A Framework for Building Hypertext Based Diagnostic Systems

Pierre Morizet-Mahoudeaux[1]   **Paul Terray[1,2]**   **Vincent Brunie[1,3]**   **Gilles Kassel[1]**

[1] UMR CNRS 6599 Heudiasyc
University of Compiègne
60205 Compiègne, France

[2] Institut National de l'Audiovisuel,
4, avenue de l'Europe
94366 Bry-sur-Marne, France

[3] Clinical Computer Eng. Unit
Hôpital de la Pitié-Salpétrière
75634 Paris CEDEX 13, France

ABSTRACT

Among the artificial intelligence techniques useful in diagnosis the integrated diagnostic techniques are often presented as the most efficient to resolve practical problems. This approach may lack however some flexibility to be applied in many application domains, when the role played by users takes an important part of the diagnosis process. Hypertext systems offer now an interesting approach whenever users' interactions are important. However, to fulfil the requirements imposed by practical and professional diagnosis, hypertext systems must satisfy constraints, which generally are not considered for common applications of hypertext. They are, for example, the ability of offering different *projections* of the same document; effective automatic navigation tools, and a well formulated representation of involved knowledge concepts. This paper presents a framework to build hypertext system-based diagnosis. First, we propose a definition of hypertext systems, which is more appropriate to account for the structural properties, which exist in any document. These structures become perceptible by the means of projections. A projection is decomposed into two steps: an interpretation operation, which is the computation of the document, and an instantiation operation, which is the transformation of the digital document into a perceptible document. Second, we show that by separating nodes content from hypertext documents structures, it becomes possible to implement efficient tools for document reading and synthesis. Last we refer to a model ontology building method, which we use for building domain dependant diagnosis ontology to model hypertext-system based diagnosis. The project of building a hypertext-based system for aiding in the acoustic and vibration diagnosis of rotating machines is finally used to discuss a possible application of the system.

1. INTRODUCTION

Artificial intelligence techniques useful in diagnosis can be classified into five categories. They are: fault-based techniques(FBR), model-based techniques (MBR), case-based reasoning techniques (CBR), machine learning for knowledge acquisition, and integrated diagnostic techniques. The integrated diagnostic approach is often presented as superior to each of the four preceding ones, in the sense that it takes advantage of each of their strength [1]. In addition to integrating AI techniques, efficient results can be obtained by integrating AI techniques with more algorithmic techniques such as real-time (RT) approaches [11, 12].

Although very attractive one major weakness of the integrated approach is that the role played by each of the integrated techniques and their relationships are more or less fixed by the application domain. Even if the employed AI techniques use non-predictable search-based problem-solving approaches, selections are made from these alternative problem-solving techniques given a priori knowledge of the most appropriate ones.

It is easy to imagine however that the same diagnostic system may be used differently according to the occurrence of a given failure with respect to the general state of the device. The same diagnostic system may also present information, results and explanations differently according to the needs of the user. The same user may want to have different points of views of the same resolution process. The user may be interested in having some freedom for building relations and cooperation between different diagnostic techniques.

Hypertext systems have been thought as systems, which are able to represent and support the association of different information sources. Their purpose is to provide the user with the enriched information resulting from the association of the information sources. This objective is obtained by the use of navigation tools. Navigation tools use nodes and links. Each node is an information source. Links are built from one part of a node to another node (or a component of another node). If the information sources were diagnostic techniques such as presented above, and if the links were the representation of how they interact to resolve a given diagnostic problem, the resulting hypertext system may be seen as an integrated diagnostic system. However, to be more effective than integrated diagnostic techniques presented above, the hypertext systems techniques should propose improved solutions.

Improvement can be the result of developing a specific approach to hypertext based diagnostic systems. More precisely it is necessary to provide the users with methods and the corresponding tools for developing such systems. These methods can be based on the one hand, on a thorough study of the electronic document and its use in a hypertext environment. On the other hand, they can be based on a

formalized approach for building a diagnosis typology and diagnosis ontology of the domain. Such an ontology can be built in the frame of the previous hypertext documentary approach, and thus lead to the development of a coherent hypertext based diagnosis system equipped with efficient navigation tools.

