Crafting
ckboard
entations

Blackboard systems are so named because they organize and process knowledge in a fashion analogous to a group of people working around a blackboard. The *blackboard* is used as a repository for the knowledge being assembled by the group. Each person represents a *knowledge source*, a specialized source of knowledge about some aspect of the problem. A leader provides a *control* function, guiding or focusing the activities of the knowledge sources as well as sequencing their access to the blackboard.

Knowledge can be represented on the blackboard and in the knowledge sources in various ways. A blackboard system is not so much a particular form of knowledge representation as it is a way of organizing and processing knowledge represented in other forms. Thus, a blackboard approach can be thought of as a problem solution process.

The Blackboard Concept

The blackboard concept was the basis for the HEARSAY II speech recognition system developed by Reddy and his colleagues(1,2) at Carnegie-Mellon University, which in turn has served as a guide for a number of other blackboard-based applications. The first system for building blackboard systems (AGE) was constructed by Nii and Aiello(3) at Stanford University. A very good description of the blackboard methodology, the history of its development, and the use of this technology in several applications is contained in a two-part article by Nii(4,5) in *The AI Magazine*.

A blackboard system can be thought of as a framework in which knowledge can be arranged so that it can be distributed and yet shared among a number of cooperating processes. That is, the knowledge about a problem can be distributed to a set of specialists called *knowledge sources*, each of which has a particular area of expertise. Part of the knowledge is encoded on the blackboard, which is the shared portion of the knowledge base through which the specialists communicate. The remainder of the knowledge resides with the individual specialists who operate independently of each other (except for the implicit communication that takes place through the knowledge placed on the blackboard).

Most of these specialists can be viewed as domain experts who deal with the subject matter of the application. Each works to further the state of collective knowledge as represented on the blackboard, thereby contributing to the problem's solution. However, in addition to the domain specialists, at least one specialist must have expertise relating to the solution process. This specialist (or cooperating set of specialists) performs a *control* function, guiding the activities of the other specialists and sequencing their access to the blackboard to make modifications. Thus, an application using a blackboard structure can be viewed schematically as shown in Figure 17-1.

Like the multiple-context form of representation described in Chapter 15, the blackboard representation is not a complete form of representation; it is only a structure within which knowledge can be placed using

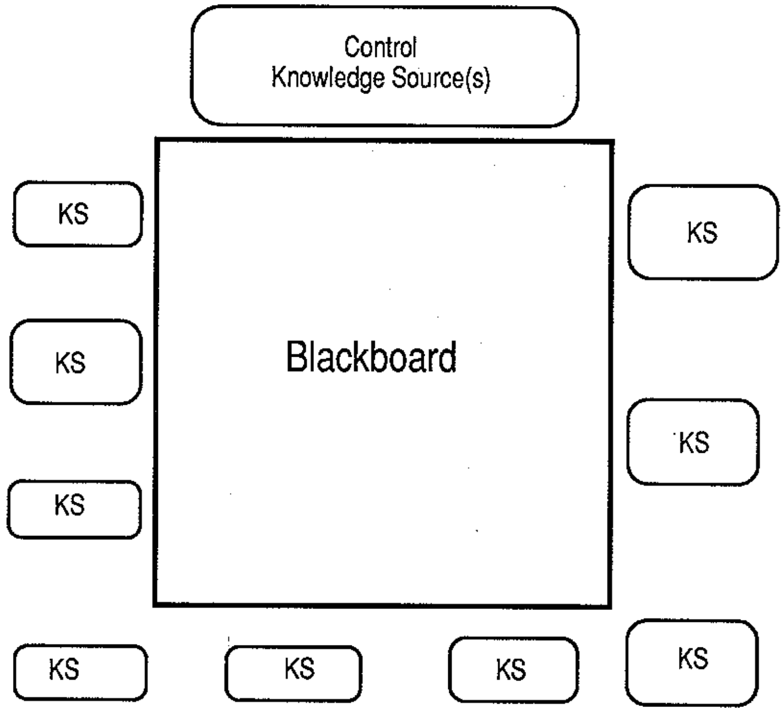

Figure 17-1 Schematic of a Blackboard System

other forms of representation. Because the structure of the knowledge within the framework is not part of the blackboard specification, a variety of knowledge representations are compatible with, and thus can be used with, a blackboard structure.

