Information Processing & Management Vol. 28, No. 3, pp. 371-387, 1992
Printed in Great Britain.

0306-4573/92 $5.00 + .00

A HYPERTEXT ENVIRONMENT FOR INTERACTING WITH LARGE TEXTUAL DATABASES

M. AGOSTI,[1] G. GRADENIGO[1] and P.G. MARCHETTI[2]
[1]Dipartimento di Elettronica e Informatica, Universitá di Padova,
Via Gradenigo 6/a, 35131 Padova, Italy
[2]European Space Agency—IRS, Via Galileo Galilei, 00044 Frascati, Italy

Abstract—This paper presents a design and implementation project based on a two-level conceptual architecture for the construction of a hypertext environment for interacting with large textual databases. The conceptual architecture has been proposed to be used for a semantic representation of the informative content of a collection of documents and for the organisation of the document collection itself. The hypertext environment is based on a set of functions that permits one to exploit the potential capabilities of the two-level architecture. Those functions are presented in detail. The paper reports some results of a more general project whose final goal is the definition of a new model for information retrieval: a model with information retrieval capabilities embedded within a hypertext environment. Finally, an outline is presented of the characteristics of a prototype, named HYPERLINE, of the hypertext environment. This prototype has been developed by the Information Retrieval Service of the European Space Agency (ESA-IRS)

INTRODUCTION

In Agosti *et al.* (1989, 1990) a conceptual architecture has been proposed that can be used for a conceptual representation of the informative content of a collection of documents and for the organisation of the document collection itself. The main scope of this architecture is to support the user with a reference model that makes explicit the semantic representations of the documents that are managed and used by the system to solve the user's query. Work is under way to make use of that conceptual architecture for two different requirements:

1. to experimentally verify the soundness of the architecture as a basis for the construction of a hypertext environment to be used as a conceptual reference and interface tool for the final user; and
2. to act as a basis for the definition of a new information retrieval model and the related approach in defining and retrieving information on documents (structural and dynamic aspects).

This paper presents the results of the application of the architecture for purpose 1; thus the paper introduces the guidelines of the design and implementation project that uses the architecture as a basis for the construction of a hypertext environment interacting with large textual databases; in Agosti *et al.* (1991) an initial report of the work was given.

The paper is structured as follows: The first section gives the motivations of the work. Then the main characteristics of the reference architecture are presented together with the description of the basic model of the interface and the different functions that have been designed and implemented. Finally, the highlights of the usable prototype are given.

MOTIVATIONS

Information retrieval systems (IRS) are designed for efficient storage and retrieval of large numbers of bibliographic references or short textual documents. The architecture of

A shorter version of this paper was presented at RIAO91—Conference on Intelligent Text and Image Handling, Autonomous University of Barcelona; Barcelona, Spain, 2-5 April 1991.

these systems does not foresee a formal and explicit representation of the information items stored in the database and relationships among them. This representation (called "schema" in the database management area) is actively used by a database management system and its final user. The lack of such a formal representation of the database content managed by an IRS deprives the user of an *explicit reference* for formulating the query and the IRS of the possibility to capture the semantic of the query. This is not, however, a serious drawback when the query involves information of a deterministic type (e.g., "find all documents with date of publication = 1990"), but it becomes very limiting when the information sought concerns the informative content of documents (e.g., "find documents about database modelling and documents about related concepts").

On currently available information retrieval systems, the user is able to see a single indexing term or a list of indexing terms used in a representation of the informative content of the documents. Browsing, during a query, through the indexing terms related to a term of interest for the user and through the structure of the indexing terms would be helpful for a proper understanding of the semantic context in which the *meaning* of each term is defined by its relationships with other terms.

Few information retrieval services support the user, providing the possibility to see the structure of indexing terms used in representing the semantic content of documents, for example, to see the connection of an indexing term with other related indexing terms. A facility like this is very useful, but it is not sufficient to really inform the user on the structure of the indexing terms. In fact:

- the facility is available only on a stand-alone basis; that is, if users browse the semantic structure of the indexing terms they are unable to see related documents, because browsing through the thesaurus or some other semantic structure is not entirely integrated with the bibliographic database search function; and
- the user is required to know at least an indexing term in order to begin searching for the semantic structure. Otherwise access tot he structure would be impossible.

Some interesting research efforts (see, for example, Croft & Thompson, 1987, and McMath *et al.*, 1989), have been addressed towards making the structure of the indexing terms more explicit to the user. The research work based on a two-level architecture (Agosti *et al.* 1989, 1990) here reported, presents a conceptual tool that makes the structure of the indexing terms explicit and directly available to the final user; together with the research results, a prototype implementation, which is available for public access and use through public networks is also presented.

The issue of defining an architecture is discussed here. This architecture is able to support the explicit conceptual modelling of information retrieval data in order to produce a schema of concepts that describes the informative content of a document collection. This schema is defined as a network of concepts for a specific information retrieval application domain. This schema can be used in an active manner by the user, because this network has to provide the user with a frame of reference in the query formulation process.

Furthermore, the interface provides some elements of information transfer and interaction mechanisms that can be considered complementary to the usual search strategy development process (Belkin & Marchetti, 1990). The present implementation of the HYPERLINE hypertext environment has been urged by the functional elements made available by the two-level model. These hypertext functional elements will have to melt with the traditional information retrieval ones in a future interface implementation. A cognitive task analysis (CTA) approach will be used to cast the large resulting interactivity (Marchetti & Belkin, 1991).

THE REFERENCE CONCEPTUAL ARCHITECTURE

The part of an informative service that is automated by means of an information retrieval (IR) system presents highly complex characteristics due to the complex nature of the

data such a service handles. The database managed by an operational IR system is usually made up of:

1. a collection of documents, and
2. a structured collection of auxiliary data.

The *collection of documents* usually consists of a large collection of textual documents. It is possible to characterise the stored documents by means of structured data such as the author's name and the date of publication. The structured data is therefore a deterministic representation of the documents of the collection.

The *structured collection of auxiliary data* is associated to each document in order to represent its semantic content, and is also used to select and retrieve the documents from the database in response to the user's queries. The auxiliary data by which the content of a document is represented is associated to the document itself by means of an indexing process. Each auxiliary data item is assumed to describe the document content only to a certain extent, neither completely nor uniquely, and different auxiliary data items may be assigned to each document. In fact, the description of the content of a document derives from a subjective process, because either the content can be visualized and described in different ways by different persons or different descriptions simply reflect different users' requirements for information.

