November 1988

Report No. STAN-CS-88-1229

Also Numbered KSL-88-50

Distributing Intelligence within an Individual

by

B. Hayes-Roth, M. Hewett, R. Washington, R. Hewett, and A. Seiver

Department of Computer Science

Stanford University
Stanford, California 94305

Knowledge Systems Laboratory
Report No. KSL 88-50

October 1988

DISTRIBUTING INTELLIGENCE WITHIN AN INDIVIDUAL

by

Barbara Hayes-Roth, Michael Hewett, Richard Washington,
Rattikorn Hewett

Stanford University: Knowledge Systems Laboratory

Adam Seiver

Palo Alto Veterans Administration Medical Center

KNOWLEDGE SYSTEMS LABORATORY
Computer Science Department
Stanford University
Stanford, California 94305

To appear in:
Distributed Artificial Intelligence, Vol 2, L. Gasser and M.N. Huhns (Eds.),
Morgan Kaufman, 1988.

ACKNOWLEDGEMENT

This research was supported by DARPA contract #N00039-83-C-0136 and NIH contract 5P41-RR-00785 and by gifts from Rockwell International, Inc. and FMC Corporation, Inc. The Palo Alto VAMC supports Dr. Adam Seiver's participation in the project. Mr. Reed Hastings and Mr. Nicholas Parlante worked on an earlier version of Guardian. We have had many discussions of the intensive care monitoring task with our colleague, Dr. Lawrence Fagan, and his students. We thank Dr. Edward Feigenbaum for sponsoring the work at the Knowledge Systems Laboratory.

1. A Metaphor for DAI: The Individual

Distributed artificial intelligence (DAI) refers generally to systems in which decentralized, cooperative agents work synergistically to perform a task. Within this general description, however, there is considerable variability in the operational definitions of terms. "Agents" may refer to arbitrary numbers of more or less sophisticated computatational entities. "Decentralized" may refer to the distribution of knowledge, data, control, or computational resources among different agents. "Cooperative" may refer to a purely discretionary exchange of a small subset of available information or, at the other extreme, to an inevitable sharing of most information.

These alternative definitions of terms entail a space of DAI system models, many of which bear metaphorical resemblances to biological or social systems, such as neural networks [6, 19], complex problem-solvers [5, 11, 13], teams [1, 2, 4], contract nets [3, 20], and societies [16, 18]. None of these models is "correct" or "incorrect." Rather, they capture different, complementary kinds of intelligence, with each model supporting different design objectives and task requirements.

Our DAI model, previously proposed and discussed in [7, 8, 9], metaphorically resembles a single intelligent individual. Its design objectives and associated architectural provisions may be summarized as follows:

1. To support adaptation to a dynamic environment, the model provides locally controlled agents for asynchronous and concurrent perception, action, and cognition.
2. To support performance of multiple complex reasoning tasks, the model provides task-specific sets of functionally independent reasoning agents.
3. To support a range of reasoning strategies, the model provides dynamic control of task-specific reasoning agents.
4. To support concurrent performance of multiple reasoning tasks, the model provides interleaving of their respective reasoning agents.
5. To support an orderly and explainable reasoning process, the model provides dynamic global control of all reasoning agents.

6. To support coordination of loosely coupled reasoning tasks, the model provides a globally accesible representation of all knowledge and inferences.

7. To support global coherence, the model provides central determination of control parameters on all perception, action, and cognition functions.

We have instantiated the single individual model in the Guardian system for patient monitoring in a surgical intensive care unit (SICU). Although conventional SICU monitoring practice instantiates the team model of distributed intelligence, our analysis in section 2 suggests that effective SICU monitoring entails the design objectives indicated above. Therefore, as shown in section 3, our design for Guardian instantiates the single individual model. Section 4 illustrates Guardian's performance in a typical SICU monitoring scenario. Section 5 discusses our preliminary conclusions regarding the value of the single individual model for the Guardian system.

