ACM PRESS

SIGIR '91

Proceedings of the Fourteenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval

edited by

A. Bookstein, Y. Chiaramella, G. Salton, and V. V. Raghavan

Sponsored by ACM SIGIR

held in cooperation with

AICA-GLIR (Italy)
GCS-IRSG (UK)
GI (Germany)
INRIA (France)

major corporate sponsor

Andersen Consulting

with additional support from

Bellcore

Chicago, Illinois USA
October 13-16, 1991

Special issue of the
SIGIR Forum
October 13-16, 1991

A TWO-LEVEL HYPERTEXT RETRIEVAL MODEL FOR LEGAL DATA

Maristella Agosti(*) - Roberto Colotti(+) - Girolamo Gradenigo(*)

(*) Dipartimento di Elettronica e Informatica, Universita' di Padova, Via Gradenigo, 6/a, 35131 Padova, Italy
(+) Softline, Via Piovese 108, 35127 Padova, Italy

ABSTRACT

This paper introduces an associative information retrieval model based on the two-level architecture proposed in [Agosti et al, 1989a] and [Agosti et al, 1990], and an experimental prototype developed in order to validate the model in a personal computing environment. In the first part of the paper, related work and motivations are presented. In the second part, the model, entitled EXPLICIT, is introduced. EXPLICIT is based on a two-level architecture which holds the two main parts of the informative resource managed by an information retrieval tool: the collection of documents and the indexing term structure. The term structure is managed as a schema of concepts which can be used by the final user as a frame of reference in the query formulation process. The model supports the concurrent use of different schemas of concepts to satisfy information needs of different categories of users. In the third part of the paper, the main characteristics of the experimental prototype, named HyperLaw, are presented.

1. ASSOCIATIVE INFORMATION RETRIEVAL AND MOTIVATIONS OF THE WORK

Associative information retrieval methods are all those retrieval methods which have been proposed and experimented since 1975 [Doyle, 1975] in order to expand query formulation by adding to an initial query some new terms related to the terms of the initial query, and similarly expanding the retrieved document set using terms related to the already used terms [Salton & Buckley, 1988]. [Salton & Buckley, 1988] contains an up-to-date situation in designing and usage of associative information retrieval methods and shows that the difficulty encountered in applying associative retrieval methods still resides in the identification of related terms and documents which would improve retrieval operations.

Concerning the currently available information retrieval systems, the most common situation is that of the user who is unable to actively interact with the indexing term structure used by the system. That is, the user is able to see a single indexing term or a list of indexing terms used in a document informative content representation, but often he cannot, at the moment of formulation of the query, effectively grasp the structure of the different indexing terms related to an indexing term of interest. A facility of this kind, if available, can prove very useful, but it is not sufficient to really inform the user of the structural organisation of the indexing terms; in fact this facility is normally available only in a non-integrated fashion and the user needs to know an indexing term beforehand in order to begin wandering through the semantic structure.

With the above-mentioned research and implementation considerations in mind, we started the work reported in this paper with the following underlying motivations:

- to give the final user the possibility to actively browse through the indexing term structure in order to acquire a proper understanding of the semantic context in which the meaning of each term has been defined by the indexers,
- to allow the user to see those documents related to possibly

useful terms at all times during interaction with the system,
- to satisfy the user's requirements in terms of transition through the connections existing between terms and documents.

This paper presents an associative information retrieval model based on a two-level architecture, proposed in [Agosti et al, 1989a] and [Agosti et al, 1990], and an experimental prototype developed in order to validate the model. This approach makes the conceptual structure of the indexing terms used in the application explicitly available to the user.

The model, called EXPLICIT, makes use of a schema of concepts for a specific information retrieval application domain; this schema can be used in an active manner by the user during user-system interaction: it provides the user with a frame of reference in the query formulation process. The model supports the concurrent use of different schemas of concepts for a specific information retrieval application domain; this capability can be very useful in providing access to the same document collection to different categories of users having different information requirements. The schema of concepts, which can be used in an active manner, operates as an associative information retrieval method giving the user the possibility to identify terms which are related to his information needs.

The experimental prototype, called HyperLaw, manages a collection of full text legal documents and a vocabulary of indexing terms expressly created by a research group of the Institute for Juridical Documentation (IDG) of the Italian Research Council (CNR) in Florence, Italy. The development of this prototype is motivated by the validation of the model in a personal computing environment. A previous prototype called HYPERLINE, was developed to experiment and validate the model in a completely different setting: an on-line information retrieval service managing large bibliographic collections. In this second type of setting, the model has been used to design and implement a semantic interface with hypertext capabilities to interact with the Information Retrieval Service of the European Space Agency (ESA/IRS). This conceptual interface supports the user through the use of very large bibliographic collections providing a facility for browsing through the structure of indexing terms and the collection of documents related to it [Agosti et al, 1991].