All specific information retrieval models differ from each other basically because of the kind and structure of the auxiliary data used in the system operations. The auxiliary data represents the vocabulary to be consulted by the IR system; thus such data exists even if the documents have not yet been inserted into the database. It is important to note that the auxiliary data items have very different semantic values; therefore this type of vocabulary takes on a rather complex structure, having to represent the complex semantic relationships that exist between auxiliary data items. Furthermore, some IR systems make use of different auxiliary data structures to permit multiple descriptions of the informative content of documents; these descriptions highlight different aspects and make use of several vocabularies for heterogeneous categories of users. Previous considerations make it clear that the complexity of data modelling in IR depends more on the auxiliary data modelling than on the modelling of the document collection.

The proposed architecture

The auxiliary data describes the document information contents, but the meaning of an auxiliary data item becomes fully defined only by means of the semantic relationship that exists between this auxiliary data item and others. As previously shown, a database managed by an IR system usually consist of two main parts: a collection of documents and a structured collection of auxiliary data. Because of the different roles these two parts play in the architecture of an IR system, it is necessary to make it explicit and to consider an architecture that permits working on at least two different levels of classification abstractions (see Fig. 1):

First level. This is the lowest level that contains the elementary objects of interest; it is therefore the level where the collection of documents D that is managed by an IR system can be placed.

Second level. This level arises from the application of the classification abstraction mechanism to the elementary objects of the first level; this level can be identified as the level of concepts—that is, the locus on which the semantically related concepts are placed; this is the plane of abstraction where the set T of auxiliary data items used by an IR system can be placed; each item represents a concept that is pertinent to a certain set of documents on the basis of their semantic content.

The classification mechanism can be further applied to produce a third level or plane of meta-classes, where by meta-class we mean a class of classes—that is, a class whose members are classes of elementary objects. A meta-class can be, for example, the class of thesauri. This level of classification arises from the repeated application of the classification abstraction process. This third level has been described in Agosti *et al.* (1990); it is men-

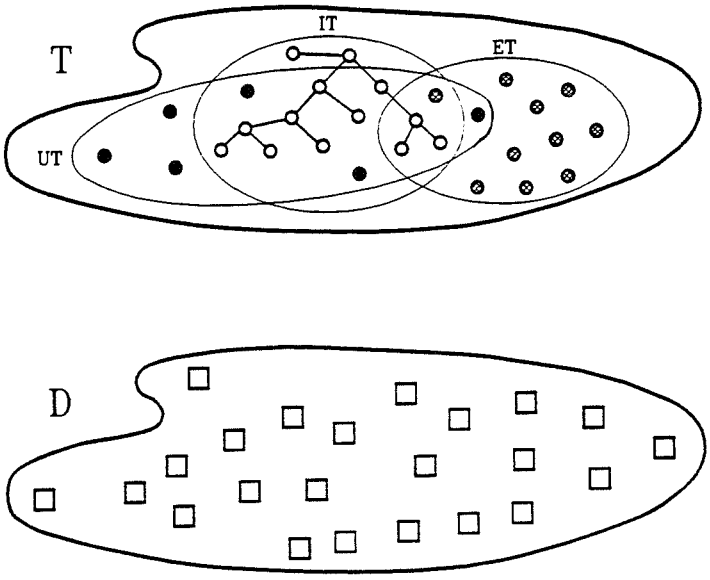

Fig. 1. Two-level architecture and basic model.

tioned here because it completes the reference architecture, but its role is out of the scope of this paper. The abstraction mechanisms (classification, generalisation-specialisation, and aggregation) included in the paradigm that underlines the proposed architecture are sufficient for the organisation of the representations of the information contained within the documents of the collection.

The collection of documents

The collection of documents is represented at the first level of the architecture. Each document has its own identity and status; the identity of the document is independent of the way it is represented or structured and the values it may assume.

The structured collection of auxiliary data

The structured collection of auxiliary data is represented at the second level of the architecture. An object-oriented approach is adopted for the design and management of the structured collection of auxiliary data because with this approach it is possible to use the multiple inheritance mechanism, which permits the implementation of this natural property of IR: An IR object can be related to many different terms and hence the object can belong to several different classes. The conceptual tool necessary for designing and managing the auxiliary data must be able to support polythetical classification of the IR objects. In this approach each auxiliary data item is viewed as a class of objects; instances of each of these classes are the documents pertinent to the specific concept expressed by the class term. Hence one of these classes is a set of documents, and from a higher abstraction level may be seen as a single conceptual object of a structure representing the semantic relationships between different concepts. Each *class* can be considered on two different abstraction levels: as a set of instances and as a whole entity.

The *properties of the class* can be subdivided into:

- the properties of the class as an object—that is, the properties of the auxiliary data as such, and
- the properties of the objects of the class—properties of the documents belonging to the class identified by the auxiliary data.

FUNCTIONS FOR A HYPERTEXT ENVIRONMENT FOR INTERACTION WITH LARGE TEXTUAL DATABASES

The introduced architecture has been used as a basis for the construction of a hypertext environment to be used as a conceptual reference and intermediary tool by the final user of an available IR system. Since the available IR system is one of Boolean nature, as is almost any IR system presently used in large applications, the functional model can be generalised for all the applications using a Boolean IR system. The first level of the architecture, the plane of the collection of documents, is implemented and managed by the available IR system; the second level of the architecture has been designed and implemented as a conceptual interface between the user and the collection, which operates as an environment with hypertext capabilities. The interface implements the interactive use of the auxiliary data conceptual structure and the relationships between the level of documents and that of concepts and vice versa; the interface makes transparent to the user the way the IR system makes use of Boolean logic. The next section describes the functions of the tool that has been designed to make a hypertext environment for browsing large textual databases available to the user. A prototype based on these general functions has been developed by the IR Service of the European Space Agency (ESA-IRS). In the following section the prototype's highlights together with examples of the prototype implementation of the environment are presented.

As has been underlined in the introduction of this paper, the complete project, of which this paper presents a part, has as its final target the definition of a new model of information retrieval, which is going to use a hypertext environment as a browsing mechanism embedded within an IR system. Many hypertext functionalities have been shown to be very useful for the retrieval of information from informative resources (Frisse, 1988; Frisse *et al.*, 1990; Nielsen, 1990; Ritchie, 1989). The problem of retrieving information from large informative resources is still an open one. This work gives some positive results in the direction of solving the problem.

Functional capabilities of the environment: The basic model

The basic model is based on the two-level architecture (see Fig. 1). At the first level is placed the collection of documents of interest; in the following the collection of documents is identified and represented by the set D. The second level can be considered as the level of concepts, that is, the conceptual plane on which the semantically related concepts are placed. This is the plane of abstraction where indexing terms used by an IR system can be placed; each term identifies a concept.

