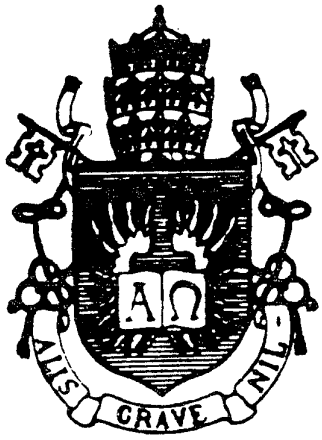

PUC

Series: Monografias em Ciência da Computação
Nº 27/89

A KNOWLEDGE-BASED BROWSER TO ACCESS COMPLEX DATABASES

Daniel Schwabe
Eduardo E. Mizutani

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO
RUA MARQUÊS DE SÃO VICENTE, 225 - CEP-22453
RIO DE JANEIRO - BRASIL

PUC/RJ - DEPARTAMENTO DE INFORMÁTICA

Série: Monografias em Ciência da Computação, 27/89

Ed: Paulo Augusto Silva Veloso September, 1989

A KNOWLEDGE-BASED BROWSER TO ACCESS COMPLEX DATABASES

Daniel Schwabe
Eduardo E. Mizutani

This work has been partially sponsored by FINEP.

In charge of publications:

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC RIO, Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453 - Rio de Janeiro, RJ
BRASIL

Tel.:(021)529-9386 TELEX:31078 FAX:(021)274-4546
BITNET:userrtlc@lncc.bitnet

A KNOWLEDGE–BASED BROWSER TO ACCESS COMPLEX DATABASES *

Daniel Schwabe and Eduardo Edison Mizutani
Departamento de Informática
Pontifícia Universidade Católica do Rio de Janeiro
R. M. de S. Vicente 225, Rio de Janeiro, RJ 22453
BRASIL

E-Mail: USERSCHW@LNCC.BITNET

June 1989

We present a knowledge–based browser that helps users access a statistical database containing data about the Brazilian social security system. The main problem in accessing this data base is the difficulty for casual users (and even expert users) to correctly identify the *names* of the data items of interest, in this case, historical time series. The browser helps the user through a *semantic model* of the domain, allowing the reference to familiar concepts, and avoiding the need to know any information about the data base itself.

1 INTRODUCTION

The Brazilian social welfare system is vast and complex, and thus difficult to manage. In order to assist administrators in the ministry to manage it, a statistical database of historical time series, named SINTESE, was built.

SINTESE contains statistical data related to the whole social welfare system, such as number of benefits granted, total amount disbursed in sick pay, etc. Every item in the database is a two dimensional time series, where the first dimension is always time, and the second dimension is usually a geographical specification, but may represent other concepts such as age, occupation, etc.

In addition to data items specific to the social welfare system, SINTESE also contains several general purpose items, such as economy indices, general population statistics, etc.

Data contained in SINTESE is meant to be used in preparing policy studies conducted by the ministry, as well as to observe and assess the effectiveness of measures being taken

*This research was partially supported by a contract with DATAPREV.

at certain times. An example of such studies is the evaluation of the impacts on the Social Welfare system caused by the recent changes in the constitution.

The current size of the database is of the order of 16000 different time series. As can be expected, the major obstacle facing users is the identification (naming) of the relevant time series for the task at hand. In other words, the user knows *what* information s/he wants, but does not know what are the *names* of the time series (if they exist) that contain that information.

To overcome this problem, a knowledge based browser was built to help users find the relevant time series for their tasks. It is knowledge based because it contains a semantic description of the task domain (the Social Welfare system) in the form of a semantic network, as well as heuristics associated with the typical types of access users make to SINTESE.

The system described here has points in common with other information retrieval assistants, notably CoalSORT [3], EP-X [1], Rabbit [4] and Backbord [6], although none of these systems has the same functionality as the SINTESE ASSISTANT.

2 The SINTESE ASSISTANT

2.1 Overview

The SINTESE ASSISTANT (SA) facilitates the access to the database by presenting a semantic model of the task domain to the user, in the form of a semantic network [5]. Thus, users are able to refer to familiar concepts and their relationships, "navigating" through the network in order to find their topic(s) of interest. As can be expected, users may also directly name concepts, or use synonims from an extensive list. In this process, users will mark all of the concepts of interest. Once this has been done, the SA collects these concepts and formulates the meaningfull queries involving them, presenting the queries to the user for selection.

Once the user has chosen a query, the SA uses the semantic network as an *index* to retrieve associated names of series, presenting this information back to the user. Unfortunately, a fairly common result of this operation is that there are no series satisfying the query, as we have found that typical users expect a level of detail beyond what is available in the database. At this point, the SA will try to formulate alternative queries, as simplifications of the original one. These alternative queries may, nevertheless, result in relevant information to the task. The user is then free to accept these alternatives or to try to reformulate the query using other criteria.

Once the user has obtained a set of series names, s/he may get further detailed information about each series, such as a short textual description of its contents, the time of last update, periodicity, etc.

