Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2002), *16*, 243–257. Printed in the USA.
Copyright © 2002 Cambridge University Press 0890-0604/02 $12.50
DOI: 10.1017.S0890060402163098

No zero match browsing of hierarchically categorized information entities

CHRIS MCMAHON,[1] ROSE CROSSLAND,[2] ALISTAIR LOWE,[1] TULAN SHAH,[2]
JON SIMS WILLIAMS,[2] AND STEVE CULLEY[1]

[1]Department of Mechanical Engineering, University of Bath, Bath BA2 7AY, United Kingdom
[2]Faculty of Engineering, University of Bristol, Bristol BS8 1TR, United Kingdom

(Received October 11, 2001; Accepted March 26, 2002)

Abstract

The two main ways to search for information in electronic document collections are by using free-text retrieval search engines or browsing information that has been organized into predefined organizational structures. However, each of these approaches has limitations. Using word or phrase search, users are faced with a compromise between overly broad searches returning an excessive amount of information or overly narrow searches that may fail to return relevant information. Browsing organizational structures is dependent on the user's knowledge of the structures, and a user may find it difficult to refine searches. This paper introduces a user interface based approach to the browsing of hierarchically organized information entities that avoids these problems by allowing the incremental narrowing down of a set of search results and by pruning the organizational structure after each user selection to show the consequences of the selection. The effect is to present to the user at all times only that part of the organizational structure that will lead to a nonnull selection. The approach is called no zero match (NZM) browsing. The paper presents the computational basis of NZM browsing before describing a trial implementation of the approach and presenting three case studies, which represent common search situations in an engineering context.

Keywords: Information Retrieval; Hierarchical Classification; User Interface; Engineering Document Management

1. INTRODUCTION

Some have argued that we are living in a postindustrial era with a progressive transition to a society "fueled" by information and populated by knowledge workers (Drucker, 1994; Hayman & Elliman, 2000). This is debatable; however, there is consensus in the literature that information overload is becoming a real problem in a broad range of disciplines, such as engineering, medicine, commerce, computing, etc. (Reuters, 1996; Edmunds & Morris, 2000). While computer technologies have made the *creation* of documents easier and provided the means to open up huge digital collections to searching, in some ways this has paradoxically made the *location* of relevant information more difficult.[1] Drucker (1994, 2000) contends that one of the most important challenges in the twenty-first century will be increasing the productivity of those involved in knowledge intensive industries. This is a challenge that can clearly be addressed by the use of improved information search and retrieval tools.[2]

In a recent review of usability issues related to information retrieval systems, Hearst (1999) notes that:

> the human–computer interface is less well understood than other aspects of information retrieval. . . . The human–computer interface is the most rapidly developing area of information access today and improvements in the interface are likely to lead the way toward better

Reprint requests to: Chris McMahon, Department of Mechanical Engineering, University of Bath, Claverton Down, Bath BA2 7AY, UK. E-mail: c.mcmahon@bath.ac.uk

[1] An *information entity*, in the context of this paper, is used to describe any information object, such as a document, electronic file, or database record. An *information collection* is used to describe a collection of records in a database or documents or files stored on a computer network.

[2] The distinction between the use of the terms *knowledge management* and *information management* is contentious and out of the scope of this paper. However, we would argue that technical reports, informal documents, CAD models, company procedures, and similar items are best described as information objects rather than knowledge objects. (Knowledge is considered to be more associated with the interpretation of information.)

search results and better-enabled information creators and users.

The field of human–computer interaction (HCI) is a broad one, and this article only touches on a small subset of pertinent issues. As such, it is not the intention to focus on classical "front-end" HCI and usability concepts. Instead, we hope to make a contribution by presenting a new highly interactive and intuitive user interface (UI) based approach to the browsing of hierarchically organized information entities. This approach, embodied in a prototype engineering system, facilitates the interaction between human information seekers, information retrieval systems, and repositories of information.

1.1. Information retrieval strategies

Several models of information retrieval strategy have been proposed in the literature (e.g., Bates, 1986; Waterworth & Chignell, 1991; Meadow, 1992). Meadow's list of search types includes the categories of known item search, specific information search, general information search, and database exploration. These can be fulfilled by two fundamentally different approaches to information retrieval (Chen et al., 1998):

- *Keyword-based directed searching*: At search time, users enter sets of free-text terms and phrases that, in their opinion, best characterize their information needs (e.g., the AltaVista and Google internet search engines at www.altavista.com, www.google.com, etc.).
- *Browsing*[3] *organized information*: In this process, information is organized into predefined organizational structures (typically, hierarchical subject categories) and users browse the organizational structures to search for information (e.g., the hierarchical display in Microsoft Explorer or the Yahoo internet directory).

1.1.1. Keyword-based directed searching

Keyword-based directed searching has traditionally been the most common search in computer-based systems. This approach is particularly suitable when a user's need for information can be easily translated into a text string that can be interpreted by the computer system. However, without taking into account the semantic content of the document, it is often difficult to precisely specify a query unless it is very straightforward or the user is familiar with the information that is being searched. In addition, for users searching a structured database, familiarity with the terms used in the database is important, as is knowing how to refine a query in the event of an unsuccessful search. Typically, poorly formed queries lead to highly indiscriminate search results (either far too many hits in relation to a search or no hits at all). However, studies have shown that system users are generally poor at formulating suitable queries, even when the system allows for the construction of complex queries (Waterworth & Chignell, 1991; Anick, 1994). This tendency toward indiscriminate results can be explained by considering the precision and recall of search strategies. Precision and recall are frequently used to measure the effectiveness of information retrieval, and there is always a trade-off between the two (Cleverdon, 1967). *Precision* is a measure of the proportion of relevant documents in the results set that are presented to users, whereas *recall* is a measure of the proportion of all the relevant documents that are contained within the results set.[4] Generally speaking, keyword-based search strategies have a tendency to return high-recall but low-precision results.[5] Nielsen (1999), in discussing usability issues related to the web, notes that, "On the Web, nobody will ever have the time to read all the relevant documents . . . it is more important to guide the user to a small number of high quality documents than to achieve completeness."

1.1.2. Browsing of classifications

Allowing users to browse classified information in a meaningful way generally produces a smaller, more precise results set. Browsing is particularly suitable for performing less focused, complex searches or those that are difficult to translate into a single query. Users are given the freedom to explore the organizational structures of meaningfully organized document collections; although reducing a user's problems, this does not eliminate them. Browsing typically involves traversing of hierarchically arranged categories. If an information entity is uniquely classified into a single category (e.g., in a library or a computer file structure), then the user may fail to select the correct branch of the categorization hierarchy into which the information of interest has been located.[6] Alternatively, if information can be classified into multiple categories, then the number of documents in each category will be larger and the number of nodes in the hierarchy (i.e., determined by the depth and breadth of the hierarchy) will have to be increased to provide a suitable level of granularity. In addition, the user is again faced with the problem that if a search has retrieved

[3] Locating or acquiring information without necessarily knowing of the existence or the format of the information being sought.

[4] Blair (1980) proposed the concept of futility point to describe people's attitudes toward recall. The number of documents through which users will be willing to browse before giving up is their futility point. Depending on the nature of the document collection, this can vary between 10 and 30 for many users.

[5] Note that useful metrics beyond precision and recall are often required to assess systems. These might include the time to learn a system, time to meet a benchmark search or retrieval task, error rates, and so on. However, human interactions with modern computer systems are complex and particularly difficult to measure and characterize. Nielsen (1993) presents an overview of appropriate usability and assessment methods.

[6] This problem is illustrated by the inconsistency with which even highly trained librarians classify books (Larson, 1992).

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

too few or two many documents, they do not know how to refine their search (i.e., in terms of other nodes in the hierarchy that might be worth investigating) to achieve a better result.

1.1.3. Other retrieval applications

A number of other computing applications incorporate document retrieval, particularly, for example, in case-based reasoning (CBR), in which a description of a new problem is compared with case bases of precedent cases (for example, concerning the description of fault conditions or design concepts) to identify potential candidates for solving the problem (Maher & Pu, 1997). Where the case base comprises text documents, comparison may be made using standard information retrieval techniques such as term–frequency matching and similarity vectors (Lenz & Burkhard, 1997). These techniques are limited in the extent to which they allow reasoning about similarities between cases in textual CBR, and the derivation of better case representations using natural language processing methods is a current research topic (Brüninghaus & Ashley, 2001).

1.2. No zero match (NZM) browsing

In this paper an improved system that allows a novel approach to the browsing of hierarchically classified information entities is presented. This approach carries out set-theoretic manipulation of the classification hierarchies presented to the user to ensure that the computer never presents a null result set. For this reason, the approach is termed NZM browsing. NZM browsing involves the user selecting nodes of interest from a classification hierarchy (against which documents have been classified) on a graphical display. As each additional node of interest (or combination of nodes) is selected, the hierarchical display is pruned and the number of entities associated with the nodes in the hierarchy is updated to reflect the selections made by the user. In this way, the user is guided by the graphical display in the selections that may be made, and there is no possibility of ending with a null result.