Since different structures of auxiliary data (or different indexing techniques) can provide different conceptual representations of the same collection of documents to satisfy different users' requirements, a basic model has been designed that foresees this possibility. Thus, this model permits users to choose at "runtime" in a transparent way the indexing and retrieval techniques most familiar to them or that most suit their informative requirements.

At the second level there is the set T, which represents the universe of possible usable terms. The set ST (System Terms), a subset of T, is the set of terms used and managed by the system. This is the set resulting from the union of all terms used by the different structures of auxiliary data managed by the interface. Two different auxiliary data structures used concurrently by the system have been taken into consideration in this work: the set ET and the set IT, where:

1. ET (Extracted Terms) is the set of terms produced by the application of an automatic parsing algorithm to the textual parts of the managed documents; the terms of the set are all the terms extracted by the algorithm not included in a list of stop-words or non-significant words; and
2. IT (Indexing Terms) is the set of indexing terms of an auxiliary data structure adopted by the system; for the first prototype of the interface, a thesaurus has been used as an example of a complex auxiliary data structure. A thesaurus consists of a complex se-

mantic structure of indexing terms associated with the documents by experienced indexers. In this context the thesaurus is seen as a repository of human knowledge and ability in concept classification. The fundamental types of semantic relationships expressed in a thesaurus are: scope, equivalence, and hierarchical and associative relationships; see Aitchison & Gilchrist (1987) for further details on a thesaurus structure.

The set of possible usable terms T could contain other auxiliary data structures, but it always contains also another set of terms: the set of User Terms (UT). The elements of the set UT are the free terms that the user of the system can insert into a query. That is the set of terms that are not necessarily present in the set of terms extracted from the documents of the collection or in the set IT of indexing terms. It is important to note that the ET, IT, and UT sets are not necessarily disjoined. The architecture of the basic model is depicted in Fig. 1, where

$$T = ET \cup IT \cup UT.$$

The proposed architecture can be used as a frame of reference by the user in the process of query formulation. Through the architecture the structure of the auxiliary data is made available to users, so they can see and navigate through the semantic structure of the indexing terms describing the informative content of the documents.

The functions made available in the environment have been designed taking into account some hypertext functionalities that have been shown to be useful and relevant for the user; see, for example, Nielsen (1990) and Shneiderman & Kearsley (1989). Furthermore, as a first step towards the integration of hypertext and information retrieval environments, it has been provided the possibility of using the result of the concept navigation process to construct a search strategy for the user. The hypertext environment makes the functions described in the following sections available to the user:

1. semantic association;
2. navigation;
3. sequential reading;
4. associative reading from a single document or from many documents;
5. backtracking;
6. history; and
7. support of search strategy development.

Semantic association. The semantic association function operates in the following way: When the user expresses interest in a subject using a specific term, a list of conceptually related indexing terms that are concepts of the auxiliary data structure are suggested to the user. The aim of the semantic association function is to make more transparent and to communicate to the user the meaning the system gives to the term used by the user in the expression of his or her information needs. In the formulation of a query, the user can use natural language; each word given by the user is mapped by the system in a set $IT(i)$ of semantically close concepts which is a part of the auxiliary data structure managed by the system. Moreover, the set $IT(i)$ given to the user by the semantic association function operates for the user as an entry point to the auxiliary data structure; the user can via this entry point start interacting with the system. The aim of this interaction is the acquisition of information via concept navigation and document browsing. In this way we do expect to help the various IR searcher groups by means of an extensive support of their conceptual knowledge (Ingwersen, 1986). The semantic association function moves from the classical term analysis potentially useful for probabilistic relevance feedback (Robertson *et al.*, 1986) and in other semi-automatic feedback modes (McAlpine & Ingwersen, 1989). The semantic association function performs a concept analysis (Belkin & Marchetti, 1990) and knowledge extraction as described in the following.

Suppose the term ut that the user initially enters is itself an indexing term, in this case: $ut = it$. Thereby the connections that the term has with the other indexing terms (e.g., the thesaurus relations) are presented to the user and made available to the other environment

functions (e.g., navigate, show, . . .). If the term *ut* that the user initially enters is not a term of *IT*, but a term of *ET*, then the list of conceptually related indexing terms is constructed making use of the documents and through an inference mechanism on the indexing terms. The procedure is based on the fact that each term of the set *ET* has been extracted during the indexing procedure (parser) from a set of documents and its relationship with them is maintained. These functional relationships are acknowledged and made usable in an active way. Thus, a term *et* is related to a set of documents $D(i)$. When the set $D(i)$ has been constructed by the interface, the interface can construct and make use also of the set of indexing terms with which the documents of the set $D(i)$ have been associated during the indexing procedure. Since the resulting set of terms of the set *IT* could have too high cardinality, an inference mechanism is applied to reduce the cardinality of the set and to present the user only the set $IT(i)$ of the most pertinent indexing terms. The way this function operates is shown in Fig. 2.

If the term *ut* that the user initially enters is not a term of the set *IT* nor of the set *ET*, a stemming algorithm is used for suggesting an alphabetical list of terms morphologically close to the term *ut*. After the users' choice of a term from those suggested, the interaction continues in one of the two previously presented ways. In the present implementation a list of conceptually related indexing terms (elements of *IT*) is suggested only if the used term *ut* is also an element of the set *ET*, that is:

$$UT - ET = \varnothing, \quad \text{where the symbol } \varnothing \text{ denotes the empty set.}$$

When the semantic association function has completed its operations, the set $IT(i)$ is available to the user as input to one of the other functions.

Navigation. Before presenting the navigation function, it is important to recall that the user of an IR system is usually looking for documents only to find information stored in the collection of documents. The navigation function gives the user the possibility to browse the structure of the semantic concepts representing the information content of the collection of documents, that is, the conceptual structure of auxiliary data items and of the complex relationship that exists between them. This function enhances the user-system communication, because the user has the possibility to navigate within the semantic structure of concepts (the auxiliary data structure), and a more powerful interaction with the information stored in the collection of documents managed by the IR system is made possible.