2.2 The semantic description of the domain

To allow users to refer to familiar concepts of the task domain, the Social Welfare system was modeled using a semantic network. Nodes represent *concepts*, and arcs represent *relations* between concepts. The usual "a-kind-of" (AKO) relation is used: nodes related

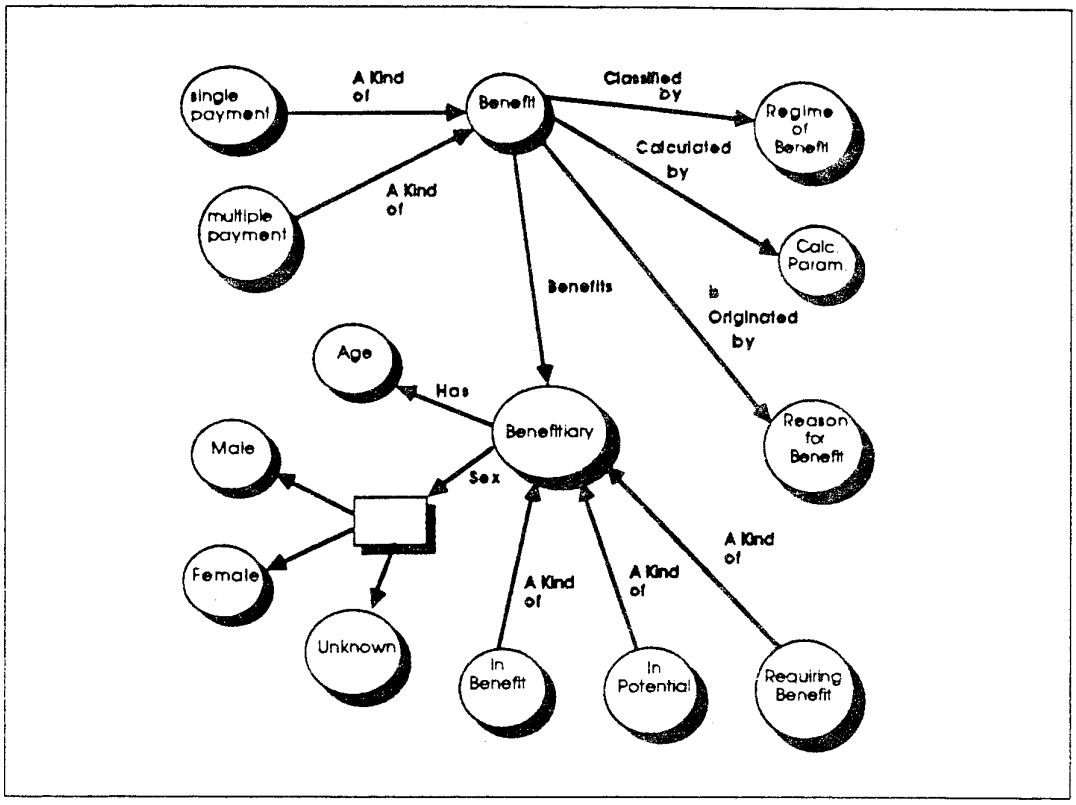

Figure 1: Example of a semantic network

through this relation form a hierarchical sub-network. Some nodes represent concepts that can be seen as *properties* of another concept or as an *attribute* of some relation. For example, the concept `form of payment` can be considered as a property of the concept `benefit`; and `date` an attribute of the relation `makes payment to` between `benefitiary` and `bank`.

A concept and its properties may be regarded as forming a *frame* [2], where the name of the frame is the concept, the slot names are the relations, and the slot values are the concepts to which the given concept is linked through the relations. The AKO relation is used to allow factoring of properties along paths of the corresponding hierarchy. Thus, concepts at lower levels inherit properties from concepts further up in the hierarchy. Figure 1 shows an example of a part of the semantic network used.

When presenting a concept to the user, the SA forms the corresponding frame from the concept, the associated properties (including inherited ones), and the corresponding values as present in the semantic net. In addition, it may also present slots corresponding to properties of descendant nodes in the AKO hierarchy; this is controled by the value of a system parameter which can be set according to the level of user expertise. This last feature is included to allow users to refer to concepts indirectly via their properties; in

cases in which this reference is not unique, the system will enter a dialogue with the user in order to disambiguate the reference.

The actual modelling of the Social Welfare was done by experts from the Ministry, based on a preliminary model done at the data processing company of the Ministry, Dataprev. The model thus generated was then validated by another group of experts.

2.3 The semantic model as an index

Once the modelling phase was concluded, a team of experts from Dataprev, who are very knowledgeable about the series contents, took each concept in the semantic model and associated to it each series deemed relevant to that concept. A series was, in general, associated with more than one concept, and vice-versa.

This indexing step is crucial to the success of the system; improper indexing will most likely cause relevant series to be missed. The greater the number of concepts a series is associated with, the bigger the probability of it being found during a search.

Our experience with this process has been that indexers find it quite natural to associate series and concepts; for the vast majority of series, the relevant concepts were immediately identified. Although we have not done a systematic study yet, an experiment with a small subset of the series showed that two different experts associated these series with exactly the same concepts in the model.

3 The Implemented System

The functioning of the system can be regarded as being composed of several distinct phases: Browsing, query formulation, query refinement, presentation of results. The user may move from one phase to the other at will.