2. THE TWO-LEVEL HYPERTEXT MODEL

2.1 ARCHITECTURE AND BASIC MODEL ELEMENTS

The EXPLICIT hypertext retrieval model is based on a two-level architecture, which holds the two main parts of a database managed by an information retrieval system: the **collection of documents**, and the **auxiliary data**. By the term auxiliary data we mean the data describing the document information contents. It is important to note that the meaning of an auxiliary data item becomes fully defined only by means of the semantic relationships existing between this auxiliary data item and other terms. The model itself is based upon specific information modelling constructs and tools for management and retrieval of information.

To make the presence of those two parts (i.e. the collection of documents and auxiliary data) of the information resource managed by an information retrieval system explicit, we have considered an architecture which permits operations on two different levels of abstraction:

1st) the level which contains the networks of documents of interest (e.g. full text documents);
2nd) the level which contains the network of the semantically related concepts; this is the plane of abstraction where indexing terms together with their structure are placed; the objects of this level result from the application of the classification abstraction mechanism to the objects of the first level;

the connections established between the first and second level carries out the relationship between concepts and the documents described by these concepts.

The system which is going to be used for implementation of the two-level architecture and the EXPLICIT model needs to be able to operate both as a document collection management tool and also a means by which the semantic representation of the collection contents may be handled.

The abstraction mechanisms of classification, generalisation/specialisation, and aggregation included in the paradigm underlining the proposed architecture (the paradigm has been presented and justified in [Agosti et al, 1990]), have proved sufficient for operations concerning modelling and organisation of the representations of information which may be contained within the documents of the collection.

To fully present the characteristics of the architecture and the EXPLICIT model their presentation has been organised in the following three parts:

- the initial part (2.2) illustrates all the characteristics of the architecture and the capabilities of the model at the first and second level, including relationships existing between the two levels;
- the middle part (2.3) shows the type of user-system interaction supported together with the retrieval capabilities; and
- the final part (2.4) introduces the management and updating capabilities which permit the administration of the informative resource.

Each of these three parts presents the structural or static characteristics, and the management or active capabilities.

2.2 ARCHITECTURE, MODEL AND DATA MANAGEMENT CAPABILITIES

2.2.1 FIRST LEVEL

The collection of interest is represented and managed at the first level of the architecture. The collection is made of objects of the real world; in the common practice of information retrieval these objects are textual documents, but they could be other types of objects: photographs, tapes, motion picture films, pictures, images, etc..

Each real world object of the collection is a physical entity which is directly modelled to a 1 to 1 correspondence with an object of the representation. Each object has its own identity and status. The identity of the object is independent of the manner in which it is represented or structured and of the values it may assume; the identity makes each object distinct from all the other objects of the collection.

The representation of an object of this level is made by means of:

- a set of structured data which represents the different deterministic properties of the object (e.g.: date of publication, title, list of authors, etc.);
- a text fragment: i.e. an abstract of the document if it is present within the document itself, or a summary which is prepared as a representation describing the informative contents of the document;
- connections to documents which are related to it; connections can be activated with documents which are somewhat related by means of: their informative content (e.g.: documents on the same subject), membership of the same series (e.g.: the series of the ACM SIGIR proceedings), etc.; and
- connections to the auxiliary data items (i.e. indexing terms) which represent the informative content of the object; the auxiliary data items and their structure are represented and managed at the second level of the architecture.

The collection of document objects is organised at the first level of the architecture as a "hyperdocument", that is in the form of a lattice structure. Each node of the hyperdocument is an informative item consisting of the document representation, which follows the previously introduced structure, together with the text of the document. A complete document may be represented by a node or a set of nodes.

The hyperdocument is made up of a network of structural links combined with the network of reference links. It is important to note that for links, the directional semantics are considered significant. This means that the user may choose to follow along one path or another even in consideration of the direction of the references present within the semantic units. The significance of such possibility is evident if we consider the difference in the semantics for a text being referred to, or in one referring to another. The features of structural and reference links are presented below in the following subsections.

Structural links permit organisation of the nodes because they represent and implement physical relationships existing between parts of documents. Structural links reflect the hierarchical structure existing in the original documents. The structural links are obtained by connecting a father node to its offspring in order to form a branching diagram within the global multidimensional diagram of the hyperdocument. It is important to note that these links are set up by the designer and administrator of the hyperdocument; the final user can merely use them.

Reference links permit the semantic relationships existing in the informative content of nodes to be represented. Reference links can also be handled by the final user according to his information interests of linking documents.

The EXPLICIT model supports some functions that the user can activate during usage of the hyperdocument; these functions permit the active interaction between user and system at the first level of the architecture; these functions are presented in the remaining part of this section.

The model supports navigability through the document collection. Due to the fact that specific cross-references are often present between the documents of the collection, the system must explicitly be able to support navigability through these connections. Furthermore, assessment of one item of information generally stimulates request for other information in further depth. The implementation of a hypertext network between the various information items permits their direct consultation.