The navigation can start from any term of the set *IT*. If the user has had in response

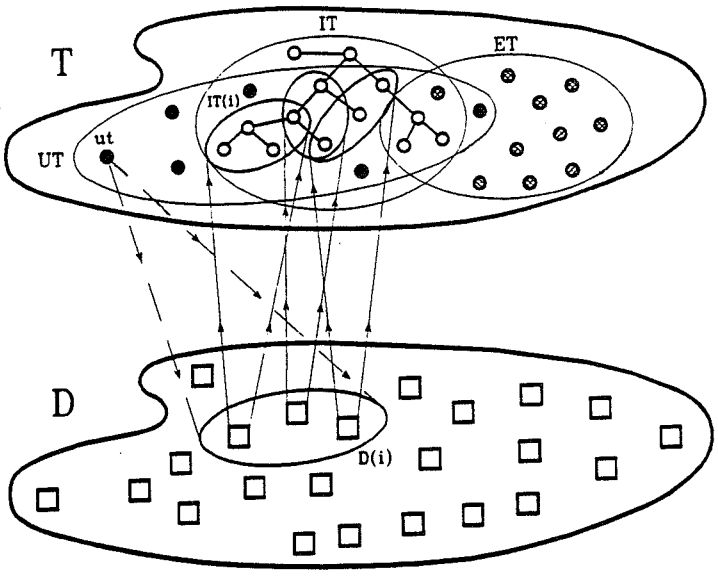

Fig. 2. Semantic association.

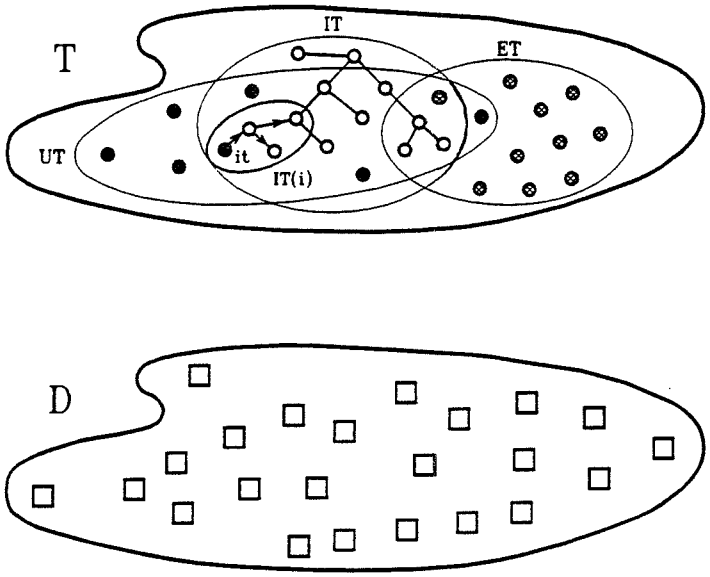

Fig. 3. Navigation.

to a previously entered term a set $IT(i)$ of indexing terms, or has directly entered a term *it*, the user can navigate the semantic structure of the auxiliary data in order to find out the way in which the terms have been connected and chosen by the indexers. Figure 3 shows the functional relationships in the navigation procedure.

Sequential reading. The sequential reading function allows browsing of the documents associated with any specific term of the set *IT*; in fact, each term of the auxiliary data structure is connected to a set of documents that are semantically related to the term itself. Sequential reading allows the user to move from the second level of the architecture (level of the auxiliary data structure) to the first level (where the documents are managed). In Fig. 4 the sequential reading function is graphically represented. Reading or browsing of the set of the related documents can be done in a sequential manner, one after the other; this function is usually defined as a "documents browsing facility" by the operative information re-

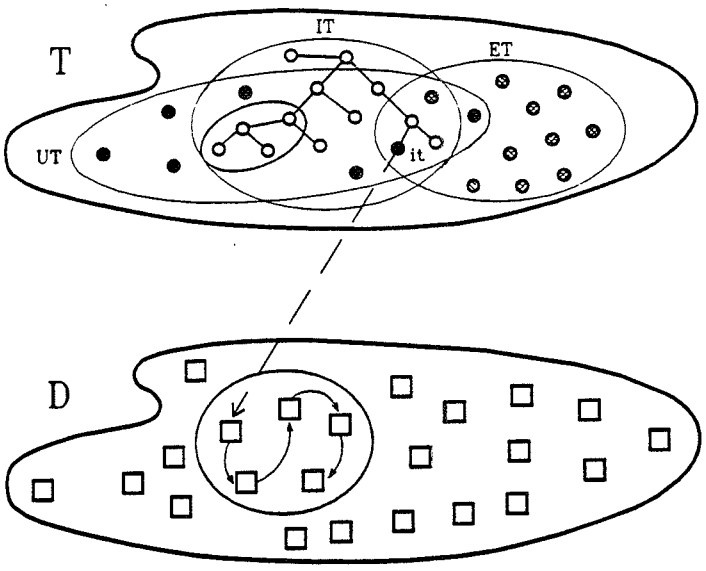

Fig. 4. Sequential reading.

trieval services. It is important to note that this facility is available from operative information retrieval services when the user has formulated a query that has retrieved a set of documents; the user can "browse" (sequentially read) the documents of the set identified as a result of the query. The sequential reading function, instead, permits reading of the documents of the set related to every indexing term of the auxiliary data structure without any query formulation.

Associative reading from a single document or from many documents. The user can invoke the function of associative reading from documents to return from the level of documents to the level of auxiliary data and navigate along the auxiliary data structure. This facility permits navigation between the two levels of the architecture; depending on the level where the user is (first or second), he or she can navigate on that level using the suitable functions. The associative reading function has been designed to be made available from a single document or from many documents.

Two different modes of associative reading can be triggered by the user when using the sequential reading function:

1. First mode of associative reading from a single document: During sequential reading of a set of documents, the user can ask to see the set of indexing terms $IT(k)$ associated with the document *Dk* being read at that precise moment. In Fig. 5 this mode of associative reading is graphically represented by the arrows connecting a document of the first level to a set of terms of the second level.

2. Second mode of associative reading from a single document: During sequential reading of a set of documents the user can leave the reading of that precise set of documents and express interest in knowing the indexing terms associated to the term *it* which indexes the document *Dk* the user is reading in that moment. The user can obtain this list of terms associated to *it* and continue reading through the documents indexed by them. The interface prepares the list of associated terms passing from the level of documents to the level of terms and back to the level of documents, thus obtaining again a semantic association from the index term *it* to other associated index terms. This semantic association is not constrained by the auxiliary data structure hierarchy, but it is driven by the choices made by the knowledge engineers who indexed the documents. In Fig. 6 this mode of associative reading is graphically represented as for the previous mode of reading, with the arrows connecting a specific term of the set of terms to a set of documents belonging to the first level.