3.1 Browsing Phase

In the browsing phase, the user is presented with three panels: inspection, query, path. In the inspection panel, the concept currently being examined is displayed, in the form of a frame as discussed previously. Whenever a concept present in this frame (as the value of a slot) has series associated with it (i.e., can be used as an index), it is marked with a '*' character next to it. If the cursor is placed over a concept, and the 'enter' key is pressed, the system 'traverses' the corresponding link in the semantic net, and presents the frame corresponding to that concept.

Alternatively, the user may choose to name a concept directly. If the named concept is known to the system, its corresponding frame is displayed, as above. The system also contains a large dictionary of synonims for the names of concepts. Whenever an ambiguous name is given, the system enters a disambiguating mode, in which the possible interpretations of the name are shown to the user, using the AKO hierarchy and the relations of the concept. The user is then prompted for specifying which of the possible interpretations s/he intends.

During the browsing phase, the user will usually run across one or more concepts of interest. Whenever this happens, s/he may signal to the system that this concept

should be remembered (selected) for later use in retrieving the series of interest. All of the selected concepts are shown in the query panel.

The path panel contains a record of all concepts previously examined. By moving the cursor to this panel, and placing it over any of the concepts it contains, the user may go back to a previous stage of the browsing phase.

3.2 Query Formulation Phase

Once the user is satisfied that all of the relevant concepts have been identified to the system (by "remembering" them), the query formulation phase may be entered by moving the cursor to the query panel and hitting the 'enter' key.

In the query formulation phase, the system examines the concepts the were selected ("remembered"), and forms all possible meaningful queries out of these concepts. In general, more than one query will result, normally due to the fact that concepts that are in different sub-trees of the AKO hierarchy must be in separate queries. The reason for this is that, by construction, there are no series that refer *at the same time* to two concepts that are (directly or indirectly) related via the AKO hierarchy. For example, there are no series that refer to both `pension` and `sick leave`, which are AKO `benefit`.

Another reason for generating multiple queries is due to the distinction drawn between basic concepts and properties. Properties do not have any meaning on their own; rather, they must be associated to some basic concept. Therefore, if a user selects concepts that are possible values of a property, then two queries are generated, each combining the basic concept and a value of the property.

For example, suppose `pension` has the relation `benefits` with `benefitiary`, which in turn has property `sex`, whose values are `male`, `female`, `unknown`. If the user selects `pension`, `male`,and `female`, then the queries `pension`, `male` and `pension`, `female` will be generated.

Again, the reason for this is due to the way the series are constructed. To select more than one value for a property is equivalent to ask for *aggregated* information; but, by construction, there are no series that directly contain this information. There are series containing aggregated information, but they will always be indexed by the basic concept *only*. In the example above, series relating to aggregated information about `pensions` will be associated only to this concept, and not to either `male` or `female`.

Once the user has chosen to proceed with one of the generated queries (s/he may work on the other ones at a later time), the system will ask for further detailment. Since SINTESE contains two dimensional time series, the user may specify a particular time and space detailment (for example, "per month" and "per city").

Another dimension, which has been encoded in the names of series by convention, indicates whether the series contains monetary values or contingents; names of series containing monetary values always begin with the character "$". The user may also indicate her/his preference in this case. In any of the detailments above, the user always has the optional value "don't care".

3.3 Query Refinement Phase

Once the user has finished formulating the query, the system will try to find the relevant series by taking the intersection of the sets of series associated with each of the concepts in the query. In many cases, however, there this intersection may be empty; this means that the level of detail the user has chosen is too great.

Precisely because SINTESE contains statistical data, it may still be meaningful to look at series that contain more aggregated data than was requested. Based on this observation, the system generates possible simplifications to the generated queries using a number of heuristics, and suggests them to the user.

For example, suppose the user has chosen the query `pension, male, age`. Depending on the detailment chosen, this query could be interpreted as *Find all series containing total amount of pensions paid to persons of the male sex, per age bracket, per month, per city*. If there are no series satisfying this query, then the queries `pension, male` and `pension, age` will be suggested.

At this point, the user may accept one of the suggestions, or go back to the previous phase and reformulate the query.

3.4 Presentation of Results Phase

Once the system has found the relevant series, their names are shown to the user. The user may ask for further details about a given series. such as periodicity, time of last update, etc.

Another type of informatin shown to the user at this phase is a summary of series that are relevant to the chosen query, but were not included in the answer due to the detailment chosen. For instance, the user may have chosen a time detailment of "per week", but there are some series that are detailed "per month". In this case, the system will indicate to the user the total number of series that were discarded because of this time detailment. The same will be done for the other detailments.

4 An Example

In this example, the user intends to get information about the amount of money disbursed on benefits granted to rural benefitiaries, detailed by sex, age and week. At first sight, the related concepts would be `benefit, rural benefitiaries, sex` and `age`; "week" should be the time dimension detailment.

The user starts browsing by selecting the concept `benefit` to be visited and chosen as an indexer (see figure 2).

Next, the user browses its neighbourhood and discovers that this concept has a relationship called `benefits` with the concept `benefitiary` . The user decides to further investigate this concept. See figure 3.

The user will find the concept `rural benefitiary, age` and `sex` in the neighbourhood of `benefitiary` . See figure 4.

The concept `rural benefitiary` , `male` and `female` were found and selected as indexers. See figure 5.