On the one hand, we have been working on a framework for developing task oriented navigation tools for hypertext systems. It is based on the notion of *contextual navigation*. Its objective is to account for the fact that the path followed by the hypertext reader is not only defined by the relations between nodes, but also by the context in which links and other possibilities of navigation appear. This led to define the notion of navigation structures, which become *perceptible* through *projections*. We have also shown that by separating nodes content from hypertext documents structures, it becomes possible to implement tools, that we call the *instrumentation of the reading*, which support the dynamic creation of documents while reading [4].

On the other hand we are working on building a diagnosis typology and diagnosis ontology in the application domain of the diagnosis of rotating machines. The approach is based on the assumption that, when limited to certain professional domain, the behavior of the user corresponds to types, which are *normalized by their practice*. We use a generic tool, supported by a programming language compatible with a hypertext representation, for building domain ontology, which is currently under development in our laboratory [9].

Thus, we come up with a framework for building hypertext based diagnostic systems. Section 2 describes the internal generic structure of a digital document and shows how it can be viewed through different projections for presenting information relevant to certain request of the reader. Then, we present the main characteristics of the implementation of a system, which support the instrumentation of reading. Section 3 presents our general approach for building domain dependent diagnosis ontology. A simple application is presented in Section 4. Section 5 is the discussion and the presentation of future work.

2. READING HYPERTEXT STRUCTURED DOCUMENTS

Numerous models of hypertext systems are built based upon the first definition given by Bush, describing the MEMEX system. It consists of a trail of pages, coming from a collection of books and articles put on microfilms. The documents are linked by the reader while reading, based on mental association [5]. This trail can later be "reactivated", to follow one's path of thoughts. If this mechanism, described just after Second World War, is now a reality thanks to digital text, it was not thought in the perspective of numeric systems.

Reading in context

We have not access to digital documents directly, since they always are an abstraction of the electronic state of the computer, which we model through a bunch of numbers. This calls for defining the operation that makes the document readable. We will call it a *projection* of the digital document.

The Spatial Model: Generally, hypertext is defined as a network of information nodes, connected by links that allow passing automatically from one to the other. This builds up a graph structure that defines the hypertext. It provides the users with the ability to create, manipulate, or examine a network of information-containing nodes interconnected by relational links [8]. This representation allows using a graph to represent a whole hypertext system.

One important shortcoming of this approach is that nodes have *a fixed display*. Although not explicitly stated, the Dexter based models suppose that the node content implies the node display. This comes from the principle that the nodes format includes the way in which the node is displayed. Consequently, most models do not take into account the content of the node itself, stating that the way in which it is displayed is not dependant on the hypertext system. These systems are only characterized by the graph of nodes. Nodes are considered as functional units, structurally and semantically complete, as if they were still on a static medium. Another characteristic of the current models is that the links do not belong to the description of the nodes. A link is "anchored" in the node, but this anchoring is considered to be dependent on the structure and the format of the node. We call this kind of models the *spatial models*. It is however possible to describe the nodes layer at a generic level.

The model of the interaction of the reader is the description of a path in the graph of nodes. The problem to be solved is then to find the relevant path, i.e. the relevant actions of the reader for reading the hypertext, during a single session. Instead of models of the reader's path in the graph, we can build models of the reader's reading action. They are based on structuring the instantaneous state of the hypertext system during the reading action. These models use a statechart, which represents the possible states of the hypertext system [14].

The action of reading can be limited neither to the study of paths in graph of nodes, nor to the study of statecharts of the hypertext. Reading includes also the context in which navigation functions appear in a node. We mean that the information, which appear in a hypertext document, influence the information supported by the link. Thus the meaning of a link depends both on the structure and on the content of a node.

Hypertext nodes *embed* their own way of navigation for browsing documents. Therefore, they are highly dependent on the content of the node. By putting this navigation tools into context, hypertext allows the reader to navigate while reading. It results in a *contextual navigation*. This implies that the path taken by the hypertext reader is not only defined by the relation of the nodes, but also by the context in which links and other possibilities of navigation appear.