To reduce the common problems of disorientation and knowledge overload (see for example [Nielsen, 1990] for a presentation of these common problems of user-hypertext system interaction) which face the user during the use of the hyperdocument, a simple searching technique for detection of text strings located within the full text information items has been introduced. In fact the opportunity of locating with a certain approximation the whereabouts of some nodes and to use them as starting points for one's own queries has been considered quite important. Following the results of the user's requirements analysis which has been initially conducted, it has been decided to include in the model only a simple string search function, because the results of the analysis have indicated that it was not considered really important to include particularly sophisticated search functions.

Since it has been judged that a system based on the EXPLICIT model needs to provide the possibility of keeping reference of nodes which are considered as being interesting in order to be later reconsidered for a deeper inspection of the hyperdocument, the model supports the possibility of creating reference "marks" to information items which are considered relevant. Re-activation during a later time, or even in a later work session, of these reference marks can take the user back to any specific point of the path that has been constructed during a previous inspection of the hyperdocument. The reference mark can to all effects become a starting point for subsequent analyses and search operations. It is important to note that the role of a reference mark is to mark a node that is semantically relevant to a user's demands for information.

2.2.2 SECOND LEVEL

The auxiliary data structure is represented and managed at this level of the architecture. In order to effectively design and manage auxiliary data, the model offers constructs supporting the fundamental abstraction mechanisms. Everything existing in the application world, even those linguistic entities used to describe other entities, are modelled as real objects. All abstractions, having as their purpose the conceptual organization of objects, are represented as links between objects.

Objects of the second level result from the application of the classification abstraction mechanism to the objects of the first level; they denote concepts which are variously interrelated, for example through a classification hierarchy, a specific case of which may be the IS_A hierarchy.

Objects and links of this level form the "hyperconcept", that is, a parallel structure to the "hyperdocument", whose task is to handle the semantic structure of concepts used to describe the contents of document collection. This level is conceptually located above the hyperdocument and performs the same functions performed by the usual auxiliary data of an operative information retrieval system.

In the adopted approach for data design and management, each auxiliary data item is viewed as a class of objects; instances of each of these classes are the documents which are pertinent to the specific concept expressed by the class term. Hence each 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 from two different points of view: as a set of instances or as a whole entity. Thus the properties of the class can be separated into:

- the properties assigned to a class as an object, that is, the properties of the auxiliary data as such, and
- the properties of the objects of the class: i.e. properties of the documents belonging to the class, identified by the auxiliary data.

The process of associating an auxiliary data item to a document object, in order to describe its informative content, corresponds, in the model, to setting up an instance-of relationship between a class term, i.e. an object of the second level, and a document, which represents an object of the first level. Being a document generally indexed by more than just one single term, a document object proves to be an instance of various different term classes. These modelling capabilities support the notion of polythetical classification (as defined in [Van Rijsbergen, 1979]).

One can imagine each auxiliary data item as defining a set of first level objects: the documents which are an "instance-of" specific auxiliary data, that is the documents indexed by that particular auxiliary data. On the other hand, document sets defined by two or more distinct terms are not necessarily disjointed so the common elements of the intersection are documents which belong to different classes.

While links between concepts and document objects model the classification abstraction mechanisms, links connecting conceptual objects express generalisation and specialisation abstraction mechanisms. If two concepts are related at the second level by a specialisation relationship, the two corresponding sets of documents at the first level are associated by means of a sub-set relationship; the relationship between sets of documents is an inclusive-set one if the concepts are related by means of a generalisation mechanism. As for the generalisation/specialisation relationships, all the structuring mechanisms of auxiliary data items corresponds at the first level to a set structure of the collection of documents.

At this level navigability through the semantic structure permits formulation of a query by means of the identification of a semantic path through the reference structure. The use of simpler structures (i.e. classification schemes) or of more complex nature (i.e. thesauri) has no essential significance in determining the construction of the mechanism.

2.2.3 RELATIONSHIPS BETWEEN FIRST AND SECOND LEVEL

This section describes what kinds of relationships are necessary between the first and the second level of the architecture and the operations which need to be supported by the EXPLICIT model.

Each of the two levels of the system's architecture represents a distinct network of nodes and links. The relationship between the terms included in the hyperconcept and the related documents present within the hyperdocument are described by a peculiar type of link. By means of this type of link it is possible to set up a mechanism of access to the information items, for example, through a thesaurus of the hyperdocument domain.

The semantic link between two different second level class objects corresponds to a set relationship between corresponding class instances of the first level. Depending on the type of link existing between class objects of the second level, different kinds of set relationships can be established between the extension sets of the first level.

The experimental system was planned and set up to manage the connections located within, as well as between, the two hypertext levels. According to the EXPLICIT model, the hyperconcept and the hyperdocument which compose the two levels of the architecture are each independent from the other. This means, for example, that the insertion of a new descriptive term into the hyperconcept does not imply any modification of the hyperdocument; in the same way, insertion of a new document doesn't entail any variation in the hyperconcept; the only consequence is an activation of new connections between the hyperconcept and the hyperdocument.