2. Guardian's Task: Monitoring Patients in the SICU

The sickest surgical patients in the hospital are cared for in the surgical intensive care unit (SICU). Most of these patients have failure of one or more organ systems--usually lung or heart. Organ system failure is treated with life-support devices which assume the fundamental functions of the ailing system until it can heal. For example, the ventilator (see Figure 2-1) is an artificial breathing machine that augments the patient's own respiration. Life-support devices are adjusted based upon frequent patient observations. Some of these observations are made continually by automatic measuring machines, for example, measurements of air pressures and air flows in the patient-ventilator system. Some of the observations are made intermittently. Chest X-rays, for example, are usually taken once or twice a day. The short-term goal of SICU monitoring is to keep the patient comfortable and progressing toward therapeutic objectives. The long-term goal is to gradually withdraw life-support devices so that the patient can function autonomously.

Current SICU monitoring practice instantiates the team model of distributed intelligence. Lead by the surgeon, different experts on the critical care team cooperate to interpret and synthesize large amounts of patient data. The surgeon, who performs the operation and is legally responsible for the patient, has the best grasp of the cause of the patient's problem, the surgical management of the disease, and the overall patient care strategy. Nurses, who are present at the bedside, have continuous access to automatically measured patient data and the

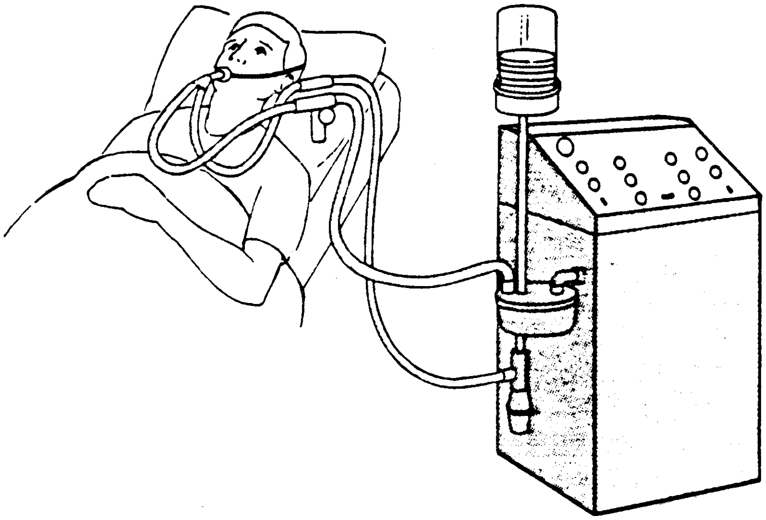


Figure 2- 1: Patient Supported by a Ventilator.

best grasp of minute-to-minute details of the patient's condition. Other consultants have the best understanding of particular aspects of the patient's condition within their specialty. For example, respirator therapists have detailed knowedge of the functioning and use of the respirator. Radiologists are expert at reading chest X-rays. High-quality patient monitoring requires cooperation among critical care team members to continuously interpret patient data and determine therapeutic actions.

The team model of SICU monitoring reflects both organizational and economic considerations. As medical knowledge has grown, the profession has distributed that knowledge among increasingly specialized practicioners. Each of these specialists is exceptionally well prepared to handle a part of the SICU monitoring task, but none is adequately prepared to handle the entire task. In addition, physician specialists are too valuable to take responsibility for the routine 80% of SICU monitoring activities.

Nonetheless, the team model of STCU monitoring has serious limitations. Given the distribution of knowledge and skills among different experts with multiple responsibilities, these experts are rarely present in the SICU at precisely the moment their expertise is required. As a result, the following kinds of problems can occur:

Scenario 1

It is 3 am. Mr. Stone returned from the operating room 12 hours ago following an *emergency* replacement of the major blood vessel in his abdomen, the aorta. Now his urine-output is 15 cc per hour. Since it has remained constant for the past 3 hours the nurse has not given that number much attention. He is covering another patient who is much more unstable and consequently is a much higher priority. It is now 6 am and the surgeon has returned for morning rounds. When she reviews the chart, she notices that the urine output has been 15 cc per hour since midnight. Anything less than 60 cc is a significant problem! She is quite distressed that the nurse had not called her when it first developed. Now the patient. has a significant chance of developing renal failure with a 90 percent mortality. If only the nurse had recognized the abnormal urine output, the crisis could have been completely avoided.

