Report 83-30
Stanford -- KSL

Scientific DataLink

The Blackboard Architecture: A General Framework for Problem Solving?
Barbara Hayes-Roth,
May 1983

card 1 of 1

Heuristic Programming Project
Report No. HPP-83-30

May 1983

The Blackboard Architecture:

A General Framework for Problem Solving?

Barbara Hayes-Roth

Heuristic Programming Project
Computer Science Department
Stanford University

HPP-83-30

The Blackboard Architecture:

A General Framework for Problem Solving?

Barbara Hayes-Roth[1]

Heuristic Programming Project

Computer Science Department

Stanford University

May 20, 1983

---

[1] This research was supported by Defense Advanced Research Projects Agency under DARPA Order No. N00039-83-C-0136. However, the contents of this report should not be interpreted as necessarily representing the official policies, either express or implied, of the Defense Advanced Research Projects Agency or the United States government. Several colleagues contributed to the ideas in this paper. Edward A. Feigenbaum, H. Penny Nii, and Michael R. Genesereth generously argued with me about the blackboard architecture and probably disagree with parts of the paper. They, along with Bruce Buchanan, Douglas Lenat, and Peter Friedland, provided helpful comments on an earlier draft of the paper.

Table of Contents

Abstract

The blackboard architecture is a problem-solving framework originally developed for the Hearsay-II speech-understanding system [Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80]. Since then, it has been used to structure a variety of artificial intelligence systems and cognitive models. Nonetheless, our understanding of it remains limited and fragmented. Previous applications employ idosyncratic variations on the architecture and nowhere is the architecture defined and evaluated independent of a particular application. This paper is an attempt to fill that gap. It defines the blackboard architecture's basic components: entries, knowledge sources, the blackboard, and the control mechanism. It enumerates the basic assumptions underlying these components and shows how they give the architecture computational advantages, psychological plausibility, and general scientific merit. The paper concludes that, while further research is needed, the blackboard architecture is a viable candidate for a general problem-solving framework.

BLANK PAGE

1. Introduction

The blackboard architecture is a problem-solving framework developed for the Hearsay-II speech-understanding system [Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80]. Subsequently, researchers used the blackboard architecture to structure artificial intelligence systems for vehicle tracking and planning [Lesser, V.R., and Corkill 81], sonar signal interpretation (HASP: [Nii, H.P., Feigenbaum, E.A., Anton, J.J., and Rockmore, A.J. 82]), multiple-task planning (OPM: [Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. 79]), protein crystallography (CRYSALIS: [Terry, A. 83]), and scene analysis [Nagao, M., Matsuyama, T., and Mori, H. 79]. Other researchers used the blackboard architecture to structure psychological models of reading [McClelland, J.L., and Rumelhart, D.E. 81, Rumelhart, D.E., and McClelland, J.L. 82], text comprehension (Kintsch, personal communication), composition planning [Rose, M. 81], and multiple-task planning [Hayes-Roth, B., and Hayes-Roth, F. 79]. This variety of applications suggests that the blackboard architecture may be one of exceptional power and generality.

Despite its many applications, the blackboard architecture remains an informal construct. Individual system builders freely adapt the architecture to suit their particular problem domains, system requirements, and system building styles. Similarly, individual psychologists tailor the architecture to suit their problem-solving domains, scientific goals, and theoretical convictions. Thus, while the blackboard architecture is implicit in all of these applications, nowhere is it defined and evaluated independent of application. This paper is an attempt to fill that gap. Section 2 characterizes the fundamental elements of the blackboard architecture and to enumerate the assumptions underlying these elements. Section 3 characterizes the strengths of the blackboard architecture from computational, psychological, and scientific perspectives, and to show how these strengths derive from the assumptions. Section 4 draws conclusions bearing on the question in the title of this paper: Is the blackboard architecture a general framework for problem-solving?

2. Elements of the Blackboard Architecture

The blackboard architecture has four definitive elements: (a) entries, which are intermediate results generated during problem-solving; (b) knowledge sources, which are independent, event-driven processes that produce entries; (c) the blackboard, which is a structured, global data base that mediates knowledge source interactions and organizes entries; and (d) an intelligent control mechanism, which decides if and when particular knowledge sources should generate entries and record them on the blackboard. These elements interact to produce a problem-solving style that is characteristically incremental and opportunistic. Partial solution "islands" emerge, one entry at a time, in different structural partitions of the blackboard. New solution islands appear and existing islands grow wherever the opportunities are most promising. Eventually, mutually supportive partial solutions merge to form a complete solution.

Sections 2.1-2.5 below discuss in more detail the four elements of the blackboard architecture and the dynamics of problem-solving within the architecture. Examples from Hearsay-II [Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80] and OPM [Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. 79] illustrate the major points of the discussion. Hearsay-II is a computer system designed to understand a subset of spoken English. It takes a parameterized speech signal as input and attempts to interpret the syllables, words, and phrases it represents. Eventually, Hearsay-II hypothesizes a coherent semantic interpretation for the entire utterance. OPM is a computer simulation of people's planning of multiple-task sequences. It takes as input a problem description specifying a starting time and location, an ending time and location, a set of tasks to be done, a spatial environment in which to do tasks, and time constraints on the tasks it performs. It develops a plan specifying which tasks to do, in what sequence to do them, and by what routes to travel between successive tasks. In addition to illustrating the major points of the discussion, these two systems illustrate how the blackboard architecture has been applied to two different goals, AI system engineering and cognitive simulation, and to two different problem classes, signal interpretation and plan synthesis.

Section 2.6 abstracts and enumerates the assumptions underlying the characterization of entries, knowledge sources, the blackboard, and control.

2.1 Entries

Entries are intermediate results generated during problem solving. They may include both elements of the problem solution and information deemed important in generating solution elements. Depending upon the problem domain, entries may include perceptions, observations, beliefs, hypotheses, decisions, goals, interpretations, judgments, or expectations. For Hearsay-II, the problem is speech- understanding and the entries it generates are hypothetical interpretations of the speech-signal. For example, Hearsay-II might generate the hypothesis: "The first word in the utterance is 'get'." For OPM, the problem is to generate a plan and the entries it produces are decisions about the entries and organization of the plan. For example, OPM might generate the decision: "From Stanford, travel east on University Avenue."