The NZM approach is essentially a hybrid of existing browsing and directed search strategies. Users initially browse the hierarchically arranged concepts in the graphical display. However, when a concept is selected, the system can be considered to be performing a directed search on the classified documents using the selected concept as the search criterion. This process of browsing and searching can be repeated until the document results set features a suitably small number of documents that the user will be prepared to look through. The key distinction is that in a directed search process the user must specify all of the search criteria that relevant documents are expected to meet (in terms of the words and phrases that they must contain) in a *single step*. The NZM approach allows the user to *incrementally* build up their search criteria[7] by selecting combinations of concepts from a browsable graphical display. In addition, prior to the user selecting the concepts, the system interface provides an indication of how discriminating each concept is as a search criterion by showing the number of documents that will be included in the results set if a given concept is selected. In this paper alternative approaches to query refinement in structured and unstructured information sets are first described. The computational basis of NZM browsing is then presented before a trial implementation of the approach is described. Three case studies are introduced: in the first, an application of the approach with precisely categorized documents, and in the second and third, following automatic document classification at various levels of precision. A discussion on performance and implementation issues concludes the article.

2. BACKGROUND

The work reported in this paper originated from a research program with three aerospace companies, which sought to assist design engineers with the organization of and access to their technical and commercial documents (McMahon et al., 1998). The requirement for improved search strategies has been confirmed by background work undertaken by the authors and reported in Lowe et al. (2000) and Court et al. (1996). It is best illustrated by considering some specific examples shown in Table 1 that will be discussed throughout the paper. These three cases represent instances of the search strategies identified by Bates (1986), Waterworth and Chignell (1991), and Meadow (1992) and applied, in the present case, to the engineering design domain.

Each of these cases describes a different search problem in which traditional search techniques provide insufficient assistance to the user. Without detailed knowledge of the retrieval environment, users have difficulty in forming queries that are well designed for retrieval purposes. In case 1 the database does not indicate which, if any, of the user's query terms are matched in the database and which other records may be a close match to the entered terms. In case 2, the search tool does not indicate which search terms the user might add to identify the most relevant documents among the 9000 identified. Finally, in case 3 the user needs to have an understanding of the company intranet's organizational structure in order to succeed in document retrieval. These examples will be considered in more detail in Section 5 when considering the particular features of the NZM tool.

2.1. Approaches to query refinement

Many approaches have been identified for query refinement, especially in combination with the free-text retrieval

[7] Users can also effectively relax their search criteria at any time by deselecting concepts. This has the effect of increasing the number of documents in the results set that are presented to users.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

Table 1. *Examples of typical search strategies in engineering design domain and problems commonly encountered*

Example	Description of Initial Search	Problems Encountered
Case 1: collection of database records (highly structured "documents")	An engineer is searching a catalog for components with physical and performance characteristics that are suitable for a particular application. However, upon querying the database with a query related to the *ideal* requirements, the system responds with an error message that there are no components that *exactly match* the query.	Unfortunately the engineer does not know how to refine the search to find components that may be *suitably close* to the requirements for the given application.
Case 2: "grey literature" on the internet (unstructured documents)	An engineering designer wants to find technical papers on the internet relating to quality management approaches in design. The query ⟨"quality management" AND "engineering design"⟩ is submitted to a major search engine, and over 9000 documents are returned, many of which do not appear to be relevant.	In this instance, the designer is not sure how to refine the search so as to retrieve a more focused set of search results that better matches his or her needs.
Case 3: company intranet (loosely structured documents)	A project manager is searching on the company intranet for an item of correspondence (written by a colleague who is on leave) addressed to a supplier, which is related to a particular project and technical issue. Documents are organized in a folder structure on the network, but it is not known which folder will contain the document of interest.	The manager does not know whether the colleague has organized documents into the supplier name folder, the project name folder, or the technical domain folders.

of unstructured information entities. Relevance feedback is the most well established query reformulation strategy and has been found to lead to improvements in the performance of keyword-based retrieval systems based on a number of information retrieval models (Robertson & Sparck Jones, 1976; Salton & Buckley, 1990). In a relevance feedback cycle, users are presented with lists of retrieved documents and, after examining them, mark up those that best meet their information needs. Improvements come from two mechanisms: *query expansion* (the addition of new search terms from relevant documents or from thesauri that allow the identification and substitution of synonyms) and *term reweighting* (the modification of original term weights based on the user's relevance judgment).

Research related to HCI has been one of the most rapidly developing areas of aids to query refinement (Hearst, 1999), although there is a debate on the direction that future technologies should take (Institute for Systems Research, 2001). Technological advancements now permit the presentation to the user of search results and feedback in an interactive graphical manner, for example, showing the relationship between information entities and categorizations and between the categorizations themselves. In contrast, conventional systems present search results in the form of a simple ranked list. In most cases these advanced interactive approaches have aimed at improved means of browsing information, in particular through exploiting relationships between concepts (although the approaches typically incorporate relevance feedback techniques in addition to improved visualization techniques). A number of systems that exploit advanced UI technologies in their presentation of information retrieval results are outlined below.

2.1.1. *Custom folders*

The NorthernLight search engine (www.northernlight.com) uses a patented approach (Krellenstein, 1999) in which information entities retrieved by a keyword search are organized into custom folders (where the folders represent the nodes in a hierarchy) and these folders are presented to users so that they may be browsed to refine the selection. The chosen hierarchy of categories will reflect the users' information requirement as represented by the choice of initial keyword search terms. NorthernLight folders may be organized into subfolders, and only folders that contain documents are displayed to users. However, it is not possible to explore multiple branches of the hierarchy or cross-reference between branches. Two other search engines that also attempt to categorize hits into categories of folders include Vivísimo (www.vivisimo.com) and Wisenut (www.wisenut.com).

2.1.2. *Database forms*

In many database search applications, the user enters a search by completing boxes in a form. The limitation with such "linked-box" interfaces to databases is that selections have to be made in a particular order. Essentially, selections

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

are dictated by the design of the underlying data structure and may be different from the more arbitrary sequence in which users may wish to make them. Presenting to a user (for selection, for example, from a menu box) the terms that may be used in a query and then updating the displayed term list to reflect the selections already made may be more intuitive. This approach also addresses the issue of querying a structured database and achieving a null result. It is the approach taken in the Review system (McMahon et al.,1995). In this system each information entity is associated with a set of attribute–value pairs. Search is carried out by the user selecting the desired attributes and values from on-screen lists to give filtered sets of information entities. The attribute–value pairs presented to the user for search refinement are updated to show only those pairs that are referenced by the filtered set of entities. After each selection is made, the filtered set and the attribute and value lists presented to the user are updated to reflect the previous selections. Attributes may be selected for query in any desired order, so, for example, in the component selection task described above, the user would select the most important component parameters first and could then search on the other parameters for these selected components. In the review at each selection stage the user may select one or more attribute–value pairs and the filtered document set is the intersection between the previous filtered set and the union of the document sets for each newly selected attribute–value pair.

2.1.3. Hierarchy refinement

A similar principle to review is incorporated in the active navigation portal engine (www.multicosm.com) developed from the Microcosm system (Davis et al., 1992; Hall et al., 1996; Crowder et al., 1999). Active Navigation indexes documents into topics or themes. When a search is made by keyword, the most important themes referenced by the selected set of entities are presented to the user for refinement of the selection. As each selection is made, the topic list is updated. At each selection stage the filtered document set is the intersection between the previous document set and the document set associated with the newly selected theme. Hearst and Karadi (1997) describe Cat-a-Cone 3D, an interactive interface for specifying searches and viewing retrieval results using a large category hierarchy. This incorporates a browsing tool in which only parts of the hierarchy relevant to a current document set are displayed. A similar approach is made in the querying of database contents by HiBrowse (Pollitt, 1997), in which precoordinated indexing and coordinated viewing are used to assist in query refinement for entities stored using a relational database in a faceted classification.

3. NZM BROWSING OF CONCEPT HIERARCHIES

The NZM browsing approach presented here builds on that incorporated in Review (see Section 2.1.2) by extending the concept of filtering from attribute–value pairs to hierarchically decomposed classification categories that we term *concept hierarchies*. These networks may be constructed by the manual or automated classification of information entities.

3.1. Terminology and basic concepts

The purpose of the NZM approach is to support the search and retrieval of information entities, where an information entity is an object such as a document, file, or database record. Information entities are organized for the purposes of retrieval into concept hierarchies, where a concept is a category used for the purposes of categorizing information entities and is represented by a descriptive string. Concepts are organized into parent–child hierarchies representing taxonomies of concepts in particular subject domains. Each concept may have multiple parents; hence, the structures used are strictly concept networks, but the networks are visually represented to the user by trees, where a concept with multiple parents has multiple representations, each appearing below one of its parents. Each concept may have zero, one, or many child concepts. There may be one or many concept hierarchies used to categorize any given set of information entities.

Each information entity may be associated with any number of concepts in the concept hierarchies, from zero upward. Figure 1 shows a representation of a concept hierarchy together with a small number of information entities and the relationships that associate them with nodes in the concept hierarchy.