The blackboard thus serves a dual role in a knowledge-based application. Like other forms of knowledge representation, the blackboard structure governs access to and modification of the knowledge represented on the blackboard. However, the blackboard also serves as a solution methodology, setting forth an approach to developing a solution to the problem at hand. Like the structure aspect of the blackboard, the methodology aspect is only partially specified. Thus, a blackboard structure provides room for many different solution approaches. The blackboard approach can be viewed more as a philosophy or a set of guidelines than as a carefully specified process.

Components

As shown in Figure 17-1, a blackboard knowledge representation is constructed from three major components:

1. Knowledge sources (expertise)
2. Blackboard (knowledge storage and communication)
3. Control (problem-solving strategy).

Because the three types of components play different roles, each offers different facilities to the knowledge crafter.

Knowledge Sources

The knowledge sources represent expertise that is captured in the overall knowledge-based system. Each knowledge source represents some particular specialized knowledge pertaining to the problem being solved. This knowledge may be represented in different ways, for example, as a collection of logical relationships or rules, or as a subroutine or function programmed in a conventional procedural programming language. Whatever form of representation is used within a knowledge source, however, the knowledge reflects an action (i.e., a change to the blackboard) under appropriate circumstances.

A knowledge source is itself made up of a number of subcomponents. For example, a knowledge source in BB1(6), the blackboard system developed at the Stanford Knowledge Systems Laboratory, has 16 different subcomponents. Conceptually, however, two major subcomponents need to be recognized—the knowledge to be applied and the knowledge about when that knowledge should be applied. These are referred to as the *action* part and the *condition* part of the knowledge source. Like a specialist, the knowledge source consists of some expertise as well as a knowledge of when that expertise is applicable.

Knowledge Sources: Condition

The condition part of the knowledge source specifies when the knowledge source would have something to contribute. That is, when the specified conditions are satisfied, the knowledge source can appropriately be fired or executed. Until those conditions are satisfied, the knowledge source can be ignored.

Consider a group of experts gathered together to determine whether a newly discovered manuscript was indeed written by Shakespeare or whether it is a forgery. As long as the conversation among the experts focuses on the text of the manuscript (e.g., word frequencies, grammatical constructions), the chemist can sit back and relax. When the subject turns to the composition of the ink on the paper and the age of the paper, however, the chemist must be alert and contribute to the discussion.

Knowledge Sources: Action

The other major part of the knowledge source specifies the actions that are to be taken based on the data (situation) presented on the blackboard. Such actions can involve the placement of new facts on the blackboard, the modification of data on the blackboard, or even the deletion of data from the blackboard (the housecleaning function).

Note that conditions can also be represented within the action part of the knowledge source. The conditions specified in the condition part of the knowledge source specify the conditions under which the expertise or knowledge contained in the knowledge source is relevant. The conditions in the action part of the knowledge source specify which pieces of knowledge are applicable to the particular situation being considered.

In terms of the previous example, the chemist's knowledge becomes applicable when the conversation turns to the materials used to record the text. If the focus of the conversation at a particular moment is on the spectrographic analysis of the ink, however, then the chemist will apply only that part of the specialized chemical knowledge that relates to ink composition, temporarily ignoring the expertise related to the chemical composition of the paper.

The Blackboard

The operation of a blackboard system can be likened to a collection of specialists that have gathered together to tackle a problem. However, if the specialists are allowed to talk directly to each other, a number of side conversations will develop. As a result, some of the specialists would not know some of the information or conclusions being developed or contributed by the other specialists. To prevent this from happening, everyone must be included in all conversations. Each expert must work alone and then share all conclusions with the others by writing them on the blackboard for all to see. These characteristics of independent operation and globally shared information are fundamental to the blackboard approach to problem solving.

The blackboard thus represents the communication medium through which the knowledge sources (e.g., specialists) communicate their findings to each other. The blackboard is the:

- Source of all data on which a knowledge source operates
- Destination for all conclusions from a knowledge source.

The Blackboard: Types of Knowledge

The blackboard contains two types of knowledge: static and dynamic. The static knowledge is typically the domain-specific knowledge that is relevant to the problem and that will have a relatively long life during the solution process. Static knowledge generally consists of factual data relating to initial conditions, parameter values, relationships, and the like. In terms of the Shakespearean manuscript, the text of the manuscript, the size of the paper, and a count of word frequencies would be represented on the blackboard as static knowledge.