Entries may have any relationships to one another that the user finds useful. For example, for Hearsay-II, the hypothesis "The first word in the utterance is 'get'" might have a "supports" relationship to another hypothesis, "The first word-phrase in the utterance is 'get me'." For OPM, the decision "From Stanford, travel east on University Avenue" may have an "elaborates" relationship to another decision, "Go from Stanford to The Good Earth." As these examples illustrate, related entries may represent a single concept at different levels of abstraction. Entries at low levels of abstraction support or elaborate entries at higher levels.

Entries are represented as objects of user-determined complexity. Associated attribute-value pairs describe an element's semantic content, its relationships to other entries, its history of generation and modification, and any other useful information.

2.2 Knowledge Sources

Knowledge sources are the cognitive processes that produce entries. Each knowledge source has two parts, a condition and an action. The condition describes the circumstances under which a knowledge source can contribute to the problem-solving process. Ordinarily, the condition requires the existence of certain previously generated entries. The action of a knowledge source generates new entries or modifies previously generated entries. The condition-action format corresponds to the two stages of knowledge source operation: (a) triggering, during which a knowledge source's condition is evaluated; and (b) execution, during which its action is carried out. Knowledge sources are event-driven because only triggered knowledge sources (those whose conditions evaluate to "true") can execute their actions. Knowledge sources operate independently and do not communicate with one another. However, they influence each other indirectly whenever the action of one knowledge source generates or changes a entry that satisfies or partially satisfies the condition of another knowledge source.

The following knowledge sources, MOW from Hearsay-II and FIRST-LEG from OPM, illustrate these points.

KS: MOW
CONDITION:
There are previously generated syllable hypotheses
ACTION:
Hypothesize words that comprise sequential subsets of the syllable hypotheses
Adjust their begin-times and end-times
Rate their credibilities

MOW is invoked when other HEARSAY-II knowledge sources hypothesize sequences of syllables likely to be present in the speech signal. When executed, MOW first hypothesizes all words in its vocabulary containing subsequences of the hypothesized syllables. Because of noise in the speech signal and ambiguity in syllable interpretation, MOW usually hypothesizes more than one word for a given syllable sequence. For multisyllabic words. MOW requires hypotheses for all syllables in at least one pronunciation of a word to be present before it hypothesizes that word. MOW then adjusts the begin-time and end-time within the speech signal for each hypothesized word and rates each word's credibility given the data.

KS: FIRST-LEG
CONDITION:
Two tasks, t1 and t2, have been planned in temporal order t1,t2
ACTION:
Plan the first leg of the route t1-t2 as:
  start = t1
  leg1 = the direction-on-street from t1 that minimizes angular
    disparity from the straight line connecting t1 and t2
  end-of-leg1 = the first intersection encountered on leg1

FIRST-LEG is invoked when other OPM knowledge sources decide to perform two tasks, t1 and t2, in the temporal order t1,t2. When executed, FIRST-LEG establishes t1 as the starting point for the route from t1 to t2. It then enumerates the paths leading from t1 in all directions on all streets. This set ordinarily includes at least two paths, one leading in each of two directions on t1's street. It includes additional paths if t1 lies at an intersection. FIRST-LEG determines which of these paths minimizes angular disparity from the hypothetical straight line connecting t1 and t2 and plans that path as the first leg in the route. It then identifies the first intersection encountered on the planned path and plans that intersection as the end of the first leg.

As these examples illustrate, knowledge sources frequently transform entries at one level of abstraction into entries at another level. Some knowledge sources, like MOW, operate bottom-up. They aggregate several lower-level entries (syllables) into a smaller number of higher-level entries (words). Other knowledge sources, like FIRST-LEG, operate top-down. They expand higher-level entries (task sequences) into a larger number of lower-level entries (route segments). Other knowledge sources operate entirely within a single blackboard level or between different blackboard "panels" [Hayes-Roth, B., and Hayes-Roth, F. 79, Terry, A. 83]. Thus, the blackboard architecture can combine knowledge sources embodying different inference mechanisms in a single problem-solving system.

Knowledge source representation follows the condition-action format. The condition component (a) requires at least one previously generated entry to have particular attributes and values; and (b) evaluates to "true" or "false." The action comonent generates one or more new entries or modifies one or more previously generated entries. Beyond these specifications, knowledge sources may be viewed as "black boxes" within which computations of user-determined complexity are applied to user-determined representations.

2.3 Blackboard

The blackboard is a global data base containing all entries generated by all knowledge sources during the problem-solving process. It serves two functions. First, it mediates all knowledge source interactions. Although knowledge sources do not communicate directly with one another, they influence one another indirectly by recording and responding to entries on the blackboard. Thus, an entry recorded by the action of one knowledge source may satisfy the condition of another knowledge source. Second, the blackboard organizes all partial and complete solutions generated for the problem under attack. These solutions comprise configurations of related entries on the blackboard.

The blackboard may have a user-determined internal structure to define important relationships among entries. These relationships might include, for example, generalization, aggregation, elaboration, support, temporal sequencing, or spatial arrangement. The blackboard structure also focuses knowledge source activity. Typically, a knowledge source's condition refers to previously generated entries in a particular area of the blackboard, while its action generates or modifies entries in some other area. Knowledge sources need not consider entries in areas of the blackboard not mentioned in their conditions or actions. Thus, the blackboard structure is a theoretical framework for organizing, inspecting, and generating entries.

Although the blackboard can have any additional structure the user desires, two orthogonal dimensions are definitive. The "vertical" dimension distinguishes entries at different levels of abstraction. As discussed above, entries at a given level of abstraction support or elaborate entries at the next higher level. The "horizontal" dimension represents distinctive, possibly overlapping intervals in the solution. These intervals may reflect temporal, spatial, conceptual, or other groupings. In some cases, the second dimension may expand into several interacting spatial/temporal dimensions. Together, these two dimensions define an aggregation hierarchy. Entries at higher levels of abstraction aggregate sets of entries at lower levels.

The Hearsay-II and OPM blackboards illustrate these points. First, consider the Hearsay-II blackboard:

Data Base Interface	
Phrase	
Word-Sequence	
Word	Hypotheses
Syllable	
Segment	
Parameter	

Temporal Locus in the Speech Signal

The Hearsay-II blackboard has seven levels of abstraction, each providing a different interpretation of the speech signal: parameters, segments, syllables, words, word-sequences, phrases, and data base interface Because the speech signal is linear with time, the blackboard also distinguishes overlapping temporal intervals within the speech signal. Taken together, the two dimensions, levels of abstraction and temporal intervals, define an aggregation hierarchy. Sequences of parameters are aggregated into segments, sequences of segments are aggregated into syllables, sequences of syllables are aggregated into words, and so forth.