Scenario 2

It is 8 AM, Dr. Payne, the radiologist, is trying to read the chest film on Mr. Jones, All that the X-ray requisition says is “Post-op chest”. This provides very little contextual information. Dr. Payne needs to know how high the filling pressures are to differentiate pulmonary edema from adult respiratory distress syndrome. Although he is up in the SICU, he does not have the time to go through the patient chart to find the necessary data. He is inexperienced with intensive care bedside practice and always has a difficult time finding the relevant information. Because he does not have a good background summary on Mr. Jones he cannot give a definitive reading and therefore has to “hedge”.

In fact, Dr. William Knaus, a noted intensive care researcher, concluded from a study of over 5000 patients in thirteen medical centers that the likelihood of patient survival was related more to the exchange of information among SICU team members than to other factors, such as the amount of specialized treatment used [17]. Thus, the team model appears to be a suboptimal approach to the distribution of expertise for intensive-care monitoring. Given these limitations and freedom from the organizational and economic constraints on human critical care teams, we decided not to replicate the team model in our design for the Guardian system.

On the other hand, the SICU monitoring task does present the design objectives associated with the single individual model:

1. Guardian must adapt to a dynamic environment. It must perceive asynchronously sensed patient data, reason about the patient’s dynamic condition, and perform therapeutic actions under appropriate patient conditions. It cannot afford to interrupt any of these functions while performing the others, but must perform all of them asynchronously and concurrently.

2. Guardian must perform multiple complex reasoning tasks. It must interpret perceived patient data, diagnose and explain patient data in terms of the underlying medical condition, predict the course of the patient’s condition, and dynamically

replan the patient's therapy.

3. Guardian must employ a range of reasoning strategies. It must use contextual information to focus its search among plausible diagnoses. It must use "quick and dirty" reasoning methods when more precise methods exceed the available time. It must fall back on first principles when faced with problems that fall outside its clinical knowledge.

4. Guardian must perform multiple tasks concurrently. 'For example, it must continue to interpret newly perceived patient data while diagnosing previously perceived signs and symptoms. In general, it must be prepared to perform variable subsets of its several tasks, as required by the SICU situation.

5. Guardian must emply an orderly and explainable reasoning process. It must establish goals and either pursue them to a satisfactory and timely conclusion or determine that they are no longer worth pursuing. It must produce persuasive explanations of its reasoning behavior and associated conclusions.

6. Guardian must coordinate loosely coupled reasoning activities. It must reconcile the results of related reasoning activities to produce an internally consistent patient model and treatment plan.

7. Guardian must produce globally coherent behavior. It must coordinate its perception and cognition to focus dynamically on the most critical aspects of the changing patient situation. It must coordinate its cognition and actions to address the most critical aspects of the patient situation in a timely fashion.

Accordingly, we conceive Guardian as a single intelligent individual. Unlike the individual members of the human critical care team, Guardian must integrate all relevant knowledge and skills and it must be dedicated to performing the SICU monitoring task vigilantly and continuously.