Dynamic knowledge is typically the knowledge that is generated during the execution of the application system. It will consist not only of new facts but also of short-term communications such as hypotheses, goals to be pursued, requests for data (from other sources), and suggestions. The dynamic data will frequently be modified or deleted after a short period of time.

The Blackboard: Multiple Blackboards

The discussion thus far has referred to "the blackboard" as if there were only a single blackboard. However, since knowledge crafters often want to characterize knowledge in structured relationships, the blackboard can be similarly "divided." Whether a knowledge crafter chooses to view one blackboard containing a set of subareas or to view several individual blackboards is irrelevant. The concept is the same.

For the Shakespearean manuscript problem, a blackboard might be placed on one wall of the room for communication and data regarding the text of the manuscript, while a blackboard on another wall might be reserved for data on paper and ink.

The blackboard form of representation is often used in problems where the knowledge has a hierarchical structure. In such a case, the blackboard can be divided into subblackboards, with each representing a horizontal slice through the hierarchy. Raw data might be placed at the lowest level of the blackboard. Knowledge sources acting at this level would produce intermediate results that are placed at a higher level on the blackboard. Other knowledge sources, operating on the partially processed data, would leave their results at the same or a higher level, and so forth.

The Blackboard: Housekeeping

In analyzing one particular problem, given one set of facts, the knowledge crafter may reasonably permit the blackboard to contain a complete record of the data and communications related to the solution of the problem. That is, the blackboard is not erased. However, for real-time systems that involve continuing analysis of a problem, such as the monitoring of input signals about a refinery production process over a period of hours or days, the blackboard can become quite cluttered. Even if the blackboard is "very large," so that it is not "filled up," it can become filled to the point where analysis becomes quite difficult, where it takes too long for a knowledge source to find a relevant piece of data on the blackboard.

Thus, a specialized knowledge source responsible for "cleaning" the blackboard is needed. In some cases sufficiently "old" information is merely deleted. In other cases, selected information must be copied down (archived in a history file) before being erased from the blackboard. The particular mechanism used and the conditions under which information will be deleted are a function of the particular problem being addressed by the application. Nevertheless, the knowledge crafter should consider the blackboard clean-up problem during the design stage of the application.

Control

A group of experts left in a room without direction will likely wander around somewhat aimlessly in their search for a solution to the problem. The solution process needs to be guided or controlled. Consider the approach to the Shakespearean manuscript problem if a forgery were suspected. Initially word frequencies might be examined in an attempt to quickly disqualify the manuscript. If this fails, it might be much more productive to shift the focus of the inquiry to the age of the paper than proceed with a long series of quibbles about sentence construction. Such a shift in focus is part of the control function.

Control: Provision

Knowledge sources are used to provide the required control functions within the context of the blackboard system. Structurally, these knowledge sources are the same as the domain-related knowledge sources discussed previously, but they focus on the control of the problem-solving process rather than on the domain knowledge.

The control knowledge sources provide control functions in two ways—by placing information on the blackboard that will influence the knowledge sources and by selecting the knowledge sources to be fired (executed) from among those that qualify. Control knowledge sources can place a variety of information on the blackboard. For example, information about goals might be disseminated in this way, guiding the individual knowledge sources on the conclusions they should work toward or on the direction they should pursue. In the Shakespearean manuscript problem, the task of disproving authenticity (as opposed to proving authenticity) would represent such a goal.

Similarly, state information can be used to influence the knowledge sources, since certain knowledge sources might be applicable only in certain states. In the Shakespearean manuscript problem, changing the state from "words" to "ink" would suffice to activate the chemist knowledge source and to inactivate some of the grammar-related knowledge sources.

Control: Strategy

The mechanics of the control function are fairly straightforward, as described in the previous subsection. The real issue concerns the approach that the knowledge crafter should embed in the control function. This approach will be a function of the application and the solution strategy used by the expert(s). However, the following examples of control strategies should illustrate the possibilities:

- **Event-driven**—The control function reacts to the occurrence of particular events. Thus, for example, a monitoring application might react to the event "a new data item has arrived" by selecting those knowledge sources that can process newly arrived data. Similarly, it might react to the event "alarm condition" by selecting those knowledge sources that can process alarms.
- **Expectation-driven**—The control function operates from a model of the solution process. Based on the available knowledge, it can develop expectations about the character of the solution or at least about the next appropriate steps to be taken. The control function can select as firing candidates those knowledge sources that might corroborate the expected information. Thus, for example, in a diagnostic system the failure pattern may be such that an electrical failure is more likely than a hydraulic failure. The control function would then select those knowledge sources that might either confirm or refute this expectation.
- **Request-driven**—The control function selects knowledge sources that are likely to provide information that has been re-

quested by other knowledge sources. Thus, in the case of the Shakespearean manuscript, were the chemist to request information about the spectral analysis of the ink, the moderator of the discussion would turn the floor over to those individuals who could comment on the findings of that analysis.