Now consider the OPM blackboard:

Outcomes
Designs
Procedures
Operations

Tentative Decisions

Temporal Locus in the Plan

The OPM blackboard has four levels of abstraction, each providing a different specification of the plan: outcomes the plan should achieve, designs for the general spatial/temporal organization of the plan, procedures sequencing individual tasks, and operations for performing individual tasks and making inter-task transitions. Because the plan is linear with time, the blackboard also distinguishes temporal intervals within the plan. Taken together, these two dimensions define an aggregation hierarchy. Operations are aggregated into procedures, procedures are aggregated into designs, and designs are aggregated into outcomes. (The OPM blackboard also has other structural partitions not discussed in this paper.)

2.4 Control Mechanism

During the problem-solving process, many different knowledge sources may be triggered (have their conditions satisfied) simultaneously. In a serial implementation of the blackboard architecture, only one knowledge source can execute on a single problem-solving cycle. In a parallel implementation, several knowledge sources could execute simultaneously. In either case, an intelligent control mechanism determines which of the currently triggered knowledge sources should execute next.

In Hearsay-II and OPM, the control mechanism is operationalized as an agenda-based scheduler. On each problem-solving cycle, the agenda lists all pending knowledge source activities. The scheduler decides which of these activities to execute next. (Some blackboard models, such as HASP [Nii, H.P., Feigenbaum, E.A., Anton, J.J., and Rockmore, A.J. 82] and Crysalis [Terry, A. 83], operationalize control differently. However, their control mechanisms can be implemented with the kind of agenda-based scheduler discussed here.)

The scheduler's intelligence reflects its understanding of the system's repertoire of knowledge sources and their applicability to particular aspects of the problem under attack. Depending on its knowledge, the scheduler can exhibit a range of behaviors. At one extreme, the scheduler can adopt a single well-defined strategy. For example, it might adopt a strategy to proceed bottom-up through the levels of abstraction, first scheduling only knowledge sources that generate entries at the lowest level, then, when these are exhausted, scheduling only those that generate entries at the next level, and so forth. Alternatively, the scheduler can integrate multiple strategies. For example, it might adopt top-down and bottom-up strategies simultaneously, adhering to whichever one currently prescribes the smaller number of triggered knowledge sources (in other words, whichever one more effectively reduces the search space). The scheduler also can adopt arbitrary criteria and, based upon them, deviate opportunistically from its chosen strategy. For example, it might schedule recently triggered knowledge sources or knowledge sources that generate particularly important entries even though those knowledge sources do not implement its current strategy. It can even modify its strategy and other scheduling criteria dynamically in the course of problem-solving. The capacity for and disposition toward opportunistic behavior are characteristic of the blackboard architecture.

Consider the Hearsay-II scheduler. It maintains an agenda of pending "knowledge source activation records" (KSARs). Each KSAR specifies a knowledge source, triggering events, and a proposed activity. The proposed activity may be either: (a) a pattern-matching activity to determine whether other triggering conditions are met; or (b) a blackboard-modification activity to compute and execute the knowledge source's action. The scheduler decides which one of the KSARs to execute on each problem-solving cycle.

The Hearsay-II scheduler distinguishes two phases of system activity. During the first phase, it uses a bottom-up strategy. It schedules only KSARs that contribute to parameter-level blackboard modifications until all such KSARs have executed. It then schedules all segment-level KSARs, all syllable-level KSARs, and finally, all word-level KSARs. This phase teminates when all KSARs at these four levels have executed.

During the second phase of system activity, the scheduler becomes more opportunistic. On each cycle, the scheduler calculates a priority for each pending KSAR, using three criteria: (a) local effects of the proposed activity, including the nature of the hypotheses it would generate and the cumulative credibility and duration of the generated hypotheses and their predecessors; (b) global effects of the activity, in terms of cooperative and competitive relationships between the hypotheses it would generate and existing hypotheses on the blackboard; and (c) the computational resources required by the activity. On each cycle, the scheduler executes the pending KSAR with the highest priority.

Now consider the OPM scheduler. It maintains a similar agenda of pending KSARs. However, all KSARs propose blackboard modifications. None propose pattern-matching activities because all unsatisfied conditions for KSARs on the agenda are evaluated against each subsequent blackboard event. The OPM scheduler decides which one of the fully triggered KSARs to execute on each problem-solving cycle.

Unlike the Hearsay-II scheduler, the OPM scheduler does not distinguish sequential phases of problem-solving activity. However, it can adopt either of two global strategies. Its top-down strategy prescribes initial scheduling of outcome-level KSARs, then design-level KSARs, then procedure-level KSARs, and finally, operation-level KSARs. Its left-to-right strategy prescribes initial scheduling of KSARs affecting the first task in a sequential plan, then KSARs affecting the second task, and so forth through the final task in the plan. These strategies are proposed by "control knowledge sources" and recorded on a "control blackboard," not discussed in this paper. Ordinarily, a control knowledge source proposes a strategy appropriate for the problem at hand early in the problem-solving process and the scheduler uses it for the remainder of the process. Occasionally, another control knowledge source subsequently replaces or modifies the initial strategy. Unlike Hearsay-II, the OPM scheduler does not adhere strictly to its current strategy. Instead, it uses the strategy as one of several scheduling criteria, deviating from it opportunistically throughout the problem-solving process, sometimes abandoning it altogether. Other scheduling criteria, recommended by other control knowledge sources, include temporary attentional foci and specific scheduling policies (e.g., prefer recently triggered knowledge sources, prefer effective knowledge sources). Thus, the OPM scheduler effectively combines the two phases of Hearsay-II activity in a uniform approach best characterized as "strategic opportunism."

On each cycle, the OPM scheduler rates each pending KSAR against its three classes of criteria: (a) consistency with the scheduler's global strategy; (b) consistency with current attentional foci; and (c) consistency with scheduling policies. Unlike the Hearsay-II scheduler, the OPM scheduler does not always choose the activity with the highest combined rating. Instead, it chooses among pending activities with probabilities proportional to their ratings.

2.5 Dynamics of the Problem-Solving Process

Within the blackboard architecture, problem-solving proceeds as a sequence of cycles, each of which may generate one or more entries on the blackboard. Each cycle begins with an agenda of KSARs. The scheduler rates the KSARs and selects one or more of them. The selected KSARs execute, evaluating conditions, recording new entries on the blackboard or modifying existing entries. These events may trigger additional knowledge sources, causing new KSARs to be added to the agenda. They may invalidate conditions of pending KSARs, causing them to be deleted from the agenda or given a special status on the agenda. They may obviate the actions of pending KSARs, causing them to be deleted from the agenda or given a special status on the agenda.

