# A Nine Month Report on Progress Towards a Framework for Evaluating Advanced Search Interfaces considering Information Retrieval and Human Computer Interaction

Max L. Wilson

April 25, 2007

## 1 Introduction

This is a nine month progress report detailing my research into supporting users in their search for information, where the questions, results or even their environment requires functionality beyond the scope of keyword search alone. While keyword search environments, such as Google, have become the familiar standard for search on the web, research has shown that users, who may not know which keywords to use, adopt coping strategies because they have no other methods to express their query (Pirolli and Card, 1995). Typically, these coping strategies involve submitting tentative queries with general keywords to learn more about the meta-data, to inform a more specific keyword search later. More recently, the term *Exploratory Search* has been used to describe search environments that provide alternative search functionality for when users may have poorly defined goals or complex questions, have insufficient pre-search knowledge, or may be using a system with poorly defined or unpredictable indexing (White et al., 2006a).

While early exploratory search interfaces have been developing for some time (Hearst, 2000), recent reports have discussed the need to find metrics for evaluating their success (White et al., 2006b). As these more interactive models of search provide increasingly versatile combinations of functions, Marchionini suggests that such success is not achieved by simply adding more features but by combining them to produce synergetic designs(Marchionini, 2006). To evaluate exploratory search interfaces, designs need to be measured in terms of their support for known search tactics. The research below investigates the history of Information Retrieval (IR) research into the human element of interactive search, to develop a framework that produces a measure for such support. Then, the weaknesses of rich search designs can be identified and mitigated, before they are put forward for complex and expensive user studies[1].

This report describes the progress of this investigation into using Information-Seeking models of users, their needs and their search behaviour, to design a

[1]The search interaction stream of the Text Retrieval Conference was abandoned in 2002 due to the difficulty of comparing highly interactive systems between research institutions. The framework described in this paper is shown to provide means for such an evaluation.

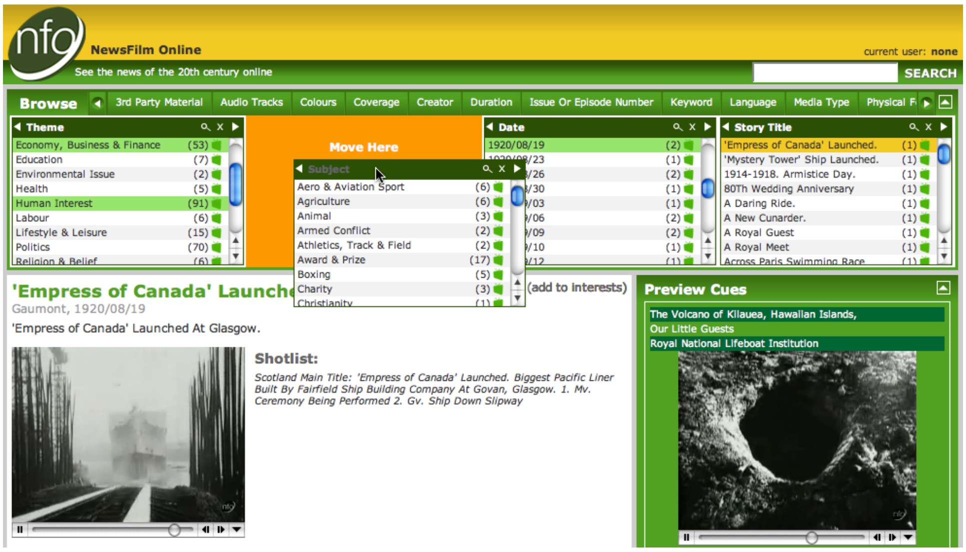



Figure 1: mSpace applied to the NewsFilm Online project dataset - in development

framework for motivating and validating designs of exploratory search interfaces. In Section 2, the motivation for such a design approach is supported by my previous research into search environments. Section 3 describes research into both advanced search environments and potential IR models that may be utilised in the evaluation framework. In Section 4, an initial framework is described and tested on three faceted browsers, as examples of more advanced search systems. The open research questions of the current framework are discussed and a plan for further research into supporting users is presented in Section 6.

# 2 Research Motivation

My interest in supporting search began in my undergraduate degree in computer science, which became focused on user interfaces and, primarily, on a search interface called mSpace. Subsequent to the end of my undergraduate, my research into mSpace and its development continued and are described in the following sections.

## 2.1 mSpace

The majority of my research, starting with my undergraduate dissertation, has involved research afforded by the development of an advanced search interface: mSpace[2]. Figure 1 shows the most recent application of mSpace to the NewsFilm Online (NfO) project[3], which provides moving footage of news broadcasts from the 20th Century for download.

[2]http://mspace.fm

[3]http://mspace.fm/projects/nfo/

mSpace provides a number of features to support different search methods. As stated above, it is designed to provide as much information and interaction persistently in view. Traditionally, mSpace has four main sections that can be plugged into a website to support search: a column browser, an information panel, a PC space and a collection space (now called the Interest Panel). These four sections, and the supplementary features, are discussed below.

The most familiar search method for most people, is the the standard Google style keyword-search[4]. mSpace provides a keyword search box, shown at the top right of Figure 1. However, this best serves people who know what they are looking for and what words to use to find it. The mSpace columns, seen across the top of the browser, are an alternative to this approach. Normally a keyword search scans the meta-data and core objects of an information system for the chosen keywords and ranks the matches. The column browser presents this meta-data to the user so that they can recognise and choose keywords rather than guess what they might be.

These columns are a visualisation of facets in a faceted browser (FB), which have become popular since the turn of the century (Hearst, 2000). A facet is one attribute of some meta-data and when selecting one instance from the facet, the set of system objects are filtered. For example, Google Directory is a category browser, where users select one category, and the set of links that can be found using the directory are reduced to the set covered by that category. Category is an example of one facet of the meta-data about the links found in the directory. Date of last modification might be another facet of web links. In a FB, when a selection in one facet is made, the choices in the other facets is reduced to those associated with the new subset of data. In a movie database, the information system object is the Movie and the facets may be Actor, Director and Year. If you select a certain year, the choices in the remaining facets will be limited to those actors and directors who made the movies in that year.

In mSpace, these facet columns create a hierarchy through the meta-data from left to right across the browser; we call this a slice. When the browser loads, all four facet columns are fully populated with all the options. If a user starts by clicking on *Education* in the first column, the columns to the right are filtered to show the subjects, dates and story titles of the stories associated with *Education*. However, the user may click on something in any column. By next selecting the 23rd of June 1920 in the date column, the columns to the right are filtered again: the user will see education stories from this date. However, any selections to the left that the user could have made are highlighted instead. That is, all the subjects associated with education clips from this date are highlighted. This presents the user with extra information with no extra action and further communicates the structure and relationships between the meta-data. This feature is currently called *Backward Highlighting*.

As the order of columns matters, mSpace provides easy tools to let the user change the order. Users may add, remove and reorder the columns through direct manipulation. To remove a column, they can click the **x**; this matches familiar software design of most operating systems. This column then gets listed with the set of optional facet columns found above the open columns. Any one of these can be added to the end of the columns by clicking on it. Alternatively, they may hold the mouse down on the chosen new column and drag it to a

[4] http://www.google.com

particular place in the slice. To move a column around, they can either click-hold-and-drag or they can use the left and right arrows, which will swap its position with its neighbour.

To help the user find items in a column, they can use the in-column filter. This filter can be opened by pressing the small magnified glass on each column. As a character is typed into this box, the list is filtered to only items that contain that character.

Each item in the columns has a number, currently this number shows the number of system objects (like movie in the example above) that can be found by making that selection. This topic is investigated further below in Section 2.2.3. Each item also has a PC icon; hovering over this icon will trigger a PC in the PC panel, seen below the columns and on the right in Figure 1.

The information panel is often a large part of an mSpace design, as it provides space for a large portion of content to be displayed about an item. For example, if an Actor is selected, information about that actor may be displayed in the information panel.

The final key section of the mSpace is the collection space. This supports information triage (Marshall and Frank M. Shipman, 1997) by allowing users to keep any item in any column for later. This is similar to the work done by schraefel et al. (2002), as smaller-than-page sized 'nuggets' of information can be stored. Users can double click on any item in the columns and it will be added to the Interest panel. Alternatively, users can drag individual items into the Interest panel.

Until recently, these panels were presented in a fixed space so that, at all times, each panel was accessible. This old version is shown in Figure 2. However user feedback showed that this was both unfamiliar for users and inflexible for close integration into existing websites. Figure 1 is an example of version 0.8, where these features can be plugged into a more familiar web page view. To support this familiar web style, two further panels are now available: Recommendations and Advanced Search. The recommendations can be set by the administrator and, when selected, show information about them in the information panel and also highlight their place in the browser columns. The advanced search allows users to enter keywords into boxes that represent different columns. Like the standard keyword search, results are displayed in the information panel. Each of these items can be selected to show more information and the items place in the columns.

## 2.2 mSpace Research

### 2.2.1 Preview Cues

My dissertation, summarised by schraefel et al. (2004), researched the notion of Preview Cues (PCs) further by investigating the extend to which PCs should be provided. The hypothesis was that more PCs would be improve the search experience for users. To test this, two versions of the mSpace interface, at that point in development, were made. One had a single preview cue as before and the other had three preview cues, shown in Figure 3.

The users were given tasks to find four songs that they liked and put them in their shopping basket (the metaphor used for the Interest panel). Participants were measured for time taken, number of PCs triggered, number of clicks and

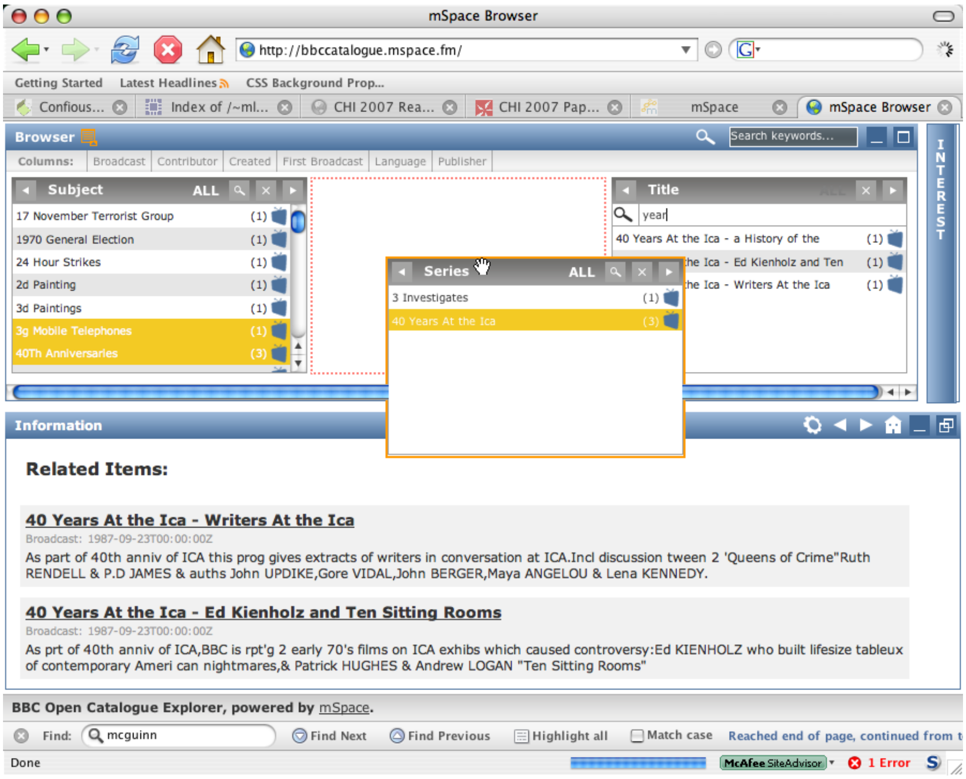



Figure 2: mSpace version 0.6

number of *back-outs*. A back-out is a sequence of user actions and is supposed to be reduced by PCs: when a user goes back on a search decision. This can be measured directly by the number of times they select an item in a column further left than their previous selection, unless after adding a song to their shopping basket. Thus by improving the search experience, we mean: less back-outs, less clicks overall and less time required to find the songs.