- **Goal-directed**—The control function selects knowledge sources likely to contribute to a particular goal it is working toward. Goals may be established to attempt to prove a hypothesis, triggering backward reasoning. They may be established to try to generate hypotheses, triggering forward reasoning. In the case of the Shakespearean manuscript, the initial goal was to attempt to prove a forgery. However, if little progress is made toward reaching this goal, the control function might switch to another goal, such as trying to prove the manuscript authentic.

Strategies may focus on objects on the blackboard or on knowledge sources (or on a combination of these). For example, in trying to identify the cause of equipment failure, the control function could focus on an object, the electrical system. In this case, any knowledge source dealing with the electrical system would be favored. Alternatively, the control function could focus on a knowledge source, the circuit resistance tester. This knowledge source would then be applied to all objects on the blackboard having the property of electrical circuits.

Control: Other Dimensions

Several other dimensions to the control function may be relevant to a particular application. In a truly parallel system, for example, all of the eligible knowledge sources could proceed to act simultaneously. However, most systems are actually executed on serial processors. Thus, control is not just choosing those knowledge sources to execute that will tend to make the best contributions to the solution of the problem, but it is also picking the single knowledge source to execute next. Consequently, some type of tie-breaker, such as the following, may be necessary:

- Priority of knowledge source
- Alphabetical order of knowledge source name
- Number of other knowledge sources whose execution might be enabled as a result of a knowledge source's contribution
- Likelihood that a knowledge source would contribute to removing a key bottleneck.

Following the execution of the selected knowledge source, the control knowledge source must perform two functions:

- Requalifying all previously ready-to-execute knowledge sources to make sure that the actions of the one knowledge source that was executed did not render any of these no longer executable
- Identifying all other knowledge sources that might now have become executable as a result of the actions taken by the knowledge source that was executed.

Note that a given knowledge source may appear on the executable list more than once. That is, for each object or set of conditions appearing on the blackboard for which a knowledge source is applicable, that knowledge source can be placed on the executable list with an indication to apply itself to the particular target object or situation. The variables in a knowledge source are bound to particular values for each situation in which the knowledge source was instantiated.

An example may clarify this point. Consider an older brother, George, whose expertise is putting on his younger brothers' and sister's boots on snowy winter days. The condition for activating George is the existence of the proper state (snowy winter day) and the proper object (child wanting to go outside). Thus, if Mary, Peter, and Jeff want to go outside on a snowy day, George would be placed on the executable list three times—once for Mary, once for Peter, and once for Jeff. A particular instance of George would then be selected for application from among the candidates. If the selection were performed with the criterion of females first, then the instance of George bound to Mary would be activated. This particular instantiation of George would check Mary's feet for the presence of boots (i.e., to see if his expertise could be applied) and, if lacking, would proceed to assist Mary in putting her boots on.

Another specialized control requirement arises in real-time environments. In this situation the knowledge source selection problem is a bit more complex. Rather than just sequencing the candidate knowledge sources, the system may need to determine which instantiated knowledge sources shall be executed at all. If processing is beginning to fall behind real time (i.e., more computation is to be performed than there is time to perform it), then some candidate knowledge sources will have to be "ignored," or dropped from consideration, because of the lack of processing power. Thus, part of the control function is to select the knowledge sources to be applied so that the more critical ones are processed in real time and the less critical ones are processed only as time is available. This process is not straightforward, however, as the criticality of a knowledge source may change as a function of how much time has passed since it was last executed.

The discussion thus far has been directed toward the ideal case, with each knowledge source operating independently of all the rest, communicating only via the blackboard. Practical applications often depart from the ideal, however. For example, the knowledge crafter may not be able to eliminate all dependence between knowledge sources. The control knowledge sources will therefore need to provide some type of sequential control in order to prevent sequence-dependent errors from arising.

An Analogy

To illustrate the blackboard process in terms that may be more familiar, consider a noncomputer-based problem solved using a blackboard methodology. One we particularly like was described by Barbara Hayes-