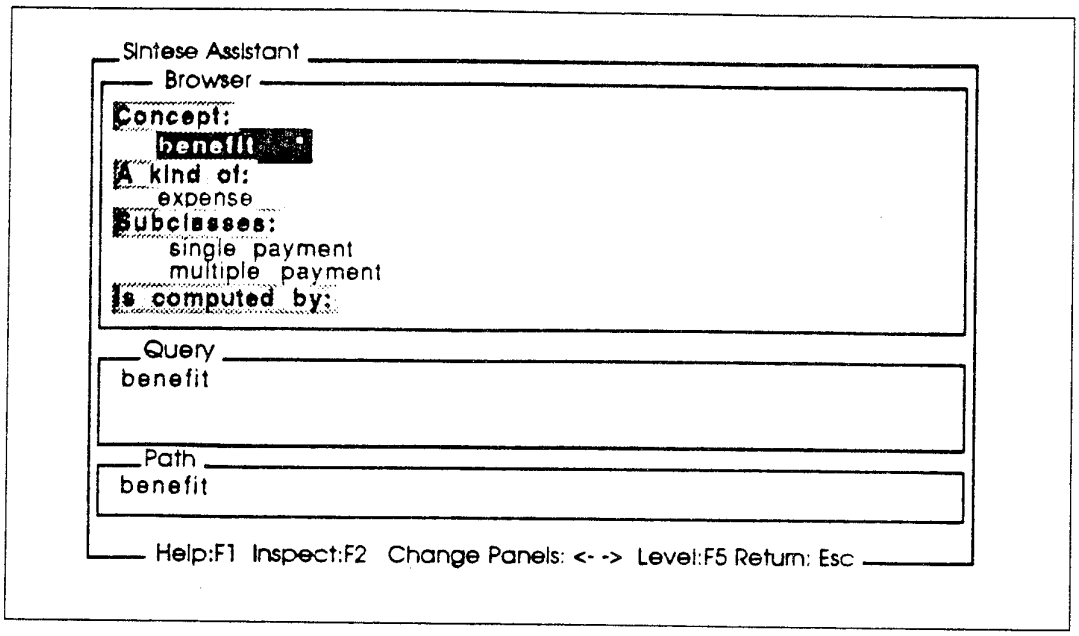

Figure 2: The concept benefit is being visited now. It is th e first one to be browsed, therefore the "path"panel only contains it. This concept has been selected as an indexer, as shown in the "query" panel.

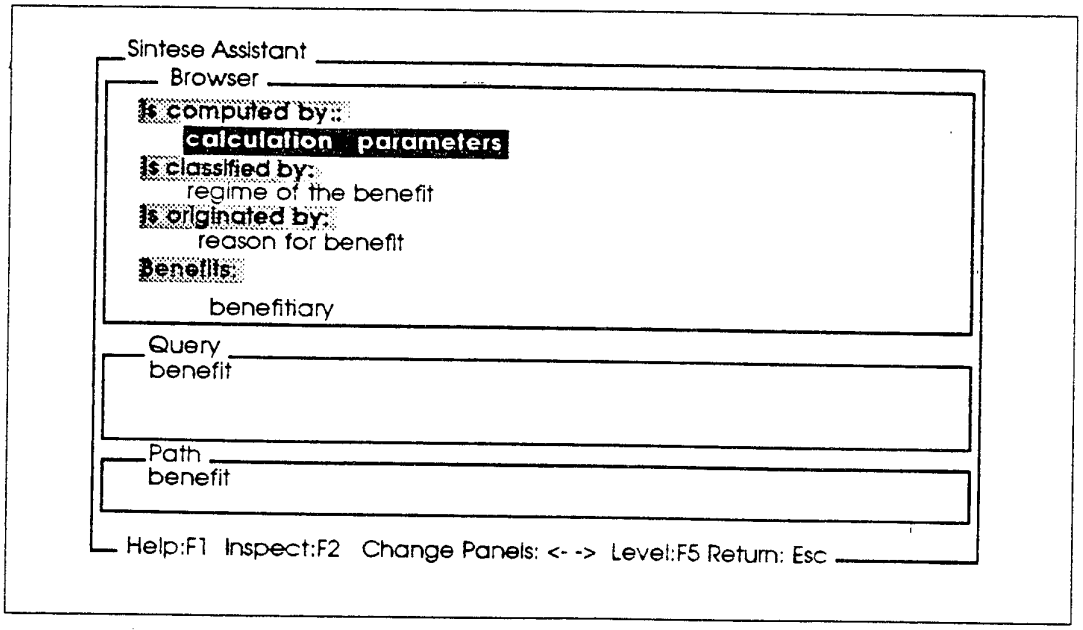

Figure 3: The user finds a relationship which leads her/him to a related concept.