The model supports navigation between the two levels by means of the navigability function. In this way it is at all times possible to pass from the hyperdocument to the hyperconcept and back again.

2.3 USER INTERACTION AND RETRIEVAL CAPABILITIES

The independent nature of the two levels of the system's architecture allows us to take a step further, that is, it offers us the opportunity to construct different and distinct hyperconcepts upon the same hyperdocument. In this way it is possible to obtain different semantic descriptions of the same document collection, that is different views for different categories of users. This feature is quite significant, because a user specialised in a specific field tends to use different terminology compared to that used by a generic user. This means that we are given the opportunity to construct different access mechanisms and different types of user interaction according to the different access requirements of the various categories of users.

Such freedom of movement requires the support of an appropriate function which makes the users' interaction easier. The EXPLICIT model therefore supports the backtracking function. With this function the user is supported in finding the way to go back, step by step, along the path from whichever point in the connection network he has reached. This is important in that it limits the requirement for user know-how, imposed by the presence of various alternative paths during navigation. This function is supported also because the user needs to be provided the possibility to take any path without being afraid to lose reference, for instance to the other alternative directions. The backtracking movement is possible for any path into and between the two levels of the architecture.

2.4 UPDATING CAPABILITIES

An important feature has been devise in order to permit automatic updating of the hyperconcept and hyperdocument. The model allows insertion and removal of single items of information maintaining the integrity of reference within the two level structure. The image is that of a binding fabric woven around the information items. The model supports automatic generation of the necessary bonds for insertion of new nodes and removal of others which no longer have any significance.

Each object handled by the system is a carrier of information essential to the system itself, in order to place it in its appropriate location within the binding fabric. This approach makes it possible to resolve the difficulties concerning insertion and deletion of nodes within a hypertext network. When the object is inserted in the network it becomes a node of the structure. The data which the object contains are modelled as property values of the object and become, when inserted, actual node attributes. Some node attributes can be, for example, name, node type (e.g. legal authority documents, law documents, auxiliary data items), or the link type.

As soon as the user enters a node, the system acquires information about the node by means of these attributes, as well as the relations binding it to the rest of the hyperconcept and the hyperdocument. Thanks to the information identifying the node type, the system may automatically find out which functions it must activate while consulting the node.

A structure of this kind is truly fundamental in handling the evolution of information managed by the system. Operations such as acquiring new documents or deleting nodes which no longer have any significance are performed automatically. For a traditional hypertext system instead, the operation of inserting new nodes and establishing the appropriate links within an existing network often involves a lot of manual work on behalf of the user.

3. HYPERLAW: A PROTOTYPE IMPLEMENTATION FOR LEGAL DATA

3.1 VALIDATION OF THE MODEL IN A PERSONAL COMPUTER ENVIRONMENT

The validation of the architecture and of the EXPLICIT model has been accomplished by means of an experimental system expressly devised for a personal computer environment. We have implemented this experimental system on the basis of the model and architecture previously presented.

The main objective was to obtain a tool which would make good use of the characteristics of the reference concept tools in a stand-alone environment. Such a tool has been specifically designed for a personal computer environment, which is generally distinguished for its handling capabilities with document collections being not too extensive in dimensions but often of a non-homogeneous nature. In this environment the possibility to make use of more highly sophisticated technology devices and at lower prices brings about a growth in demand for powerful personal information retrieval tools.

Personal computer users normally pay a special attention to the problems related to user interfacing, sometimes giving up somewhat on the performance requirements. Those who make use of document collections on personal computers normally interact directly with the document set itself and manage any query in an interactive manner. The access tool to information contained within this environment must therefore be capable of balancing the handling efficiency with the most natural usage of the system.

Hypertext technology has a built-in set of characteristics which are suitable for carrying out these requirements, as underlined also in [Wilson, 1988] specifically for legal documents. The tools currently available in the personal computer environment are already mature for this purpose and present all the necessary characteristics of which the hypertext model is made. Their usage is therefore already possible even beyond the need to develop a prototype.

Specific hypertext characteristics can be briefly summed up in the following points:

- hypertext systems permit an information consultation approach to research operations by means of a matching process similar to that performed by the human brain: the user "navigates" through the information in order to gain access to it;
- utmost flexibility in defining the kind of information which can be handled (the limits are set practically by the machine-system being used); and
- a special attention dedicated to interfacing requirements.

The freedom of managing potential offered by a hypertext model generally presents a few drawbacks, more precisely:

- creating some difficulty in performing efficient research operations on information collections of moderate size, due to the limited power of the functions offered to support information retrieval;
- an overloading of the knowledge requirements on behalf of the user who must be able to carry out the necessary choices between alternative search paths and keep in mind the semantics of the path taken.

The EXPLICIT model seems to offer the necessary opportunities, balancing out the required characteristics an information retrieval system in a personal computer environment should have, as described previously, together with the characteristics of hypertext technology. Although hypertext systems represent a valid tool in information handling operations, these are in fact rather limited on a functional basis for retrieval of the same information. The architecture here being proposed allows the user to make adequate use of the interconnection capabilities between information items comprised in the hypertext within a precision structuring system.