The details of the problem-solving cycle depend upon whether the architecture is implemented as a serial model or with parallel components. Because Hearsay-II and OPM are serial models, they iterate the cycle:

1. schedule one pending KSAR
2. execute the scheduled KSAR
3. generate blackboard event(s)
4. trigger knowledge sources
5. update the agenda

Parallel implementations with scheduling synchronization might iterate a similar cycle:

1. schedule n pending KSARs
2. execute the scheduled KSARs
3. generate blackboard events
4. trigger knowledge sources
5. update the agenda

Parallel asynchronous implementations might iterate a set of interleaved cycles:

1. on availability of a free processor, schedule a pending KSAR
2. execute the scheduled KSAR
3. generate blackboard events
4. trigger knowledge sources
5. update the agenda
6. free the processor

Other cycles are also possible.

As defined above, the blackboard architecture supports problem-solving behavior that is more or less opportunistic. At one extreme, a rigorously strategic scheduler can use the available knowedge sources to generate entries systematically, gradually filling the blackboard with a coherent and parsimonious solution to the problem at hand. At the other extreme, an extremely opportunistic scheduler may appear to use the available knowledge sources haphazardly, scattering the blackboard with partial solutions that converge only at the last moment into a connected whole. While none of the implementations with which I am familiar exhibits behavior at either extreme, current implementations represent a variety of points along the continuum.

2.6 Assumptions of the Blackboard Architecture

This characterization of the elements of the blackboard architecture and its problem-solving dynamics embodies a number of specific assumptions. In order to make the architecture more explicit and amenable to scrutiny, this section abstracts and enumerates those assumptions.

1. Problem-solving activity generates a set of intermediate results, called entries, that are represented as objects with attributes and values.
2. Entries may have user-determined relationships to one another.
3. All entries have the relational attributes: abstracts/refines and adjacent-to.
4. All entries are recorded in a global data base, called the blackboard.
5. The blackboard structure includes partitions for different levels of abstraction and solution intervals.
6. The blackboard may have additional user-determined structure.
7. Independent processes, called knowledge sources, generate, modify, and record entries on the blackboard.
8. Each knowledge source has a condition that matches a hypothetical configuration of entries on the blackboard, performs computations, and evaluates to "true" or "false."
9. Each knowledge source has an action that performs computations and generates blackboard modifications.
10. Only a triggered knowledge source (one whose condition is true) can execute its action.
11. An intelligent scheduler determines which triggered knowledge source(s) should perform pattern matching functions or execute their actions.
12. The scheduler can base its decisions on user-determined criteria, including: (a) characteristics of the triggered knowledge sources; (b) characteristics of the events that trigger knowledge sources; (c) global or local problem-solving strategies; (d) temporary attentional foci; (e) general scheduling policies; (f) problem characteristics; (g) the current set of entries on the blackboard; (h) other information on the blackboard; (i) knowledge of its own previous behavior.

3. Motivations for the Blackboard Architecture

The blackboard architecture appears promising as a general framework for problem-solving because of its computational advantages, psychological plausibility, and general scientific merit. In Sections 3.1-3.3 below I attempt to characterize properties in each of these categories and to show how they derive from specific assumptions listed in Section 2.6, above.

3.1 Computational Advantages

The blackboard architecture was developed originally to cope with three common, but challenging computational problems: integration of multiple sources of knowledge; problems whose solution depends on heuristic methods and noisy data; and computational complexity. It provides two other computational advantages in its capacity to integrate different problem-solving methods and its potential for organizing parallel problem-solving activities. Each of these advantages is discussed below.

3.1.1 Integration of Multiple Sources of Knowledge

Solution of many tasks requires the application of multiple sources of knowledge. For speech-understanding, these include knowledge about the pronunciation of words, the syntax of the language, and the kinds of utterances a speaker might make. For multiple-task planning, they include knowledge about the relative benefits of performing alternative tasks, the expected resource requirements for specific tasks, and the geometry of efficient routes. It is not clear if or how such diverse knowledge could be integrated in a single, generally applicable problem-solving method.

The blackboard architecture finesses this problem by preserving the distinctions among different sources of knowledge. It permits different knowledge sources to embody qualitatively different sorts of expertise, to operate independently, and to contribute entries when they can (assumptions 1, 7, 8, 9, 10). Knowledge sources integrate the entries they generate with related entries already on the blackboard by recording appropriate relational links among them (assumptions 2, 4, 7). Thus, the blackboard architecture integrates the results, rather than the processes, of multiple knowledge sources.

3.1.2 Heuristic Methods and Noisy Data

Solution of many problems relies upon heuristic methods to evaluate noisy data. Both of these factors introduce error into the problem-solving process. The blackboard architecture compensates for noisy data and heuristic methods by enabling multiple, redundant knowledge sources to converge on the most promising entries (assumptions 1, 4, 7). For example, Hearsay-II improves the reliability of its word hypotheses by integrating the hypotheses of a bottom-up knowledge source, MOW, whose inputs are at the syllable level, and a combined top-down-bottom-up knowledge source, VERIFY, whose inputs are at the phrase and segment levels. Similarly, OPM improves the quality of its operation plans by integrating the suggestions of top-down knowledge sources like FIRST-LEG with a number of single-level knowledge sources that perform spatial/temporal chaining. In both cases, the convergence of expertise from two or more knowledge sources produces more reliable entries than would be produced by a single knowledge source.

3.1.3 Management of Computational Complexity

Many knowledge-based systems must cope with extensive search spaces. The blackboard architecture architecture incorporates two powerful methods for reducing problem complexity. First, it integrates problem-solving activities at multiple levels of abstraction (assumptions 3, 5). This enables the problem-solver to solve a simplified version of the problem at hand and then use that solution to guide and limit exploration of the space of more detailed solutions [Newell, A., Shaw, J.C., and Simon, H.A. 59, Stefik, M., Aikens, J., Balzer, R., Benoit, J., Birnbaum, L., Hayes-Roth, F., and Sacerdoti, E. 82]. Second, it provides opportunistic search methods (assumptions 11, 12). This permits the problem-solver to focus attention dynamically on the most productive aspects of the problem-solving process. More importantly, it permits the independent generation and convergence of solution "islands," potentially reducing the search space dramatically [Minsky, M. 61, Stefik, M., and Conway, L. 82].