Results were both mixed and insignificant almost across the board. Firs this indicates that a single PC is statistically just as useful overall as having multiple PCs. We did, however, notice a difference in search patterns of users. In order to discover the real effect of multiple PCs, we had to consider the results in more depth than the per-participant measurements that we originally collected. Triggered this pattern difference and by the comments of users in the post-task interviews, we discovered that our hypothesis became more applicable under two conditions. First, if a user was knowledgeable about the domain, they discarded the PCs altogether and browsed by the keywords they recognised. Second, more of the multiple PCs were triggered in the columns further to the left. Thus we concluded, through qualitative analysis, that more PCs are required in larger and less familiar spaces. Subsequently, for example, more PCs should be made available when representing entire periods of classical music than when representing cello pieces by a particular composer.

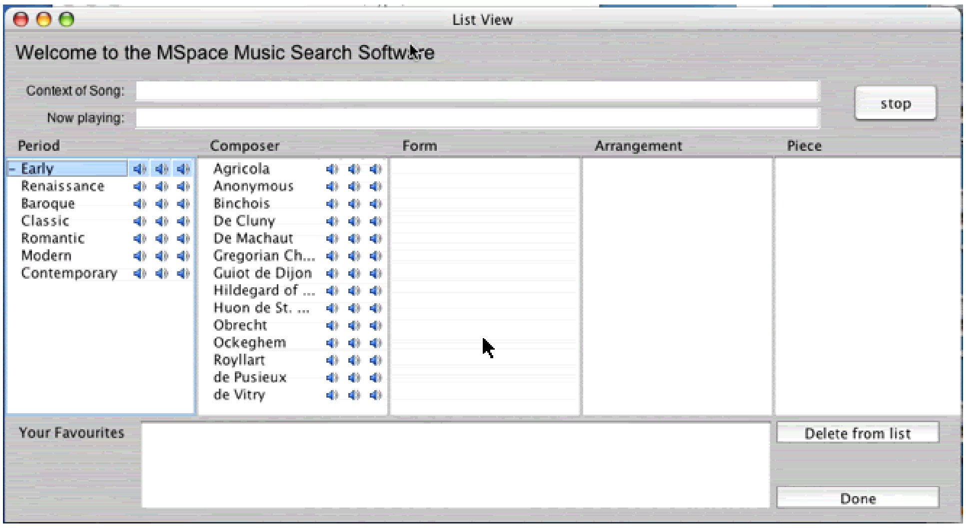



Figure 3: Old mSpace Interface with three preview cues available

### 2.2.2 mSpace Mobile and Mobility

As the research into search conditions and situations develops, environment has been identified as a key aspect of affecting user search (Pharo, 2004). Mobile computing is a key example of this. Mobile access to information has many constraints on search systems and the most researched of this is screen space (Forman and Zahorjan, 1994). However there are many other aspects, including tapping accuracy and the effect of movement (MacKay et al., 2005; Mizobuchi et al., 2005). Consequently, we investigated the support that mSpace Mobile provided for people users are on the move (Wilson et al., 2006). Our hypothesis was that the direct manipulation interface would support mobile users, who would otherwise be using keyword services such as Google Local; Google Local is a website that has been formatted for small screen use. Keyword searching is not a direct manipulation approach, as it involves text entry on otherwise static pages, which provide links and buttons to carry out actions.

We took nine participants through all four conditions and counter-balanced them to remove learning effect. Four scenarios were generated, each with equal tasks that could be achieved with each interface. These tasks grew progressively harder throughout the scenario. Users were given identical mobile devices and were asked to walk around an indoor course. The reasoning for this was two-fold. First, it allowed us to control safety concerns; we did not want to be liable for road accidents while participants were looking at our mobile devices. Second, it allowed us to provide constant wireless. Third, it afforded more opportunities to record information.

Previous publications showed that mSpace provides an interface that well supports search, so here we were concerned with how search is affected by a mobile environment. We showed that mSpace was significantly faster for users to carry out the scenarios than when they had to use Google Local, in both mobile and stationary conditions. mSpace mobile did not take significantly longer in the mobile condition than when stationary, which is a positive result.

This indicates that mSpace Mobile is just as easy to use sitting down as it is walking around. Sadly, there was also no significant difference between Google Local in the mobile and stationary conditions. However, the difference was larger for Google Local than mSpace Mobile.

mSpace provides a new data-transfer paradigm for the web, which is particularly useful for mobile technology. Instead of transferring entire pages, where the page is the smallest item, mSpace affords the transfer of data-subsets, so that only the changes need to be transfered over the network. As mobile computing often uses low-bandwidth communication, such as 3g or GPRS, this paradigm is particularly useful.

### 2.2.3 Multi-targeted Environments

mSpace can be described as supporting multi-targeted information environments, something that is becoming more frequent as the Semantic Web develops. A TO is the focus of a dataset, around which the meta-data is collected, such as: a movie, a publication document, an audio file. In the Semantic Web vision (Berners-Lee et al., 2001), information is connected through relationships, which are defined in a data schema document. Semantically stored information will then be interconnected to the point where fixed datasets are only bounded by their schemata but are actually part of a global information resource. Then, when fixed datasets were oriented around single Target Objects (TOs), they become part of a larger multi-targeted environment. Imagine a movie dataset, where the movie is the TO. The actors, directors and countries could all be TOs of other datasets. Further, they are all related, so the movie dataset might make up part of the search for Actors. As mSpace allows for the reordering of facet columns, any object, such as an Actor, can be set as the focus. This challenges many key assumptions in user search and interface design. With no particular TO in a dataset, faceted browsing can be reconsidered.

Numerical Value Indicators (NVIs) are a standard design feature for targeted search environments; they count the number of TOs available under specific selections. When there are multiple TOs, what do NVIs represent? They could represent the TO of a dataset, defined by provenance. They could be dynamic to represent a user TO choice. Finally, with no specific, they could simply be used to indicated the breadth of the choices at the next level. These were our three hypotheses (Wilson and schraefel, 2006).

To test this, we set up an mSpace layout, showing a movie dataset, but having director as the last column rather than movie. This was designed so that each of the three hypotheses were possible. We then created three more conditions that could potentially indicate an answer using small visual cues. For example, one condition suggested that the third 'next level' hypothesis was wrong, because the number of items in the column is clearly less than the NVI of the previous selection.

Our results were that, despite the conditions, the majority of people expected the numbers to still count movies, as the focus of the domain. In the reality of the Semantic Web, the may well be deduced by provenance and the focus of its contribution. Interestingly, no one in the study thought the NVIs were counting the number of directors. Yet after discussion, each participant expressed the wish to control the object that the NVIs were counting and would be pleased to use it to their advantage. In interface condition three, it seemed possible that the NVI could be representing the breadth hypothesis, because only part of the next column could be seen. This lead to many more participants selecting the breadth hypothesis, and was reduced slightly when this column count was explicitly labelled.

There is clearly scope for these design standards to be re-examined for the future interconnectivity of the Semantic Web.

## 2.3 Research Direction: Exploratory Search

Within each of these studies, the motivational question always has a *When* statement. mSpace is significantly motivated by the support for users *when* they do not much about a domain. PCs are useful, for example, *when* participants could not translate the keywords available into a decision. The direct manipulation and mSpace paradigm is advantageous *when* users are interacting with a mobile device and perhaps on the move. NVIs were investigated for *when* users can search for any object in a semantic graph.

At the International Workshop on Semantic Web User Interaction (SWUI) 2005, one of the break out discussion groups was dedicated to discussing the benefits of different graph visualisations. After much discussion between the group, the answer was summarised as *For what?*. This was perhaps summarised cleverly by schraefel and Karger (2006), which discusses this issue of misrepresentation through graph visualisations. Essentially, *when* is a graph visualisation useful? Further, other research is beginning to investigate this *when* clause in their publications. Wang and Parsia (2006) noted that Crop Circles, an Ontology Visualisation, was stronger than other visualisations *when* performing Topological tasks: finding deep subtrees, comparing sub-tree shapes, finding broad subtrees in a graph, etc.

This common *when* attribute to this continuing research has become communally known as *Exploratory Search* (ES), where ES Interfaces (ESI) are being developed to support users who frequently have *when* conditions. ES has become increasingly popular and has been discussed in three workshops[5] (two in internationally renowned conferences: SIGIR2006 and CHI2007) and was the core theme of the Communications of the ACM, April 2006. In the Guest Editors' introduction, White et al. (2006a) reference mSpace and the Classical Music scenario in their description. In the report of the original workshop, this is used as an example of exploratory search, specifically stating: *Example taken from http://www.mspace.fm , a site describing the mSpace exploratory search browser.* The following quote is taken from the homepage of the CHI2007 Exploratory Search Workshop.

> Search engines, bibliographic databases, and digital libraries provide adequate support for users whose information needs are well defined. However, there are research and development opportunities to improve current search interfaces so users can succeed more often in situation *when*: they lack the knowledge or contextual awareness to formulate queries or navigate complex information spaces, the search task requires browsing and exploration, or system indexing of available information is inadequate.

[5] http://research.microsoft.com/ ryenw/xsi/index.html, http://research.microsoft.com/ ryenw/eess/index.html, http://research.microsoft.com/ ryenw/esi/index.html

# 3 Related Work

The research into mSpace, to date, has shown that it clearly supports users in exploratory search conditions. However, the focus of work into exploratory search is now on evaluating the support for search they provide. Many search environments have been developed through research into digital libraries, information retrieval and human-computer interaction. But with so many options, we need to define *when* they are beneficial. In line with our own research, we want to discover *when* mSpace is strong and where it can be strengthened. Simultaneously, in the same edition of the CACM, Marchionini (2006) stresses that the development of advanced search functionality needs to be applied so that, in combination, they produce synergetic systems rather than confusing and complicated designs.

To investigate exploratory search for my PhD, related reading was performed in two areas: search environments and IR models of user search behaviour. This investigation has been the main activity for the last nine months, however Section 4 discusses the application of the research in an evaluation of three exploratory search systems. First, related work on search environments are presented, in terms of their motivations and solutions, and in the context research into exploratory search. Following this, related work on users, their needs, and their search behaviour is discussed in terms of their use for developing search requirements.

## 3.1 Search Environments

### 3.1.1 Exploratory Search

In the Communications of ACM issue on ES, while we wrote about the mSpace ESI (schraefel et al., 2006), Marchionini (2006) began to formalise and define what ES is. He suggests that the Web has encouraged very interactive and iterative browsing strategies, but require search tactics such as trial-and-error, navigation and selection. He goes on to suggest a more user-interactive view of search, requiring both HCI and IR fields and labels this Human-Computer Information Retrieval (HCIR). With this, he proposes an overview of ES, shown in Figure 4, which highlights a series of more complex Learn and Investigate ES activities such as: Comparison, Aggregation, Synthesis, Transformation etc.