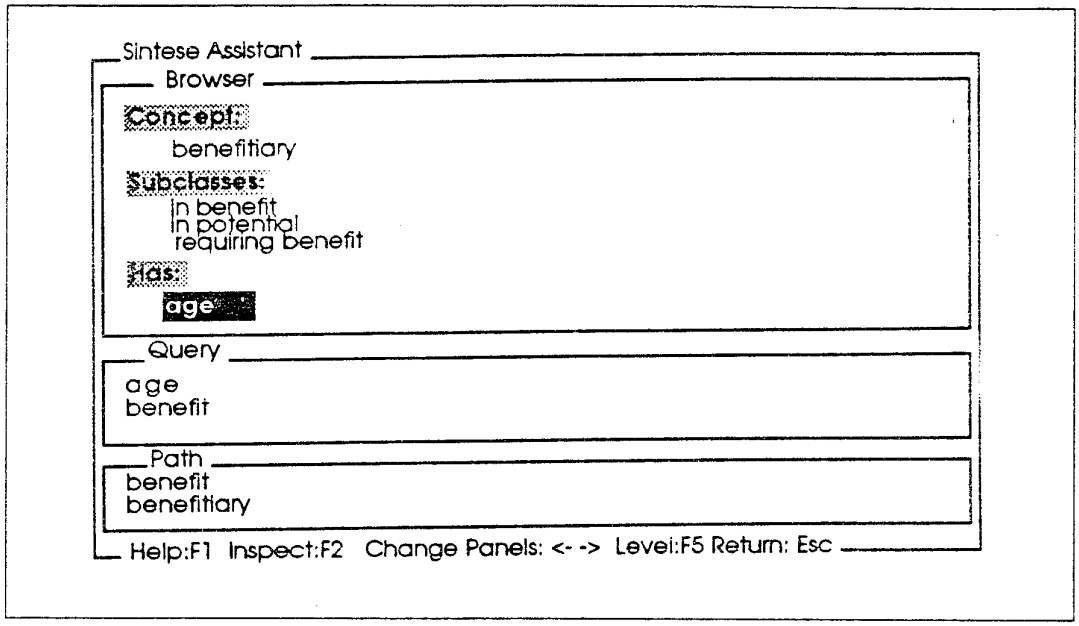

Figure 4: The concept benefitiary is browsed and the concept age is selected as an indexer.

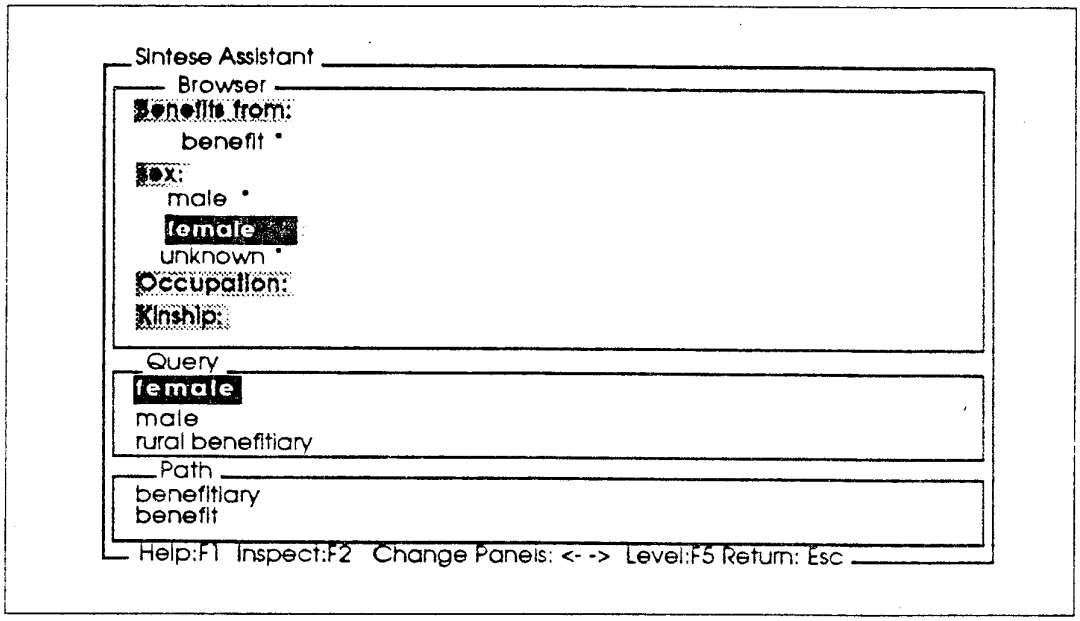

Figure 5: The user has selected the concepts of interest, and is now requesting the system to generated the possible queries.

