Knowledge Engineering as a Problem Solving Process

Christopher R. Jardine
Department of Computer Science
George Mason University
Fairfax, VA, 22030

Larry W. Brooks
Training Research Laboratory
Army Research Institute
Alexandria, VA, 22333

Abstract
A prototype software tool for collecting, organizing, and analyzing complex problem solving is presented. The design of the tool is grounded in cognitive science findings concerning the processing styles that experts use to solve problems in domains that are knowledge rich and ill-defined. Four salient characteristics of expert problem solving were selected as guides for building features into the software tool: opportunistic search, multiple views of the same information, chunking of information, and schema recognition. The prototype currently supports the first three of these characteristics in some form. The fourth characteristic, schema recognition, is currently being investigated for inclusion into the tool.

1. Introduction
In this paper we describe a new software tool called KEEN (Knowledge Engineer's Electronic Notebook), that we are developing for the purpose of aiding knowledge engineers (KEs) and subject matter experts (SMEs) in the collection, understanding and organization of knowledge related to complex problem solving, particularly in the area of design. Design problems, such as mechanical, instructional, and computer network design, are knowledge intensive in nature, and the requirements of these tasks frequently overwhelm the resources of the problem solver. These conditions indicate that computer job aids can potentially facilitate the problem solving process by helping manage the cognitive requirements of the KEs or SMEs.

2. KEEN and Problem Solving
In this section, we will briefly discuss the role of knowledge organization in complex problem solving such as design, and how KEEN fits within the problem solving process.

Problem Solving. The problem solving process can be divided into two major phases: understanding and search. In the understanding phase, the problem solver attempts to gain the information necessary to solve a problem, and to organize that information so that the solution of the problem will be as simple as possible. In the search phase, the problem solver searches the problem space until either a solution is found or generated, or until it is determined that no solution is available.

A salient element of design problems is that the majority of effort spent in solving the problem is in the understanding phase because this type of problem is ill-defined. That is, there is frequently not enough known about the solution to a design problem to determine in advance what will constitute a solution. It is therefore critical that the knowledge one has of the problem be organized so that the best problem space for generating a solution can be constructed.

KEEN is intended to be a foundation for a set of tools that will facilitate the information or knowledge organization sub-process in the first phase of the problem solving task. It is built with the assumption that it will be expanded later to incorporate elements that are domain specific. For instance, a knowledge organization tool could be built on top of KEEN that would have a representation for specific rules, frames, etc. that are not a part of KEEN itself.

Characteristics of Expert Problem Solvers. KEEN is designed to provide some of the types of representations that experts find useful in solving complex problems. We briefly describe the representations that the current version of KEEN supports (see Figure 1).

A major characteristic of expert designers is that they are opportunistic in their approach to problem solving [8]. In generating a solution to a design problem, experts tend to use both top-down and bottom-up strategies to derive a solution. In response to this characteristic, KEEN is designed to allow for the specification and organization of information in both a bottom-up and a top-down fashion.
Another characteristic of experts is that they tend to work with chunks of information [4]. Knowledge organized as chunks is easier than non-chunked knowledge to manipulate at a global level without losing reference to detailed information. KEEN is built so that users can conveniently chunk or abstract information, thus providing designers with the ability to organize knowledge at a global level and maintain easy access to local level information.

Experts also view the same information from multiple viewpoints [9], [10]. By changing viewpoints when solving a problem, the problem solver can see the current problem from the perspective of another domain and use the information in that domain to obtain a solution.

Alternately, the problem solver may notice new relationships among items in the problem domain when a different perspective is used to view the relevant knowledge. Consequently, KEEN is designed to have the capability to show the same information from multiple perspectives by selectively showing or not showing different associations among items stored in a KEEN knowledge base.

Expert problem solvers often use schemas, cases, or analogies in solving problems [6]. We are currently conducting research on the use of these mechanisms to create an adaptive tutor for teaching novices problem solving skills. The results of this research will be used to design and implement a mechanism that would allow KEEN to determine if a user is using a case or schema.
that is already known. Once a schema or case recognizer is incorporated into KEEN, then the system could help the user structure the new information to fit the already-known structure. It would also be valuable to have KEEN learn about new schemas from its interactions with the user. The next version of KEEN will include a prototype schema recognizer. Including a schema learning component in KEEN is a long term goal.