As we noticed before, most hypertext models and systems stand on a single representation of the node. Several systems use however multiple views of the same document to

circumvent the disorientation problem. The possibility for the reader to look at different representations of the same node is complex to describe with the spatial model. Therefore, we need another definition of the hypertext model.

The Projection of a Document: A digital document is composed of a limited and linear set of values. Each document follows a format, which allows its interpretation into a more complex structure. For example, a bitmap graphic file contains the size of the picture and the values of the pixels. By knowing the width of the picture and the number of bit per pixel, any program can convert it into a matrix of values, which can be transformed further into a screen picture. The format specifies the structure of the file, and the use of the data.

We cannot access the document itself, as a set of values. We only access it through what we call *projection*. It is the means by which a format is made perceptible. The digital document is a virtual document in that perspective: it does not possess its own means of perception. Even if we look at the bitmap file with a byte editor, we still access it through one projection. It can be shown as a list of values, a histogram of color intensity, a picture, etc. There is no unique way of projecting a file, and no projection is more authentic than another is.

A projection can be exhaustive, partial, or *synthetic*. The table of contents from a structured text is a projection of the document, even if it omits a great part of the file, since it is still a view of the same file. Another synthetic projection can be an index of a document; any diagram based on statistics, a graph of a resolution procedure, an intensity diagram of a graphic file.

A projection can be decomposed into two steps:
- An interpretation operation, which is a computation of a document.
- An instantiation operation made by the hardware, which is the transformation of the digital document into a perceptible document.

The second operation is the actual transformation of the document from its digital format to a perceptible form. This step corresponds to making *actual* the virtual document. That is why we call it an *instantiation operation*. The interpretation is based in the *calculability* of digital documents. By allowing operations on the numerical content of documents, the computer makes them active. A schematic presentation of links between document content and structures is presented in figure 1. The elements corresponding to the black thick lines only will be projected.
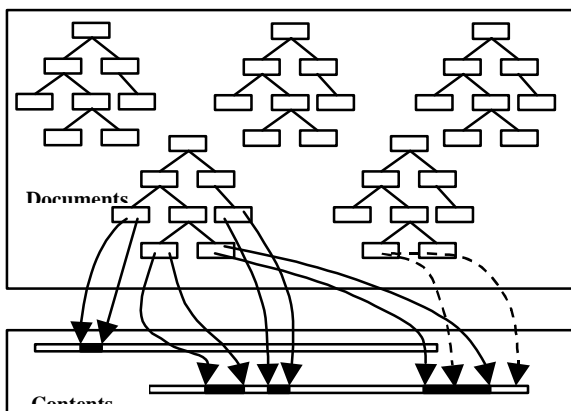

Figure 1 : Representation scheme of links between structures and contents

As we have seen above, hypertext and hypermedia differ from other kinds of interaction with digital document by the localization of the navigation tools in the document. Thus, we can define hypermedia documents from this characteristic. The navigation tools of the hypertext are produced by the projection of the node content.. With this approach hypertext is considered from the point of view of the nodes. This leads to consider that the main characteristic of hypertext is that the navigation structures are localized in the nodes. In the next sub-section we show an implementation framework, which accounts for this separation of structure from content.

Reading Structured Documents

The most general way for implementing a hypertext system is to use a structural markup scheme, for example SGML. It permits to build representations of documents. Rather than relying on explicitly marked links, navigation is driven by the structures of the documents. We have used this frame for developing a prototype of electronic patient record management with a hypertext system, which is shortly presented in this section (more details are presented in [4]). The documents, which constitute the electronic patient record, are defined according to the paradigm, which has been introduced in the previous section.

Reading a *Dossier*: We call *hyperdocument* a set of documents connected together by a pragmatic interdependence. This interdependence is related to their participation to a whole created by a practice. This is why a hyperdocument is more than an arbitrary collection of documents. For example, medical records are hyperdocuments generated as the consequence of the medical activity concerning a given patient. Thus all the documents appearing in a given medical record are related to the same patient. However, a hyperdocument is less than a document because it does not ensure a sufficient cohesion between its parts to establish a canonical reading order.