Backtracking. The user could be interested in the possibility to come back to a previous situation generated during the user's interaction with the environment. Depending on

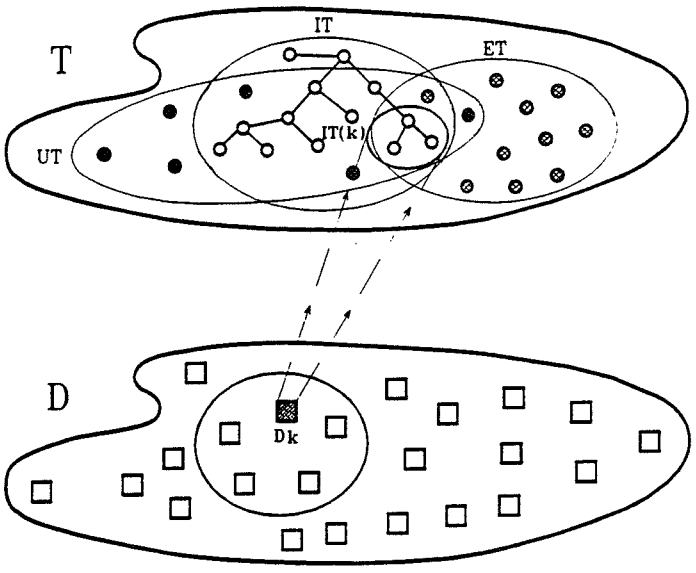

Fig. 5. Associative reading: First mode of operation.

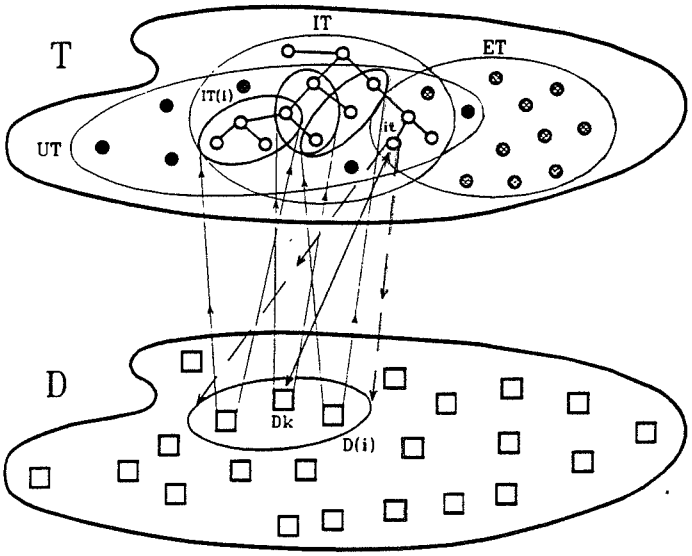

Fig. 6. Associative reading: Second mode of operation.

the level and on the specific function the user has just used, the backtracking function implements different ways of returning to the previous situation or possibility to navigate through the previously selected set of terms or documents.

History. The history function keeps details on the history of the user-system interaction during an interactive session. Since one of the major usability problems connected with tools that permit navigation in an information space (e.g., hypertext and hypermedia) is the user's risk of disorientation (Nielsen, 1990), the history function has been designed to be used in case the user loses the sense of context. Another useful aspect of this function is to keep details of all steps of the interaction, so the user can read through the interaction history to review the different interaction steps and to formulate new interaction strategies for future usage of the tool.

Support to search strategy development. At any time during the auxiliary data structure navigation or during the associative reading the user can identify some terms (indexing terms) as good candidates for a subsequent search strategy formulation. A function has been implemented in order to save these terms in a term pool. The experienced users or professional searchers can then re-use the terms in the term pool for search strategy development. In this way the classical query formulation techniques benefit also from the browsing mechanisms of the hypertext interface.

PROTOTYPE HIGHLIGHTS

A prototype based on the general functions of a hypertext environment described in this paper has been designed and developed. The prototype is available as one of the facilities of ESA-QUEST, the information retrieval system of ESA-IRS. The purpose of the project has been the integration under a unique user framework of two basic information retrieval elements: reference browsing and the concept-to-concept navigation. The project has been developed in the framework of the unifying model previously presented and with the specification of a functional environment capable of making the model and the functionality effectively available to the ESA-IRS users. The operative online environment in which the prototype is running entailed several constraints, as well as a variety of practical drawbacks. In the ESA-IRS service there is an availability of large, good bibliographic collections specifically designed for professional usage. The collection size can span up to about 3.5 million references, and the auxiliary data structure that we use is identified with a network of indexing terms reaching the size of 15 thousands terms. The user population

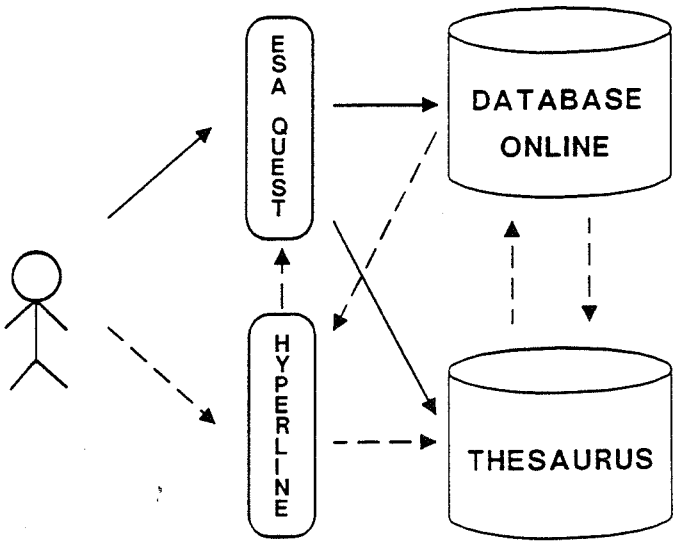

Fig. 7. Scheme of the prototype.

is rather heterogeneous in interests, level of knowledge of the IR system, and of the query language. This kind of environment, with the richness and variety of online data and the presence of a large user population, has made the steps of formative design and usability testing easier. On the other hand, the distribution of the user population over several countries, as well as the difference in equipment used for the online searches (it ranges from teletypes to personal computers and mainframes), has imposed strong constraints in the design. In order to make the prototype available to all ESA-IRS users, it has been decided to implement the prototype for accessing the ESA-IRS service from "dumb" terminals.

The prototype has been named HYPERLINE and has been officially and publicly presented in December 1990 in London during the 14th International Online Information Meeting. The scheme of the prototype and its relationships with the ESA-QUEST system are presented in Fig. 7. At present the prototype can be used with 10 different large textual databases. It is important to note that HYPERLINE is domain-independent, and it works on topics ranging from Aerospace to Metallurgy. Some tests have been performed on other multilingual collections; first results on these tests are encouraging, but further research needs to be carried out. It seems important to show here some examples of the most interesting features of how the prototype implements the semantic association, document reading, and navigation functions.