In further discussion with Marchionini, the activities in Figure 4 are emergent terms from Information Science and are rarely defined in formal terms; or even in common terms. With the introduction to these terms, Marchionini cites a reference to Saracevic (1997) for a good summary of formal theoretical models of information search. This small move towards formalising what ES actually subsumes, suggested that the Information Retrieval research field was a good place to investigate models of users, user needs and search behaviour. Another strong attempt at formalising activities in search was provided by Goncalves et al. (2004), who tried to mathematically formalise the difference between searching and browsing. The combination of theorems and lemmas led to specific enhancements in their system, which was shown to enrich the search experience provided. As cited for a good overview of IR models, research began with Saracevic (1997). The following subsections discuss some exploratory search systems. Following this, Section 3.2 discusses the relevant IR literature

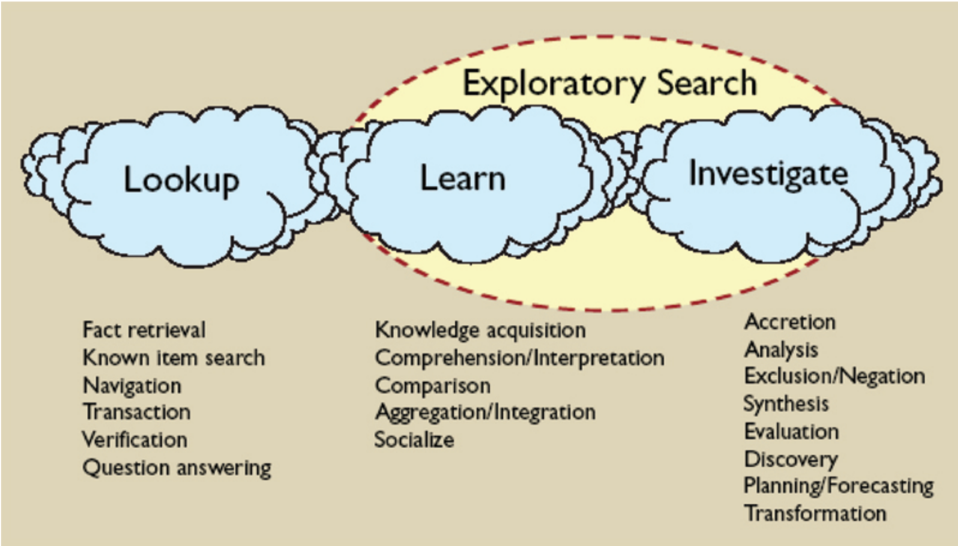



Figure 4: Search Activities cf. (Marchionini, 2006)

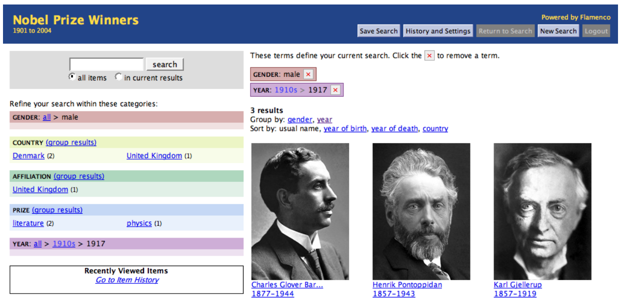



Figure 5: The Flamenco Interface on the Nobel Prize Winners Dataset

used to generate and contribute to a combined model, which has been applied to three FBs to evaluate their support for ESIs.

### 3.1.2 Faceted Browsing

In Section 2.1, mSpace is described as a type of faceted browser. Faceted browsing can be considered as an exploratory search approach, thus making mSpace and other faceted browsers example ESIs. Section 2.1 already discusses the merits of the faceted browsing approach. Below, two further FBs are presented: Flamenco and RB++.

**Flamenco**

The Flamenco browser[6] (Yee et al., 2003), shown in Figure 5, supports both keyword search and faceted browsing, accounting for both those who know

[6]http://flamenco.berkeley.edu/

their target and those who don't have much knowledge about the domain. The initial display shows all the possible facets in too columns, with vertical scroll as necessary. Here the user can either make an initial selection from the facets or use the search box, which is consistently at the top left (unless viewing a TO). By entering a search query or selecting an item in one of the facets, the user is moved away from the initial view to one where all the facets are listed vertically down the left column, with the search box remaining at the top left. A "breadcrumb" (Bonnie Lida and Pilcher, 2003) is presented at the top right, which presents a visualisation of the path of selections made by a user. A search term acts as a domain filter and the search results (displayed in the remaining space at the bottom right) may still be browsed using the facets, much like the motivation of the work by Shen *et al.* described above. If the search term can be matched to particular items in the facets, these are presented to the user above the breadcrumb.

When a selection is made in a facet, the sub-categories within the facet are shown and a per-facet breadcrumb displays the selection made. If there are no sub-items, the facet is effectively minimised (facet representations grow endlessly with the number of options within it). If facets are hierarchical, results are automatically clustered into the sub-categories of the latest selection. The user may optionally group the results by any other facet through a single interaction provided by the presence of a new link along side of each facet name. Any potential option for selection is accompanied by numeric volume indicators (Wilson and schraefel (2006)), to estimate the number of TOs that can be reached by its selection.

When TO selections are made, the user is moved away from the faceted browser display to one that shows a summary of the data associated with their choice. From here, the user is given options to return to the faceted browser: extra facet selections can be made to expand or further narrow their constraints and view similar objects. Users may also reset the interface by pressing the 'New Search' button.

**RB++**

The relation browser, named RB++[7] has been developed through research into Information Science, and is shown in Figure 6 (Zhang and Marchionini, 2005). mSpace and Flamenco have been development mainly through advancing research into Human Computer Interaction.

In RB++, the interface currently presents all the facets and their contents persistently: these facets are listed across the top of the UI and grow/shrink to fit on the screen. Users can reorder columns, but for no obvious reason, using a drop down list that formulates as both a mechanism for changing the facet and also for acting as its label. The user can make facet selections in any order and the temporary hierarchy built is controlled by this selection order: this breadcrumb order is not currently visualised. NVIs are represented as an in-place bar graph. The population of the bar represents both the number of achievable TOs from making that selection and, uniquely, the number of total TOs in the dataset. The exact figure is represented as an NVI to the left of each label. Hovering over items in each facet previews the affects of the selection on each of these NVIs and is made persistent by actually clicking.

[7] http://idl.ils.unc.edu/rave/ - Interactive Design Laboratory Presents RAVE

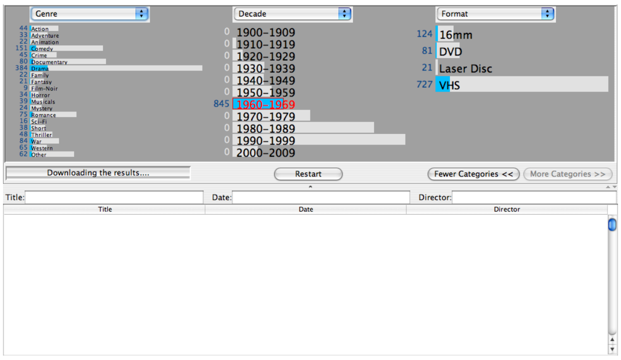



Figure 6: The RB++ Interface on the UNC Movie Catalogue

By pressing the *search* button, results are displayed in the lower half of the screen, where items can be filtered, sorted and individually selected. Once the search results are displayed, the previous selections above are transformed into a label representing the selections, much like a breadcrumb but without temporal order. The facet browser is also transformed to represent the subset of TOs that had been previously achieved through facet selection. Thus NVIs represent the number of TOs in the new subset. Any subsequent facet selections automatically filter the search results. Upon selection in the results, the TO is displayed in a new window.

### 3.1.3 Other Advanced Designs

Although FBs can be seen as a particular class of ESIs, other research has produced interesting design features to enrich search environments. These features are designed to be embedded in existing search systems to augment the coping strategies discussed in this report's introduction. That is, methods adopted by users to achieve more complex goals with the limited functionality of keyword search (Pirolli and Card, 1995).

**Polyarchies**

Robertson et al. (2002) developed a concept of multiple intersecting hierarchies called Polyarchies. With this, he developed a browser that allowed users to rotate between hierarchies; this rotation was directly visualised as in Figure 7. This allows users to identify the same information in multiple datasets, providing the option to go off on tangents in the information space. McGuffin and schraefel (2004) indicated that faceted browsing is a form of polyarching, as any number of hierarchies can exist in the information and so every point is a pivotal point to rotate around. However, as this polyarchic ability is apparently inherent in faceted browsing, its concept could be better applied to multiple datasets, allowing users to follow tangents and switch between information areas, rather than within information areas.

**Augmenting Search**

Based on the ideas of information foraging by Chi et al. (2001), the authors continued to develop a project called Scent Trails (Olston and Chi, 2003), where

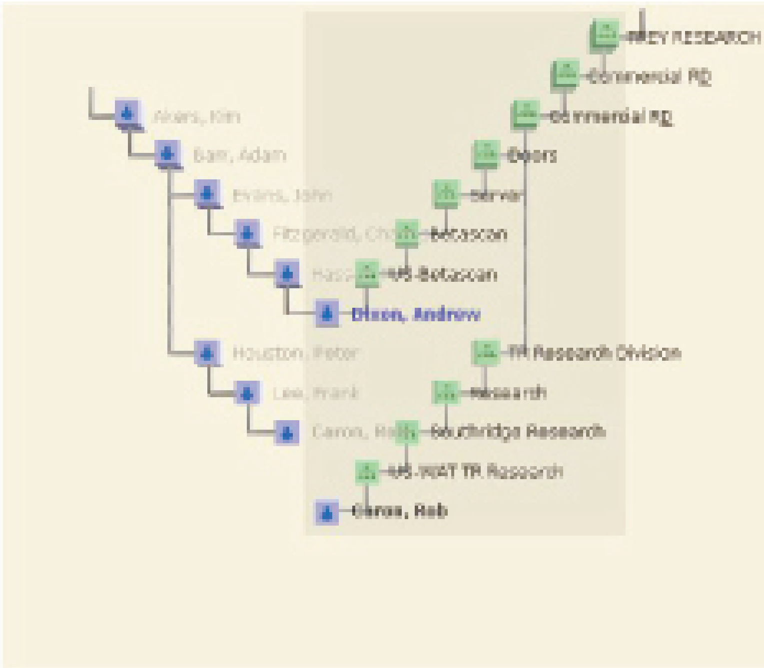



Figure 7: A Polyarchy Rotation around an intersection point between two hierarchies

links in a website are augmented in someway, such as size, to indicate their potential support for the user: a larger link represents a more useful source of information. The idea is that standard web objects are augmented to give clients more context.

Huynh et al. (2006) carried out some very exciting research in augmenting the content of websites to empower the user. By scanning websites that have consistent and repeated content, they are able to convert simple web content into faceted content. The user is then able reorder and filter the content of the website using a sidebar extension and continue to browse the website under their own preferences. There is an especially compelling video to accompany this publication[8]. The interface can be seen in Figure 8, where the content of Amazon[9] is being filtered by the media types: DVD and Paperback.

#### 3.1.4 Summary

Each of these interfaces provide strong support for users, but are simultaneously very different. Their variation in design and content, combined with their overall versatility, makes such interfaces very hard to compare. The search interaction stream of the Text Retrieval Conference (TREC) was abandoned in 2002 because the comparison between systems developed at different institutions was too difficult. Gary Marchionini and his team have carried out a study comparing

[8]http://people.csail.mit.edu/dfhuynh/research/papers/uist2006-augmenting-web-sites.mov

[9]http://www.amazon.com

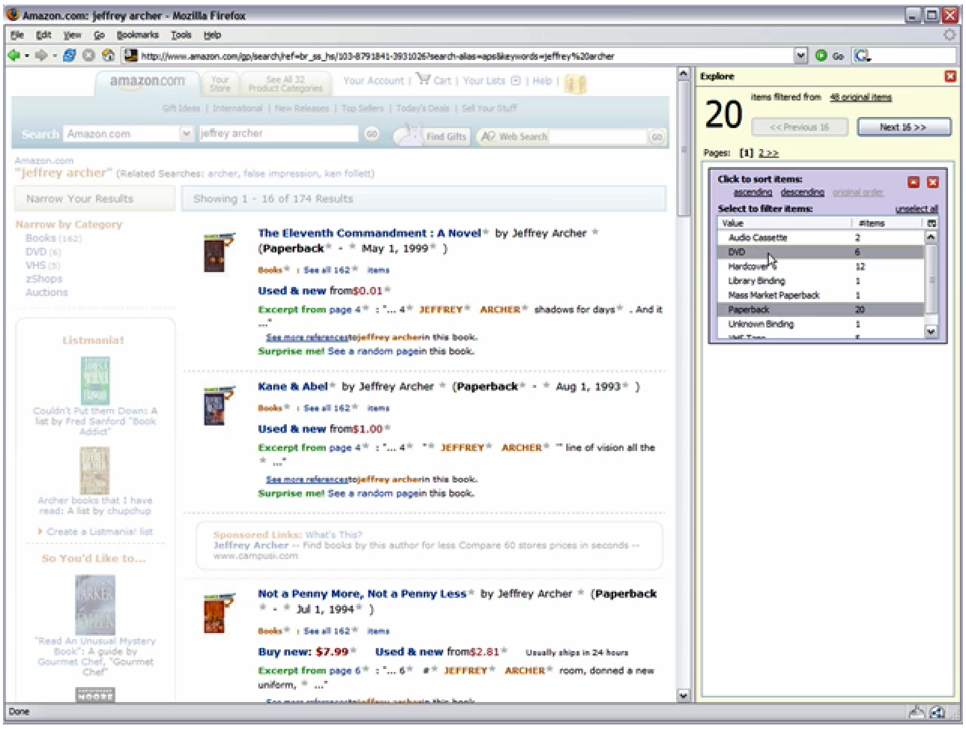



Figure 8: Amazon being augmented by the MIT Firefox Extension

three faceted browsing approaches. In discussion with Gary, the complexity of the study was approximately of order $n\hat{2}$. We hoped that mSpace could be included in the study, but the oder of complexity meant that the study would triple the size of the evaluation, for both participants and organisers. Although they yet to publish their results, email communication suggested that there were no strong differences found between the browsers. Marchionini also noted that thy ran into some difficulty, despite months of planning, which delayed the project further.