3.2 HIGHLIGHTS OF THE PROTOTYPE'S MAIN CHARACTERISTIC FEATURES

In setting out the construction of a system of this kind, it has been found necessary to implement the set of identified and previously presented functions and capabilities of the architecture and of the EXPLICIT model.

The system thus created, called HyperLaw [Colotti, 1990], is an experimental tool for handling legal collections of full text and reference documents: laws, case law, legal authority. The question of information management in legal spheres is quite problematic: the information patrimony to be used is quite vast and growing daily.

The handling of legal documents has a rather peculiar feature: the logical connections between them are very tight-binding. For example, a norm text can almost never be properly interpreted without making reference to other normative law documents; in a similar manner, a judicial sentence applies one or more norms and can therefore be fully understood only by tracing back to the original texts of these norms. It is therefore practically impossible to totally satisfy one's information requirements by consulting a single document. This extensional feature of the legal information patrimony has always strongly urged the need for suitable automatic information retrieval tools.

The results of a study, [Agosti et al, 1989b] whose purpose was to identify the real requirements of the user of legal data and of the applicability of the hypertext technology in the management of legal documents has shown that, since the 1960s, in this field the offer for specialised data banks has enormously proliferated. For various reasons however the available tools have never been used to their full capacity; some justification for this situation have been: a certain mistrust towards these new and unfamiliar tools, a significant cost factor (even up to the present date), and a certain objective difficulty in performing consultation operations.

The latter problem must be studied in further detail. The information tools which managed and currently manage these data banks permit access to a very vast information patrimony. Unfortunately, they generally present an operative difficulty which usually ends up frustrating the operator and sometimes even the more expert user. On top of this, usually we find that the required information is scattered about in various different data banks. This of course implies that, in order to carry out a complete search operation, a jurist is forced to enter and exit various information retrieval tools (i.e. various data banks), each having its own operative environment, procedures and language. There is, therefore, an enormous load of user know-how necessary, and this often prevents an adequate use of the resources offered by the information retrieval tool on behalf of the jurist.

Another aspect to be underlined is the fact that the information retrieval systems traditionally used for handling these data banks do not make it possible to move directly from one point ot another within the network of cross-references and quotations which are a peculiar feature of legal information, as previously said. Search operations are therefore made much heavier owing to the extra work load for the human operator in following these references.

The characteristics which hypertext systems display, together with the architecture and EXPLICIT model, can provide an adequate answer to the specific information requirements of a jurist, to whom we are making reference. HyperLaw has therefore been planned and developed bearing in mind the need to experiment a tool capable of handling, in an integrated manner, the vast, non-homogeneous legal information patrimony. The document collection used includes norm texts (State, Regional, Provincial laws, etc.), judicial norms (extracts and sentences) and indications of doctrinal documents, such as summaries or extracts. The vocabulary used takes the form of a classification scheme and has been expressly created for experimentation by a research group of the Institute for Juridical Documentation (IDG) of the Italian Research Council (CNR) of Florence, Italy.

3.3 THE WORK ENVIRONMENT

HyperLaw has been developed using HyperCard as a software development tool [Goodman, 1988]. The user's work environment inside HyperLaw is represented in a totally graphic manner and the user-system interaction takes place almost exclusively by means of a mouse. The interface used by this system divides the screen into two separate sections:

- the right sector contains a series of push-buttons which activate a certain number of functions, such as performing print-outs, displaying node references, navigation along a predetermined structure, etc.;
- the central section of the screen is reserved for display of information.

The user must therefore identify and indicate the object he is interested in by pointing it with the cursor: this usually consists of an icon representing a set of documents, a text string indicating a document, or some other kind of identification. By clicking the mouse button the object pointed is activated, i.e. the system receives the order to move in the direction indicated and to present the pertaining information or to execute the requested function.

3.4 HYPERLAW'S MAIN FUNCTIONS

Thanks to the hypertext characteristics implemented in the system, every access to information is performed through direct movement or navigation. HyperLaw's opening screen presents the architecture of the system being referred to, isolating the document collection and semantic structure onto two separate levels (see Figure 1). This screen display represents both the point of access to the whole system and the reference point which can be recalled at any moment of operation. To activate one of the icons present on the screen means shifting into the relative hyperdocument and starting consultation. It is possible to shift directly from any point of the hypertext network to other hyperdocuments by making use of the links existing between them.