3. Guardian: A DAI Individual

3.1. System Overview

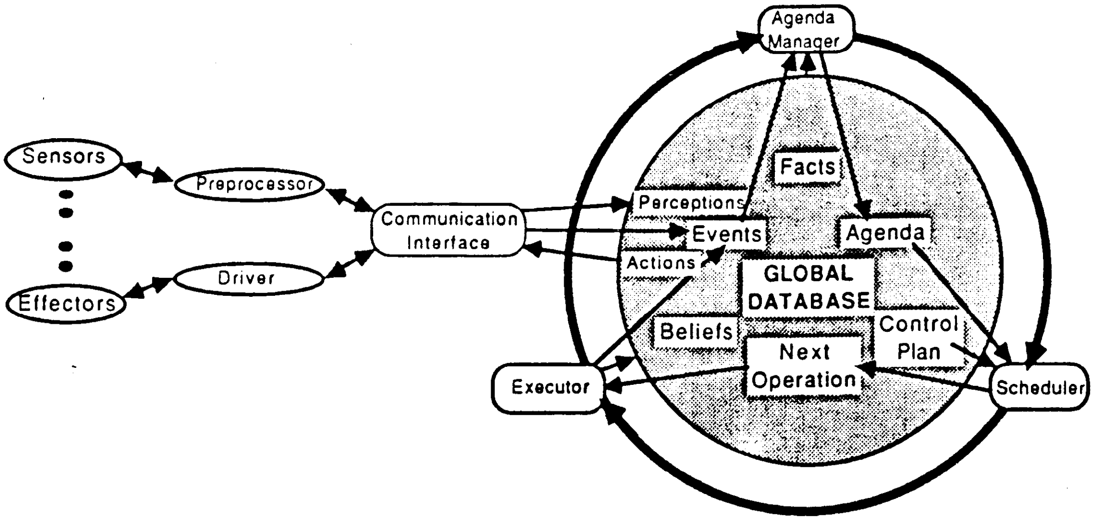


Figure 3-l: A Generic AIS Architecture.

We begin with the generic architecture for a DAI individual put forth in [7, 8, 9]. The architecture provides three general categories of function: (a) perception to acquire information from the environment; (b) action to affect entities in the environment: and (c) cognition to interpret perceived information, solve problems, make decisions, and plan actions. As illustrated in Figure 3-1, the architecture distributes the intelligence underlying these functions among three corresponding categories of agents: (a) multiple locally controlled perception agents; (b) multiple locally controlled action agents; and (c) a centrally controlled system of diverse cognitive agents.

Although the architecture provides for local control of perception and action agents, the cognitive system acts as the top-level controller for all of three categories of agents. It allocates its own limited resources among competing cognitive agents. It imposes attentional parameters on perception agents, which they incorporate in their own local control. It imposes task demands and performance parameters on action agents, which they incorporate in their own local control. To support interactions among cognitive, perception, and action agents, the model provides asynchronous communications along the paths indicated in Figure 3-1.

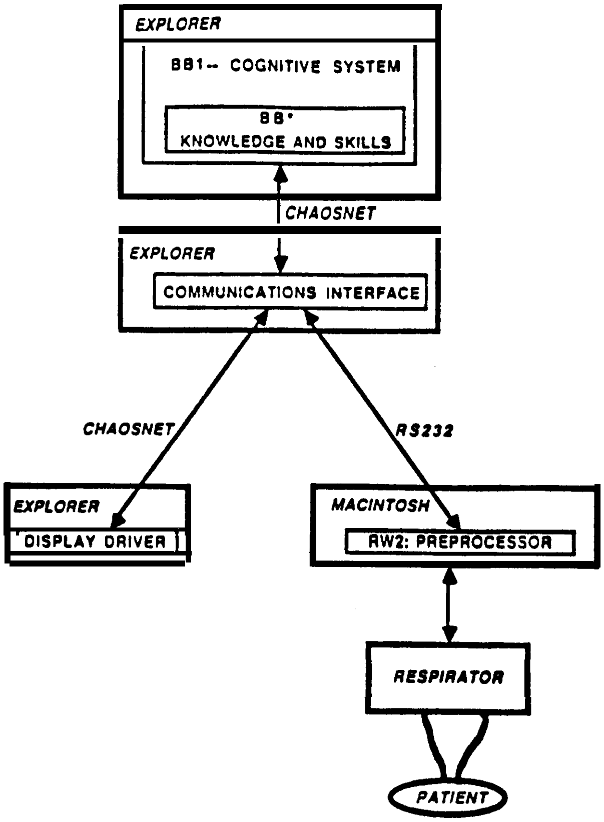


Figure 3-2: Guardian System Organization.

As illustrated in Figure 3-2, the current version of Guardian instantiates the generic architecture as: (a) a perception system for acquiring patient data; (b) a perception/action system for managing a user-oriented graphical display; and (c) a cognitive system for: focusing attention on relevant patient data, classifying perceived patient data, diagnosing observed signs and symptoms, reacting to urgent signs and symptoms, and explaining the structure/function mechanisms underlying the patient's condition. We expect future versions of Guardian to incorporate additional perception/action subsystems. The actual machines specified in Figure 3-1 are incidental to the current implementation.