In further discussion on the choice of metrics, the comparison was again complex. Where time efficiency might be a usual metric, the study was of browsers that allowed users to explore information in great detail. It was arguable, therefore, that a successful system might encourage greater time spent on the given tasks. For example, if the user was asked to investigate an information set to find three important TOs, the user may be encouraged to spend more time interrogating all of the options. Instead, the study analysed system logs and measured preferences for the systems. The lack of key differences in the results perhaps shows that the benefit provided by the different systems has not been effectively captured. The related work below investigates the potential for using models of information-seeking behaviour to quantify the strengths and weaknesses in design, to produce a metric that clearly defines the range of support provide for users in search by a search environment.

## 3.2 User Search Models

While the designs above have been generated through mainly HCI research, an entire research field (IR) has been dedicated to furthering search systems specifically. Research into Information Retrieval has concentrated on two main areas: systems and users. Robertson and Hancock-Beaulieu (1992) suggest that these have been investigated through several revolutions of focus: the interactive, cognitive and relevance feedback revolutions. Throughout these revolutions, and dating back to the mid 1970s, many authors have written in particular about the user and user models (Bates, 1990; Hsieh-Yee, 1993; Pejtersen, 1989). Yet recent papers (Kriewel, 2006; Marchionini, 2006) are still proposing further investigation into the needs and activities of users. While some key literature is concerned mainly with various forms of relevance feedback (Rocchio, 1971) to support the computer end, other established research is modelling the users specifically (Bates, 1979). Between these interests, some authors are modelling entire IR systems: human-computer interaction.

My research proposal is that some of these models from the history of IR research into users, their needs and their search behaviour, can be used to evaluate the success of search environments. Some key established models are discussed below, in the context that they might be useful to quantify or measure the support provided by a search system.

### 3.2.1 Stratified Models

Stratified models are an approach to modelling all the aspects of a system (human and computer) and the ways in which they interact at different levels; the view is stratified across the aspects that make up the whole system. Saracevic (1997) produced one of the first main stratified models, which remains to the seminal model of this kind; shown in Figure 9 below. Bates (2002) also presented a stratified model of a system, but expands further on the technological side, rather than the user aspect.

The conclusions of stratified models is that any level of the system may affect the support provided to the user. Thus, poor indexing may limit the capabilities of the search algorithms and slow hardware may limit the speed of response. Similarly, Saracevic models users across four levels, where the ability to construct an appropriate complex query might be affected by the state of their existing knowledge. Similarly, they poor internal definition of a task might affect their intent for using an interface.

In order to evaluate exploratory search interfaces in respect to their support for users effectively, the users have to be modelled carefully.

### 3.2.2 Interactive Models

While Saracevic's model can be used to evaluate the concurrently running layers of an IR system to identify at what level support for user search is constrained, Belkin et al. (1995) have produced an episodic model to define and understand the flow in scenarios of human-system interactions: these flow definitions are called scripts. However, to do this, Belkin *et al.* first highlights four binary dimensions that define 16 unique Information-Seeking Strategies (ISS). They have calculated separate scripts for each of these 16 ISS conditions, which allow for

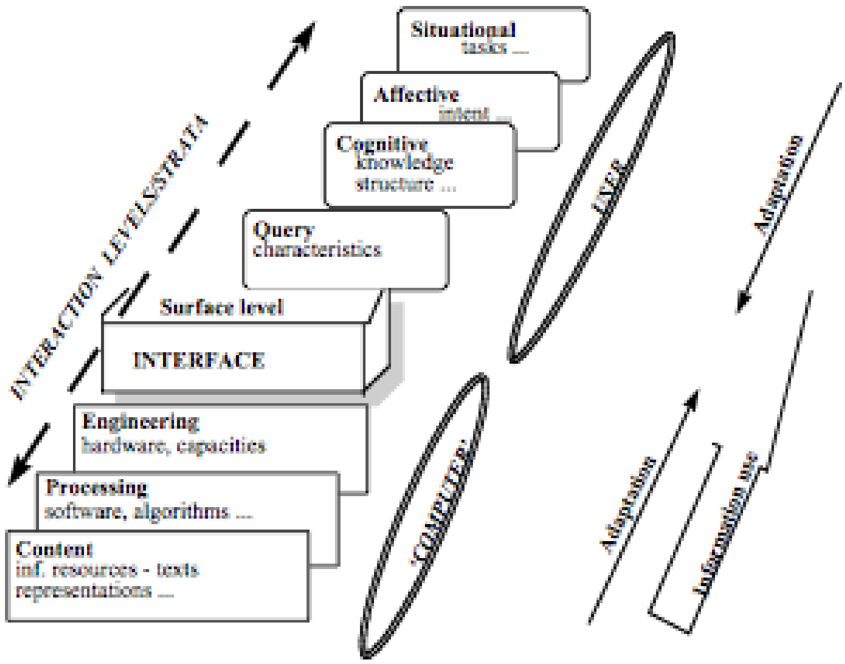



Figure 9: Saracevic's Stratified Model of IR Systems

switching between them. The dimensions are *Method*, *Goal*, *Mode* and *Resource* and in combination produce sixteen unique conditions shown in Table 1.

*Method* describes whether a user is either searching for an information object, or scanning a set of information objects. This is easily differentiated by finding a specific paper in order to get its reference details, or by searching for an possible paper, which may not exist, that can be used to support a point. *Goal* describes whether a user is learning about something or selecting something. Using the bibliographic example differentiates these as researching a topic, or finding a reference. *Mode* is between recognising and specifying something. One might remember that there was a useful publication at CHI2005 and so is trying to identify it in the proceedings, or may have known the author, title and year and has typed them into the ACM Portal. *Resource* is between wanting information items or meta data about an information item. Usually, with a bibliographic repository users are trying to find specific papers, but it is possible that the user is trying to find out first what workshops existed in a conference so that they can better define a search query at a later point in time.

For example, search engines like Google poorly support users in finding meta-information (*Resource*), as a user must know which words to use in advance before she can begin to view items of information. It also provides poor support for recognising as a *Mode* because a user has to specify meta-information in the query. This means that Google primarily supports only half of the potential search conditions of users. Marchionini (2006) notes that search engines are usually concerned with precision (maximising the accuracy of the top re-

| ISS | Method | Goal | Mode | Resource |
|---|---|---|---|---|
| **1** | Scan | Learn | Recognize | Information |
| **2** | Scan | Learn | Recognize | Meta-Information |
| **3** | Scan | Learn | Specify | Information |
| **4** | Scan | Learn | Specify | Meta-Information |
| **5** | Scan | Select | Recognize | Information |
| **6** | Scan | Select | Recognize | Meta-Information |
| **7** | Scan | Select | Specify | Information |
| **8** | Scan | Select | Specify | Meta-Information |
| **9** | Search | Learn | Recognize | Information |
| **10** | Search | Learn | Recognize | Meta-Information |
| **11** | Search | Learn | Specify | Information |
| **12** | Search | Learn | Specify | Meta-Information |
| **13** | Search | Select | Recognize | Information |
| **14** | Search | Select | Recognize | Meta-Information |
| **15** | Search | Select | Specify | Information |
| **16** | Search | Select | Specify | Meta-Information |

Table 1: Information Seeking Strategies (cf. Belkin et al. (1995))

sult) than recall (maximising the number of relevant results), and so the extent of support for ISS conditions is further reduced by poor support for learning (*Goal*). This is validated by work done in 2004, which estimated that around 81% of search engine users viewed only one result page (Beitzel et al., 2004). Relevance feedback efforts, such as Google's 'Related Pages' suggestions, have tried to support the user in terms of meta-information. Yet the user would still have to fire off at least one query and then process the results before any support is provided. Google is best used, therefore, for ISS15, where the user is searching (*Method*) to select (*Goal*) by specifying (*Mode*) attributes of a specific information object (*Resource*). Subsequently it least supports users who are scanning (*Method*) to learn (*Goal*) by recognising (*Method*) some meta data about an information object (*Goal*): this is ISS2. Faceted browsing tries to support users by presenting all the meta-information to the user in advance and letting them choose. Conversely, this best supports ISS2, but may poorly support ISS15: useful meta-data can be embedded in long lists and it may require more effort to find them than to simply type them into a search box.

Cool and Belkin (2002) realised that these dimensions were not completely exhaustive and so extended and expanded upon these four dimensions with much greater detail. They produced a multifaceted hierarchical taxonomy of search conditions, which can be used to describe search situations in much greater detail. This taxonomy is very extensive and is extremely useful to evaluate real life needs and scenarios.

The scripts produced for each of Belkin's ISS conditions recognise that, at various points, users may have gained insights or have varying external conditions that cause their goals and motivations (Saracevic's situational and affective levels) to change over time. This is what makes the model by Belkin *et al.* episodic, as the scripts account for these early exit and entry points into other scripts. Research by Kriewel (2006) summarises research into the situational aspects of IR, noting that accumulated effort, previous actions, remaining needs, technical problems, and accuracy of results are all factors that can effect the search situations of users. These search conditions were used in conjunction with the scripts defined by Belkin *et al.* to develop the Digital Library system: DAFFODIL, which was designed to recognise some standard situations and recommend various functions that support the different ISSs. The automatic estimation of user situations and goals is well summarised by Chi *et al.*, which then describes a method of user modelling called *information scent*. This algorithm performed well at estimating the goals of the users, and then suggested paths for further achieving them (Chi et al., 2001).

### 3.2.3 User Search Activities

While the research above investigates search systems and patterns of interactions, other research has investigated the activities that may prompt them. The aim is to model users in order to describe why they have chosen systems and what causes the patterns of typical interactions.

In 1990, Bates (1990) described in detail two models for both the different levels of search activities and the different levels of system automation. First she identifies 5 levels of system ranging from complete user action to complete system automation. This is then combined with four levels of search activities: Move, Tactic, Stratagem and Strategy. The first of these is a single action performed by the user, either physically or mentally: mental actions may be deciding or reading. A tactic is a combination of moves and there are endless combinations of moves that can be used to support a tactic, which depends on system implementations. She defines 32 specific information search tactics. For example, one tactic is CHECK, which is the ability to check the decisions you have made. This is well supported by faceted browsers such as mSpace, as the current path taken by the user is usually visible at all times. Stratagems are a larger combination of both individual moves and tactics: some examples include performing a citation search or following a footnote. The ES activities identified by Marchionini (2006) are examples of stratagems: Comparison, Evaluation, Aggregation, etc. Strategies are again higher and involve a combination of moves, tactics and stratagems: this might be finding relevant work for a paper and depends heavily on what the user is currently working on.

Bates suggests that by combining these, a system that supports fully automated strategy is one that reads a user's mind and knows what she is doing and that most systems provide no automatic support for strategies, allowing complete human control. A system that has no automation for moves is a pile of unsorted paper. The work on scent above is trying to automate and support stratagems and most software systems automate moves at least.