3.2. NZM browsing

The NZM browsing of hierarchically classified information entities involves selecting concepts of interest from a visual display of all the concept hierarchies relevant to an initial set of information entities. Each displayed concept in the initial concept hierarchy is either directly associated with at least one information entity, or at least one of its child or descendent concepts is directly associated with at least one

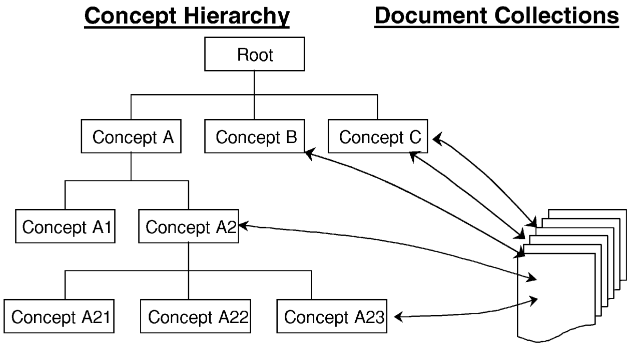


Fig. 1. A concept hierarchy and classified information entities.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

information entity. By selecting one or more concepts from one or more concept hierarchies, the user selects a set of one or more information entities associated either with those concepts or with the children or descendants of those concepts.

By selecting a set of concepts, which we term a *concept selection set*, the user selects the *filtered set of information entities*, which will always be a subset of the initial set of information entities but will never be null if the user is only presented for selection with concepts for which documents are associated. A second, filtered, set of concepts may be generated from this filtered set of information entities by computing the union of all the concepts associated with the filtered set of information entities.

We term this the *filtered set of concepts*. In NZM browsing of the concept hierarchies, this filtered set is then used to generate the display of the concepts that may be selected by the user to refine the search. Search refinement is achieved by selecting from the filtered set of concepts one or more further concepts, comprising a further concept selection set. In doing so, the user selects and is presented with the set of information entities that is the intersection of the set of information entities associated with the new concept selection set and the set of information entities associated with the previously selected concept selection sets.

This is a new *filtered set of information entities*, and this set will again always be smaller than or equal to the initial filtered set of information entities, but it will never be null. This is shown, for a new concept selection set containing two concepts, in Figure 2. A new filtered set of concepts may again be generated from this filtered set of information entities by computing the union of all of the concepts associated with the new filtered set of information entities. The display of selectable concepts is again updated to show this new filtered set of concepts, and the user may then select one or more further concepts to refine the search. This process may be repeated as required.

The effect from the point of view of the user is that he or she is selecting concepts of interest from a hierarchical display of concepts. The UI offers the possibility of selecting concepts individually or in groups called *selection sets*. Where concepts are selected as a selection set with several members, this implies that the user is interested in information entities associated with any of the concepts in the set (i.e., a logical OR of the selected concepts). When a new individual concept or selection set of concepts is selected, this implies that the user is interested in information entities associated with both the previous selection sets and with the new selection set or individual concept (i.e., a logical AND of the selected concepts). In the practical implementation, as will be explained in the next section, the user is assisted in the selection by the display with each concept of the number of information entities associated with it, or the number of entities associated with the concept and all of its child or descendent concepts. The document titles and/or contents may also be displayed. As concepts are selected from the hierarchy, the hierarchical display is pruned and the number of documents listed is updated to reflect the user's selection. In this way, the user is guided by the graphic display in the selections that may be made, and there is no possibility of ending without any search results.

4. BUILDING OF A NZM INDEX

A computer implementation of the NZM browsing application has been developed as part of the project described earlier (McMahon et al., 1998). In this implementation, the information entities are documents which include word processor documents, HTML pages and other document file types. These documents are first classified into multiple categories, and the categorization and category–document relationships are stored in a database repository. A NZM index tool is used to build an index of the concepts and documents in the repository. This index is then used in a

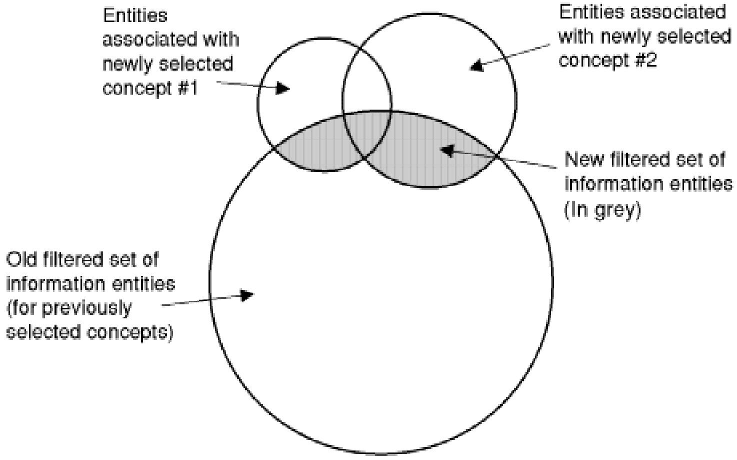

Entities associated with newly selected concept #1
Entities associated with newly selected concept #2
New filtered set of information entities (In grey)
Old filtered set of information entities (for previously selected concepts)

Fig. 2. The new filtered set of information entities, when a concept selection set of {concept#1, concept #2} is added to the existing set of selected concepts.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

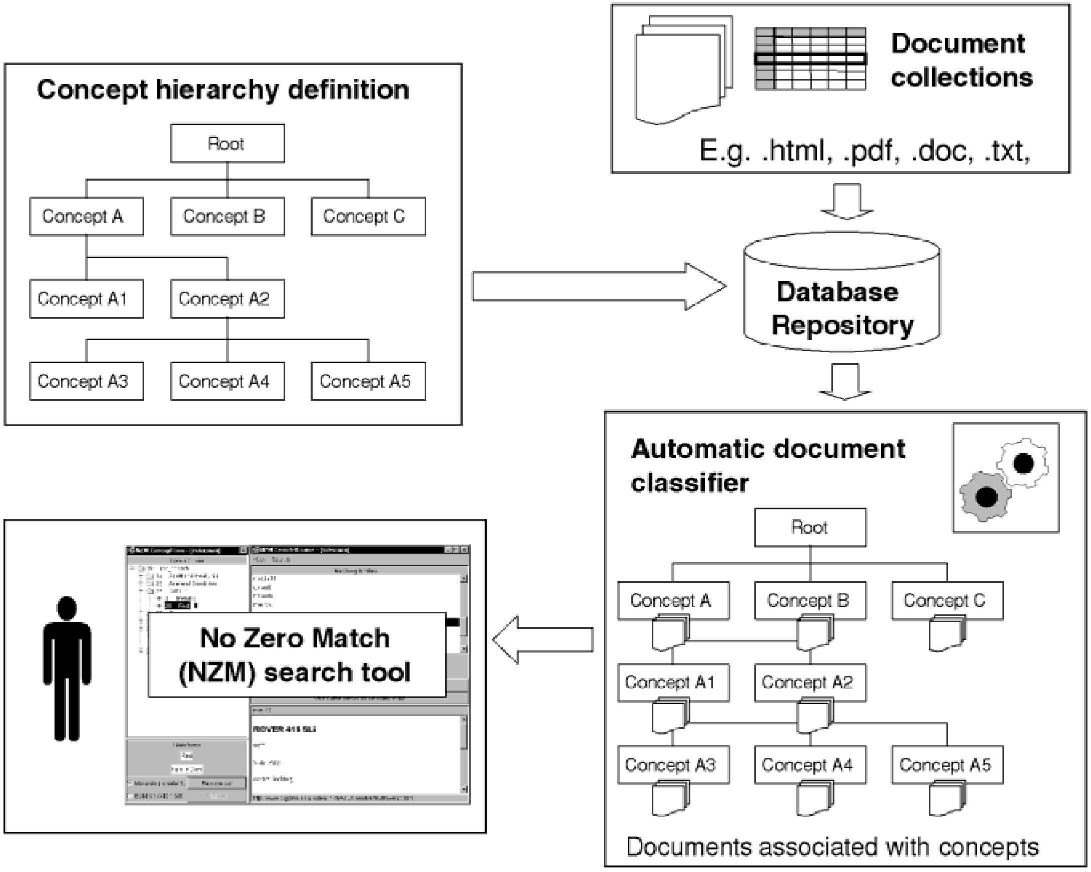


Fig. 3. Elements in the no zero match browsing system.

NZM application to allow browsing of the categorizations using a drill-down folder display. The index may be distributed independently of the underlying database (e.g., for static document sets such as on a CD-ROM). The relationship between these system elements is shown in Figure 3.

4.1. The NZM index

The concept-information entity database is used to construct an index (the NZM) that includes a matrix of bits representing relationships between concepts and information entities. Each row in the matrix represents an information entity, and each column represents a concept. A bit in the matrix is set to 1 if the relevant concept is associated with the relevant information entity. An additional row of bits identifies which concepts are currently in the filtered set of concepts; this is termed the *used concept* row. An additional column of bits identifies which entities are currently members of the filtered set of information entities; this is termed the *used entity* column. The NZM index also includes a 1-dimensional array of objects representing the concepts, and pointers between them represent the concept structuring relationships and a 1-dimensional array of objects representing the information entities. Figure 4 shows all of these data structures.