Significative examples of prototype functions

Semantic association. As stated above, the hypertext environment is designed to allow concept navigation through the auxiliary data structure. If the concept the user is interested in is not an index term, the hypertext environment performs a semantic association from the user term to the index terms conceptually associated to it. In the example shown in Fig. 8 the user is interested in the meaning and getting information concerning the acronym "SGML." SGML is not an index term for the bibliographic file (INSPEC) used in this example. This means that the user would not be able to know which concepts are associated to it without having an in-depth knowledge of the query language. What HYPERLINE does is provide the user with a list of suggested index terms conceptually related to the term expressed by the user. As previously introduced, the semantic association is performed using the knowledge stored in the auxiliary data of the bibliographic file. The ranking of conceptually related index terms is performed according to their occurrence in the most recent references. So, for the term SGML, the suggested concepts are those depicted in Fig. 8.

```
Your Input:

   1 - SGML
-------------------------------------------------------------------------------
Related thesaurus terms :

   2 - STANDARDS
   3 - ELECTRONIC PUBLISHING
   4 - DESKTOP PUBLISHING
   5 - ELECTRONIC DATA INTERCHANGE
   6 - WORD PROCESSING

-------------------------------------------------------------------------------
Enter (C)ontinue, (N)avigate, (S)how item, (G)et term
(T)op list, h(I)story, (H)elp or (Q)uit : n5
```

Fig. 8. Example of semantic association.

Navigation. If users wish to navigate the auxiliary data structure, they can choose, for example, the term "ELECTRONIC DATA INTERCHANGE." In this case the user obtains the answer reported in Fig. 9.

Document reading. At any time during auxiliary data structure navigation the user can choose the option of showing document references. This option allows reading of the reference items indexed by the actual indexing term (Fig. 10).

The first results of the feedback coming from formative evaluation of HYPERLINE are extremely encouraging: Users understand quickly the meaning of the messages in the menus and are amazed at being free to browse multi-million reference databases without knowing a single search command. The navigation aids that are usually in a hypertext context and implemented also in HYPERLINE (backtracking and history functions) seem to be sufficient to avoid the common feeling of "being lost." In this regard we feel that a strong support to the user is given by the explicit presence of the network of indexing terms built up with the thesaurus whose visibility has been maximised in this environment.

User session example

In the following an example from a user session interaction is given. The user is interested in "INFORMATION RETRIEVAL USER INTERFACE." In Fig. 11 the concepts HYPERLINE found semantically associated to it are shown. The user is interested in reading the bibliographic references sought (Fig. 12). The first reference read (Fig. 12) is cen-

```
----- thesaurus term ELECTRONIC DATA INTERCHANGE ----------------------------
Ref   Items term                                  Relationship
  1    1026 ELECTRONIC DATA INTERCHANGE
  2    5100 DATA HANDLING                         Previous
  3    1047 EFTS                                  Narrower term
  4    6481 CAD/CAM                               Related term
  5     823 INTEGRATED SOFTWARE                   Related term
  6   41817 STANDARDS                             Related term
  7    5100 DATA HANDLING                         Broader term
  8    5100 DATA HANDLING                         Top term
  9         DATA CONVERSION (SOFTWARE)            Used for
 10     396 DATA EXCHANGE                         Used for
 11      97 DATA INTERCHANGE                      Used for
 12      31 DATA INTERFACES                       Used for
 13      11 DATA PORTABILITY                      Used for
 14       6 DATA TRANSFER FORMATS                 Used for
 15     115 EDIF                                  Used for
 16         PORTABILITY, DATA                     Used for
 17         TRANSFER FORMATS FOR DATA             Used for
-------------------------------------------------------------------------------
Enter (N)avigate, (B)ack navigate, (S)how item, (G)et term,
(T)op list, h(I)story, (H)elp or (Q)uit : s1
```

Fig. 9. Example of navigation.

```
Show item 1 of 1026
Quest Accession Number: 91180726
  891052097, C91053012  INSPEC  Issue 9117
  Communications support for EDI
  Author(s): Debenham, M.J.
  IEE  Colloquium  on  'Standards and Practices in Electronic Data Interchange'
(Digest No.106), p. 8/1-3
  Published: 1991, IEE, London, UK
  Pages: 34
  Country of Publication: UK
  Meeting: 21 May 1991, London, UK
  Sponsor(s): IEE
  Document Type: CA (Conference Article)
  Treatment: P (Practical)

  There  are  many  parallels  between  passing  structured commercial messages
between  computer  systems  (which  is the objective of EDI communications) and
passing  unstructured  text between users' terminals (which is the objective of
                                                                        -More-
--------------------------------------------------------------------------------
Enter n(E)xt item, (V)iew thesaurus,
(A)ssociated terms in this item,
Associate(D) terms to <ELECTRONIC DATA INTERCHANGE>
or (Q)uit :
```

Fig. 10. Example of document reading for the index term: "ELECTRONIC DATA INTERCHANGE."

tered around the hypermedia concept, and the user tries to understand which are the possible associated concepts (Fig. 13). At this moment the user is therefore performing an associative reading, jumping to the concept "HYPERMEDIA" showing a bibliographic reference related to it (Fig. 14). Reading of the reference opens a new scenario, since the title is now talking about hypertext standardization. The user feels a bit too far away from the original starting point and asks for the list of concepts associated to "HYPERMEDIA." There (Fig. 15) the user receives proof that "HYPERMEDIA" and "USER INTERFACE" are associated concepts and this stimulates the user to return to the original list proposed by HYPERLINE (Fig. 11). A list of concepts is there to elicit associations. The user this time chooses to see the "DATABASE MANAGEMENT SYSTEMS" (DBMS) associated terms. Figure 16 gives the list of concepts associated to "DATABASE MANAGEMENT SYSTEMS." The user reads a few references related to "OBJECT-ORIENTED DATABASES" and then decides to browse the semantic structure again (Fig. 16). Then, getting the idea that "OBJECT-ORIENTED DATABASES" could be an interesting issue to look at, the user selects to navigate the thesaurus structure, choosing the path 6 in Fig. 16: "OBJECT-ORIENTED DATABASES." Figure 17 shows the associated concepts. Among the related concepts there is something that is again worth exploring, like "OBJECT-

```
EsaQuest is looking for related candidate terms

Your Input:

   1 - INFORMATION RETRIEVAL USER INTERFACE
--------------------------------------------------------------------------------
Related thesaurus terms :

   2 - INFORMATION RETRIEVAL
   3 - USER INTERFACES
   4 - INFORMATION RETRIEVAL SYSTEMS
   5 - EXPERT SYSTEMS
   6 - DATABASE MANAGEMENT SYSTEMS

--------------------------------------------------------------------------------
Enter (C)ontinue, (N)avigate, (S)how item, (G)et term
(T)op list, h(I)story, (H)elp or (Q)uit : s1
```