The medical record belongs to a class of hyperdocuments that we call *dossier*. A dossier is a special hyperdocument, which has neither a unique author, nor a unique editor. In the dossier, there is no room for an editorial team to build reading tools. The reader, who is the only one who has a global view of the current dossier at the time of its reading, must do this work. We call this work the *instrumentation of the reading*.

Structuring Documents: The generics of the uses of the dossiers are one of their major characteristics. They can be used for very different tasks, and so they can be considered as working tools. In a professional context, it is possible to work out working tasks relying on the different uses of a dossier. This classification leads to a low number of typical consultations of the dossier. The type of the consultation determines a reading strategy. An example of classification of the medical record into four types can be found in [13]. These types correspond to different reading strategies, driving to different kinds of readings. In the same way as reading situations are standardized by their professional context, reading types are also standardized as corresponding to well

defined activities. We can note that these reading types rely on the same structures, which are used in very different ways. An important characteristic of these structures is the generics of their use.

We have established that some readings are standardized enough to be useful to many potential readers of the record. Then, the result of this reading may be collected in a new synthetic document. We call such a document a *structuring document*, because it proposes a reading of the record by selecting parts and organizing them into a new structure. A structuring document offers a direct access to selected contents. Therefore, structuring documents can become reading tools, with the potentiality of dramatically increasing the reader' productivity.

We have seen in the previous sub-section that, once they are digitized, hyperdocuments become computable hypertexts. Specifically, this means that structuring documents may be automatically or semi-automatically computed, dramatically reducing the user's effort supplied for writing a new structuring document. This is an essential potential gain for the reader's productivity. Namely, we have used the reading skills based on document structures that the physicians have developed.:

The implementation of these synthesis tools has been made using the Standard Generalized Markup Language (SGML). Documents are described by a SGML Document Type Definition (DTD), using medical content tags, *e.g.* `<medical-history>`, and each tag used in the DTDs has a type defined by a generic tag of the architectural form, *e.g.* `<section>`, `<section-title>`, etc.

When new documents are added to the documentary database, synthesis tools are activated to generate the corresponding structuring documents. Pieces of documents are copied and organized into new structured documents. Theses new documents are added to the documentary database with the same status as the previous ones.

Reading tools: The generation of new documents, especially synthesis documents, implies the duplication of parts of the content of the generic documents. Actually, the content is not duplicated straightforward, but links are built in the generated document toward the generic document. It is however necessary to have links between duplicated and original contents in both direct and reverse direction to have a satisfying document *genesis*.

The prototype is currently tested with a corpus of medical records (10 records, 30 main document types, about 2,500 documents). A HTML projector gives generic information for structuring documents and a list of documents where the contents appear. These informations are given to users by the way of links with a special typography.

One interesting drawback of this study is that the system has been built based on an empirical ontology of the physicians' practice. It shows that our new perspective of hypertext system makes it possible to represent and *to instrument* such professional behavior. In the following section, we discuss the possibility of developing hypertext system based diagnosis, based on a model ontology representing the use of certain diagnostic techniques. However instead of an empirical domain ontology, it would use a more formal approach.

3. BUILDING A DIAGNOSIS ONTOLOGY

This discussion is based on our experience in building an application ontology for developing model-based diagnostic systems. We have developed a tool, based on a formal study, for building an application ontology composed of two sub-ontologies: the "domain-ontology", and the "problem-solving ontology" [9]. This, in turn, is used for building the diagnostic system [10].

Domain Ontology

In this paper, the conception of a domain ontology is understood as the description of the meaning of the concepts, which will be used for describing different situations of a given world. (e.g., physical object, component, state, process).

The ontological knowledge concerns intensional objects and not extensional objects. This conception of ontology corresponds to what is generally defined in AI as the specification of a conceptualization, where conceptualization depicts a set of concepts or more precisely of intensions. The different methods, which have been developed, for building ontology follow from this definition.