4.2. The NZM application UI

Within the literature there are many suggested guidelines to help software developers improve the usability of computer applications (e.g., Nielsen, 1993; Shneiderman, 1997).[8] Where possible, these have been taken into account when designing the graphical UI (GUI) for the NZM application. In addition, attempts have been made to provide a human–computer interface with a similar "look and feel" to commonly used applications with which most users are already familiar (the free availability and ease of adaptability of standard GUI components, such as Java's SWING classes, helps in this regard).

The prototype NZM application has a UI comprising four panes, as shown in Figure 5. The first pane (the *concept hierarchy pane*) shows browsable concept hierarchies with tree controls similar to those found in common file managers. In addition to the concept names, this hierarchy records the number of documents within the current filtered list, referenced by each concept node or by a concept node and all its children and descendants (selectable by radio but-

[8] Common examples of such guidelines include reducing the user's working memory load, offering informative feedback, offering shortcuts for expert users, and so on.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

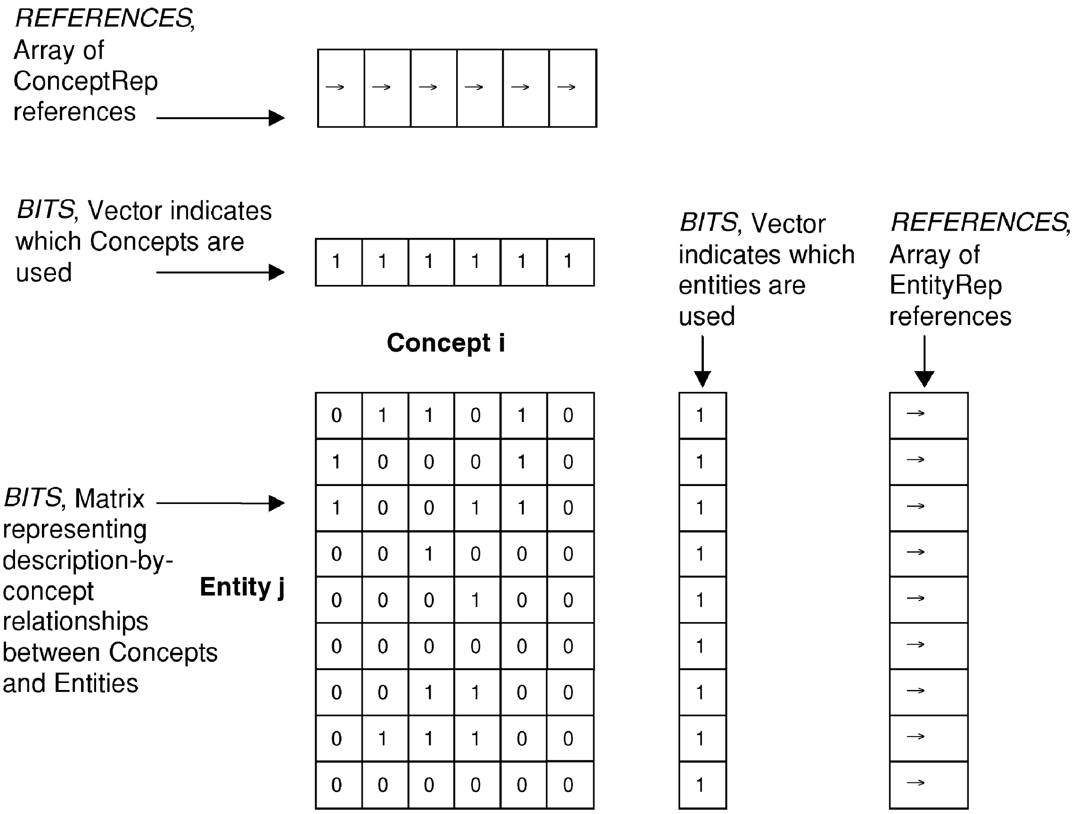

REFERENCES, Array of ConceptRep references
BITS, Vector indicates which Concepts are used
1 1 1 1 1 1
Concept i
BITS, Vector indicates which entities are used
REFERENCES, Array of EntityRep references
BITS, Matrix representing description-by-concept relationships between Concepts and Entities
Entity j
0 1 1 0 1 0
1 0 0 0 1 0
1 0 0 1 1 0
0 0 1 0 0 0
0 0 0 1 0 0
0 0 0 0 0 0
0 0 1 1 0 0
0 1 1 1 0 0
0 0 0 0 0 0
1
1
1
1
1
1
1
1
1

Fig. 4. The NZM index data structures.

ton). The second pane (the *concept selection set pane*) shows the members of the current concept selection sets. The third pane (the *filtered document set pane*) shows a list of document titles. The fourth pane (the *document content pane*) shows the contents of the document currently selected in pane 3. Additional controls include a facility to display the concepts into which the currently highlighted document is indexed, a facility to modify the tiling of the panes, and a capability to search for words and phrases in the concept hierarchy, the document titles, or the document full text contents.

In the concept selection set pane, the user may choose between two modes of operation: "make single selections" or "build selection set." In the first mode, when a selection is made in the concept hierarchy pane, this is assumed to comprise a complete concept selection set containing just one element, and so the filtered list in the filtered document set pane is updated immediately, as is the concept hierarchy in pane 1, to reflect the selection. In build selection set mode, the user may repeatedly make selections in the concept hierarchy pane, building up a concept selection set with several members; it is not until he or she selects the "add set" button that the new concept selection set is added and the filtered document set pane and concept hierarchy pane are updated to reflect the addition of the new concept selection set. Color coding is used to relate the box borders in the concept selection set pane with the icons in the concept hierarchy pane and hence to indicate to which selection sets the concepts belong.

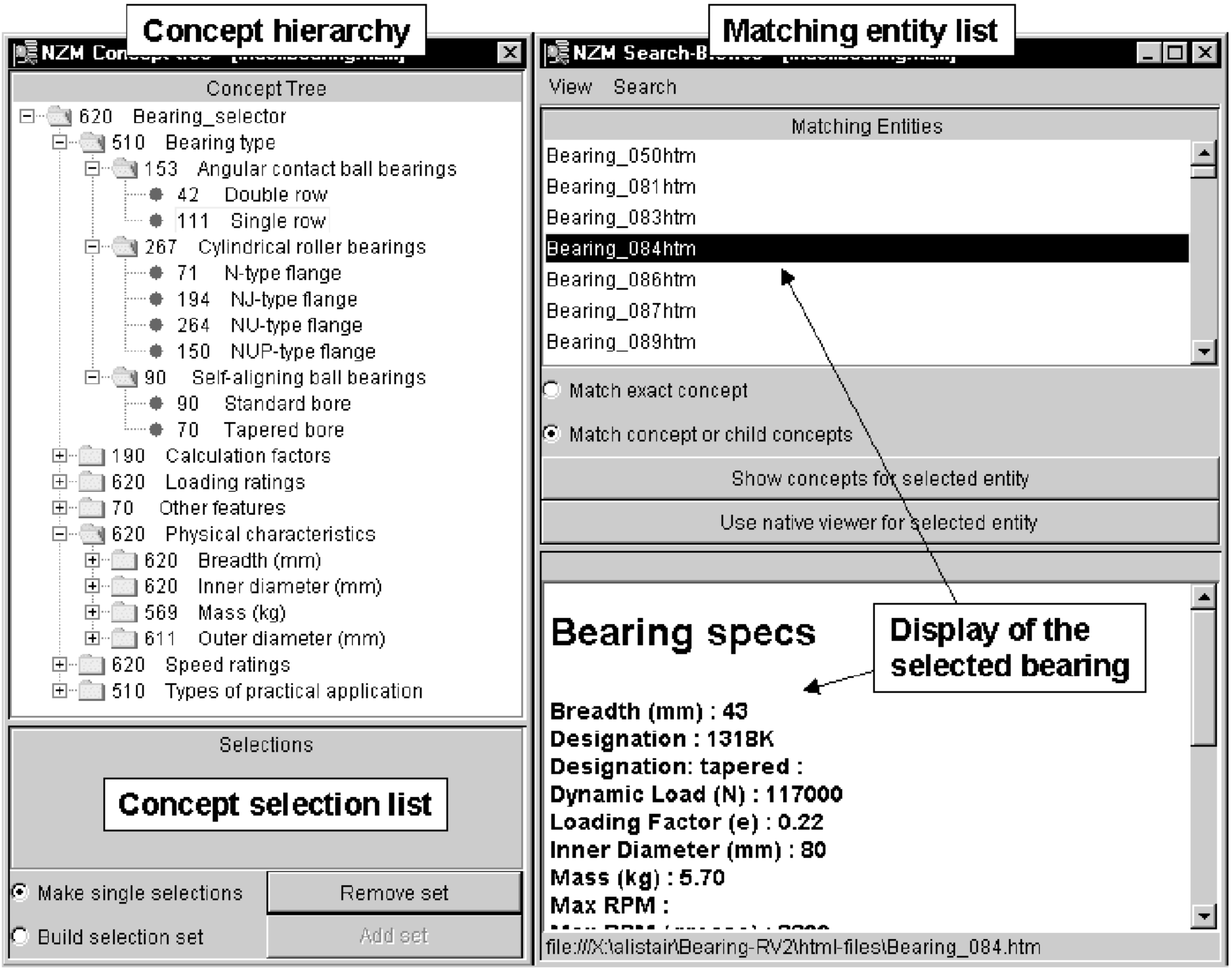
5. THE APPLICATION OF NZM TO DIFFERENT SEARCH PROBLEMS

The NZM browser and the associated document classification systems have been developed as part of a research program with three aerospace companies, aimed at assisting design engineers with the organization and browsing of their technical and commercial documents. In addition to these applications the approach has been applied to automobile advertisements, fault diagnosis data, the papers from an international conference (Culley et al., 2001), catalogue data, and sets of web pages identified by conventional search. The latter two applications, together with an application from one of the aerospace companies, correspond to the three cases described in Section 2 and form the examples described in this section.