Fig. 11. Terms semantically related to "INFORMATION RETRIEVAL USER INTERFACE."

```
Show item 1 of 46
Quest Accession Number : 91075354
  C91023775  INSPEC  Journal Paper  Issue 9107
  The rhetoric of hypertext
  Carlson, P.A.
  Human Resources Lab., Intelligent Systs. Branch, Brooks AFB, TX, USA
  Hypermedia (UK)
  Hypermedia  vol.2, no.2, 1990, p.109-31, 8 Refs, ISSN: 0955-
8543, Country of Publ.: UK
  Treatment: P (PRACTICAL)

  As a fundamental orientation, the author adopts the view that
hypertext may eventually bring about a paradigm shift in
text delivery and in human information processing. However,
paradigm shifts do not occur overnight; they are evolutionary
rather than revolutionary. Because of the considerable
commitment of western knowledge and culture to the written word
and to linear text, it seems likely that successful hypertext
systems will electronically emulate many of the strategies a
sophisticated reader uses in dealing with hard
                                                          -More-
--------------------------------------------------------------------------------
Enter n(E)xt item,(V)iew thesaurus,
(A)ssociated terms in this item,
 Associate(D) terms to <INFORMATION RETRIEVAL USER INTERFACE>
or (Q)uit : a
```

Fig. 12. Display of the first bibliographic reference indexed by "INFORMATION RETRIEVAL USER INTERFACE."

```
Associated terms :
--------------------------------------------------------------------------------
   1 - APPLE COMPUTERS
   2 - HYPERMEDIA
   3 - INFORMATION RETRIEVAL SYSTEMS
   4 - WORD PROCESSING

--------------------------------------------------------------------------------
(S)how item, (V)iew thesaurus, or (Q)uit : s2
```

Fig. 13. Terms associated to the bibliographic reference shown in Fig. 12.

```
Show item 2 of 789
Quest Accession Number : 91079387
  C91023811  INSPEC  Conference Paper  Issue 9107
  An interchange format for hypertext systems: the Intermedia model
  Riley, V.A.
  Inst. for Res. in Inf. & Scholarship, Brown Univ., Providence, RI, USA
  Proceedings of the Hypertext Standardization Workshop (NIST SP 500-178)
  Gaithersburg, MD, USA  16-18 Jan. 1990
  1990, p.213-22, 10 Refs, Country of Publ.: USA
  Publisher: NIST. Gaithersburg, MD, USA
  Pages: vi+269
  Moline, J.; Benigni, D.; Baronas, J. (Editors)
  Sponsor: NIST
  Treatment: P (PRACTICAL)

  Realization of the potential for information sharing that
is inherent in hypertext systems depends on the ability to
readily exchange data between those systems.  A format for
exchanging link-related data between first-order
                                                          -More-
--------------------------------------------------------------------------------
Enter n(E)xt item, p(R)evious-one,(V)iew thesaurus,
(A)ssociated terms in this item,  Associate(D) terms to <HYPERMEDIA>
or (Q)uit : d
```

Fig. 14. Bibliographic reference indexed by the "HYPERMEDIA" term.

```
Associated terms :
-------------------------------------------------------------------------------
   1 - INFORMATION RETRIEVAL
   2 - USER INTERFACES
   3 - INFORMATION RETRIEVAL SYSTEMS
   4 - EXPERT SYSTEMS
   5 - DATABASE MANAGEMENT SYSTEMS
   6 - MICROCOMPUTER APPLICATIONS

-------------------------------------------------------------------------------
(S)how item, (V)iew thesaurus, or (Q)uit : v
```

Fig. 15. List of terms associated to "HYPERMEDIA."

```
----- thesaurus term DATABASE MANAGEMENT SYSTEMS -----------------------------
Ref   Items term                                    Relationship
  1   15009 DATABASE MANAGEMENT SYSTEMS
  2    4755 FILE ORGANISATION                       Previous
  3    9503 MANAGEMENT INFORMATION SYSTEMS          Previous
  4     228 DEDUCTIVE DATABASES                     Narrower term
  5    1444 DISTRIBUTED DATABASES                   Narrower term
  6     153 OBJECT-ORIENTED DATABASES               Narrower term
  7    4208 RELATIONAL DATABASES                    Narrower term
  8     515 APPLICATION GENERATORS                  Related term
  9     677 CONCURRENCY CONTROL                     Related term
 10     706 DATA INTEGRITY                          Related term
 11    2055 DATABASE THEORY                         Related term
 12    4021 DECISION SUPPORT SYSTEMS                Related term
 13     935 GEOGRAPHIC INFORMATION SYSTEMS          Related term
 14     789 HYPERMEDIA                              Related term
 15     782 INTEGRATED SOFTWARE                     Related term
 16     110 MULTIMEDIA SYSTEMS                      Related term
 17    1399 QUERY LANGUAGES                         Related term
 18     612 TRANSACTION PROCESSING                  Related term
                                                   -More  0.10-
-------------------------------------------------------------------------------
Enter (N)avigate, (B)ack navigate, (S)how item, (G)et term,
(T)op list, h(I)story, (H)elp or (Q)uit : s6
```

Fig. 16. The thesaurus structure for the term "DATABASE MANAGEMENT SYSTEMS."

ORIENTED PROGRAMMING" (OO), so the user performs some sequential reading on references related to (OO).