## 4 Evaluation Framework

The challenge set above, is to evaluate complex and versatile exploratory search browsers, in some meaningful way. Typically, user studies might use time efficiency as a measure of success. However, exploratory search browsers are not necessarily designed to get the best answer first. In fact, spending more time exploring the information with a particular design might be considered a measure of success. In reality, the choice of measure should depend on what the users are trying to do with the search system and its information. The research above tries to model users, their needs, and the information-seeking behaviour. With the intention of evaluating a system in terms of its support for these factors, we propose an evaluation framework that combines the models above. A combination of Bates' and Belkin's models, to achieve this challenge, is discussed below.

In combination, these models also represent various levels of Saracevic's model of a user. The following discussion uses terms from these models to explain the proposed approach.

A user's strategy (*Bates*) may have led the user to a particular search system, as it usually represents a certain type of resource, such as a journal archive or a product collection. Bates' definition of a strategy relates quite well to Saracevic's situational tasks. Once using a particular resource, the user may wish to employ a set of stratagems (*Bates*) to achieve their goal, and should relate directly to Saracevic's affective intentions level. Similarly, Bates' tactics, may also be considered part of Saracevic's affective intentions. Finally, Bates' moves can relate to Saracevic's query generation, as each action should contribute towards exploring the information set. The only un-modelled level of Saracevic's model is the cognitive level, defining existing knowledge, for example. Cognition can be partially modelled by using Belkin's user conditions, which incorporates things such as previous knowledge. This model actually touches on a few levels of Saracevic's model, including intention.

Bates' moves are used to quantify the support for tactics by interface features. Then these tactics are applied to support Belkin's dimensions, so that the support for the sixteen conditions can be calculated. To achieve this, an exact six-stage procedure is described below.

## 4.1 Six-Phase Design

#### Stage 1: Identify Features

First, the interface features and their interactions must be identified. For example, mSpace has a set of features including: browser columns, a collection space, a preview player and an information panel. Also, Flamenco and RB++ provide the ability to sort search results. The features of each design should be incorporated, so that their full support for users are measured.

#### Stage 2: Calculate Support for Tactics

Each interface feature is addressed one at a time, for each design. For the current feature of the current design, the moves required to support each tactic is calculated. This produces a series of tables, one for each design, where tactics are listed across the top and the interface features down the side. The count of moves is noted in the appropriate cross section between feature and tactic. No support for a tactic counts as 0. Repeat and Optional moves are ignored. For example, selecting multiple items involves selecting 2+ items, selecting 3+ is considered a repeat move of selecting 2 items. Optional moves include scrolling: a desired item may be the first or last item. The optimum situation is that it is one of the items that is visible without scrolling.

#### Stage 3: Summarising Metrics

As no support is represented by zero, support in a single move is represented by 1 and support in ten moves by 10, all values above 0 must be inverted. Thus a feature that supports a tactic well approaches the value of 1 and a poor support approaches 0. These inverted metrics can then be summed by feature and by tactic. This calculates the support provided by a feature for all tactics and the

| RB++ | | CHEC | WEIG | PATTI | CORF | RECO | BIBBI | SELE( | SURV | CUT | STRE | SCAF | CLEA' | SPEC | EXHA | REDU | PARA | PINP( | BLOC | SUPE | SUB | RELAT | NEIGH | TRAC | VARY | FIX |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| preview cue | 4.5 | | 1 | | | | | | 1 | 2 | | | 2 | | | | | | | | | | | | | |
| favourites | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| make selecti | 9.41666667 | 1 | 1 | 1 | | | | 2 | 1 | | | | | 2 | | | | | | 1 | 2 | 2 | 2 | 4 | 3 | |
| multiple sele | 4.83333333 | | | | | | | | 1 | | 2 | | | 2 | 2 | 2 | 2 | 2 | 3 | | | | | | | |
| facet organis | 0.25 | | | 4 | | | | | | | | | | | | | | | | | | | | | | |
| breadcrumb | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| change selec | 7.5 | 1 | 3 | 1 | 3 | | | 3 | 1 | | | | | 3 | | | | | | 3 | 2 | 3 | 3 | | 3 | |
| keyword sear | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| view item | 0.25 | | | | | 4 | | | | | | | | | | | | | | | | | | | | |
| filtering | 8.33333333 | 1 | 1 | 1 | 3 | | | 2 | 1 | 2 | | | 2 | 2 | | 2 | | 2 | | | 2 | | | | | |
| sorting | 1.66666667 | | | | | | | 3 | | | | | 3 | 3 | | | | | | | | 3 | 3 | | | |
| NVIs | 5 | 1 | 1 | | | | | | 1 | 2 | | | | | | | | | | | | | | | | |
| | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | | | | | | | | | | | | | | | | | | | | | | | | | |

Figure 10: Example table of the moves being counted for the RB++ browser

support provided for a tactic across all features, respectively. An example table for collecting these metrics for one interface or design can be seen in Figure 10.

#### Stage 4: Tactic Support Analysis

A graph can be produced including the summed values for each tactic in each design. Again, tall bars indicate strong support for a tactic. This comparison may identify tactics which may require improved support through redesign.

#### Stage 5: Feature Strength Analysis

A graph can be produced including the summed values for each feature in each design. An example can be seen in the following section. Strong features will score produce tall bars, and a quick comparison of user effort can indicate a strong feature design.

#### Stage 6: User Conditions Analysis

Each tactic supports particular ends of Belkin's dimensions of user conditions. CHECK, a tactic for users checking their decisions so far, supports users who are trying to Learn as their *Goal*. The support for a tactic by a design is added to the total support for a dimension. Then for each of the sixteen conditions, the sum of the total support values are calculated. This value for each condition can be graphed showing the difference in support for different user conditions.

This is carried out for every *tactic* for every feature of all three interfaces and is stored in a table, as shown in Figure 10.

## 4.2 Applied

To both present and test the model, it was applied to three FBs, which are inherently ESIs. As stated above, an FB provides an alternative to keyword search, so that options may be compared and meta-data may be browsed. The three browsers compared were those discussed above: mSpace, Flamenco and the Rave Browser (RB++). Each have been developed in academia and motivated by improving access to information. Other faceted browsers exist. Endeca [10] is a commercial faceted browser that is not publicly accessible for research purposes. More recently,
facet has been developed to use faceted browsing for supporting information

[10]http://www.endeca.com - Endeca

architecture and evolution (Hildebreand et al., 2006). These have not been included during this initial trial of the framework, but could be easily included if access, to Endeca for example, were permitted. While each of these three FBs have been studied independently, they have rarely been compared or evaluated against each other. The results and conclusions of applying this evaluation framework to these browsers are discussed below.

## 4.3 Results

### 4.3.1 Tactic Support Analysis

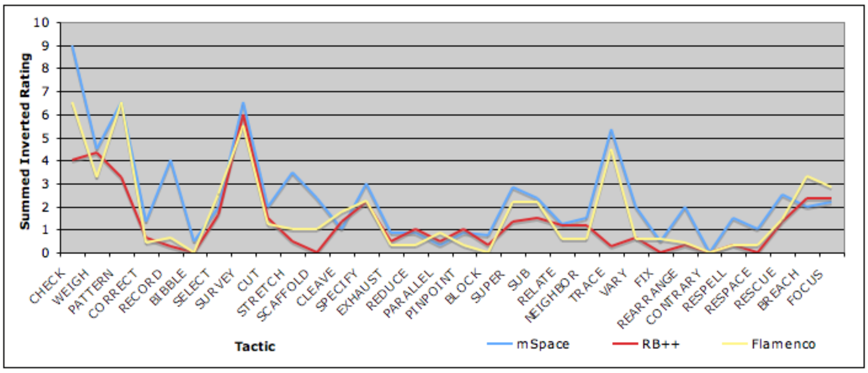



Figure 11: Graph Showing the Summed Inverted Metrics of each Tactic for the Three Browsers. Strong support for a Tactic is shown by a high metric value.

According to **Stage 4**, Figure 11 was produced and provides a number of notable observations. First, each interface has a high peak for SURVEY. This is an expected peak when evaluating faceted interfaces because the user is presented with optional selections at each stage. This peak would not be so visible in keyword only interfaces. We now continue by investigating significant differences between interfaces in this graph.

The first tactic, CHECK, has different levels of support in all three interfaces: this tactic is to see what actions have made to corroborate them with the current aims. In RB++, although previous selections are highlighted in the interface, no representation of order is given and so a lower support for checking ones actions is provided. In Flamenco, this feedback is given in a breadcrumb, and is visible when navigating through the facets. To view a TO in Flamenco, the user is moved to a new page with a summary of that object. Thus, before the user can view the breadcrumb, they must first return to search: this requires two moves. In mSpace, breadcrumbs are embedded into the ordered facets. As mSpace is a focus+context browser, the user can view the facets and their previous actions at all times, including when viewing a TO. This leads to a higher peak for mSpace and then Flamenco in Figure 11.

There is a significant peak for the mSpace interface, which supports the RECORD tactic. The mSpace interface includes a within-browser collection space that can store any object in the facets. Although any state reached in Flamenco and mSpace can be saved using the parent application[11], and pages

[11] Usually an Web Browser such as Firefox or IE

displaying TOs in all three interfaces can be saved in this way, a single double-click move can store facet items in the Interest panel of the mSpace browser at any point: even when viewing a TO it can be saved with by double-clicking or dragging the item into the box.

There is also a significant peak over the STRETCH and SCAFFOLD tactics for the mSpace browser. STRETCH, reusing objects in unintended ways, is highly supported because of the explicit ordering of facets. The reordering of facets allows users to see the effects of meta-data on other meta-data: this reordering involves a single dragging action. SCAFFOLD, finding quick paths to TOs, is highly supported, because selecting preview cue objects will brings up not only information about its TO, but can also be used to see its position in the facets. Users may recover a path used to find items in the Interest panel by dragging it onto the columns or double clicking the item, displaying a quick jump to a previous path.

It may be noted that mSpace is specifically higher over all of the Term Tactics (SUPER to RESPACE). It may also be noted that no interface supported CONTRARY, an antonym of a selection. After investigation, these higher ratings are supported mainly by a combination of features. While it is easy in Flamenco to use the SUPER tactic, by simply removing an item from the breadcrumb, users of mSpace have two options: they may simply identify and click on a different item, or they may reorder the columns so that a selection is placed higher up the temporary hierarchy. The former of these two is not achievable in Flamenco, as alternatives of a selection are hidden and the exact selection is only displayed in the breadcrumb. The RELATE and NEIGHBOR tactics are also poorly supported in Flamenco due to the aforementioned four step process to change a selection. REARRANGE is also well supported by mSpace due to the ease in reordering facets. Finally, tactics like RESPELL are well supported by mSpace because changes to misspellings and unrecognised words in the keyword search are suggested and can be applied by a single click.

Finally, SCAFFOLD and TRACE are both poorly supported by RB++ as the facet columns are used for two purposes: making facet selections and, once TOs have been listed, filtering TOs. The selections made before TOs are listed are hidden. It is a unique feature that this separation exists, as making facet selections are by nature filtering the TO list and most browsers merge these conditions.

### 4.3.2 Feature Strength Analysis

Figure 12 shows the significant contribution of different interface features. Certain elements of the previous discussion can be seen here clearly. Flamenco's four steps to change a selection are reflected in the slight drop of their bar. It may also be noted that Flamenco has no preview cue, and thus the bar is absent from the graph. The ease of multiple selection in RB++ is also clearly shown. One feature to compare is 'View Item'. RB++ has a significant drop in support here, as the implementation has a significant separation between TOs and Browser. TO Pages may be simply launched from the browser, but there are no ways in which the user can interact with the browser when viewing them. The only option is to return to the browser. In Flamenco and mSpace, users can make further selections from the TO page that force automatic interactions with the facets: this is most direct in mSpace where the facets are always present.