The following sections describe generic subsystems for cognition, perception, and action and their instantiations in the current Guardian system

3.2. The Cognitive System

The cognitive system, which is framed within the BB1 architecture [10, 14], is responsible for all knowledge-based reasoning required to perform the overall task. In Guardian's case, it must interpret, diagnose, predict, and explain the patient's condition, and plan therapeutic actions. It must dynamically focus its own limited computational resources on the most important and urgent of these tasks and it must focus its subordinate perception and action agents on important patient data and therapeutic actions.

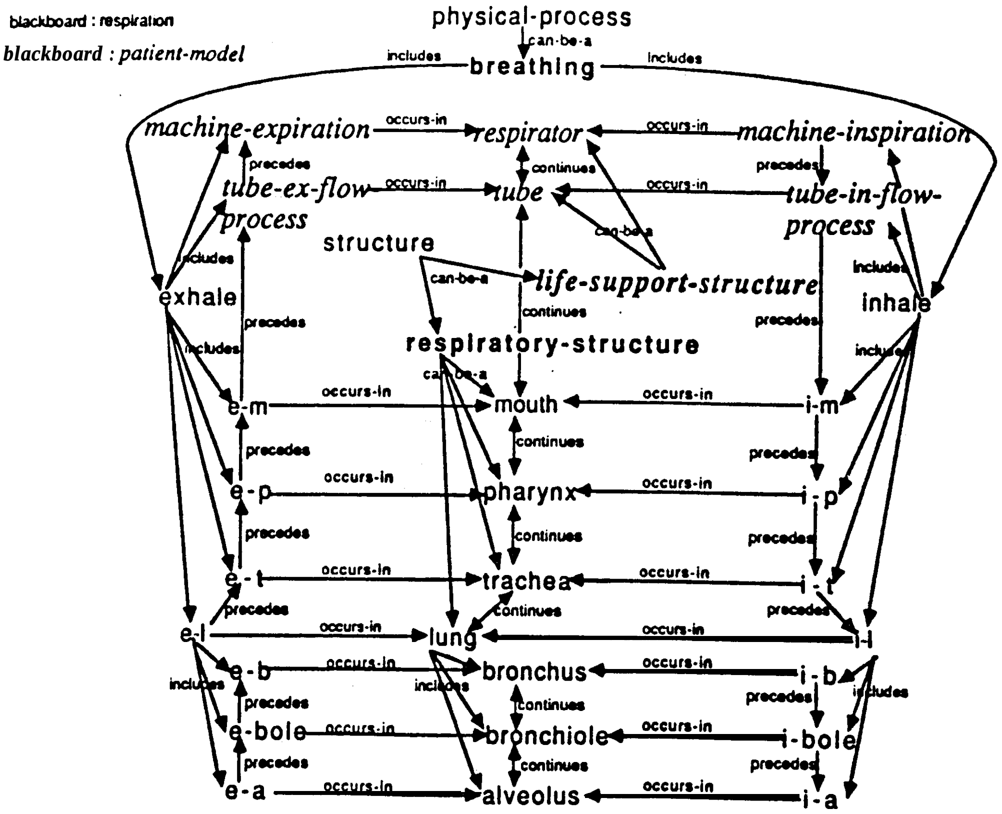


Figure 3-3: Some of Guardian's Knowledge of the Anatomy and Physiology of the Human Respiratory System

As illustrated in Figure 3-1, BB1 represents all knowledge in a *globally accessible knowledge base.* It uses a conceptual network representation [12], which provides predefined architectural concepts, such as operations, events, perception and action buffers, control plans, strategies, facts, cognitive skills, etc. It also provides an editor for building application-specific modules representing factual knowledge and cognitive skills. For example, Figure 3-3 illustrates some of Guardian's knowledge of the anatomy and physiology of the human respiratory system. Although BB1 represents all knowledge declaratively, cognitive skills embody performance knowledge for particular tasks and may be viewed as agents in the DAI model. In fact, each cognitive skill may include a number of knowledge sources, each of which defines the triggering conditions for one of the operations involved performing in the associated task. Each knowledge source may be viewed as a smaller-grained DAI agent.