The user has grasped a certain number of ideas and therefore decides that it is time to go back again to the concept network (Fig. 17) and to view the history of his or her navigation (Fig. 18). The user, starting from the concept "INFORMATION RETRIEVAL USER INTERFACE," has explored areas like hypermedia, DBMS, and OO databases and programming. A final look at the concept network around the DBMS concept (Fig. 16)

```
----- thesaurus term OBJECT-ORIENTED DATABASES -------------------------------
Ref   Items term                                    Relationship
  1     153 OBJECT-ORIENTED DATABASES               Narrower term
  2   15009 DATABASE MANAGEMENT SYSTEMS             Previous
  3    2233 OBJECT-ORIENTED PROGRAMMING             Related term
  4   15009 DATABASE MANAGEMENT SYSTEMS             Broader term
  5    6447 COMPUTER APPLICATIONS                   Top term
  6    4755 FILE ORGANISATION                       Top term
-------------------------------------------------------------------------------
Enter (N)avigate, (B)ack navigate, (S)how item, (G)et term,
(T)op list, h(I)story, (H)elp or (Q)uit : s3
```

Fig. 17. Terms associated to "OBJECT-ORIENTED DATABASES" in the thesaurus.

```
----- Navigation History ----------------------------------------------------
 0 Your input  - INFORMATION RETRIEVAL USER INTERFACE
 1 Show        - INFORMATION RETRIEVAL USER INTERFACE
 2 Show        - HYPERMEDIA
 3 -Top List - - ---------------
 4 Navigate    - DATABASE MANAGEMENT SYSTEMS
 5 Show        - OBJECT-ORIENTED DATABASES
 6 Navigate    - OBJECT-ORIENTED DATABASES
 7 Show        - OBJECT-ORIENTED PROGRAMMING
 8 B-Navigate  - DATABASE MANAGEMENT SYSTEMS
(V)iew list again or (Q)uit : v
```

Fig. 18. History of the navigation conducted by the user.

shows that since the original interest was centered around the user interface, it is worth looking at references dealing with "MULTIMEDIA SYSTEMS." With this last browsing operation the user decides to quit the session.

This session example shows some of the peculiar characteristics of the hypertext environment when embedded in an information retrieval system. It is very easy to jump from concept to concept and perform sequential and associative reading. The concepts proposed are related to the one entered by the user, but the retrieved documents do not necessarily overlap the original topic. Therefore the hypertext tool can lead the user far away from his original aim. The richness of concepts and references available in a large operative information retrieval system can then require a considerable intellectual effort on behalf of the user and challenge one's ability to perceive and classify concept associations. On the other hand, it seems that concept association is a straightforward task for human beings, and it is the most basic unit of thinking and learning (Iran-Nejad, 1989). We do expect, therefore, that users will appreciate the large wealth of associations elicited by the HYPERLINE hypertext environment for large bibliographic collections.

CONCLUSIONS

This paper has described the conceptual design and functions of an operational prototype, named HYPERLINE, of a hypertext environment for interaction with large textual databases. The functions that are made available by the environment have been introduced and specified. Scope of the prototype is to make explicitly available to the final user the auxiliary data structure, which in this environment is a thesaurus structure. The prototype is based on a two-level architecture that permits the conceptual separation between auxiliary data structure and document collection. The prototype permits the direct use of the auxiliary data structure by the user together with the possibility of navigating the document collection, and the relationships between these two conceptual levels. This is made available through a set of functions that have been described. HYPERLINE is already in experimental usage by ESA-IRS users. From the beginning of the experimental usage of HYPERLINE, log data of the user-system interaction are collected. From an initial statistical analysis of the log data, it emerges that the effectiveness of the retrieval is improved by the possibility of moving between the two conceptual levels and a navigation in each of them.

Acknowledgements—The work of Maristella Agosti and Girolamo Gradenigo has been partly supported by the Italian National Research Council (CNR) under the project, "Sistemi informatici e calcolo parallelo–P5: Linea di Ricerca Coordinata MULTIDATA."

REFERENCES

Agosti, M., Gradenigo, G., & Mattiello, P. (1989). The hypertext as an effective information retrieval tool for the final user. A.A. Martino (Ed.), *Preproceedings of the III International Conference on Logics, Informatics and Law*, Vol. I, Firenze, 1–19.

Agosti, M., Crestani, F., Gradenigo, G., & Mattiello, P. (1990). An approach for the conceptual modelling of IR auxiliary data. *Ninth Annual IEEE International Phoenix Conference on Computers and Communications*, March 21–23, 1990, Scottsdale, Arizona, 500–505.

Agosti, M., Gradenigo, G., & Marchetti, P.G. (1991). Architecture and functions for conceptual interface to very large online bibliographic collections. *Intelligent Text and Image Handling, RIAO 91*, Barcelona, Spain, April 1991, Vol. 1, 2–24.

Aitchison, J., & Gilchrist, A. (1987). *Thesaurus construction—A practical manual* (2nd Ed). London: Aslib.

Belkin, N.J., & Marchetti, P.G. (1990). Determining the functionality and features of an intelligent interface to an information retrieval system. In J.-L. Vidick (Ed.), *Proc. 13th ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, Brussels, Belgium, 151–177.

Croft, W.B., & Thompson, R.H. (1987). I^3R: A new approach to the design of document retrieval systems. *Journal of the American Society for Information Science, 38*(6), 389–404.

Frisse, M.E. (1988). Searching for information in a hypertext medical handbook. *Communications of the ACM, 31*(7), 880–886.

Frisse, M.E. (Chairman), Agosti, M., Bruandet, M.F., Hahn, U., & Weiss, S. (1990). Panel Session: Hypertext: "Growing Up?" In J.-L. Vidick (Ed.), *Proc. 13th ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, Brussels, Belgium, 343–347.

Ingwersen, P. (1986). Cognitive analysis and the role of the intermediary in information retrieval. In R. Davies, (Ed.), *Intelligent Information Systems: progress and prospects*, (pp. 206–237) New York: Wiley.

Iran-Nejad, A. (1989). Associative and nonassociative schema theories of learning. *Bulletin of the PSYCHONomic Society* (US), *27*(1), 1–4.

Marchetti, P.G., & Belkin, N.J. (1991). Interactive online search formulation support. *Proc. Online Meeting N.Y.*, May 1991.

McAlpine, G., & Ingwersen, P. (1989). Integrated information retrieval in a knowledge worker support system. In N.J. Belkin & C.J. van Rijsbergen (Eds.), *Proc. of ACM-SIGIR 1989 Conf.*, Cambridge, MA (pp. 48–57).

McMath, C.F., Tamaru, R.S., & Rada, R. (1989). A graphical thesaurus-based information retrieval. *Int. J. Man-Machine Studies. 31*, 121–147.

Nielsen, J. (1990). The art of navigating through hypertext. *Communications of the ACM, 33*(3), 296–310.

Ritchie, I. (1989). HYPERTEXT—Moving towards large volumes. *The Computer Journal, 32*(6), 516–523.

Robertson, S.E., Thompson, C.L., Macaskill, M.J., & Bovey, J.D. (1986). Weighting, ranking and relevance feedback in a front-end system (information retreival). *Journal of Information Science, 12*(1–2), 71–75.

Shneiderman, B., & Kearsley, G. (1989). *Hypertext hands on! An introduction to a new way of organizing and accessing information*. Reading, MA: Addison-Wesley.