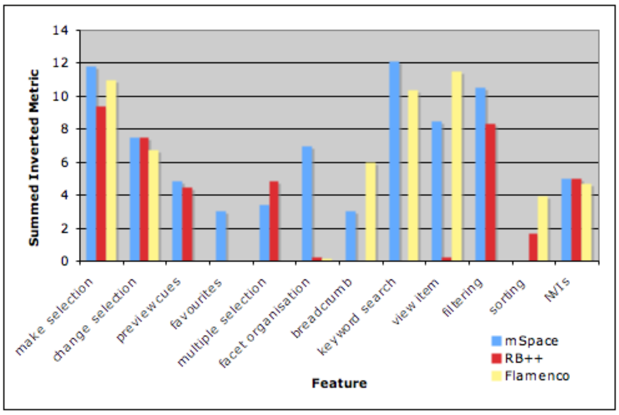



Figure 12: Graph Showing the Summed Inverted Metrics of each Feature for the Three Browsers. Strong support is indicated by a high metric value.

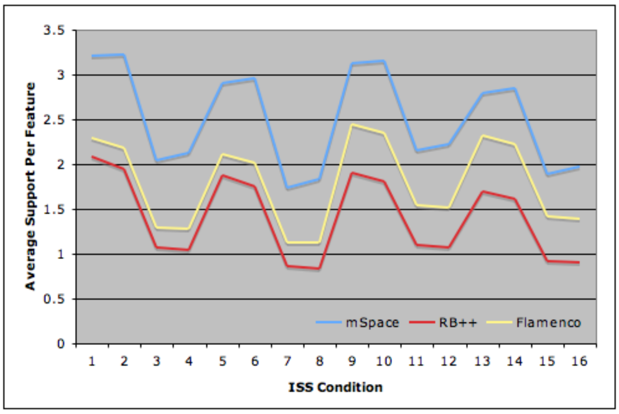



Figure 13: Graph Showing the Normalised Support for each ISS Condition by Faceted Browser

mSpace has no sorting function, which is shown clearly on the graph, but is well supported by RB++ and Flamenco. In Flamenco, a user is able to group the results by any of the facets in the system and provides the strongest implementation of a sorting method. However, Flamenco does not support filtering. In mSpace, user can filter long lists of items in facets to jump quickly to selections. RB++ also provides the filtering of TOs by reusing the facets for filter selections: this support is only for TOs and presents weaker support for the interface. The in-browser collection space in the mSpace interface clearly provides support for the interface but is also unique to mSpace.

### 4.4 Discussion

Two observations can be drawn from these graphs. First, there is better support for search in mSpace due to the wider number of implemented features. Second, each interface has strong features that prevail in the group. These two points are considered in greater detail below.

| ISS | Tactics | mSpace | | RB++ | | Flamenco | |
|---|---|---|---|---|---|---|---|
| 1 | CHECK, WEIGH, RECORD, SURVEY, EXHAUST, PARALLEL, SUPER, RELATE, NEIGHBOUR, RESCUE, BREACH | 35.25 | 3.20 | 22.92 | 2.08 | 25.20 | 2.29 |
| 2 | CHECK, WEIGH, RECORD, SURVEY, STRETCH, EXHAUST, PARALLEL, SUPER, RELATE, NEIGHBOUR, RESCUE, BREACH | 38.75 | 3.23 | 23.42 | 1.95 | 26.20 | 2.18 |
| 3 | CHECK, CORRECT, RECORD, CUT, SPECIFY, CLEAVE, EXHAUST, PARALLEL, BLOCK, SUPER, RELATE, NEIGHBOUR, REARRANGE, CONTRARY, RESPELL, RESPACE, RESCUE, BREACH | 36.83 | 2.05 | 19.25 | 1.07 | 23.23 | 1.29 |
| 4 | CHECK, CORRECT, RECORD, CUT, STRETCH, SPECIFY, CLEAVE, EXHAUST, PARALLEL, BLOCK, SUPER, RELATE, NEIGHBOUR, REARRANGE, CONTRARY, RESPELL, RESPACE, RESCUE, BREACH | 40.33 | 2.12 | 19.75 | 1.04 | 24.23 | 1.28 |
| 5 | WEIGH, RECORD, SELECT, SURVEY, SCAFFOLD, EXHASUT, PARALLEL, SUPER, RESCUE, BREACH | 29.00 | 2.90 | 18.75 | 1.88 | 21.12 | 2.11 |
| 6 | WEIGH, RECORD, SELECT, SURVEY, STRETCH, SCAFFOLD, EXHAUST, PARALLEL, SUPER, RESCUE, BREACH | 32.5 | 2.95 | 19.25 | 1.75 | 22.12 | 2.01 |
| 7 | CORRECT, RECORD, SELECT, CUT, SCAFFOLD, SPECIFY, CLEAVE, EXHAUST, PARALLEL, BLOCK, SUPER, REARRANGE, CONTRARY, RESPELL, RESPACE, RESCUE, BREACH | 29.42 | 1.73 | 14.58 | 0.86 | 19.15 | 1.12 |
| 8 | CORRECT, RECORD, SELECT, CUT, STRETCH, SCAFFOLD, SPECIFY, CLEAVE, EXHAUST, PARALLEL, BLOCK, SUPER, REARRANGE, CONTRARY, RESPELL, RESPACE, RESCUE, BREACH | 32.92 | 1.83 | 15.08 | 0.84 | 20.15 | 1.12 |
| 9 | CHECK, WEIGH, PATTERN, BIBBLE, SURVEY, REDUCE, PINPOINT, SUB, RELATE, NEIGHBOUR, TRACE, VARY, FIX, FOCUS | 43.75 | 3.13 | 26.67 | 1.90 | 34.25 | 2.45 |
| 10 | CHECK, WEIGH, PATTERN, BIBBLE, SURVEY, STRETCH, REDUCE, PINPOINT, SUB, RELATE, NEIGHBOUR, TRACE, VARY, FIX, FOCUS | 47.25 | 3.15 | 27.17 | 1.81 | 35.25 | 2.35 |
| 11 | CHECK, PATTERN, CORRECT, BIBBLE, CUT, SPECIFY, CLEAVE, REDUCE, PINPOINT, BLOCK, SUB, RELATE, NEIGHBOUR, TRACE, VARY, FIX, REARRANGE, CONTRARY, RESPELL, RESPACE, FOCUS | 45.33 | 2.16 | 23.00 | 1.10 | 32.28 | 1.54 |
| 12 | CHECK, PATTERN, CORRECT, BIBBLE, CUT, STRETCH, SPECIFY, CLEAVE, REDUCE, PINPOINT, BLOCK, SUB, RELATE, NEIGHBOUR, TRACE, VARY, FIX, REARRANGE, CONTRARY, RESPELL, RESPACE, FOCUS | 48.83 | 2.22 | 23.5 | 1.07 | 33.28 | 1.51 |
| 13 | WEIGH, PATTERN, BIBBLE, SELECT, SURVEY, SCAFFOLD, REDUCE, PINPOINT, SUB, TRACE, VARY, FIX, FOCUS | 36.33 | 2.79 | 22.00 | 1.69 | 30.17 | 2.32 |
| 14 | WEIGH, PATTERN, BIBBLE, SELECT, SURVEY, STRETCH, SCAFFOLD, REDUCE, PINPOINT, SUB, TRACE, VARY, FIX, FOCUS | 39.83 | 2.84 | 22.50 | 1.61 | 31.17 | 2.23 |
| 15 | PATTERN, CORRECT, BIBBLE, SELECT, CUT, SCAFFOLD, SPECIFY, CLEAVE, REDUCE, PINPOINT, BLOCK, SUB, TRACE, VARY, FIX, REARRANGE, CONTRARY, RESPELL, RESPACE, FOCUS | 37.92 | 1.90 | 18.33 | 0.92 | 28.20 | 1.41 |
| 16 | PATTERN, CORRECT, BIBBLE, SELECT, CUT, STRETCH, SCAFFOLD, SPECIFY, CLEAVE, REDUCE, PINPOINT, BLOCK, SUB, TRACE, VARY, FIX, REARRANGE, CONTRARY, RESPELL, RESPACE, FOCUS | 41.42 | 1.97 | 18.83 | 0.90 | 29.20 | 1.39 |

Table 2: Table Showing Bates' Tactics for each of Belkin's ISS Conditions With Standard and Normalised Scores for each Interface

It is clear from Figure 11 that there are few tactics that are not better supported by the mSpace browser. This is explained from Figure 12 by the number of strong contending features: there are comparably high mSpace bars for almost every feature. This is arguably representative of the focus+context design, which aims to present as many options and features to the user as possible and at all times.

In line with the second observation, however, there are clearly some features of each interface that have stronger implementations of the three browsers. For example, multiple selection is easiest in RB++, yet keyword search is missing from RB++ and the implementation is strongest in mSpace. One feature missing from the mSpace implementation is the ability to sort items. The strongest implementation of this is the ability to group the results by any facet, as seen in Flamenco.

#### 4.4.1 User Support Analysis

One question that was considered at the start of this paper is when a browser well supports user intention: under what user conditions does a browser provide good support. By attributing each Tactic to support one of Belkin's four dimensions of search, such as supporting learn (*Goal*) or meta-information (*Resource*), the support for each ISS condition can be quantified using the Summed Inverted Metrics as before: this is shown in Table 2 and in Figure 13.

The pattern that is seen almost identically for each interface in Figure 13 is indicative of the mapping between Bates' tactics and the pattern of ISS conditions defined by Belkin *et al.* Predictably, as was shown in Figure 11, there are three distinct lines, showing that mSpace provides the widest support for search. This height difference does not show us new information. Instead what should be drawn from the graph is hidden within this pattern and shown in the differences in peaks and troughs for each interface condition. Quite clearly the graphs rise and fall in alternating pairs. This represents the alternation between recognise and specify (*Mode*) and is perhaps a predictable outcome for faceted browsers. By including more lessons learnt from the information seeking work on keyword search, such as relevance feedback, we might see a balance between these two conditions. Within each of these alternating pairs, the mSpace line marginally increases where the others fall. This indicates an increased support for meta-information (*Resource*). Considering individual browser lines, while RB++ and Flamenco follow a similar pattern for the first 8 ISS conditions, Flamenco notably improves this gap in the final 8 conditions. These two halves are made unique by the *Method* dimension and indicates that Flamenco provides better support for search, which is defined by having a known TO to exist: this might be knowing that an academic paper exists and just trying to find it. This significant increase, also sharper than mSpace, may be present due to the better support for making further selections and the lower support for changing selections.

The final pattern we draw from Figure 13 is shown every four conditions and is controlled by Belkin's *Goal* dimension. The Learn aspect of this dimension is shown by height differences between ISS1-4 and ISS5-8, and again between ISS9-12 and ISS13-16. This is characterised by the ability to see options in faceted browsers. The persistence of these options shown throughout to the user of mSpace is highlighted by the exaggerated difference in the first and third troughs compared to the second and fourth.

#### 4.4.2 Implications for Design

In some cases, simple feature upgrades would lead balance some identified differences. For example, keyword search in mSpace is stronger that Flamenco, but only because of the relevance feedback that is provided (Rocchio, 1971). By simply upgrading this with more recent IR research into relevance feedback (White et al., 2005a), both interfaces will become stronger. Other features may be intrinsic to a design: Facet Organisation in mSpace is one of these. Facet organisation has little use, but is possible, in both RB++ and Flamenco and is mainly used to bring popular facets to the forefront. However, the ability to order columns in mSpace supports a number of different Tactics. Multiple selection is also supported by both mSpace and RB++, but not supported in the

Flamenco design. However Flamenco has purposely supported faster selections towards a TO and adding multiple selection would slow this down. In order to provide this feature, making a normal selection would require more Moves.

First, it is clear from Figure 12 that keyword search should still be integrated into faceted browsers to support users in both methods. These can be optimised by including the enhancements detailed by information-seeking theory. Second, while supporting a user in selection making is important for users who are confident of their target, optimising the ability to change selections and make multiple selections is important for users who are searching for a potentially relevant but unknown object. Third, we suggest that representing the temporary hierarchies to the user is important for keeping track of user actions and understanding the effects of facets on each other. Although facets can be used in any order, the spatial ordering of them has been shown to support a number of search tactics. Fourth, the interoperability of viewing information pages with the browsing of facets is a key element in maintaining the search context. Fifth, the sorting and filtering of lists is an important part interacting with data. Sixth, the collection of information during search is important, especially when users are trying to locate relevant information rather than specific information. Finally, previewing the affects of actions is important for making decisions in search and should be shown as soon as possible.