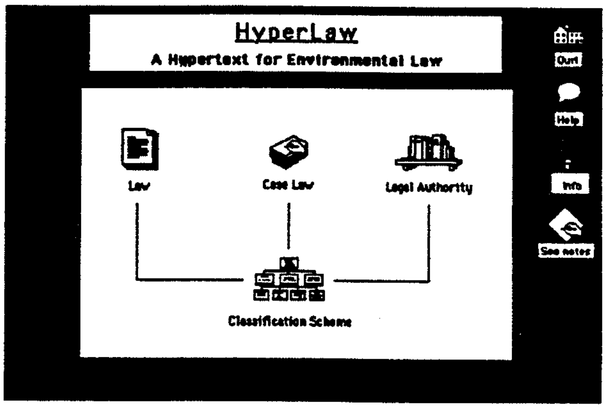

Figure 1. HyperLaw's opening screen

We present hereunder a few examples of the implementation of the EXPLICIT model functions in HyperLaw.

<u>Navigation through the semantic structure</u>

In the current version of HyperLaw a classification scheme of strictly hierarchical nature has been used. The semantic terms which correspond to the terminating "leaves" of this structure are the actual access points of the documents. In the system's opening screen the icon of the corresponding semantic structure must be activated. Thus the list of options which represent the first level of the scheme are displayed. The option relating to the environment of interest has to be selected. According to the choice made, another subsequent hierarchical level display or the list of terms contained within the chosen option can be obtained (see Figures 2 to 4).

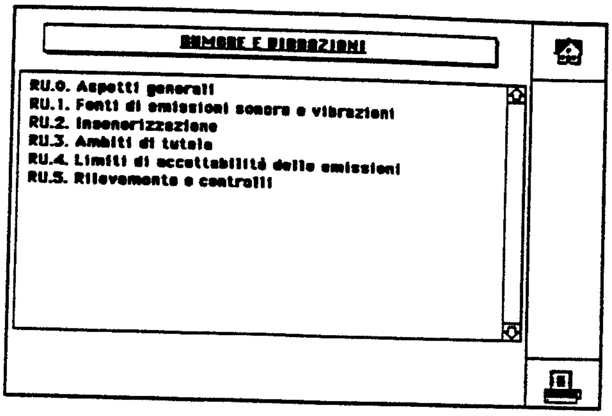

Figure 2. Classes of the classification scheme (the first level of the hyperconcept)

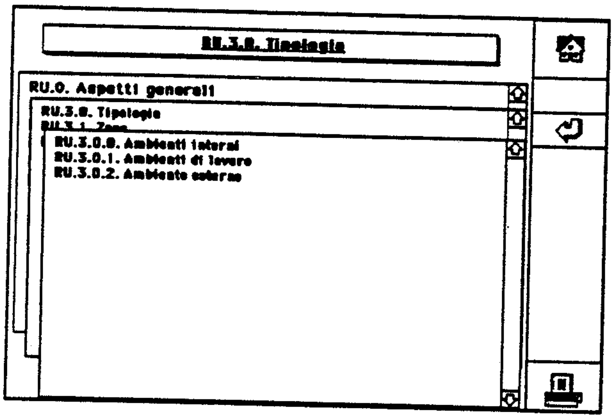

Figure 3. Moving through the semantic structure of the hyperconcept

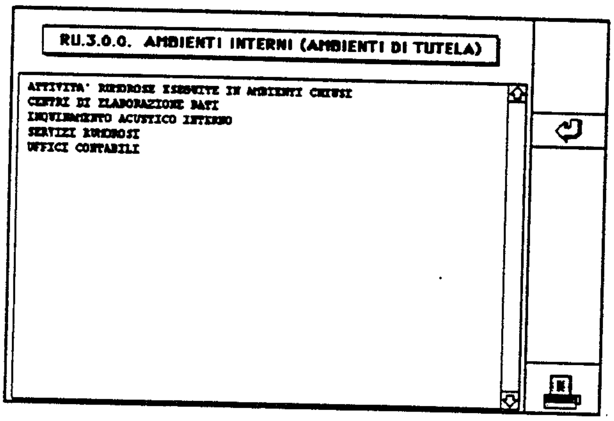

Figure 4. Terminal leaves" of the classification scheme

By activating the term, access is gained to the intermediate node which permits discrimination of access only to the documents semantically represented by that specific term. This function allows separate access to the different types of documents or attention being paid directly to a single document collection. We indicate along the bottom the cardinality of each cluster of documents bound to the specific term (see Figure 5).

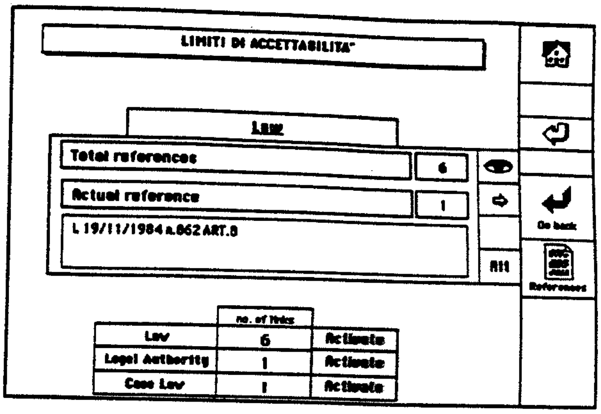

Figure 5. The intermediate node between the hyperconcept and the hyperdocument. This node permits descrimination of access to the different kinds of document

<u>Navigation through the documents</u>

The nodes included within the single documents contains a function which allows all the links which bind that single document to the others to be displayed. A first classification of these links can be made into active and passive links, according to the direction of the link. By the term <u>active link</u> we mean a binding connection which has its origin in the current node; on the other hand a <u>passive link</u> is a recollection to a section of another document.