3.1.4 Integration of Problem-Solving Methods

Current AI systems embody a variety of problem-solving methods, combining particular inference mechanisms (e.g., forward chaining, backward chaining, goal-driven, data-driven, top-down, bottom-up) with particular search strategies (e.g., exhaustive, heuristic, left-to-right, right-to-left, opportunistic, island-merging). The typical AI system focuses on a single problem-solving method, presumably the combination of inference mechanism and search strategy that best suits the problem domain under attack. However, some problems benefit from multiple problem-solving methods. For example, the speech-understanding problem appears to benefit from exhaustive, bottom-up recognition early in the problem-solving process and from opportunistic, top-down and bottom-up methods later in the problem-solving process [Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80]. The multiple-task planning problem appears to benefit from opportunistic, top-down and bottom-up methods throughout the problem-solving process [Hayes-Roth, B., and Hayes-Roth, F. 79]. Both tasks appear to benefit from opportunistic generation and merging of solution islands for different data intervals (temporal intervals in the speech signal or plan) throughout the problem-solving process.

The blackboard architecture differs from other problem-solving frameworks in the ease with which it integrates multiple problem-solving strategies. Its independently defined knowledge sources can embody any of the prototypical inference mechanisms (assumptions 7, 8, 9). Its control mechanism can sequence knowledge source pattern-matching or action computation according to any single search strategy or any combination of search strateggies (assumptions 11, 12). Thus, the blackboard architecture can accommodate problem-solving systems that use radically different problem-solving methods as well as systems that use arbitrary combinations of methods.

3.1.5 Organization of Parallel Problem-solving

Although the possibility of parallel computing has provoked interesting speculations and experiments [Lesser, V.R., and Erman, L.R. 79, Fahlman, S.E. 82, Minsky, M. 77, Smith, R.G., and Davis, R. 78, Yonezawa, A., and Hewitt, C. 77], only recently have researchers had access to the types and numbers of processors they need to build parallel systems. The blackboard architecture is a natural framework for several kinds of parallel activities. For example, with access to multiple processors, the control mechanism could schedule condition matching, action execution, and blackboard modification for several knowledge sources in parallel (assumptions 7, 8, 9, 10, 11). There has been some experimentation along these lines [Fennell, R., and Lesser, V. 77, Lesser, V.R., and Corkill 81, Lesser, V.R., and Erman, L.R. 80], but the potential gains from a parallel blackboard architecture remain largely unexplored.

3.2 Psychological Plausibility

Although an AI architecture need not mimic the organization or functioning of human information processing, doing so increases its credibility as a general framework for intelligent problem solving. The blackboard architecture corresponds in several ways to current models of human information-processing, as discussed below.

3.2.1 Symbolic Nature

Newell and Simon [Newell, A., and Simon, H.A. 72] argued for the symbolic nature of human problem-solving and identified these basic components: symbol structures, a memory to contain symbol structures, information processes to produce symbol structures, and an interpreter to control the information processes. These components correspond to the blackboard architecture's entries, blackboard, knowledge sources, and scheduler (assumptions 1, 4, 7, 11). The main difference between components in the two systems derives from Newell and Simon's concentration on low-level, computationally simple components, contrasted with the blackboard architecture's accommodation of arbitrarily complex components.

3.2.2 Distinctive Memory Components

Memory is one of the most thoroughly investigated aspects of human intelligence (see, for example, [Anderson, J.A., and Bower, G. 73]. Although the blackboard architecture does not embody a model of memory search, its components correspond to widely recognized functional memory stores. For example, very recent modifications to the blackboard and the KSARs they trigger (assumptions 1, 10) correspond to the contents of short-term memory. A more inclusive subset of relatively recent blackboard modifications and pending KSARs corresponds to the contents of working memory for the task at hand. Knowledge sources (assumptions 7, 8, 9) define some of the contents of long-term memory. Additional long-term memory contents could be accommodated in special blackboard partitions defined for that purpose (assumption 6). Through the activities of particular knowledge sources (assumption 7), the architecture provides access to information in external memory.

3.2.3 Production-Like Character

Newell and Simon [Newell, A., and Simon, H.A. 72] demonstrated the production-like character of human information processing. The blackboard architecture embodies this character in its independent knowledge sources and their condition-action format (assumptions 7, 8, 9, 10).

3.2.4 Serial and Parallel Components

Different components of the human information-processing system have distinctive serial or parallel performance capabilities. For example, Newell and Simon argue that the central processor is inherently serial, but acknowledge that long-term memory access operates in parallel. Functions underlying peripheral processes like pattern recognition and motor performance also operate in parallel [Anderson, J., and Hinton, G. 81, Dodwell, P.C. 70]. While the blackboard architecture does not entail any particular configuration of serial and parallel components, its flexible and intelligent scheduling of independent knowledge sources (assumptions 7, 11, 12) permits it to model arbitrary configurations.

3.2.5 Goal-driven Behavior

Newell and Simon [Newell, A., and Simon, H.A. 72] describe the goal-driven nature of human behavior as reflected in six behavioral characteristics: interruptability, subgoaling, depth-first subgoaling, equifinality, and consummation. Typical blackboard models exhibit all of these features, as illustrated by the following examples from OPM.

Consider interruptability. OPM's preference for scheduling recently invoked knowledge sources (assumption 12b) produces sequences of closely related actions that serve a common goal. However, OPM exhibits interruptability when it uses other criteria to schedule actions that are only indirectly related to their immediate predecessors (assumption 12). The scheduler typically, though not always, resumes the interrupted sequence of actions later in the planning process.

Consider subgoaling behaviors. OPM's knowledge sources base decisions about the plan on previous decisions regarding goals that characterize desired plan elements and data that instantiate the goals (assumptions 7, 8, 9). It exhibits subgoaling whenever goal decisions invoke knowledge sources that generate the data necessary to instantiate them and vice versa. The scheduler's preference for recently invoked knowledge sources (assumption 12b) promotes depth-first subgoaling.

Equifinality, avoidance of repetition, and consummation refer to the system's application of alternative problem-solving methods until it achieves its first-order goals. OPM's first-order goal is to generate a satisfactory plan and its methods are embodied in knowledge sources that generate and instantiate second-order goals for individual plan elements (assumptions 7, 8, 9). When OPM schedules knowledge sources that fail to produce satisfactory plan elements, it exhibits equifinality by scheduling other knowledge sources that generate alternative goals or respond to previously generated alternatives (assumption 12 g, h, i). OPM avoids repetition by choosing not to schedule previously executed activities (assumption 12). It exhibits consummation by choosing not to schedule redundant, alternative activities as soon as one of them produces a satisfactory plan element (assumptions 12g, h, i).