# 5 Conclusions

This evaluation approach has made two contributions. First it has presented the application of a combination of models to evaluate three faceted browsers. Although these models have been designed to encompass elements of user search, applied in combination they can be used to identify the strengths and weakness of a browser. Second, by applying this evaluation to three interfaces it has been used to quantify these strengths and weaknesses over: the support for tactics, the support provided by interface features, and the support for sixteen unique user conditions.

Both moves and well defined tactics, from the model of strategic search interaction defined by Bates (1990), have been used to quantify the support for each tactic provided by the features of three faceted browsers: mSpace, Flamenco and RB++. These metrics have first been summed by tactic to show which of Bates' tactics are particularly supported by a browser. Second, by summing the metrics by feature, it has shown the support provided by its implementation. Identifying weak features can promote changes and advances in implementation to support more tactics or reduce the moves required to achieve each tactic. Finally, by summarising and normalising these metrics into Belkin's model of Information-Seeking Strategies (Belkin et al., 1995), it has identified particular strengths and weaknesses of the three faceted browsers in different search conditions.

The goal of this approach is to enhance the design phase of a system before expensive and complex user studies are employed to assess versatile systems that are increasingly difficult to compare. Evaluation in IR has focused on designing experiments that are: *insightful*, to assess the attributes, on which they focus, successfully; *affordable*, in respect to the cost of creating and running experiments; *repeatable*, so that others can build on results; and *explainable*, to guide subsequent improvements (Liu and Oard, 2006). The evaluation framework adheres to these four principles, and through simulating core user interactions, allows for refinements made to designs during their formative development. In some respects, the approach mirrors that of White et al. (2005b), but is more theoretically grounded.

# 6 Work Plan

So far, the evaluation framework above has been used to identify and quantify the strengths and weaknesses in the design of different faceted browsers. Uniquely, this approach measures these strengths and weaknesses in terms of the support provided for users in varying search conditions. Yet, while these results appear to be promising, there are still several non-trivial phases of research required before the approach is finished: validation, extension, evaluation and refinement. The first of these two are expected to be varied out before the mini-thesis and the final two thereafter.

## 6.1 Pre-Mini-Thesis Work Packages

The Pre-Mini-Thesis work will concentrate on the validation and extension of the outstanding activities mentioned above. The main deliverables for the validation phase will be documentation on: planned user studies, the resources used to carry them out and the results of statistical analysis. The main deliverables for the extension will be a report describing available models, their potential and my reasoned choices. The results and documentation from these stages can be reused to produce a submission to CHI2008. The expected tasks within each activity are described below.

### 6.1.1 WP1 - Approach Validation Plan

Validating the approach and the results of the evaluation method described in Section 4 has some unique challenges. The model takes a new tact in evaluating search systems that uses novel methods. Some of these methods may be transferable from other approaches. For example, the counting of Bates' moves is much like the counting in the keystroke model (Card et al., 1980). However, by using these models to quantify the support for search, the challenge is to validate a novel approach using existing methods that do not measure support with the same metrics. In particular, the metrics used in the framework above do not correlate directly to time, but some of the results may.

**Method (6 weeks total)**

Discover how other approaches have been validated, such as the Keystroke model. In the worst case, this may involve planning careful user studies to validate each identified result in Section 4.3.

**Tasks**

- Literature Search (3 weeks)

  I will revisit literature on the keystroke model and explore publications on similar research methods.

- Interview Experts (1 week dispersed)

  To guide reading, I will try to get in contact with various experts. Stuart Card, author of the keystroke model, is still publishing his research into IR. I have an existing contact in IR, who has co-authored my submission to SIGIR2007 (Wilson and schraefel, 2007b) and is running the Exploratory Search workshop at CHI2007; I have had a position paper accepted in this workshop (Wilson and schraefel, 2007a).

- Produce Validation Plan (3 weeks) The plan must contain targets to validate and for each, the exact method and expected results must be planned.

**Deliverable**
A written formal plan for validating the evaluation framework.

### 6.1.2 WP2 - Approach Validation Action

Once planned, this activity will be built up of a series of practical and analytical tasks. Other models have been validated through user studies. The GOMS model was validated by Gong and Kieras (1994) using user studies, as was the Keystroke Model by Koester and Levine (1994). The approach used by Koester and Levine was to carry out a between-participants study on the model, one informing the model with random information about the users and the other group using participant-specific information. The accuracy of the model, using time, was calculated using time efficiency. Time efficiency was an appropriate measure as the model aims to reduce the number of keystrokes needed to achieve a goal. In the framework above, however, the approach doesn't necessarily support time efficiency. Instead, it measures what could be described as "Search Efficiency". Thus, if this user study approach was used, each result would have to be validated using a study, where an appropriate measure has been selected to prove the result. Gong and Kieras's validation of the GOMS model, a model which is based on cognitively defined activities and needs, carried out a *before and after* style study. An interface was redesigned, having been informed by applying the GOMS model, and both the learning time and using time were reduced significantly. This holistic approach, rather then per result, may be applied to our model, assuming that an appropriate metric, like learning time, can be measured. It may not be enough to measure the differences in the metrics of this approach, and may need validating with other metrics.

It is clear that, one way or another, the model will be validated through using user studies to show that the results found are correct. In the worst case, this may involve identifying the contribution of each of its results, and validating each one with a different study. However, these results will depend greatly on the interfaces tested, and so each the interfaces tested would have to challenge every dimension of the framework to be effective. This is in turn a challenge to find or perhaps, if necessary, build. It would be ideal to summarise the results into types of contribution and validate these different types, so that the validity can be confidently accepted outside of the test interfaces used.

Another important validation is to speak to experts, or even the original authors of the models chosen, to investigate the strengths and weaknesses that have noted in the models used in the framework above. Discussion of their research since, might enlighten refinements or extensions to the model.

Finally, the continual application of the framework may identify weaknesses in the procedures or results. A number of opportunities are due to arrive during this period. For example, I will be participating in the RichTags project[12], which is due to start in February 2007.

**Method (18 weeks estimate)**

The exact method is not yet known, but will be produced as the output of the Validation Planning. The tasks will be primarily practical: planning, arranging, performing and analysing user studies. This effort will be supported by sources of expertise that include both interviews and extended reading. This estimate assumes potentially three studies that together validate the model. Three have been predicted because of the three types of results that are produced by the framework: tactic analysis (Stage 4), feature analysis (Stage 5) and user condition analysis (Stage 6).

**Expected Tasks**

- Study Planning (1 week minimum per study)

  Once the approach has been identified, a series of studies will be carried out to validate the model. Each study will be motivated by a specific hypothesis, and with this each study can be designed in terms of: method, structure, participants, expected outcome, analysis method. This will require a lot of careful designing and scripting.

- Developing Test Conditions (3 week minimum per study)

  For each study, an independent variable will have to be introduced. In the validation of the GOMS model (Gong and Kieras, 1994), two versions of the interface we compared, one of which was informed by the GOMS model. This must have required developing two versions of the interface. If we used this method to test the validity of our model above on mSpace, mSpace would have to be re-designed, and possibly re-implemented; hopefully prototypes will be sufficient to reduce the amount of development required. Further, this development may require collaboration with other researches who are closely involved with the code. It will be a challenge to design and coordinate the development of variations in the designs. The accessibility of code may also pose a challenge, as both RB++ and Flamenco have been developed by other institutions.

- Performing Studies (1 week per study)

  Carrying out the studies, once carefully planned, will be time consuming. If numerous studies are required, then a large number of participants may be involved, each contributing for, perhaps, an hour. It is vital that the studies are planned carefully, so that no time is wasted.

- Analysing Results (1 week per study)

  Again, the analysis has to be carefully planned, so that time is not wasted on statistically analysing results that produce useless results. The stats

[12]http://mspace.fm/projects/richtags

will be designed ahead so that, once performed, they provide specific information regarding the hypotheses that are being investigated. Even with previous experience in analysing the results of user studies, the process is usually more time consuming and challenging than expected.

**Known Tasks**

- Directed Reading

  Directed reading will be carried out to validate the models included in this framework. While a significant amount of reading has been described above in Section 3, other researchers may have investigated, extended or critiqued the models in recent or on-going publications.

- Community Discussion (2 weeks dispersed)

  I will be attending the SIGIR2007 conference in Amsterdam, where more experienced researchers will be present. I expect there to be relevant presentations and workshops, and even original authors of the models used. This provides opportunity to learn and expand my horizon for this approach.

- Further Application of Framework (2-3 days per interface)

  Weaknesses in the approach and framework may be identified by further applying it to exploratory search interfaces. New designs and search systems are being developed by colleagues for their projects, including enhancements to mSpace.

**Deliverables**

A written report on the validity of the evaluation approach. Ideally, statistical evidence will indicate strengths and potential weaknesses in the approach that will guide refinement and extension.

#### 6.1.3 WP3 - Model Extension

The results from validating the approach should identify missing or weak areas in the evaluation framework. This task is also has specific challenges. The challenge here is related to the method of integration. There are numerous models of information-seeking behaviour that were not used in the current version of the framework. The models used were chosen because of their specific attributes that make them usable in a combined approach. This challenge of integration becomes harder as more models are added, because each model may constrain the ability to integrate with any potential additions. Further, this integration challenge assumes that their is a single model that will contribute directly to the weakness in the combined approach that has been identified by the validation process.

**Method (15 weeks total)**

Further research into published literature will be carried out to identify existing models that can be included to develop the approach, while encouraging synergy. While the approach appears to involve mainly reading, the challenge is in assessing their potential integration into the framework.

**Tasks**

- Directed Reading (8 weeks)

  Based on the results of the previous report, further reading can be directed at specific topics. For example, cognitive load has already been identified as an aspect that affects the synergy of a design. That is, the amount of information and features that are included in a design may make it crowded and subsequently harder to use.

- Documenting Potential (on-going)

  As potential models are identified, they will be documented, including: research area, description, previous validation, potential for extension.

- Comparison of Options (4 weeks)

  Where multiple models may viable options for extending the evaluation approach, the documented differences, advantages and weaknesses can be compared and formatively tested on user interfaces.

- Reporting Results (3 weeks)

**Deliverables**

A report will produced containing: Documented Models, Comparison Analysis, Proposed Changes, and the Extended Model.

#### 6.1.4 WP4 - Related Reading

While my on research is on-going, closely related work is also being produced. In particular, there will be a large amount of related reading included in the Exploratory Search and HCI workshop at CHI2007. Certain key researchers in Exploratory Search, have also recently produced theses that will contain both relevant content and relevant further reading.

### 6.2 Post-Mini-Thesis

In the final stages of my PhD, I will be evaluating the final stages of the approach, designed to evaluate exploratory search interfaces using a validated novel combination of key research. The aim will be to establish the framework as a recognised technique that can be used to evaluate complex and versatile designs in their early stages. This will be especially useful for occasions where the benefits of a design are produced through synergism. The expected activities are described below.

#### 6.2.1 WP5 - Full Model Evaluation

Once the approach has been extended, its new state will have to be evaluated. As the approach will have already been validated carefully, the evaluation may take the form of applying it to a new set of browsers. However, the new additions to the framework may have to be evaluated in the potential ways described above, in which case the number of interacting models may make the validation methods even harder to plan carefully. Hopefully, the general approach to validation will be clear based on the planning done for the mini-thesis.

#### 6.2.2 WP6 - Recommendations for Use

The key challenge here will be to produce guidelines for using the framework. While the process can be defined, it will be useful to have thresholds that can be used to decide on the significance of design strengths and weaknesses. At this time, it is impossible to say what may be used to define this exactly, but it may be produced by analysing the results of continued applications of the framework on exploratory search interfaces.

#### 6.2.3 WP7 - Refinement

The evaluation and preparation of the final framework is likely to identify areas for refining the procedures and documentation involved.

# References


Marcia J. Bates. Information search tactics. *JASIS*, 30:205–4214, 1979.