5.1. NZM application with precisely categorized catalogue entities

In the first example, data relating to rolling element bearings are indexed into hierarchical categories describing bear-

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press



Concept Tree

- 620 Bearing_selector
  - 510 Bearing type
    - 153 Angular contact ball bearings
      - 42 Double row
      - 111 Single row
    - 267 Cylindrical roller bearings
      - 71 N-type flange
      - 194 NJ-type flange
      - 264 NU-type flange
      - 150 NUP-type flange
    - 90 Self-aligning ball bearings
      - 90 Standard bore
      - 70 Tapered bore
  - 190 Calculation factors
  - 620 Loading ratings
  - 70 Other features
  - 620 Physical characteristics
    - 620 Breadth (mm)
    - 620 Inner diameter (mm)
    - 569 Mass (kg)
    - 611 Outer diameter (mm)
  - 620 Speed ratings
  - 510 Types of practical application

Selections

● Make single selections

○ Build selection set

Matching Entities

Bearing_050htm
Bearing_081htm
Bearing_083htm
Bearing_084htm
Bearing_086htm
Bearing_087htm
Bearing_089htm

○ Match exact concept

● Match concept or child concepts

Bearing specs

Breadth (mm) : 43
Designation : 1318K
Designation: tapered :
Dynamic Load (N) : 117000
Loading Factor (e) : 0.22
Inner Diameter (mm) : 80
Mass (kg) : 5.70
Max RPM :

file:///X:\alistair\Bearing-RV2\html-files\Bearing_084.htm

Fig. 5. The NZM user interface for the bearing selection application.

ing types (ball bearing, roller bearing, taper roller, etc.), bearing dimensions, load ratings, and special characteristics. Each record refers to a description of a particular bearing described using HTML. The classification here is *precise* in that each information entity may be precisely and unambiguously classified into a series of categories. Figure 5 shows the screen display of the categorization hierarchy, a selected list, and the display for a single bearing. It may be seen that the directory style display of the selectable categories allows selection criteria to be entered in any desired order: in one case, the designer may first select external dimensions of the bearing because these may be critical to the application; in another case the first selection might be the desired load rating or the particular bearing type. In each case, after selection the display is updated to show the categories that may be used to refine the selection. Figure 6 shows the resulting display after three selections, showing that the initial 620 entities have been reduced to 5 by the selections. This rapid reduction is characteristic of NZM browsing, as is the continuous provision of feedback to the user of the degree of filtering that will result from a particular selection.

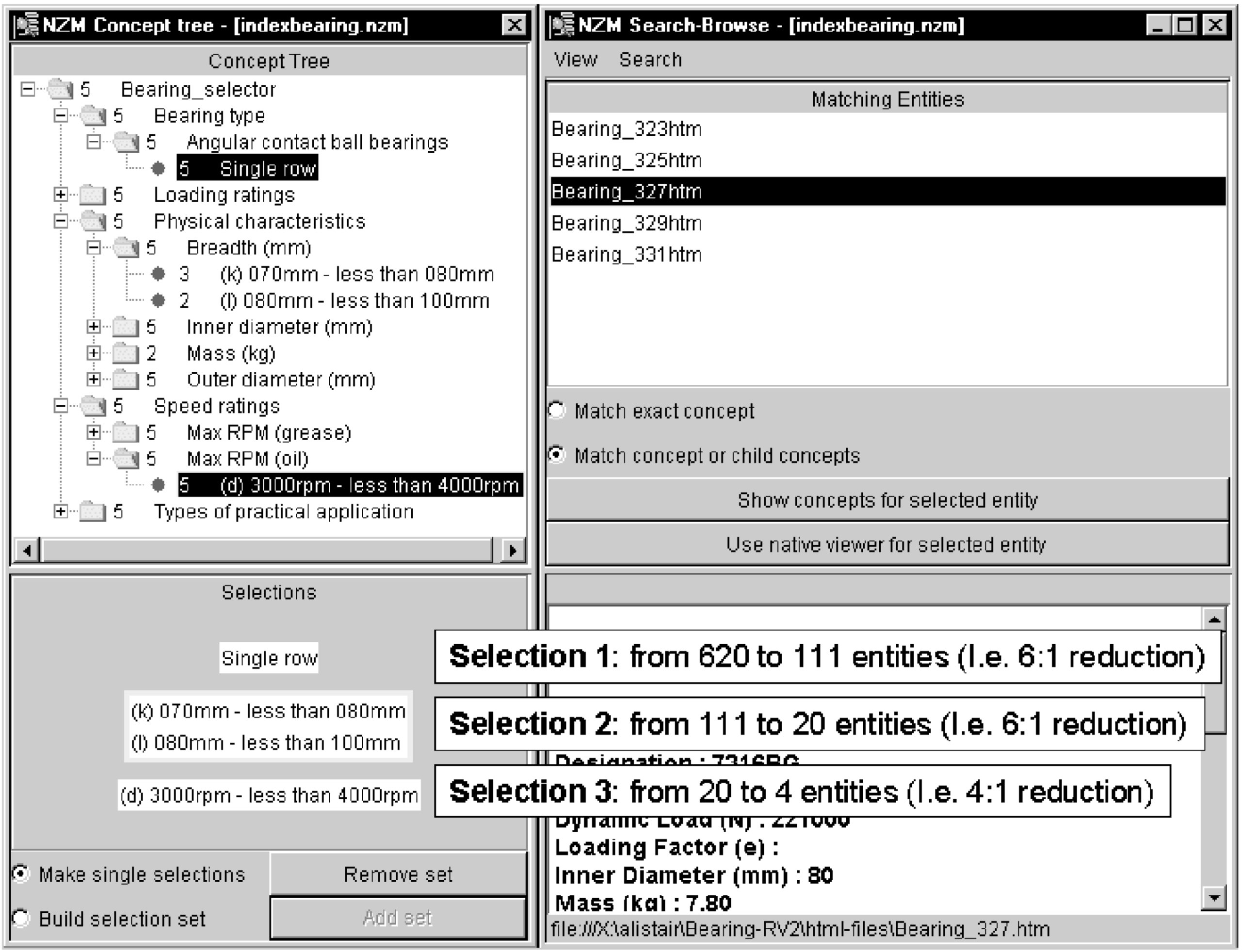
5.2. NZM browsing of categorized web pages

In this second application, NZM is used to browse categorized web pages first identified using a conventional search engine. A query string is submitted to the search engine, and the URLs returned are used in an automatic categorization system to categorize documents into predefined concept hierarchies. Once categorized, the URLs may be browsed using the standard NZM browser.

In the particular case shown here, the query term ⟨“`quality management`” AND “`engineering design`”⟩ was submitted to a popular search engine,[9] and a utility

[9] There is a limit to the number of URLs returned by most search engines in a single set, and therefore a composite set of queries was used to collect more than the ~9000 documents originally identified by the single query noted in Section 2.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press


NZM Concept tree - [indexbearing.nzm]
Concept Tree
5 Bearing_selector
5 Bearing type
5 Angular contact ball bearings
5 Single row
5 Loading ratings
5 Physical characteristics
5 Breadth (mm)
3 (k) 070mm - less than 080mm
2 (l) 080mm - less than 100mm
5 Inner diameter (mm)
2 Mass (kg)
5 Outer diameter (mm)
5 Speed ratings
5 Max RPM (grease)
5 Max RPM (oil)
5 (d) 3000rpm - less than 4000rpm
5 Types of practical application
Selections
Single row
(k) 070mm - less than 080mm
(l) 080mm - less than 100mm
(d) 3000rpm - less than 4000rpm
● Make single selections
○ Build selection set
Remove set
Add set
NZM Search-Browse - [indexbearing.nzm]
View
Search
Matching Entities
Bearing_323htm
Bearing_325htm
Bearing_327htm
Bearing_329htm
Bearing_331htm
○ Match exact concept
● Match concept or child concepts
Show concepts for selected entity
Use native viewer for selected entity

Loading Factor (e) :
Inner Diameter (mm) : 80
Mass (kg) : 7.80
file:///X:\alistair\Bearing-RV2\html-files\Bearing_327.htm

Selection 1: from 620 to 111 entities (I.e. 6:1 reduction)

Selection 2: from 111 to 20 entities (I.e. 6:1 reduction)

Selection 3: from 20 to 4 entities (I.e. 4:1 reduction)

Fig. 6. The bearing hierarchy after selection.