3.2.6 Integration of Multiple Sources of Knowledge

As discussed above, solution of many problems requires the integration of multiple sources of knowledge. Human beings integrate multiple knowledge sources in performing even the simplest of tasks. For example, in planning a day's activities, they use knowledge of the requirements and alternatives before them, cost/benefit analysis, constraint management, local routes and terrain, coordination of objectives, etc. [Hayes-Roth, B., and Hayes-Roth, F. 79]. In reading, people use their knowledge of the perceptual features of letters, the symbolic content of the letters, common letter sequences, valid words, the semantic categories words can fill, and valid phrasal sequences of semantic categories [McClelland, J.L., and Rumelhart, D.E. 81, Rumelhart, D.E., and McClelland, J.L. 82]. More sophisticated tasks require integration of more diverse knowledge sources. For example, in developing a speech, a politician may use knowledge of particular political issues, the opinions of his or her supporters and detractors, rhetorical style, his or her own strengths and weaknesses as a speaker, characteristics of the medium etc. As discussed above, the blackboard architecture incorporates multiple knowledge sources as distinct, independently operating components of the problem-solving process, integrating their results in its global data base, the blackboard (assumptions 4, 7).

3.2.7 Exploitation of Levels of Abstraction

Newell, Shaw, and Simon [Newell, A., Shaw, J.C., and Simon, H.A. 59] introduce the concept of levels of abstraction for use in problem reduction. In attempting to solve a problem, people sometimes abstract a simpler problem by eliminating some of the details of the original. They solve the abstract problem and then use the solution as a plan for solving the more detailed original. Miller, Galanter, and Pribram [Miller, G.A., Galanter, E., and Pribram, D.H. 60] do not use the expression levels of abstraction, but their hierarchical control plans also embody the concept. More recently, a number of researchers have been using levels of abstraction to model story comprehension [Thorndyke, P.W. 77], reading [McClelland, J.L., and Rumelhart, D.E. 81, Rumelhart, D.E., and McClelland, J.L. 82], and planning [Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. 79, Hayes-Roth, B., and Thorndyke, P.W. 0"]. The blackboard architecture incorporates levels of abstraction in its blackboard structure (assumptions 3, 5).

3.2.8 Strategic Flexibility and Opportunism

Bruner, Goodnow, and Austin [Bruner, J.S., Goodnow, J.J., and Austin, G.A. 56] document people's natural flexibility in adopting alternative strategies for solving related classes of problems or even for the same problem. Subsequent research replicates this observation and demonstrates that people can be induced to adopt particular strategies [Hayes-Roth, B. 80, Reed, S.K., Ernst, G.W., and Banerji, R. 74, Simon, H.A., and Reed, S.K. 76]. Hayes-Roth and Hayes-Roth [Hayes-Roth, B., and Hayes-Roth, F. 79] also observe that people opportunistically select and depart from chosen strategies. The blackboard architecture's combination of independent knowledge sources (assumption 7) and heuristic scheduling (assumptions 11, 12) maximizes strategic flexibility and the potential for opportunism.

3.2.9 Interruptability

In addition to its role as a criterion for identifying goal-driven behavior, interruptability stands out as a salient characteristic of human cognitive behavior [Hayes-Roth, B., and Hayes-Roth, F. 79, Newell, A., and Simon, H.A. 72, Reitman, W.R. 65]. The blackboard architecture permits interruptions in problem-solving activity to occur between knowledge source condition matching and action execution and immediately following blackboard modification (assumptions 8, 9, 10, 11). Its opportunistic scheduling mechanism (assumption 12) naturally leads to interruptions, some of which temporarily draw attention away from the original action sequence and some of which permanently alter the course of problem- solving.

3.2.10 Ability to learn

Perhaps the most powerful computational power of the human information processor is its ability to learn [Anzai, Y., and Simon, H.A. 79, Hayes-Roth, B. 80, Reed, S.K., Ernst, G.W., and Banerji, R. 74, Simon, H.A., and Reed, S.K. 76]. Although the blackboard architecture does not contain any operational learning mechanism, it provides a foundation for several different types of learning: acquiring new knowledge sources (assumption 7); tuning the condition or action components of individual knowledge sources (assumptions 8, 9); compiling several knowledge sources into a single knowledge source; de-compiling a knowledge source into component knowledge sources; perceiving or inventing new levels of abstraction or other blackboard partitions (assumptions 5,6); acquiring new scheduling heuristics (assumption 12); and modifying the memory inspected by knowledge sources.

3.2.11 Individual Differences

Human beings exhibit marked individual differences in information processing behavior [Bruner, J.S., Goodnow, J.J., and Austin, G.A. 56, Goldin, S.E., and Hayes-Roth, B. 80, Hayes-Roth, B. 80, Newell, A., and Simon, H.A. 72]. The blackboard architecture can model individual differences in operational blackboard structure (assumption 5), knowledge source repertoire (assumption 7), knowledge source details (assumptions 8, 9), scheduling heuristics (assumption 12), and memory contents. Goldin and Hayes-Roth [Goldin, S.E., and Hayes-Roth, B. 80] found evidence for all of these differences in people's planning behavior.

3.2.12 Intuitive Appeal

The blackboard architecture seems to correspond well to people's own introspections about their problem-solving processes. In efforts to extract and codify the expertise of doctors, lawyers, scientists, geologists and other domain experts, knowledge engineers rarely encounter individuals who can articulate a well-formed algorithmic procedure for making even routine professional decisions. On the other hand, most of these experts can describe a variety of special-purpose heuristic methods used in different combinations and sequences for different problems. These heuristics correspond to the blackboard architecture's knowledge sources (assumption 7) and the different combinations and sequences reflect different scheduling decisions (assumptions 11, 12)

3.3 Scientific Merit

A general framework for problem solving should satsify the following general scientific criteria: generality, explanation, constraint, prediction, and parsimony. The blackboard architecture represents at least a first step toward each of these criteria.