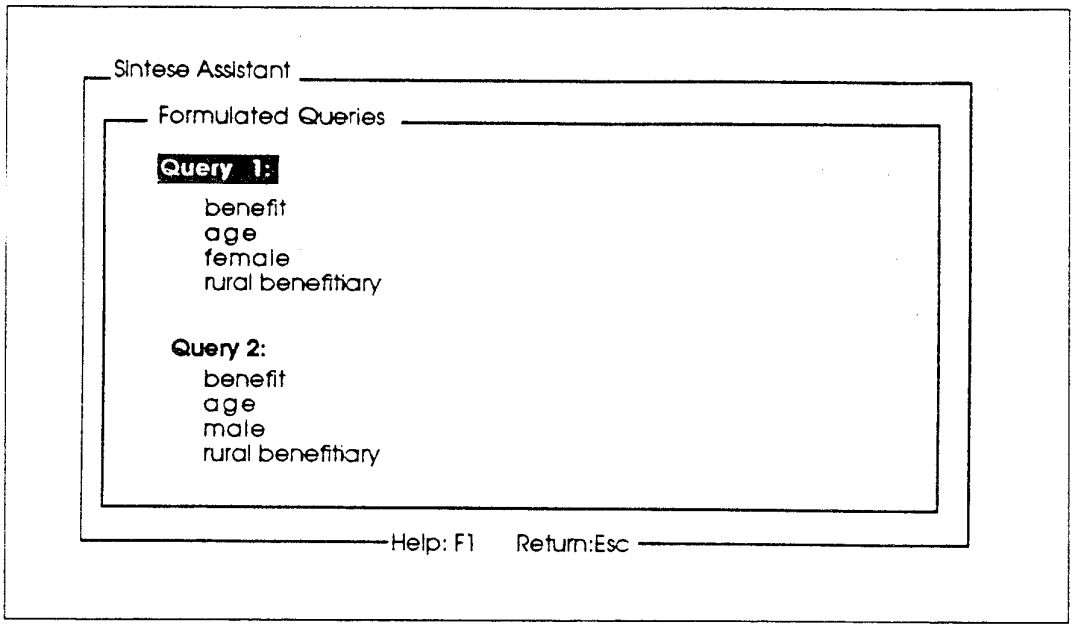

Figure 6: The system presents the generated queries. The user must choose one of them to continue. He/she has chosen query number 1.

Once the user has selected the concepts of interest, the system analyses and generates the meaningful queries. Notice that the concepts `female` and `male` were not put in the same generated query as they belong to the same AKO tree. See figure 6.

Next, the user may restrict the results to the series in which the time detail is `week` and contents are `monetary value`, by pointing out in the refinement panel on the right side. See figure 7.

The system determines that there are no series for this set of indexing concepts, and suggests as an alternative query, formed as a subset of the concepts shown in figure 8, from which the concept `benefit` has been excluded, according to heuristic rules.

The user accepts this alternative query and the system presents the results (see figure 9): there were some series associated with the concepts in the query. However, the series thus retri eved were not presented because their time detailment is not compatible to the chosen one (`week`). This is indicated in the message in the `ALERT` panel.

The user, once aware of this situation, exits the screen back to the query refinement phase, and decides to choose a `month` time detailment, instead of `week`. At this point, the system presents the relevant series. See figure 10.

To obtain more information, the user may request the description of some of the resulting series. The information presented is an excerpt of the main database dictionary. See figure 11.

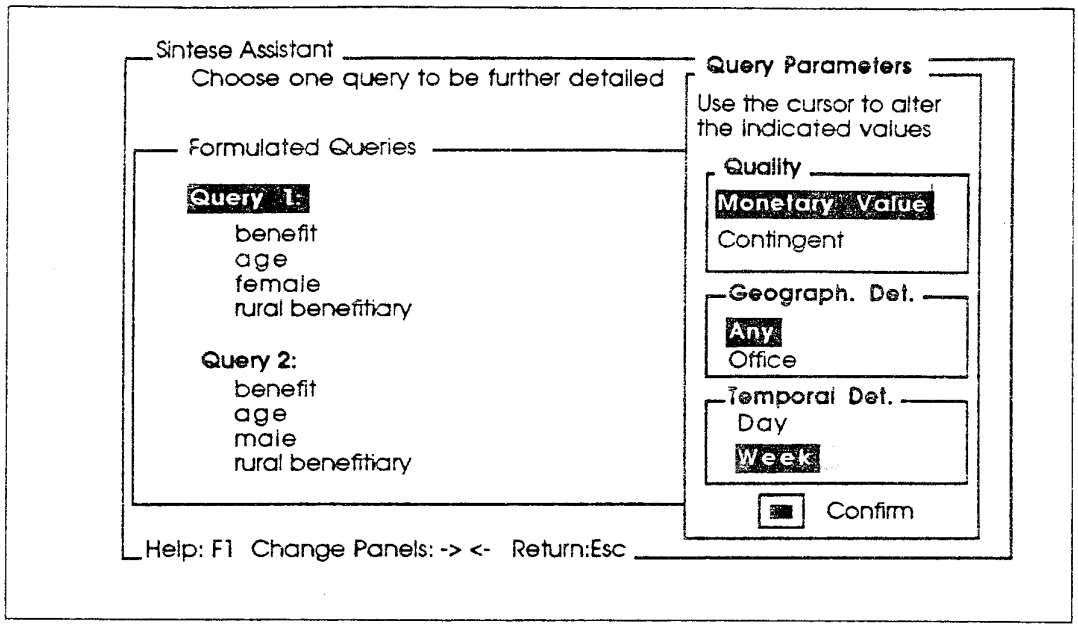

Figure 7: Refine phase – The user specifies the desired series detailments.