Building a domain ontology corresponds to study the essential characteristics of objects and not to study any hazardous property they could have in specific situations. The essential characteristics of objects are such that, if a defined object looses only one of them, it would not longer exist as such. These properties must be true by intension as long as the object exists. These properties are definitional, in the sense that objects that bear them are recognized as members of the type in every possible world [3] This notion is equivalent to the notion of the semantic rigidity of a concept. A concept is semantically rigid if it contributes to the very identity of its instances, in such a way that, if an object is an instance of this concept in a particular situation, it has to keep it to be an instance of this concept in any possible situation to keep its identity [7].

Problem Solving Ontology

Until now we have only considered concepts that denote objects, the role of which is to permit the construction of the model of a world. We have called such conceptualization "domain ontology". The *world model* is used by the diagnostic system to perform its task.

A diagnostic system contains also knowledge about how it realizes its tasks. This "meta-knowledge" is useful, for example, to explain problem-solving steps or to improve its diagnostic performance. We call this meta-knowledge the *problem-solving model* and the corresponding conceptualization the "problem-solving ontology".

We have recently proposed that the link between these two models matches exactly the link between the concept and the object. Giving the status of "meta-object" to the concept has done this [9]. This concept status, more precisely the concept intension status, stems from the fact that it corresponds to what the problem solving software knows about an object. We have also proposed to give the status of object to the concept, thus allowing expressing knowledge about the concept (e.g., the concept may play the role of a sign or of a syndrome). The concepts of "sign" and "syndrome" belong to the problem-solving ontology.

Once the causal link between concept and object has been expressed, it can be used during the diagnostic reasoning. Particularly, the problem solver can use this link to build justifications. This causal link partakes to the problem solving model and not more to the object world model. This accounts for the fact that an object of the world cannot have a direct effect on the state of a problem solver. Only the concept, i.e. what the problem solver knows about an object can have an effect on the state of the problem solver. It can, for example, launch other solving tasks by invocation of other concepts.

4. DIAGNOSIS OF ROTOR UNBALANCE

The main result of our approach is that problem solving concepts can be represented in an ontology as well as domain concepts are, and that this knowledge can be represented in hypertext structures. We have developed a model to test the feasibility of the approach. In this model we consider the case of rotor unbalance. In consists in several HTML pages describing a simple rigid rotor, some mechanical equations of movement, a table of possible causes and signs of rotor unbalance, a set of pages describing diagnostic methods, and CGI links to MatLab signal processing modules and diagnostic reasoning modules

A simplified HTML representation of the ontology is presented in figure 2. Figure 3 presents a sample of HTML pages, which were called on the screen for an interrogation of the system. From the home page describing the general mechanics of the rotor unbalance, the user can call a list of diagnostic techniques, detailed mechanical equations, or a synthesis table of causes and symptoms.

This table is built by the system by fetching in the objects and concepts ontology the elements, which compose the table. The relevant parts of the objects, which must appear in the table, have been previously marked-up by tags properly defined in the DTDs describing the content of the involved documents. The user can then select one or several causes to know, for example, how the observable signs can occur, how they can be characterized and processed, and to have more information on the repair procedures linked to the default. In this case, the interaction with the system can be seen as calling the explanation module of an expert system in rotor unbalance diagnosis.
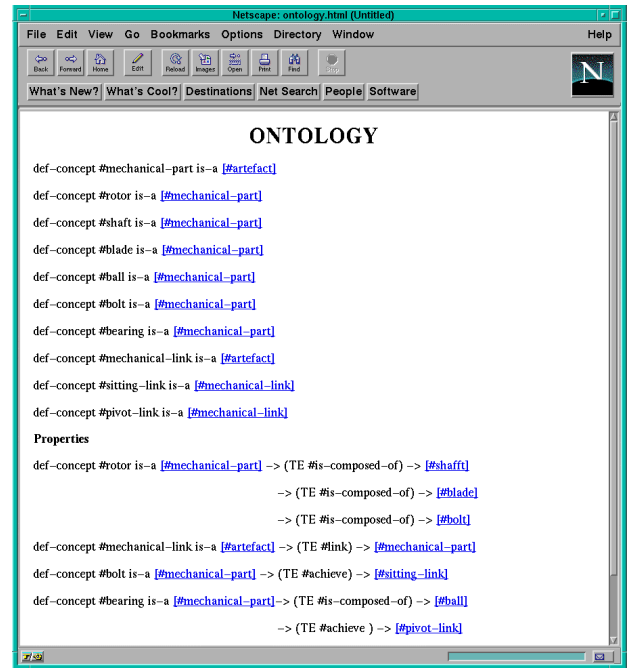