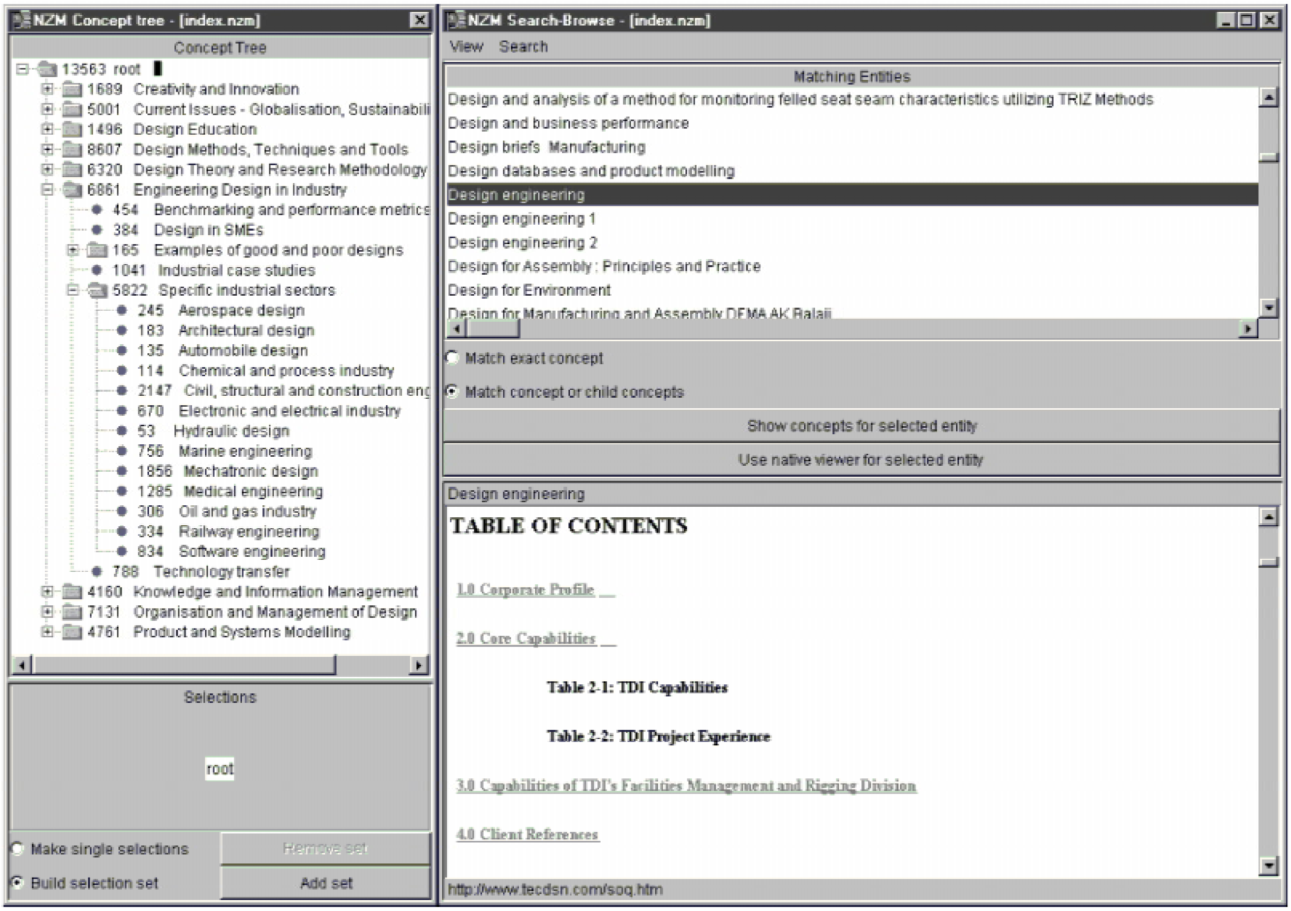
used to collect the URLs returned. This collection of URLs was then categorized into the hierarchical classification of design research topics prepared for the organization of papers submitted to the ICED01 conference (Lowe et al., 2001). In carrying out this categorization the level of precision or recall used by the categorization algorithm could be varied. As noted, precision means the proportion of relevant documents out of those returned, whereas recall means the number of returned documents out of the relevant ones (Buckland & Gey, 1994). In this context, a high recall classification algorithm classifies documents into a greater number of categories than a high precision algorithm, but the relationship between a document and a category may be more tenuous in the former case.

For the particular set of URLs retrieved, a high recall algorithm categorized all 13,500 documents (i.e., web pages) into 400 categories, as shown in Figure 7, whereas a high precision algorithm categorized 2,746 of the 13,500 documents into 290 categories. Experimenting with the precision used in the classification algorithm suggested that with NZM browsing, a high recall algorithm was more valuable when trying to find a very specific reference because it gave greater clues as to the possible content of documents and a wider variety of terms for refining a search. The presence of documents in a filtered set that are poorly related to the browsable categories is less important because further selections can quickly filter the set down to a manageable number of documents. By contrast, when browsing for generally relevant documents in a particular domain, a high precision classification algorithm eliminated documents (such as course syllabi) that had large numbers of terms but little content on any specific term.

5.3. NZM browse of general project documents

The categorization of documents into as many categories as possible is also important in the third example, taken from an index of research project documents at one of the industrial partners. In some document management systems documents are categorized into a small number of categories, with unique references within a single branch of the categorization hierarchy. A browsing path in a con-

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press


NZM Concept tree - [index.nzm]
Concept Tree
NZM Search-Browse - [index.nzm]
View
Search
Matching Entities
○ Match exact concept
● Match concept or child concepts
Show concepts for selected entity
Use native viewer for selected entity
Selections
root
○ Make single selections
● Build selection set
Remove set
Add set

- 13563 root
  - 1689 Creativity and Innovation
  - 5001 Current Issues - Globalisation, Sustainabili
  - 1496 Design Education
  - 8607 Design Methods, Techniques and Tools
  - 6320 Design Theory and Research Methodology
  - 6861 Engineering Design in Industry
    - 454 Benchmarking and performance metrics
    - 384 Design in SMEs
    - 165 Examples of good and poor designs
    - 1041 Industrial case studies
    - 5822 Specific industrial sectors
      - 245 Aerospace design
      - 183 Architectural design
      - 135 Automobile design
      - 114 Chemical and process industry
      - 2147 Civil, structural and construction eng
      - 670 Electronic and electrical industry
      - 53 Hydraulic design
      - 756 Marine engineering
      - 1856 Mechatronic design
      - 1285 Medical engineering
      - 306 Oil and gas industry
      - 334 Railway engineering
      - 834 Software engineering
    - 788 Technology transfer
  - 4160 Knowledge and Information Management
  - 7131 Organisation and Management of Design
  - 4761 Product and Systems Modelling

Design and analysis of a method for monitoring felled seat seam characteristics utilizing TRIZ Methods
Design and business performance
Design briefs Manufacturing
Design databases and product modelling
Design engineering
Design engineering 1
Design engineering 2
Design for Assembly : Principles and Practice
Design for Environment
Design for Manufacturing and Assembly DFMA AK Balaji

Design engineering

TABLE OF CONTENTS

1.0 Corporate Profile

2.0 Core Capabilities

Table 2-1: TDI Capabilities

Table 2-2: TDI Project Experience

3.0 Capabilities of TDI's Facilities Management and Rigging Division

4.0 Client References

http://www.tecdsn.com/soq.htm

Fig. 7. An NZM browse of web documents.