When a request for display of the type of link is made the procedure runs along the structure to identify the nodes located at the opposite end of the connection and these are presented according to their typology (see Figure 6). Clicking the interesting reference with the mouse button leads to the relative document (see Figure 7).

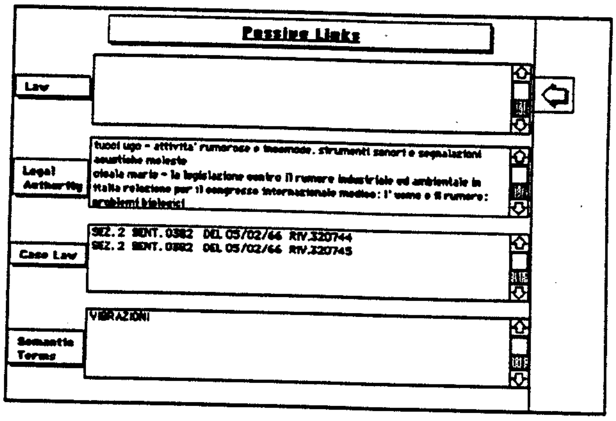

Figure 6. Following link: all the passive link of a node, descriminated by the kind of document tied

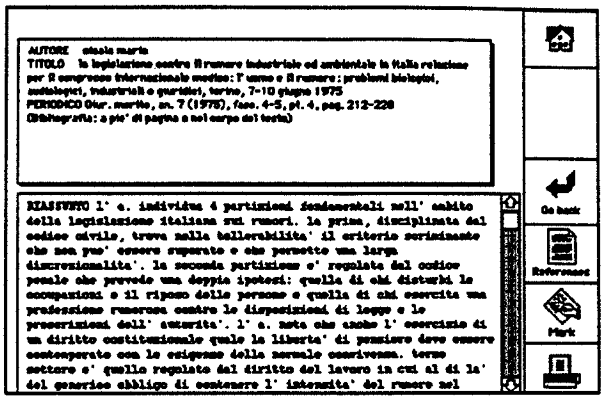

Figure 7. An example of a node: the representation of a Legal Authority document

Navigation between the two levels of the system architecture

This function is obtained by means of the two preceding ones. The nodes, i.e. the single semantic terms from which the hyperdocument links arise, are located at the extremities of the semantic hierarchy of the classification scheme. Activation of one of the links gives access to the referenced document.

From any one of the hyperdocument nodes one may skip directly to the hyperconcept by means of the active or passive link display functions of the node itself. The connection existing between the document and the concept represents a passive link for the node which contains the document. The function displays the semantic links referring to the node as a category of their own (see the bottom window of Figure 6).

Reverse path

It is at all times possible to return back along the path taken for consultation. Following the first step, in each node a specific arrow icon is activated allowing the path to be retraced back, one step at a time, towards the starting point.

Usage of reference marks

Owing to the utmost freedom of movement allowed in document consultation, it may be difficult to keep in mind the various different documents already visited and considered of interest or in any case worthy of further deeper analysis. The function described herein permits creation of a reference point to a single document in order to permit re-examination at a later time, even repeatedly. A node containing a list of references is generated and handled externally with respect to the basic hypertext network. This means these links can be reactivated either from within the system or even from the outside. An appropriate recall procedure to these reference points from outside the system reactivates it and repositions the user at a specific point of the hypertext network.

Depending on choice, these references can be stacked together by the user, one group per each node. These groups usually represent the work session. Within each node a set of references is stored away, giving the opportunity to display, delete or print them out on paper. Upon activation of a reference, the system moves the user to the relative document allowing him thereby to start a normal work session with HyperLaw, or go back to the beginning after having consulted the text. If a print-out of a reference is requested, HyperLaw retrieves the entire text of the document referred to and sends it out to the printer (see Figure 8).

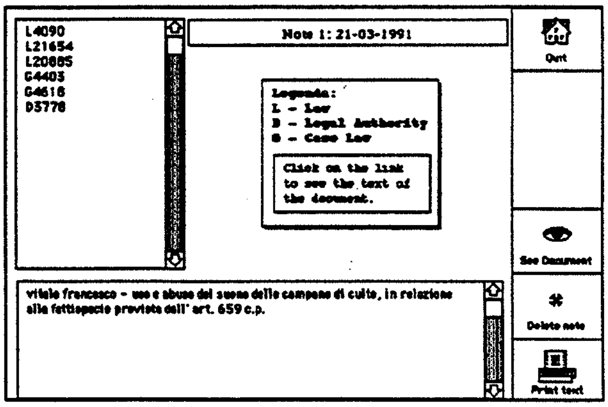

Figure 8. Usage of reference signals: on the upper left there is the list of mnemonics of the recorded references; at the bottom there is the description of the node referred by the signal actually selected

Automatic treatment of insertion of new documents

In systems like the type described herein, two aspects of the treatment of documents are particularly delicate: importation of new documents, and the creation of new links between the documents just imported to those already present within the application.

