their goals and assumptions about user behavior, however, are very similar to ours.

The scatter/gather technique of information retrieval [5, 13] is similar to ours. The technique gathers text summaries of clusters of documents, and allows users to browse them at different levels of granularity. Scatter/gather supports the exploration of a topic structure to aid in refining an ill-defined problem. The main differences between scatter/gather and opportunistic exploration is that we assume that the problem will always be ill-defined, and that we automatically determine the level of granularity based on a history of very simple user operations.

Little work has been done on visualizing large product spaces. Much of what has been done is directly based on physical world metaphors (e.g., browsing music stores by creating virtual aisles, walls, doors, and record bins [12]). This approach adopts all of the disadvantages of the physical world without adopting any of the advantages of the online world. As one critic put it [8], "Most metaphor abuse online comes from reinventing the bad bits of the physical world simply for the sake of familiarity." Our aquarium metaphor is simple, yet takes full advantage of the capabilities of online spaces.

The main goals of our work are very similar to those of information landscapes [4, 14]:

- give users access to all information and allow them to find their own emergent structures,
- simultaneously support the best of searching and browsing in a single, simple user interface [15], and
- enable the journey through an information space to be meaningful [17].

Our work differs from information landscapes in that (1) we work with photographs while they work with text, (2) we never make explicit the relationships between items, even when viewing very small subsets of items, and (3) we bring items to the user while they allow the user to move to the items.

5.2 Future Work

We have discussed three kinds of similarity measures. We have yet to implement profile or demographic similarity. There are many other kinds of similarity to be considered as well; e.g., those based on consumer intentions such as gardening or moving [6].

Next, we would like to formalize our navigation algorithm and generalize it to real-valued dimensions such as price and size. We also need to define key performance metrics for opportunistic exploration so that different navigation algorithms can be evaluated on large corpora with many users.

We conducted an initial usability test on 12 users. Each user performed two ill-defined shopping tasks, one using the aquarium and one using Wal-Mart's Web site. While the results are largely inconclusive, we found that users are inclined to search, browse, or explore when shopping, and that different user interfaces have little affect on that inclination; i.e., shopping behavior may be a personality trait. We also found that users are uncomfortable with

only pictorial feedback from the system and strongly preferred to have the top set of keywords displayed as they shop. In fact, users of the aquarium often used its keyword search feature or directly manipulated word weights, rather than clicking on pictures. A good deal of user testing and interface refinement is needed if the aquarium is to be usable to a broad audience.

6. SUMMARY

We have characterized a new class of user behavior called opportunistic exploration and differentiated it from browsing and searching. We designed a novel visual metaphor, called an aquarium, which is well suited to opportunistic exploration. Lastly, we implemented an aquarium using information retrieval techniques and demonstrated its use on a collection of 12,000 consumer products.

7. REFERENCES

[1] Card, S.K., Mackinlay, J. The structure of the information visualization design space, in *Proc. Info. Vis. Symp. 97*, IEEE Comp. Soc. Press, October 1997, 92–9.

[2] Conklin, J. Hypertext: An introduction and survey, *IEEE Computer*, 20(9):17–41, September 1987.

[3] Cox, I.J., et al. PicHunter: Bayesian relevance feedback for image retrieval, in *Proc. Int. Conf. on Pattern Recognition*, IEEE Comp. Soc. Press, August 1996, 361–69. See also U.S. Patent 5,696,964.

[4] Cooper, M., et al. *Information landscapes*, MIT Technical Note. April 1994.

[5] Cutting, D., Karger, D., Pedersen, J., Tukey, J. Scatter/gather: A cluster-based approach to browsing large document collections, in *Proc. of SIGIR'92*, ACM Press, June 1992, 318–29.

[6] Doblin Group. Corporate identity: What's next, *Design Management Journal*, Winter 1996.

[7] Doyle, L.B. Semantic road maps for literature searchers. *Journal of the ACM*, 8(4):553–78, October 1961.

[8] *The Economist.* Ban cyberspace, June 21, 1997.

[9] Kirtland, K. M. The 1996 mall customer shopping patterns report, *Intl. Council of Shopping Centers Research Quarterly*, 3(4), 1996. See www.icsc.org.

[10] Kowalski, G. *Information Retrival Systems: Theory and Implementation*, Kluwer Academic Publishers, 1997.

[11] Lohse, G.L. and Spiller, P. Quantifying the effect of user interface design features on cyberstore traffic and sales, in *Proc. of CHI 98*, ACM Press, April 1998, 211–18. See also by the authors, Electronic Shopping, *Comm. of the ACM*, 41(7):81–7, July 1998.

[12] Massari, A., et al. Virgilio: A non-immersive VR system to browse multimedia databases, in *Proc. of the Conf. on Multimedia Computing and Systems*, IEEE Comp. Soc. Press, June 1997, 573–80.

[13] Pirolli, P., Schank, P., Hearst, M., and Diehl, C. Scatter/Gather browsing communicates the topic structure of a very large text collection, in *Proc. of CHI 96*, ACM Press, April 1996.

[14] Rennison, E. Galaxies of news: An approach to visualizing and understanding expansive news landscapes, in *Proc. of UIST 94*, ACM Press, November 1994.

[15] Rennison, E. Personalized Galaxies of information, in *CHI 95 Companion*, ACM Press, May 1995. See also www.Perspecta.com.

[16] Salton, G. Developments in automatic text retrieval, *Science*, 253:974–80, 1991.

[17] Small, D. Navigating large bodies of text, *IBM Systems Journal*, 35(3&4), 1996.

[18] Stappers, P.J. and Pasman, G. Exploring a database through interactive visualised similarity scaling, in *Proc. of CHI 99*, ACM Press, May 1999, 184–6.

[19] Tou, F.N., et al. Rabbit: An intelligent database assistant, in *Proc. of the National Conf. on Artificial Intelligence*, (AAAI-82), 1982, 314–18.

[20] Wise, J.A., et al. Visualizing the non-visual: Spatial analysis and interaction with information from text documents, in *Proc. Info. Vis. Symp. 95*, IEEE Comp. Soc. Press, October 1995, 51–8. See also www.-Cartia.com.