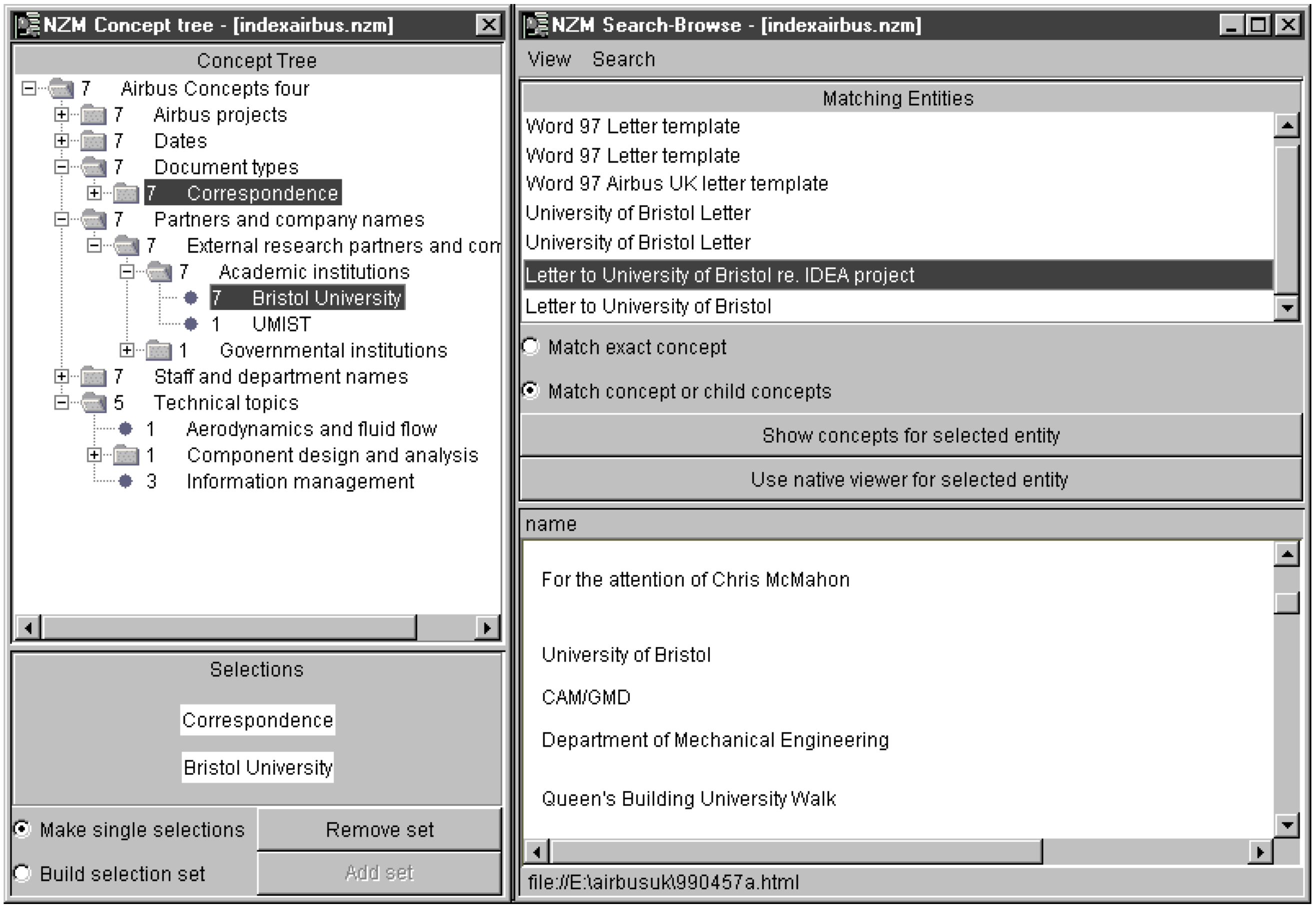
ventional browsing tool that does not give a useful result requires the user to back up and try again and to remember which pages contain which information. If the desired information is deep in a hierarchy or not available at all, this can be a time consuming and frustrating process. Because documents are conceptually stored within unique categories, users cannot create queries based on combinations of categories.

With NZM, the opposite approach is more appropriate. It provides a means of providing a browsing tool for which, at any time, browsing criteria can be interactively refined or broadened. It is appropriate, therefore, to provide a rich set of references. As an example, consider the indexing of documents from a company's research organization. Each document is indexed into hierarchies describing the technical context, the people and organizations involved in the described work and in the production of the document, the project names and dates, the type of document, and so on. By using the feedback from the NZM browser tree, the search problem described in section 2, in which a project manager is searching for an item of correspondence addressed to a supplier and related to a particular project or technical issue, becomes straightforward. The project manager may select the criteria for the search in any order and will be given visual feedback to allow further refinement. Figure 8 shows an example situation. The person browsing has shown interest in correspondence involving Bristol University. The NZM browse tree shows that the seven resulting documents may be further filtered by selecting technical criteria, dates, or company project names. The tree also provided the feedback that items of correspondence with the university are concerned with technical topics, of which one involves aerodynamics and fluid flow, one involves component design and analysis, and three involve information management.

6. DISCUSSION

In the initial implementation described here, the NZM index builder is a Java application communicating with the virtual repository using Java/RMI. The browser has been implemented as a Java application (shown in the figures in this paper) and as an Applet.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press


NZM Concept tree - [indexairbus.nzm]
Concept Tree
NZM Search-Browse - [indexairbus.nzm]
View
Search
Matching Entities
Show concepts for selected entity
Use native viewer for selected entity
name
Selections
Remove set
Add set

7 Airbus Concepts four
7 Airbus projects
7 Dates
7 Document types
7 Correspondence
7 Partners and company names
7 External research partners and com
7 Academic institutions
7 Bristol University
1 UMIST
1 Governmental institutions
7 Staff and department names
5 Technical topics
1 Aerodynamics and fluid flow
1 Component design and analysis
3 Information management

Correspondence
Bristol University

● Make single selections
○ Build selection set

Word 97 Letter template
Word 97 Letter template
Word 97 Airbus UK letter template
University of Bristol Letter
University of Bristol Letter
Letter to University of Bristol re. IDEA project
Letter to University of Bristol

○ Match exact concept
● Match concept or child concepts

For the attention of Chris McMahon

University of Bristol
CAM/GMD
Department of Mechanical Engineering

Queen's Building University Walk

file://E:\airbusuk\990457a.html

Fig. 8. NZM browsing of general technical documents within a company department.

6.1. Performance and application issues

The initial NZM browser was designed to give a satisfactory interactive performance for approximately 10,000 documents indexed into 1000 categories with 100 concept references per document. The performance with the 13,500 document collection suggests that this objective has been achieved in that the time to update the screen display after a concept selection is less than 2 s on a 500-MHz Pentium II with 128 Mb of RAM. The described method of constructing an index is limited in the number of information entities that may be indexed and in the number of concepts that may be used by, first, limitations in the size of array (binary matrix) that may be constructed and manipulated by the available computing equipment, and by, second, declining performance of the method as the size of the array increases. These limitations will become less restricting as computer equipment performance increases. The authors are currently working on techniques that they believe will give a 10- to 100-fold performance improvement.

The initial experimental applications suggest that the most valuable application of the NZM approach in its present implementation will be as a mechanism for local browsing of static or quasistatic document collections, such as papers on a CD-ROM, or for small numbers of users to share a specialist collection of documents accessed through a server. An example of the latter in engineering might be a collection of technical documents maintained by a specialist group such as a design analysis team within a company. Outside of engineering, specialists in medicine or law would be strong contenders to use the approach. Current research is exploring methods for improving the performance of the approach to allow large numbers of concurrent users in strongly database driven applications.

6.2. NZM as a UI technology

The motivation for the improvement of UIs is that the ultimate performance of information retrieval systems is highly dependent on obtaining effective feedback from and giving feedback to system users. As previously noted, the human-computer interface is typically less well understood than other aspects of information retrieval systems. While much of the core underpinning retrieval technology is relatively mature (and there has been a degree of convergence on a limited range retrieval models and ranking algorithms), UI

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

innovations are open to development. Shneiderman (1997) and Nielsen (1993) suggest that a good human–computer interface should provide informative feedback to the user and allow easy reversal of actions. The NZM approach is particularly strong in these respects. Any action may be undone simply by repeating the selection, and the user receives constant feedback from the interface on the effect of his or her actions. As each user action is made, all of the possibilities for further refinement of the selection are presented to the user, along with the number of selected or selectable entities for each node of the concept tree. The strength of the UI is such that it is being explored as a mechanism for the automatic presentation of diagnostic information by relating information entities to fault conditions and symptoms. Studies of the usage of pilot and demonstrator NZM applications by engineers from collaborating industrial partners and other trial users have led to the following observations:

- Some users, who are familiar with a conventional directory explorer interface in which the directory structure is not pruned, initially find the dynamic nature of the hierarchy presentation to be nonintuitive; however, after brief explanation of the mode of operation, users are able to use the interface without formal tutorial support.
- Selection from multiple branches of the hierarchy requires those branches containing selected nodes either to be continually displayed or, if closed, to be marked (e.g., by a change of color) to show that a child node has been selected. Selection from multiple branches of complex hierarchies causes lengthy hierarchies to be displayed, requiring scrolling of the display. Ideally, this is to be avoided, which merits the exploration of alternative approaches such as pull-down menus for the hierarchy branches.
- The choice of key and mouse key operations for selection of categories (especially differentiation between AND and OR selection and between selection and deselection) is important to the look and feel of the interface.
- The absolute performance of the hierarchy updating mechanism is less important than the consistency in its performance, reflecting observations made by Shneiderman (1997).

7. CONCLUSIONS

There is widespread dissatisfaction with the searching of large databases, and it would be hard to find anyone who is completely happy with existing Internet search strategies.

We have illustrated the problems associated with three types of search, which are representative of search types from the extensive case study and analytical work (particularly in engineering organizations) that the authors have undertaken over the last 10 years. The fundamental issues associated with searching for information are linked to inherent conflicts or trade-offs between precision and recall in searching approaches and the nature of keyword-based or browsing search strategies.

To assist in addressing some of these issues, we are proposing a UI-based NZM approach, which has been created to ensure that the searcher or user is never presented with a null result. The approach works in association with predefined concept hierarchies and has a firm theoretical foundation.

The approach has been tested with a range of data sets or documents from the authors' research partners, and three illustrative examples are included to demonstrate the way in which constant feedback is provided as the searcher or user moves through the hierarchies.

The computational performance of the approach has been shown to be very satisfactory for the data set sizes tested. Strategies for improving this as data set sizes increase are being developed.

The approach was designed and developed initially to support engineering designers. These professionals have been shown, by us and other researchers, to use bounded specialist collections of documents. Although the size of these document sets will increase, they will never approach the orders of magnitude of the 10s or 100s of millions of uncontrolled or random documents that have to be dealt with by the internet search engines (e.g., Google is currently indexing over 2 billion web pages). Thus, the NZM search method described in this paper is considered to have wide ranging applications in domains that exhibit these bounded characteristics (such as in medicine, law, government, and similar domains). Its hybrid characteristic of pruning from a predetermined concept hierarchy makes it an innovative and invaluable strategy to handle the precision or recall conflicts in these technical or quasitechnical domains.