The current Guardian system includes these factual knowledge modules and cognitive skills modules:

- Bio-Facts contains factual knowledge of the normal and abnormal anatomy and physiology of the respiratory, circulatory, pulmonary exchange, tissue exchange, and tissue metablic systems (see Figure 3-3).

- Clinical-Facts contains Bayesian belief networks relating common respiratory signs and symptoms to likely underlying faults and relating likely faults to standard treatments.

- Generic-Systems-Facts contains factual knowledge of the normal and abnormal structure and function of generic flow, diffusion, production, and consumption system models.

- Classify-Skill contains performance knowledge for classifying input data as instances of known categories of normal/abnormal parameters and parsing them into appropriate temporal episodes.

- Assoc-Skill contains performance knowledge for using belief networks to diagnose observed signs and symptoms.

- ICE-Skill contains performance knowledge for using generic system models to diagnose and explain the faults underlying observed signs and symptoms in particular systems.

- React-Skill contains performance knowledge for using association networks to generate standard treatments for commonly diagnosed faults.

- Backlog-Skill contains performance knowledge for managing dynamic imbalances in input data rates and cognitive load.

As these descriptions suggest, none of the knowledge modules is specific to Guardian or the SICU monitoring application. For example, Bio-Facts and Clinical-Facts could be used in a variety of medical and biological applications. Generic-Systems-Facts could be used in any domain involving the designated types of systems. All of the skills modules are domain-independent.

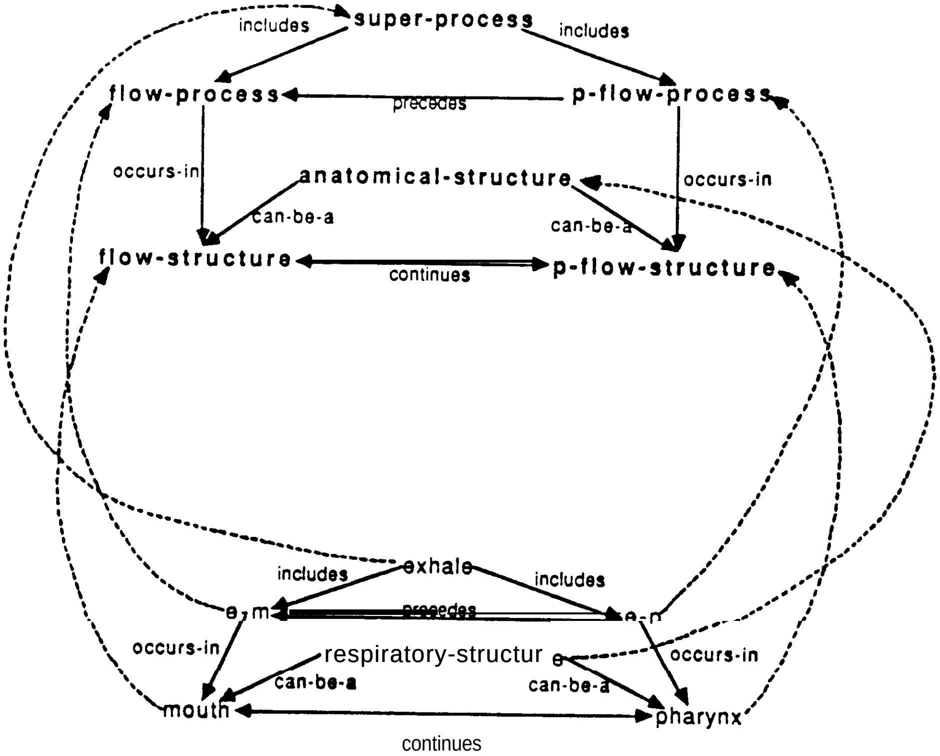


Figure 3-4: Integration of Knowledge from Two Modules: Bio-Facts and Generic-Systems-Facts.

BB1 allows the user to construct knowledge modules independently, load them in different combinations for development purposes, and selectively reuse them in other application systems. Loading a set of related modules together in BB1 integrates them in a seamless conceptual network, as illustrated in Figure 3-4. Information in the network is available for use in any cognitive operation, regardless of its module of origin. Thus, for example, operations originally defined in Classify-Skill and ICE-Skill both use information originally defined in Bio-Facts.