Figure 2: HTML representation of the ontology

5. FRAMEWORK FOR FUTURE EXPERIMENT

We are currently developing a Leonardo da Vinci European project in the domain of building a hypermedia tool for the acoustic and vibration diagnosis of rotation machines. The project is developed in cooperation with Three Europan universities and for European companies.:

This tool, based on a multi-media CD-ROM, will be realized by a multidisciplinary team. The project will cover basic methods of signal diagnosis, which apply in the relevant domain. It will be organized in modules developed according to industrial needs. The modules (e.g. basic methods of signal diagnosis, rotating machines, guide for the choice of appropriate methods, etc.) will contain interactive consultation modes, with several levels of input, complexity, and purpose. In addition to ease of use, the multimedia support will offer access to interactive simulations and training, which will show the comparative performances and limits of certain methods.

We intend to use the scope of this project to implement the approach presented above. The most difficult point concerns the building of the ontology. Examples of using a model ontology construct to guide the diagnosis already exist in the domain of model-based reasoning, for example [2]. We have also succeeded in constructing ontology for developing medical diagnosis expert systems [6, 10]. Our hypertext approach was proved satisfactory for building interactive electronic-patient record management [4]. Finally the test application presented in the preceding section has shown the feasibility of the overall approach.
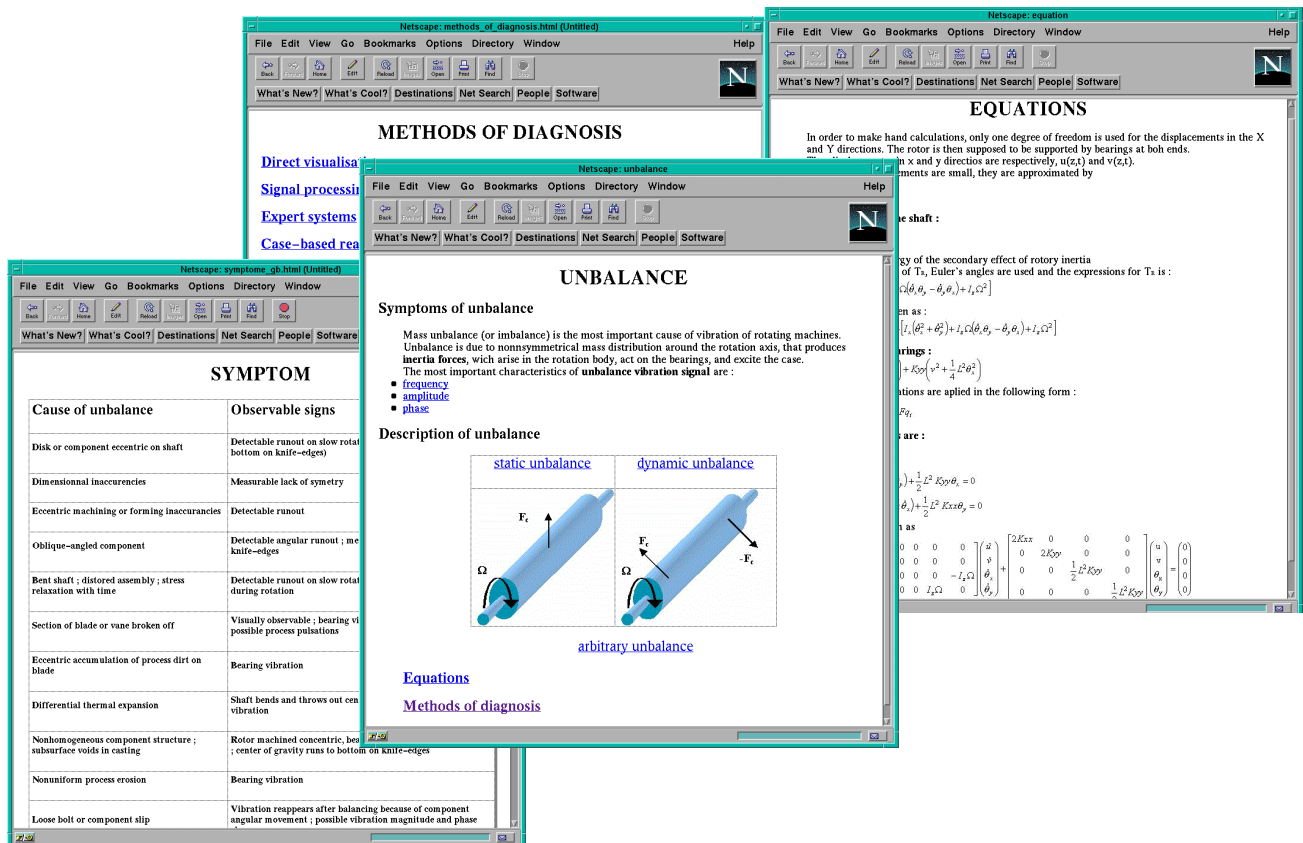
Figure 3: Presentation of original and synthetic documents (e.g., the table of causes and symptoms) built by the system