3. Description of KEEN

The KEEN system provides a graphical workspace where the user can create nodes of information/knowledge, link the nodes together in multiple ways, and visually organize the nodes in a manner that best suits the purposes of the user (c.f., [15]). KEEN currently supports about fifty functions related to the creation and manipulation of objects and their relations. These functions are invoked by special mouse-clicking sequences, or by selecting from a menu, an example of which is shown in Figure 2.

**Creating New Nodes.** A new node is created by double clicking on the graphic window. A pop-up dialog window appears immediately after the double click. This window asks the user to provide a name, type, and description for the node. The new node is then placed on the window at a location where the mouse cursor was at the time of the double click. Internally, nodes consist of a data structure that has a number of slots or attributes. Some of these attributes are visible to the user (e.g., name, type), and some are hidden from the user but used by the system to manipulate the node (e.g., parent, child).

**Relating Objects and Information.** Nodes can be linked together by double clicking inside an existing node and then, continuing to hold down on the mouse button after the second click, moving the mouse button to another node. Next, a pop-up window appears on the screen and requests that the user enter information concerning the type of link to be created, as well as a description of the link. New link types are added to the links menu in the top menu bar.

![Diagram of KEEN](image)

**Figure 3.** Different viewpoints of a television's picture circuitry

**Figure 4.** Block Diagram of Picture Signal Circuits

We will describe only a few of these functions in the context of an example which demonstrates how information might be organized for use by an expert system which diagnoses faults in televisions. We begin by discussing how objects (nodes) and relations (links) are created, and then proceed to discuss how each of the problem-solving strategies under consideration is realized in KEEN. The object nodes and relations in Figures 3 and 4 were created in the manner described below.
Representing Multiple Viewpoints in KEEN. By using different link types and node types, the user can view the same information from multiple viewpoints.

For example, Figure 3a is a structural depiction of a subsection of the picture component of a television receiver. Figure 3b is a functional depiction of the same component. In this example, the user can use different link types to depict the different relations between the objects, and in so doing describe and think about the system from both viewpoints.

Top-down and Bottom-up Structuring. The 'zoom in' and 'zoom out' features of KEEN allows the problem solver to apply a top-down or bottom-up approach, respectively, to the problem in an opportunistic fashion. The user can jump back and forth between these approaches as deemed necessary. The hierarchical structuring of the domain is maintained through the use of the parent and child slots for nodes, which are updated automatically by the system. Figure 4 shows how the visual signal circuits (from Figure 3) may be decomposed into subcomponents. Notice that the "Picture Signal Circuits" node in Figure 3 is highlighted. This indicates to the user that the node has a more detailed description at a lower level in the hierarchy.

Chunking. Abstraction/chunking of information is accomodated by the 'zoom in' feature of KEEN. This allows the user to hide information of varying complexity "beneath" any given node. The more detailed depiction of the picture signal circuits is subordinate to the "Picture Signal Circuits" node depicted in Figure 3.

4. Building a Rule Base Using KEEN

Once the KE has organized the knowledge using KEEN, he or she can then build the appropriate reasoning or inferencing structures based on this organization. For instance, a simple rule base for trouble-shooting the TV circuit shown in Figure 3 could be built to take advantage of the structural and functional viewpoints. The rules could refer explicitly to the different viewpoints, thus allowing for a more flexible inference process that should be able to solve problems that a rule base representing only one point of view could not or would do so in an ineffective manner. For example, a functional rule for diagnosing a problem associated with color appearing on the picture tube could be:

IF no_color THEN test(color_circuit).

Structural rules for the same problem could be:

IF no_color THEN test(picture_signal_circuit);
IF AND(no_color, ok(picture_signal_circuit)) THEN test(color_circuit).

5. Conclusions and Future Directions

Keen represents an attempt to construct a knowledge acquisition software tool which derives from cognitive science findings on expert problem solving strategies. In particular, KEEN proceeds from the premise that experts employ multiple viewpoints in problem solving [13]. Traditional expert systems view problems in one way. KEEN is a first step in moving toward expert systems that use multiple perspectives to solve problems. We believe that this approach will lead to more flexible and efficient knowledge-based systems.

References


