Building Web-based Knowledge Clusters

Mitchel Berberich and Stefan Bamberger
Department of Computer Science, Wuerzburg University
Am Hubland, D-97074 Wuerzburg, Germany

{mitchel, bambi}@informatik.uni-wuerzburg.de

Abstract: Today the WWW as an easy-to-use environment for all kind of applications is widely accepted. Knowledge based systems profit in many ways from this new technology. Using an internet browser as a generally available low-cost front-end makes it possible to offer the power of an expert system to many users. Depending on the declarative nature of knowledge structures it is very easy to generate dynamic html pages and to build up a knowledge server [1].

In spite of this, it is very difficult for a user to find the appropriate knowledge server for his demands. We discuss an approach how it is possible to integrate an arbitrary number of distributed knowledge servers to a knowledge cluster using only one internet server as a central communication interface to the user. The tool is based on an expert system shell tool kit called Coop-D3 [2] which can be used to build up heterogeneous diagnostic knowledge bases.

1. Introduction

Modern expert systems make high demands on processor and memory and their use is often limited to one specific operating system. The WWW as an easy-to-use environment and the use of an internet browser as a generally available low-cost and platform-independent front-end makes it possible to offer the power of an expert system to many users: The hard- and software requirements on the user’s side are drastically reduced and the high-end server(s) can be used by multiple users simultaneously. The internet also makes possible completely new applications like using expert systems for online support to reduce costs for providing a hotline: Before contacting the hotline the user can try solving his problem with the expert system first so that only the really hard problems have to be solved by the hotline staff.

In this paper we first present an approach for making an expert system available through the internet using a Java-Interface. Then we present an extension to the expert system allowing the use of multiple knowledge bases and finally we show how these two approaches can be merged together to form an even more powerful system.

Here is a list of the components used. They will be explained in detail later.

- D3 [3] is a diagnostic expert system shell kit developed at Wuerzburg University for building and using expert systems for diagnosis.
- D3WWW is a web based interface for D3 which allows many users to simultaneously use the system from their own computers. The only requirement is to have an Web-Browser installed.
- Coop-D3 is an extension to D3 that allows multiple knowledge to be involved in the problem solving process.

2. D3

The Diagnostic Expert System Shell Kit D3 has various components: knowledge acquisition, problem solving, end-user dialogues and extensions. The components themselves consist of modules which can be linked dynamically and be registered to become part of the system. [4]
The knowledge acquisition component consists of several graphical editors (forms, hierarchies, graphs, tables) allowing experts to enter their knowledge without the help of a knowledge engineer. The built-in problem solvers support categorical, heuristic, statistical, case based, set covering, and functional knowledge. The dialogue component contains various dialogues some of them being accessible through the WWW (D3WWW). The extensions consist of information systems, machine learning modules and so on.

3. D3WWW

The HTML-based D3WWW dialog can be used with any Web-Browser supporting HTML version 3.0 or above. Here we present a more advanced version using a Java interface. The server is an IBM-compatible PC with Windows NT and Internet Information Server installed. The Client can be any Java 1.1 compatible Web-Browser like Netscape 4.05 or Microsoft Internet Explorer 4.0.

Fig. 1: D3WWW user interface: The user can enter free text which is automatically mapped to internal objects (1). Values can be entered in questionnaires and popup-menus with additional information are available to all objects (2). Diagnoses are presented as soon as they are suspected or established (3). A kind of “Card Layout” is used to switch between questionnaires, explanations, and file operations / preferences to keep the interface simple (4). A hierarchical view of all questionnaires allows the user to override the system’s proposals and directly choose a questionnaire (5).

The Java-based D3WWW user interface was kept simple so that it can even be used by less experienced users. Only one window is used and only the most vital information is presented in the forms, the rest like (multimedia) explanations can easily be accessed via popup-menus. Objects and user input are buffered in the applet to reduce network traffic.

D3WWW in its current form is being used in Copernikus - an expert system on statistical quality control, a research project of the WRQC [5]. It is also being used by GTI mbH, a software tool vendor for building graphical interfaces located in Marktheidenfeld, (Germany) for a hotline support system. Several large medical applications are on the way.
The “Free Input”-field (to be completed soon) will be a big improvement to the user interface, because it offers additional flexibility: An user (e.g. a medical doctor) has three options for entering data:

- Using the questionnaires and entering the facts in the order proposed by D3.
- Using the hierarchical view of all questionnaires and knowing where to find the questions he can select the appropriate form and fill in the data whenever he wants to, but he still has to repeat this process for every question.
- Using the “Free Input”-field the doctor could just enter something like “Age 18; Size 1,70m” and the System will try to map the data to the questions\(^1\). This mapping will then be shown to the user and can be confirmed by just pressing a key.

### 4. Coop-D3

Creating large monolithic knowledge bases with several thousand rules is a difficult task. Coop-D3 makes it possible to split up problems in smaller sub-problems which can be handled by smaller and easy-to-create knowledge bases. When a knowledge base decides, that another knowledge base might be more competent, it can permanently transfer the control over to the other knowledge base together with the data and intermediate results inferred so far. Control can also be given over temporarily for clarification. The objects that can be shared are determined by a terminology knowledge base. The mapping from internal objects to shared objects is done by the mapping knowledge bases. The glossary can be used to unify the shared knowledge, so that the mapping can be kept simple. This process is totally transparent to the user.

![Co-operating knowledge bases in Coop-D3](image)

**Fig. 2** Co-operating knowledge bases in Coop-D3: At any time the active knowledge base can give over control to another more competent one.

Coop-D3 is currently used in a project with one of the world leading printing machine manufacturers Koenig&Bauer-Albert AG in Wuerzburg to support service technicians in the process of finding and remedying faults within reel-rotation printing machines. One way of dividing the problem of finding faults in the printing machine can be to develop separate knowledge bases for functional units like reversing gear, folding machine and so on. These knowledge bases again can be divided in smaller units handling the different kinds of folding machines available.

### 5. Merging the systems

By using a web-based interface and co-operating knowledge bases like those in Coop-D3 it should be possible to build web-based knowledge clusters with a central communication server and several

---

1 Currently we assume, that the user sticks to a predefined syntax and vocabulary, so we can avoid the problems of natural language understanding.
knowledge bases running on different servers. The user only has to contact the communication server and is automatically transferred to the server which seems to be the most competent one for his problem according to the facts entered so far. This server’s knowledge base can again be divided into smaller knowledge bases like in Coop-D3 or it can be another communication server just forwarding the requests to one of the other knowledge servers connected to it.

![Diagram of Web-Based Knowledge Clusters](image)

Fig. 3: Web-Based Knowledge Clusters: Users are connected to a communication server. The control is given to the knowledge server suited best for their problem.

Again free input developed for D3WWW could be a significant improvement to Coop-D3 as it might help to identify the right knowledge base more quickly: In its current form the input-process is strictly sequential. The order in which the questionnaires are presented is determined by the most competent knowledge base, which again is determined by the facts entered so far. It might be possible that the user has to answer a lot of ‘useless’ facts before the right knowledge base gets activated. Using free text input all known facts can be entered at the beginning and may lead to the right knowledge base, being classified as the most competent one much earlier.

6. Literature