REFERENCES

[1] Abu-Hakima, S. 1994. Artificial Intelligence Techniques in Diagnosis: a Review of Approaches, Applications, and Issues, Technical Report, NRC 37142, National Research Council Canada.

[2] Abu-Hanna, A.; Benjamins, R.; and Jansweijer, W. 1992. Integrating Multiple Model Types in Model Based Diagnosis. In *Proceedings of the Third International Workshop on Principles of Diagnosis*, 179-184, Rosario, WA.

[3] Bouaud, J.; Bachimont, B.; Charlet, J.; and Zweignbaum, P. 1994. Acquisition and Structuring of an Ontology within Conceptual Graphs, In Proceedings of ICCS'94 Workshop on Knowledge Acquisition Using Conceptual Graph Theory, 1-25, Univ. Maryland College Park, MD.

[4] Brunie, V.; Morizet-Mahoudeaux, P.; and Bachimont, B. 1998. Separating Textual Contents from Structures for Reading Hypertext Structured Medical Records, in Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia, Pittsburgh, PA, Forthcoming.

[5] Bush, V. 1945, As we may think, The Atlantic Monthly.

[6] Greboval, M.H.; and Kassel, G. 1994. The Production of Explanations Seen as a Design task: a Case Study, In Proceedings of the Eleventh European Conference on Artificial Intelligence, ECAI'94, 351-355, Amsterdam.

[7] Guarino, N. 1992, Concepts, Attributes and Arbitrary relations, *Data and Knowledge Engineering* 8:. 249-261.

[8] Halaz, F.; and Schwarz, M. 1994. The Dexter Hypertext Reference Model, *Com. of the ACM* 37(2):. 30-39.

[9] Kassel, G.;and Traore, M. 1997. Modelling with a Strongly Intensional Language Brings Numerous Advantages, In Proceedings of the Seventh Workshop on Knowledge Engineering, Milton Keynes, UK.

[10] Krim, G.; and Kassel, G. 1995 SATIN: un Système à Base de Connaissances pour l'Aide au Traitement des Infections Neonatales, *Revue Européenne de Technologie Biomédicale* 17(1): 19-24, Elsevier eds.

[11] Morizet-Mahoudeaux, P. 1996. On-Board and Real-Time Expert Control, *IEEE expert, Intelligent Systems and their Applications* 10(4): 71-81.

[12] Musliner, D.J.; Hendler, J.A.; Agrawala, A.K.;and Simon, H. 1995, The Challenges of Real-Time AI, *Computer* 28(1): 58-66.

[13] Nygren, E.; and Henriksson, P. 1992. Reading the Medical Records I: Analysis of Physicians' Way of Reading Medical Record, *Computer Methods and Programs in Biomedicine* 39: 1-12.

[14] Zheng, Y.; and Pong, M.C. 1992. Using statecharts to model hypertext, in ECHT.