ACKNOWLEDGMENTS

The authors would like to thank the participating engineers from Airbus (UK), TRW Aeronautical Systems, and CSC (UK) Ltd for their help in undertaking this study. The useful comments made by the anonymous reviewers are also appreciated. This work was carried out as part of the IDEA project, which is funded by the UK Engineering and Physical Sciences Research Council (Grant GR/L90170).

REFERENCES

Anick, P.G. (1994). Adapting a full-text information retrieval system to the computer troubleshooting domain. *Proc. 17th Annual Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 349–358.

Bates, M.J. (1986). An exploratory paradigm for online information retrieval. In *Intelligent Information Systems for the Information Society* (Brookes, B.C., Ed.), pp. 91–99. Amsterdam: North-Holland.

Blair, D.C. (1980). Searching biases in large interactive document retrieval systems. *Journal of the American Society of Information Science 17*, 271–277.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

Brüninghaus, S., & Ashley, K.D. (2001). The role of information extraction for textual CBR. In *Case-Based Reasoning Research and Development: Proc. 4th Int. Conf. Case-Based Reasoning (ICCBR-01*; Aha, D.W., Watson, I., & Yang, Q., Eds.), Vancouver, Canada, July 30–August 2, 2001. New York: Springer.

Buckland, M., & Gey, F. (1994). The relationship between recall and precision. *Journal of the American Society for Information Science 45, 12–19.*

Chen, H., Houston, A.L., Sewell, R.R., & Schatz, B.R. (1998). Internet browsing and searching: User evaluations of category map and concept space techniques. *Journal of the American Society for Information Science 49*, 582–603.

Cleverdon, C.W. (1967). The Cranfield tests on index language devices. *Aslib Proceedings* 19, 173–192.

Court, A.W., Culley, S.J., & McMahon, C.A. (1996). Information access diagrams: A technique for analysing the usage of design information. *Journal of Engineering Design 7*, 55–75.

Crowder, R.M., Hall, W., Heath, I., & Wills, G. (1999). Integration of manufacturing information using open hypermedia. *Computers in Industry 38*, 31–42.

Culley, S.J., Duffy, A.H.B., McMahon, C.A., & Wallace, K.M. (Eds.). (2001). *Proc. 13th Int. Conf. Engineering Design*, Glasgow, August 2001.

Davis H.C., Hall W., Heath I., Hill G., & Wilkins, R. (1992). Towards an integrated environment with open hypermedia systems. *Proc. ACM Conf. Hypertext, EHCT '2*, Milan, Italy, pp. 181–190. New York: ACM Press.

Drucker, P.F. (1994). *Post-Capitalist Society*. Oxford, UK: Butterworth Heinemann.

Drucker, P.F. (2000). The change leader. *National Productivity Review 19*, 13–20.

Edmunds, A., & Morris, A. (2000). The problem of information overload in business organisations: A review of the literature. *International Journal of Information Management 20*, 17–28.

Hall, W., Davis, H.C., & Hutchings, G. (1996). *Rethinking Hypermedia—The Microcosm Approach*. Boston, MA: Kluwer.

Hayman, A., & Elliman, T. (2000). Human elements in information system design for knowledge workers. *International Journal of Information Management 20*, 297–309.

Hearst, M.A. (1999). User interfaces and visualization. In *Modern Information Retrieval* (Baeza-Yates, R., & Ribeiro-Neto, B., Eds.), pp. 257–323. New York: ACM Press.

Hearst, M.A., & Karadi, C. (1997). Cat-a-cone: An interactive interface for specifying searches and viewing retrieval results using a large category hierarchy. *Proc. 20th Annual Int. SIGIR Conf.*, Philadelphia, PA, pp. 246–255.

Institute for Systems Research. (2001). Hendler and Shneiderman debate at ASIS meeting. Institute for Systems Research, University of Maryland. Available on-line at http://www.isr.umd.edu/ISR/publications/newsletter/ssfa01/Shneiderman-HendlerDebate2.html

Krellenstein, M.F. (1999). *Method and apparatus for searching a database of records*. US Patent 5,924,090.

Larson, R. (1992). Experiments in automatic Library of Congress classification. *Journal of the American Society for Information Science 43*, 130–148.

Lenz, M., & Burkhard, H.-D. (1997). CBR for document retrieval: The FALLQ project. *Proc. Second Int. Conf. Case-Based Reasoning, Case-Based Reasoning Research and Development ICCBR-97*, Providence, RI, July, pp. 84–93. Berlin: Springer.

Lowe, A., McMahon, C.A., Shah, T., & Culley, S.J. (2000). An analysis of the content of technical information used by engineering designers. *Proc. ASME Design Engineering Technical Conf.*, September 10–13, Baltimore, MD.

Lowe, A., McMahon, C.A., Shah, T., & Culley, S.J. (2001). The application of an automatic document classification system to assist the organisers of ICED01. Design management, process and information issues, *Proc. 13th Int. Conf. Engineering Design*, Glasgow, August 2001, pp. 179–186.

Maher, M.L., & Pu, P. (Eds.). (1997). *Issues and Applications of Case-Based Reasoning to Design*. Mahwah, NJ: Erlbaum.

McMahon, C.A., North, M.R., Sims Williams, J.H., & Culley, S.J. (1998). *Information for Design Engineering in Aerospace (IDEA)*. EPSRC Project GR/L90170/01.

McMahon, C.A., Pitt, D.J., Yang, Y., & Sims Williams, J.H. (1995). Review: An information management system for informal design data. *Journal of Engineering with Computers 11*, 123–135.

Meadow, C.T. (1992). *Text Information Retrieval Systems*. New York: Academic Press.

Nielsen, J. (1993). *Usability Engineering*. New York: Academic Press.

Nielsen, J. (1999). User interface directions for the web. *Communications of the ACM 42*, 65–72.

Pollitt, S. (1997). Interactive information retrieval based on faceted classification using views. *Proc. 6th Int. Study Conf. Classification*, University College, London, June 1997.

Reuters Business Information. (1996). *Dying for Information: An Investigation Into the Effects of Information Overload in the UK and World-Wide*. London: Reuters. Available on-line at http://www.reuters.com/rbb/research/overloadframe.htm

Robertson, S.E., & Sparck Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Sciences 27*, 129–146.

Salton, G., & Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Sciences 41*, 288–297.

Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human Computer Interaction*. Reading, MA: Addison–Wesley.

Waterworth, J.A., & Chignell, M.H. (1991). A model of information exploration, *Hypermedia 3*, 35–58.

Chris McMahon is Reader in Engineering Design in the Department of Mechanical Engineering at the University of Bath, where he teaches and conducts research in engineering design and computer-aided design. He is interested in many aspects of design and computing, in particular, the organization and delivery of design knowledge and information, risk and uncertainty in design, multiple-viewpoint feature-based modeling, and design automation. Chris has written a popular textbook on CADCAM and over 100 papers on various design topics.

Rose Crossland is a Research Associate in the Department of Engineering Mathematics at the University of Bristol. She received a BA in mathematics in 1986 from the University of Oxford and subsequently worked as a Software Engineer before joining the University of Bristol in 1994. Rose received her PhD in engineering mathematics in 1997 and has since worked on a variety of projects concerning information management in design. Her research interests include the management of risk in engineering design and the development of web-based computer tools to support the information management needs of engineering designers.

Alistair Lowe has worked for the last 4.5 years on an industrially focused research project called Information for Design Engineering in Aerospace (IDEA) with the Universities of Bristol and Bath, Airbus UK, TRW Aeronautical Systems, and Computer Sciences Corporation. In the previous 3 years he was a Design Engineer at the UK Defence Evaluation and Research Agency. Alistair graduated from the University of Bristol with a first class MEng in mechanical engineering and has recently completed a PhD thesis based on his recent research entitled, "Studies of information use by engineering designers and the development of strategies to aid in its classification and retrieval."

Tulan Shah was a Research Assistant for 4.5 years in the Department of Mechanical Engineering at the University of

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press

Bristol, where he worked with Airbus UK, TRW Aeronautical Systems, and Computer Sciences Corporation on methods for information organization and categorization. He received the MEng in mechanical engineering from the University of Bristol in 1997 and is currently working on his PhD in developing methods and software tools for supporting engineering designers in managing their information needs.

Jon Sims Williams is a Senior Lecturer in Engineering Mathematics at the University of Bristol. His principal research interests lie in risk in design projects, design information systems, and knowledge-based systems. He has lectured in operations research at both Auckland and Bristol Universities and currently leads Bristol's MEng degree in engineering design. He has developed the TAL system for the automatic generation of student tests from databases of computer markable questions.

Steve Culley is Head of Design in the Department of Mechanical Engineering at Bath University. He has researched in the engineering design field for many years. In particular, this has included the provision of information and support to engineering designers. Steve pioneered work in the introduction and use of the electronic catalogue for standard engineering components and has extended this work to deal with systems and assemblies. He has over 150 publications and has recently coauthored a book on design and changeover.

https://doi.org/10.1017/S0890060402163098 Published online by Cambridge University Press