Marcia J. Bates. Where should the person stop and the information search interface start? *Inf. Process. Manage.*, 26(5):575–591, 1990. ISSN 0306-4573.

Marcia J. Bates. The cascade of interactions in the digital library interface. *Inf. Process. Manage.*, 38(3):381–400, 2002. ISSN 0306-4573.

Steven M. Beitzel, Eric C. Jensen, Abdur Chowdhury, David Grossman, and Ophir Frieder. Hourly analysis of a very large topically categorized web query log. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 321–328, New York, NY, USA, 2004. ACM Press. ISBN 1-58113-881-4.

N.J. Belkin, C. Cool, A. Stein, and U. Thiel. Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems. *Expert Systems with Applications*, 9(3):379–395, 1995.

Tim Berners-Lee, Jim Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):28–37, 2001.

Spring Hull Bonnie Lida and Katie Pilcher. Breadcrumb navigation: An exploratory study of usage. *Usability News*, 5(1), 2003.

Stuart K. Card, Thomas P. Moran, and Allen Newell. The keystroke-level model for user performance time with interactive systems. *Commun. ACM*, 23(7): 396–410, 1980. ISSN 0001-0782.

Ed H. Chi, Peter Pirolli, Kim Chen, and James Pitkow. Using information scent to model user information needs and actions and the web. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 490–497, New York, NY, USA, 2001. ACM Press. ISBN 1-58113-327-8.

Colleen Cool and Nicholas J. Belkin. A classification of interactions with information. In *CoLIS 4: Proceedings of the 4th international conference on Conceptions of Library and Information Science*, pages 1–15, 2002. ISBN 1-59158-016-1.

George H. Forman and John Zahorjan. The challenges of mobile computing. *Computer*, 27(4):38–47, 1994. ISSN 0018-9162.

Marcos A. Goncalves, Edward A. Fox, Layne T. Watson, and Neill A. Kipp. Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries. *ACM Trans. Inf. Syst.*, 22(2):270–312, 2004. ISSN 1046-8188.

Richard Gong and David Kieras. A validation of the goms model methodology in the development of a specialized, commercial software application. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 351–357, New York, NY, USA, 1994. ACM Press. ISBN 0-89791-650-6.

Marti A. Hearst. Next generation web search: Setting our sites. *IEEE Data Engineering Bulletin: Special Issue on Next Generation Web Search*, 2000.

Michiel Hildebreand, Jacco van Ossenbruggen, and Lynda Hardman. /facet: A browser for heterogeneous semantic web repositories. In *ISWC '06: Proceedings of the 5th international conference on the Semantic Web*, 2006.

Ingrid Hsieh-Yee. Effects of search experience and subject knowledge on online search behavior: Measuring the search tactics of novice and experienced searchers. *Journal of the American Society for Information Science*, 44(3): 161–174, 1993.

David F. Huynh, Robert Miller, and David Karger. Enabling web browsers to augment web sites' filtering and sorting functionalities. In *ACM Symposium on User Interface Software and Technology*, 2006.

H. H. Koester and S. P. Levine. Validation of a keystroke-level model for a text entry system used by people with disabilities. In *Assets '94: Proceedings of the first annual ACM conference on Assistive technologies*, pages 115–122, New York, NY, USA, 1994. ACM Press. ISBN 0-89791-649-2.

Sascha Kriewel. Finding and using strategies for search situations in digital libraries. *Bulletin of the IEEE Technical Committee on Digital Libraries (TCDL)*, 2(2), 2006.

Baolong Liu and Douglas W. Oard. One-sided measures for evaluating ranked retrieval effectiveness with spontaneous conversational speech. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 673–674, New York, NY, USA, 2006. ACM Press. ISBN 1-59593-369-7.

Bonnie MacKay, David Dearman, Kori Inkpen, and Carolyn Watters. Walk 'n scroll: a comparison of software-based navigation techniques for different levels of mobility. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 183–190, New York, NY, USA, 2005. ACM Press. ISBN 1-59593-089-2.

Gary Marchionini. Exploratory search: from finding to understanding. *Commun. ACM*, 49(4):41–46, 2006. ISSN 0001-0782.

Catherine C. Marshall and III Frank M. Shipman. Spatial hypertext and the practice of information triage. In *HYPERTEXT '97: Proceedings of the eighth ACM conference on Hypertext*, pages 124–133, New York, NY, USA, 1997. ACM Press. ISBN 0-89791-866-5.

Michael J. McGuffin and m.c. schraefel. A comparison of hyperstructures: zzstructures, mspaces, and polyarchies. In *HYPERTEXT '04: Proceedings of the fifteenth ACM conference on Hypertext and hypermedia*, pages 153–162, New York, NY, USA, 2004. ACM Press. ISBN 1-58113-848-2.

Sachi Mizobuchi, Mark Chignell, and David Newton. Mobile text entry: relationship between walking speed and text input task difficulty. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 122–128, New York, NY, USA, 2005. ACM Press. ISBN 1-59593-089-2.

Christopher Olston and Ed H. Chi. Scenttrails: Integrating browsing and searching on the web. *ACM Trans. Comput.-Hum. Interact.*, 10(3):177–197, 2003. ISSN 1073-0516.

A. M. Pejtersen. A library system for information retrieval based on a cognitive task analysis and supported by an icon-based interface. In *SIGIR '89: Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 40–47, New York, NY, USA, 1989. ACM Press. ISBN 0-89791-321-3.

N. Pharo. A new model of information behaviour based on the search situation transition schema. *Information Research*, 2004.

Peter Pirolli and Stuart Card. Information foraging in information access environments. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 51–58, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-84705-1.

George Robertson, Kim Cameron, Mary Czerwinski, and Daniel Robbins. Polyarchy visualization: visualizing multiple intersecting hierarchies. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 423–430, New York, NY, USA, 2002. ACM Press. ISBN 1-58113-453-3.

S. E. Robertson and M. M. Hancock-Beaulieu. On the evaluation of ir systems. *Inf. Process. Manage.*, 28(4):457–466, 1992. ISSN 0306-4573.

J. Rocchio. Relevance feedback in information retrieval. *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323, 1971.

Tefko Saracevic. The stratified mode of information retrieval interactino: Extension and applications. In *Proceedings of the American Society for Information Science*, volume 34, pages 313–327, 1997.

m.c. schraefel and David Karger. The pathetic fallacy of rdf. In *International Workshop of the Semantic Web and User Interaction (SWUI06)*, 2006.

m.c. schraefel, Max L. Wilson, and Maria Karam. Preview cues: Enhancing access to multimedia content. Technical report, School of Electronics and Computer Science, University of Southampton, August 2004. http://eprints.ecs.soton.ac.uk/9253/.

m.c. schraefel, Max Lloyd Wilson, Alistair Russell, and Daniel A. Smith. mspace: improving information access to multimedia domains with multimodal exploratory search. *Commun. ACM*, 49(4):47–49, 2006. ISSN 0001-0782.

m.c. schraefel, Yuxiang Zhu, David Modjeska, Daniel Wigdor, and Shengdong Zhao. Hunter gatherer: interaction support for the creation and management of within-web-page collections. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 172–181, New York, NY, USA, 2002. ACM Press. ISBN 1-58113-449-5.

Taowei David Wang and Bijan Parsia. Cropcircles: Topology-sensitive visualization of owl class hierarchies. In *International Semantic Web Conference (ISWC06)*, 2006.

Ryen W. White, Bill Kules, Steven M. Drucker, and m.c. schraefel. Introduction. *Commun. ACM*, 49(4):36–39, 2006a. ISSN 0001-0782.

Ryen W. White, Gheorghe Muresan, and Gary Marchionini. Report on acm sigir 2006 workshop on evaluating exploratory search systems. *SIGIR Forum*, 40(2):52–60, 2006b. ISSN 0163-5840.

Ryen W. White, Ian Ruthven, and Joemon M. Jose. A study of factors affecting the utility of implicit relevance feedback. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 35–42, New York, NY, USA, 2005a. ACM Press. ISBN 1-59593-034-5.

Ryen W. White, Ian Ruthven, Joemon M. Jose, and C. J. Van Rijsbergen. Evaluating implicit feedback models using searcher simulations. *ACM Trans. Inf. Syst.*, 23(3):325–361, 2005b. ISSN 1046-8188.

Max L. Wilson, Alistair Russell, Daniel Smith, and m.c. schraefel. mspace mobile: Exploring support for mobile tasks. In *British Conference on Human-Computer Interaction 2006*, 2006.

Max L. Wilson and m.c. schraefel. mspace: What do numbers and totals mean in a flexible semantic browser. In *The 3rd International Semantic Web User Interaction Workshop*, 2006.

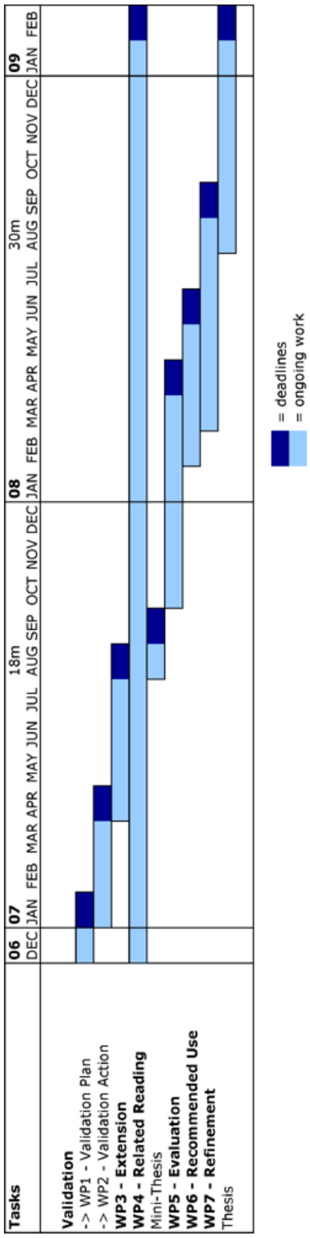
Max L. Wilson and m.c. schraefel. Bridging the gap: Using ir models for evaluating exploratory search interfaces. In *First Workshop on Exploratory Search and HCI at SIGCHI07*, 2007a.

Max L. Wilson and m.c. schraefel. A combined model for evaluating search interfaces. In *Submitted to ACM Conference on Information Retrieval 2007*, 2007b.

Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. Faceted metadata for image search and browsing. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408, New York, NY, USA, 2003. ACM Press. ISBN 1-58113-630-7.

Junliang Zhang and Gary Marchionini. Evaluation and evolution of a browse and search interface: relation browser. In *dg.o2005: Proceedings of the 2005 national conference on Digital government research*, pages 179–188. Digital Government Research Center, 2005.

# A Mini-Thesis Gantt Chart



## B Community Participation

I am also becoming active in the various communities relating to my research. In November 2006, I was ask co-chair for the International Semantic Web Conference Poster Session, in Athens, GA, USA. I attended this and other conferences (ISWC2005, HCI2006, CHI2006, JCDL2006). Having spoken to Gary Marchionini, a prolific academic in information sciences, at CHI2006, I visited JCDL2006, which was being held at his institution in North Carolina. During this conference I met with a number of parties interested in search behaviour and have set up a distributed discussion group to manage further collaboration between us[13]. My work and submission to CHI2007 has been noted and I have been selected to review a paper with similar IR based content.

My work with these groups will help support different aspects of my research. By maintaining effort in the Semantic Web group, I will develop my knowledge of data connectivity for supporting the underlying structures that I hope will support my developing search techniques. CHI is a very important source of work on interaction and is my primary research community. Also, the community that gather at JCDL and also SIGIR, are most knowledgeable about IR and the challenges accessing large datasets. Ryen White, at Microsoft Research, is a key researcher in IR and wrote the lead article for the CACM on ES. He read and commented on my CHI submission and we have discussed future collaboration.

Finally, I will soon be visiting my supervisor, while she on sabbatical at MIT. With this I have established a connection with academics some advanced search interfaces and semantic web developments, so that I can potentially collaborate with similar interests. In particular I am meeting with David Huyhn, who has been working on Piggy Bank.

[13]http://sag.ecs.soton.ac.uk/