3.3.1 Generality

The blackboard architecture is powerful and general. Its multi-dimensional blackboard structure (assumption 5) permits it to accommodate multiple problem representations. Its open-ended scheduling capabilities (assumption 12) enable it to implement arbitrary problem-solving strategies (e.g., top-down, bottom-up, breadth-first, depth-first, best-first, forward-chaining, backward chaining, goal-driven, data-driven, island-driven, means-ends analysis) and arbitrary combinations of strategies. It can implement different kinds of parallelism (assumptions 7, 8, 9, 11). It can describe performance by people or machines or any combination of problem-solvers by treating them as knowledge sources or schedulers (assumptions 7, 11). It can describe performance in a variety of problem domains.

3.3.2 Explanation

The blackboard architecture's explanatory power derives from its sufficiency to produce interesting cognition in artificial systems and to model the cognition of human beings. Previous research has exploited the architecture in developing AI systems for a variety of signal-interpretation problems [Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80, Nagao, M., Matsuyama, T., and Mori, H. 79, Lesser, V.R., and Corkill 81, Nii, H.P., Feigenbaum, E.A., Anton, J.J., and Rockmore, A.J. 82, Terry, A. 83] and planning problems [Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. 79, Lesser, V.R., and Corkill 81]. Other research has used it to develop psychological models for signal-interpretation problems, like reading [McClelland, J.L., and Rumelhart, D.E. 81, Rumelhart, D.E., and McClelland, J.L. 82] and text comprehension (Kintsch, personal communication), and planning problems, like multiple-task planning [Hayes-Roth, B., and Hayes-Roth, F. 79] and composition planning [Rose, M. 81]. All of these researchers attribute at least part of their success to their use of the blackboard architecture. On the other hand, there currently are no rigorous methods for evaluating either the computational utility or psychological validity of models as complex as blackboard models. Developing such methods is a prerequisite to extending our understanding of the architecture's explanatory power. Assessing the architecture's applicability in a still wider range of domains is another important area for future research.

3.3.3 Constraint

A useful theory of problem-solving must also constrain the set of permissible behaviors. While maximizing the number of interesting behaviors it produces, the theory should also minimize the number of uninteresting behaviors it produces. In the case of artificial problem-solving, this means restricting behaviors to those that are effective in achieving goals. In the case of natural problem-solving, this means restricting behaviors to those that human beings exhibit. The blackboard architecture is open-ended in its ability to produce behaviors, interesting or otherwise. Thus, the architecture provides extreme generality, without offering much constraint. However, particular blackboard models provide constraint in their specification of blackboard structure, knowledge source definition, and scheduling heuristics (assumptions 5, 7, 8, 9, 12). Previous blackboard models offer alternative examples. For example, the Hearsay-II blackboard comprises a single, three-dimensional plane, with seven levels of abstraction. Subsequent models expand the blackboard structure to incorporate multiple planes related to one another in different ways. See, for example, the Crysalis blackboard [Terry, A. 83] and the OPM blackboard [Hayes-Roth, B., and Hayes-Roth, F. 79]. Developing prescriptive specifications for blackboard structure, knowledge source definition, and scheduling heuristics is another important area for future research.

3.3.4 Prediction

The predictive power of a model lies in its ability to predict important parameters and characteristics of problem-solving activity. Although the blackboard architecture itself does not entail specific predictions, particular blackboard models do. For example, if we view Hearsay-II as a working hypothesis about effective speech understanding, we may translate its specifications into predictions. Thus, its inclusion of seven levels of abstraction presupposes their individual necessity and their combined sufficiency to solve the speech-understanding problem. Thus, it entails the predictions that removal of any one of the levels would seriously impair performance and that addition of a new level would not significantly improve performance. Similarly, Hearsay-II's distinction between bottom-up scheduling of knowledge sources during phase 1 and opportunistic scheduling during phase 2 entails the prediction that other scheduling strategies would produce inferior performance. Of course, the researchers who developed Hearsay-II and the other blackboard models discussed above probably would not assert such hypotheses without qualification. However, they are useful tests of the models as stages in the development of an effective model of problem- solving. This is another area for future research.

3.3.5 Parsimony

Parsimony is the fifth scientific criterion for a proposed theory. Other things being equal, we prefer theories that make the fewer and simpler assumptions. Given its power and generality, the blackboard architecture is surprisingly simple. It provides a uniform architecture for a variety of tasks, individuals (human and artificial), and strategies. It provides uniform condition-action knowledge sources (assumptions 7, 8, 9, 10), a uniform data representation and structure (assumptions 1, 3, 4, 5, 6), and a uniform control mechanism for scheduling (assumption 11).

4. Conclusions

I have tried to define the fundamental elements of the blackboard architecture, to enumerate the assumptions underlying these elements, and to show how the assumptions enable the architecture to satisfy important computational, psychological, and scientific requirements. There is one important scientific criterion that I have omitted from this analysis: necessity. For a theory to be necessary, it must be true that (a) the theory explains all relevant data; and (b) no conceivable alternative theory can explain all of the data. Given the preliminary status of the blackboard architecture, assessment of its necessity is obviously premature. At this point, I would not even claim that the architecture is demonstrably better than known alternatives. Here are the claims I would make:

1. The blackboard architecture, as defined in this paper, is a well-defined object and amenable to study.
2. The architecture's assumptions enable it to satisfy important computational, psychological, and scientific criteria.
3. The architecture's ability to satisfy this array of criteria distinguishes it from alternatives that satisfy only a subset of the criteria.
4. The architecture's range of previous applications, along with the present analysis, suggest that it may be a generally useful framework for modelling a wide range of problem-solving behavior.
5. Claims 1-4 warrant further investigation of the blackboard architecture.

In summary, I would answer the question posed in the title of this paper: "The Blackboard Architecture: A General Framework for Problem Solving?" in a single word: Perhaps.

References

[Anderson, J., and Hinton, G. 81]
Anderson, J., and Hinton, G. (Eds.).
Parallel models in associative memory.
Hillsdale, N.J.: Erlbaum, 1981.

[Anderson, J.A., and Bower, G. 73]
Anderson, J.A., and Bower, G.
Human associtive memory.
Washington, D.C.: Winston-Wiley, 1973.

[Anzai, Y., and Simon, H.A. 79]
Anzai, Y., and Simon, H.A.
The theory of learning by doing.
Psychological Review 86:124-140, 1979.

[Bruner, J.S., Goodnow, J.J., and Austin, G.A. 56]
Bruner, J.S., Goodnow, J.J., and Austin, G.A.
A study of thinking.
New York: Science Editions, Inc., 1956.

[Dodwell, P.C. 70] Dodwell, P.C.
Visual pattern recognition.
New York: Holt, 1970.

[Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. 80]
Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R.
The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty.
Computing Surveys 12:213-253, 1980.

[Fahlman, S.E. 82]
Fahlman, S.E.
Three flavors of parallelism.
Technical Report, Pittsburgh, Pa.: Carnegie-Mellon University, 1982.

[Fennell, R., and Lesser, V. 77]
Fennell, R., and Lesser, V.
Parallelism in AI Problem-Solving: A case study of HSII.
IEEE Transactions on Computers C-26:98-111, 1977.

[Goldin, S.E., and Hayes-Roth, B. 80]
Goldin, S. E., and Hayes-Roth, B.
Individual differences in planning processes.
Technical Report N-1488-ONR, Santa Monica, Ca.: Rand Corporation, 1980.

[Hayes-Roth, B. 80]
Hayes-Roth, B.
Flexibility in executive processes.
Technical Report N-1170-ONR, Santa Monica, Ca.: Rand Corporation, 1980.

[Hayes-Roth, B., and Hayes-Roth, F. 79]
Hayes-Roth, B., and Hayes-Roth, F.
A cognitive model of planning.
Cognitive Science 3:275-310, 1979.

[Hayes-Roth, B., and Thorndyke, P.W. 0"]
Hayes-Roth, B., and Thorndyke, P.W.
Decisionmaking during the planning process.
Technical Report N-1213-ONR, Santa Monica, Ca.: Rand Corporation, 1980".

[Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. 79]
Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S.
Modelling planning as an incremental, opportunistic process.
Proceedings of the International Joint Conference on Artificial Intelligence 6:375-383, 1979.

[Lesser, V.R., and Corkill 81]
Lesser, V.R., and Corkill.
Functionally accurate cooperative distributed systems.
IEEE Transactions on Systems, Man, and Cybernetics SMC-1:81-96, 1981.

[Lesser, V.R., and Erman, L.R. 79]
Lesser, V.R., and Erman, L.R.
An experiment in distributed interpretation.
Technical Report, Pittsburgh, Pa.: Carnegie-Mellon University, 1979.

[Lesser, V.R., and Erman, L.R. 80]
Lesser, V.R., and Erman, L.R.
Distributed Interpretation: A model and experiment.
IEEE Transactions on Computers C-29:1144-1163, 1980.

[McClelland, J.L., and Rumelhart, D.E. 81]
McClelland, J.L., and Rumelhart, D.E.
An interactive activation model of context effects in letter perception: Part 1. An account of basic findings.
Psychological Review 88:375-407, 1981.

[Miller, G.A., Galanter, E., and Pribram, D.H. 60]
Miller, G.A., Galanter, E., and Pribram, D.H.
Plans and the structure of behavior.
New York: Holt, Rinehart, and Winston, Inc, 1960.

[Minsky, M. 61] Minsky, M.
Steps toward artificial intelligence.
In E. A. Feigenbaum and J. Feldman (editor), *Computers and thought*, . New York: McGraw-Hill, 1961.

[Minsky, M. 77] Minsky, M.
Plain talk about neurodevelopmental epistemology.
Proceedings of the International Joint Conference on Artificial Intelligence 5, 1977.

[Nagao, M., Matsuyama, T., and Mori, H. 79]
Nagao, M., Matsuyama, T., and Mori, H.
Structured analysis of complex photographs.
Proceedings of the International Joint Conference on Artificial Intelligence 6:610-616, 1979.

[Newell. A., and Simon, H.A. 72]
Newell, A., and Simon, H. A.
Human problem solving.
Englewood Cliffs, N.J.: Prentice-Hall, 1972.

[Newell, A., Shaw, J.C., and Simon, H.A. 59]
Newell, A., Shaw, J.C., and Simon, H.A.
Report on a general problem-solving program.
Proceedings of the International Conference on Information Processing , 1959.

[Nii, H.P., Feigenbaum, E.A., Anton, J.J., and Rockmore, A.J. 82]
Nii, H.P., Feigenbaum, E.A., Anton, J.J., and Rockmore, A.J.
Signal-to-symbol transformation: HASP/SIAP case study.
AI Magazine 3:23-35, 1982.

[Reed, S.K., Ernst, G.W., and Banerji, R. 74]
Reed, S.K., Ernst, G.W., and Banerji, R.
The role of analogy in transfer between similar problem states.
Cognitive Psychology 6:436-450, 1974.

[Reitman, W.R. 65]
Reitman, W.R.
Cognition and problem-solving.
New York: Wiley, 1965.

[Rose, M. 81] Rose, M.
The Composition Process.
PhD thesis, UCLA, 1981.

[Rumelhart, D.E., and McClelland, J.L. 82]
Rumelhart, D.E., and McClelland, J.L.
An interactive model of context effects in letter perception: Part 2. The contextual enhancement effect and some tests and extensions of the model.
Psychological Review 89:60-94, 1982.

[Simon, H.A., and Reed, S.K. 76]
Simon, H.A., and Reed, S.K.
Modeling strategy shifts in a problem-solving task.
Cognitive Psychology 8:86-97, 1976.

[Smith, R.G., and Davis, R. 78]
Smith, R.G., and Davis, R.
Distributed problem solving: The contract net approach.
Technical Report HPP-78-7, Stan-CS-78-667, Stanford University, 1978.

[Stefik, M., Aikens, J., Balzer, R., Benoit, J., Birnbaum, L., Hayes-Roth, F., and Sacerdoti, E. 82]
Stefik, M., Aikens, J., Balzer, R., Benoit, J., Birnbaum, L., Hayes-Roth, F., and Sacerdoti, E.
The organization of expert systems: A prescriptive tutorial.
Artificial Intelligence 18:135-173, 1982.

[Stefik, M., and Conway, L. 82]
Stefik, M., and Conway, L.
Towards the principled engineering of knowledge.
AI Magazine 3:4-16, 1982.

[Terry, A. 83] Terry, A.
Hierarchical control of production systems.
PhD thesis, UC, Irvine, 1983.

[Thorndyke, P.W. 77]
Thorndyke, P.W.
Cognitive structures in comprehension and memory of narrative discourse.
Cognitive Psychology 9:77-100, 1977.

[Yonezawa, A., and Hewitt, C. 77]
Yonezawa, A., and Hewitt, C.
Modelling distributed systems.
Proceedings of the International Joint Conference on Artificial Intelligence 5, 1977.

FILMED FROM BEST AVAILABLE COPY