Thanks to the object-oriented approach given to this experimental system, together with a particular Data Description Language expressly defined for the purpose, it is possible to create a procedure external to HyperLaw which permits automatic treatment of these situations. It is in fact possible to insert new information items into the hypertext network of the system loading them in from an external source file. The system, making use of the information included in the file, is then able to automatically establish the new node and relative link. This function is particularly important in the handling of a set of information which evolves in time. The possibility to automatically connect the new documents permits the resolution of that which is considered the main problem concerning the treatment of a hypertext, that is, the task of creating the necessary links. Thanks to the structural set-up of the system itself, deletion of a node which is no longer significant and the resolution of its relative links becomes a very easy task.

The procedure itself lets the user add his own links to those originally foreseen in the documentary basis. If on one hand this function is fundamentally important within canonical hypertext systems, in this system it was thought better to avoid its implementation during consultation. This choice was taken in view of the fact that the set of documents and links present in HyperLaw were pre-defined and prepared by experts. The indiscriminate insertion of new links at the pleasure of the user could diminish the validity of the interconnection network semantics set beforehand. A possible solution could have been the assignment of specific attributes to links inserted by the user which would distinguish them from the others, but for the current version of the system its introduction was not considered of interest.

CONCLUSIONS

This paper has introduced the main characteristics of the associative information retrieval model known as EXPLICIT. The experimental prototype, called HyperLaw, based on this model has also been introduced together with some examples of the experimental implementation. The prototype applies the model to a personal computer environment.

The experimental results that have been collected by use of HyperLaw give further helpful suggestions for the refinement of the different aspects of the model. One important development currently under study is the enrichment of the model to make it capable of supporting different representations of the same object; the support for multiple representations of the same object would permit use of the model in the designing of multimedia objects for the effective management of multimedia object characteristics.

ACKNOWLEDGEMENTS

The HyperLaw version presented herein has been partly developed within a co-operative research project with the Institute for Juridical Documentation (IDG) of the Italian Research Council (CNR), Florence, Italy; IDG/CNR provided the documents and the vocabulary used. An initial version of the prototype was presented by Roberto Colotti in November 1989 at the 3rd International Conference on Logics, Informatics and Law, held in Florence [Colotti, 1989].

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

The authors acknowledge a partial support of TecLogic - Padova and Apple Computer Italia.

REFERENCES

[Agosti et al, 1989a] M. Agosti, G. Gradenigo, P. Mattiello. The hypertext as an effective information retrieval tool for the final user. In: A.A. Martino (Ed), Pre-proceedings of the 3rd Int. Conf. on Logics, Informatics and Law, Vol. I, Firenze, 1989, 1-19.

[Agosti et al, 1989b] M. Agosti, A. Archi, R. Colotti, R.M. Di Giorgi, G. Gradenigo, B. Inghirami, P. Mattiello, R. Nannucci, M. Ragona. New prospectives in information retrieval techniques: a hypertext prototype in environmental law. Online Information 89, London, England, 1989, 483-494.

[Agosti et al, 1990] M. Agosti, F. Crestani, G. Gradenigo, P. Mattiello. 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 et al, 1991] M. Agosti, G. Gradenigo, P.G. Marchetti. Architecture and functions for a conceptual interface to very large online bibliographic collections. Intelligent Text and Image Handling, RIAO 91, Barcelona, Spain, April 1991.

[Colotti, 1989] R. Colotti. HyperLaw: a hypertext for environmental law. Presentation at: 3rd Int. Conf. on Logics, Informatics and Law, Firenze, 1989, pp.17.

[Colotti, 1990] R. Colotti. HyperLaw: Prototipo per un sistema ipertesto per il diritto dell'ambiente. Ottobre 1990, pp.31.

[Doyle, 1975] L.B. Doyle. Information Retrieval and Processing. Melville, Los Angeles, CA, 1975.

[Goodman, 1988] D. Goodman. HyperCard developer's guide. Bantam Books (Macintosh Performance Library), New York, 1988.

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

[Salton & Buckley, 1988] G. Salton, C. Buckley. On the use of spreading activation methods in automatic information retrieval, Proc. of ACM-SIGIR 1988 Conf., Grenoble, France, 1988, 147-159.

[Van Rijsbergen, 1979] C.J. Van Rijsbergen. Information Retrieval (2nd Ed). Butterworths, London, 1979.

[Wilson, 1988] E. Wilson. Integrated information retrieval for law in a hypertext environment. Proc. of ACM-SIGIR 1988 Conf., Grenoble, France, 1988, 663-677.