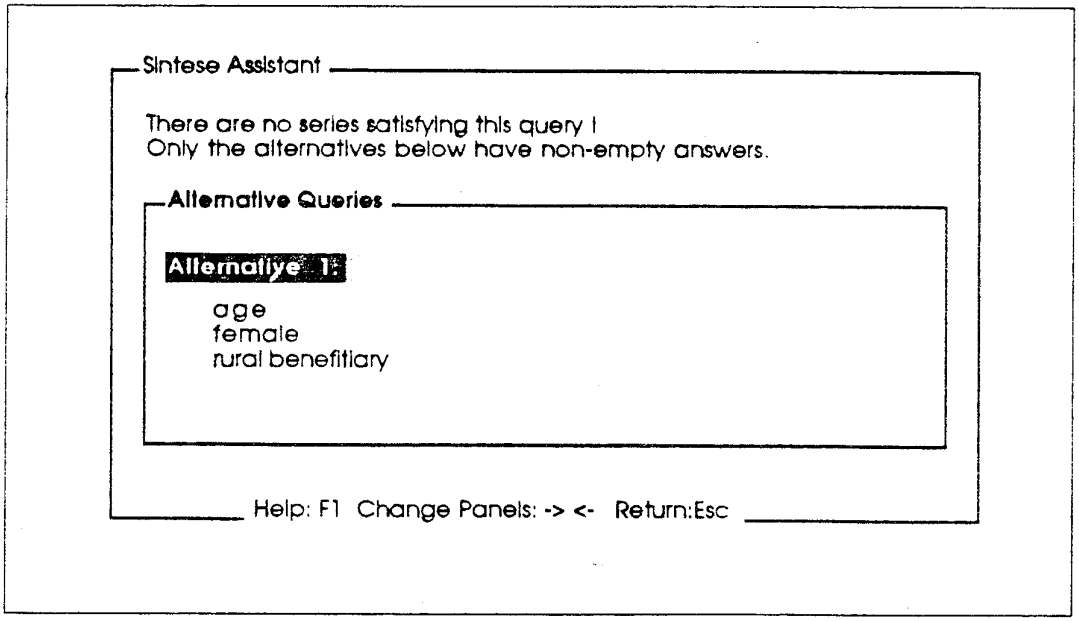

Figure 8: The Query Refinement Phase – possible alternative queries of interest were generated and presented to the user.

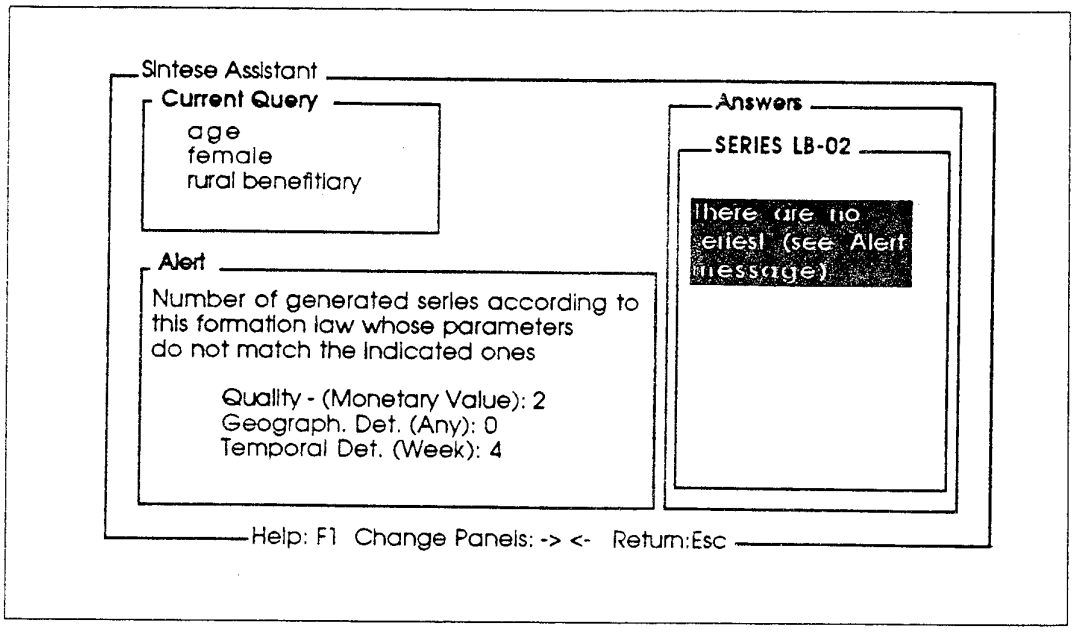

Figure 9: Some series were retrieved, but none presented, due to incompatible time detailment, as indicated in the ALERT panel.

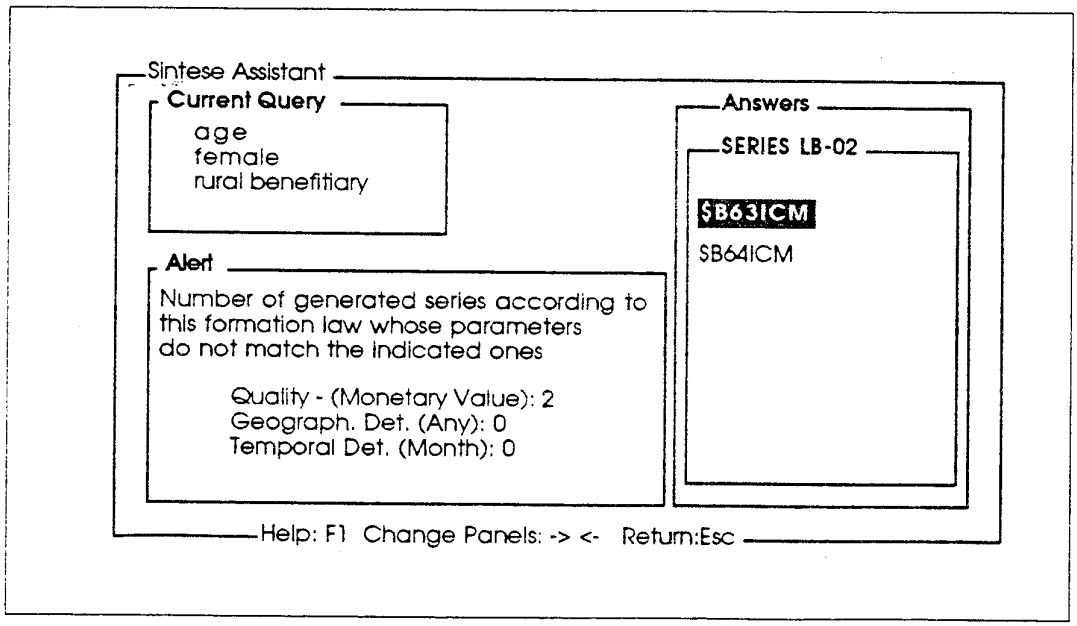

Figure 10: The resulting series.

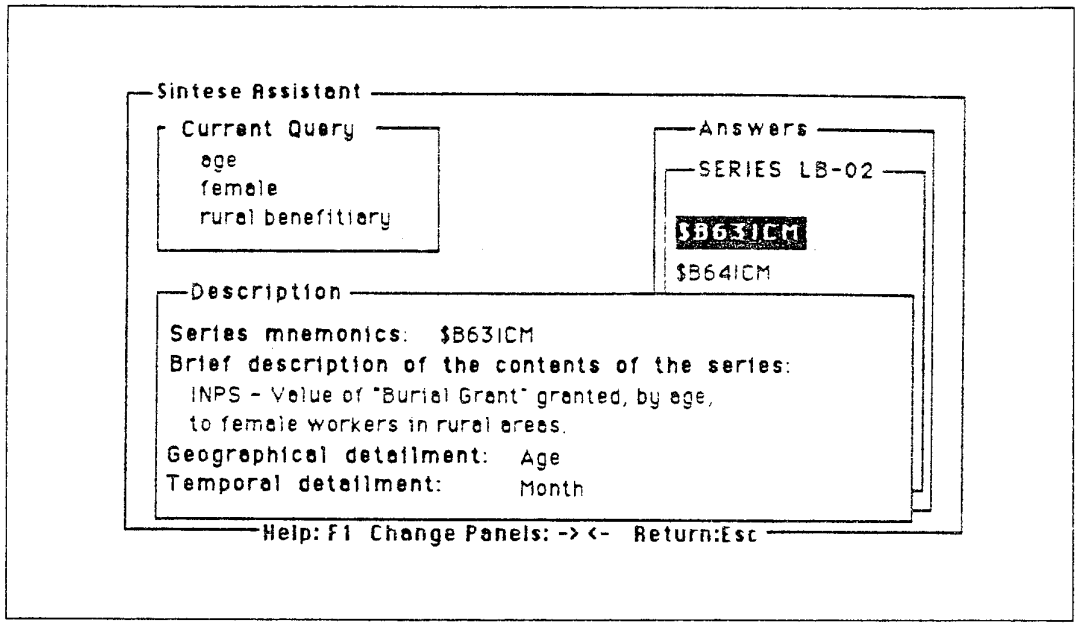

Figure 11: The description of a series.

5 Conclusions

The system has been implemented in Prolog, on an IBM PC compatible machine, and occupies 184k bytes. It has been in experimental use within Dataprev, and will soon be released to the whole ministry.

We are currently designing and implementing the maintenance component, which will include both semantic model updating function and indexing support functions. Another extension being incorporated is an online connection with the database, in such a way that the assistant will be seen as an "intelligent help" function of the dbms.

References

[1] Krawczak, D.A., et alii "EP-X: A Knowledge Based System to Aid in Searches of the Environmental Pollution Literature", in *Proceedings of the Second "Artificial Intelligence Applications - The Engineering of Knowledge Based Systems" Conference*, Miami Beach, Florida, 1985.

[2] Minsky, M. "A Framework for Representing Knowledge", in *The Psychology of Computer Vision*, edited by Patrick H. Winston, McGraw Hill, New York, 1975.

[3] Monarch, I. and J. Carbonell "CoalSORT: A Knowledge-Based Interface", IEEE Computer, Spring 1987.

[4] Tou, F. N., et alii " Rabbit: An Intelligent Database Assistant ", in *Proceedings of the AAAI*, 1982.

[5] Woods, W. A. "What's In a Link: Foundations for Semantic Networks" , in *Representation and Understanding: Studies in Cognitive Science*, edited by D.J. Bobrow and A. Collins, Academic Press, New York, 1975.

[6] Yen, J., R. Neeches and M. De Bellis, "Backbord: Beyond Retrieval by Reformulation", Information Sciences Institute report ISI/RS-88-202, Marina Del Rey, California, 1988.