BB1 iterates a three-step reasoning cycle: (1) *the agenda manager* identifies cognitive operations (associated with any known cognitive skill) that are applicable to recent perceptual inputs or other changes to the knowledge base: (2) from these, the *scheduler* chooses the operation that best matches the current *control plan* specified in the knowledge base; and (3) the executor executes the chosen operation, making associated changes to the knowledge base.

To perform a task for which it has a known skill, BB1 begins by posting a decision to do so on the control plan (see Figure 3-l). On subsequent cycles, the BB1 scheduler chooses *control operations* that construct an appropriate task strategy, as well as *base-level operations* that perform the task in accordance with the constructed strategy.

BB1 allows a system to decide to apply multiple skills to multiple tasks concurrently by posting corresponding decisions on the control plan. BB1 interleaves the operations of concurrent tasks on successive cycles. For example, applying Classify and Assoc skills concurrently, Guardian interleaves operations that classify new patient data and operations that diagnose previously classified patient data. That way, Guardian can respond immediately to newly observed signs and symptoms even though it has not finished diagnosing previously observed signs and symptoms.

To focus its attention strategically among concurrent tasks, BB1 allows a system to record higher-level control strategies on the control plan along with its task-specific strategies. For example, if Guardian were diagnosing a critical sign requiring immediate treatment, it might decide to focus on Assoc operations and temporarily ignore potential Classify operations except for those triggered by patient data directly relevant to the ongoing diagnosis.

BB1 has an independent *communication interface* to mediate data exchange between the cognitive system and various perception/action agents [15]. As illustrated in Figure 3-1, the communication interface continuously monitors physical input ports from all perception agents. It sorts input data into appropriate logical input buffers in the global knowledge base. The agenda manager uses data in input buffers, along with other internally generated events, to trigger cognitive operations. In the current version of Guardian (Figure 3-2), the communication interface relays input data from the *Mediator* and the *Display Driver* to various logical input buffers. Conversely, BB1 operations can place descriptions of intended actions in logical output buffers, from which the communication interface retrieves them and sends them to physical output ports for appropriate action agents. In the current version of

Guardian, the communications interface relays actions from logical output buffers to the Mediator and the Display Driver. Thus, the communication interface shields the cognitive system from both the details of device-specific communication protocols and, more importantly, from I/O interference in its own performance. It similarly shields perception/action agents from the details of BB1 data structures and from interference in their own performance. Although the communication interface can run as a background process on the BB1 machine, we run it on a separate machine to get the overall processing speed required by Guardian.

3.3. Perception/Action Systems

Perception/action systems perform computations for selective perception and controlled action execution under top-level control instructions from the cognitive system. In both cases, intervening agents mediate the exchange of data between the cognitive system (via the communications interface) and peripheral sensor/effector agents (see Figure 3-l).

In the case of perception, *preprocessors* monitor peripheral sensors, translate and filter data according to instructions from the cognitive system, and send the results to the cognitive system. The current version of Guardian has a single preprocessor, the Mediator, which manages patient data from multiple sensors on the respirator. (For development purposes, we replace the actual respirator and patient with a simulation.) Guardian's Backlog skill sends new filters to the Mediator to modify input data rates in response to changes in Guardian's cognitive load, its focus of attention, and sensed data rates. Thus, perceptual agents enable a system to attend selectively to available data so as to monitor the environment as closely as possible, avoid perceptual overload, and minimize interference with other cognitive activities.

In the case of action, *drivers* receive action descriptions from the cognitive system and control action execution on peripheral effectors. The current version of Guardian has a single driver, the Display Driver, which controls a graphical display of Guardian's changing interpretation of the patient's condition. The Display Driver also receives input from the user and relays that to the cognitive system. Thus, action agents enable a system to control execution of complex action programs, while minimizing interference with cognitive activities.

4. Guardian's Performance **on a Typical** SICU Scenario

4.1. Monitoring a Stable Ventilator-Assisted Patient

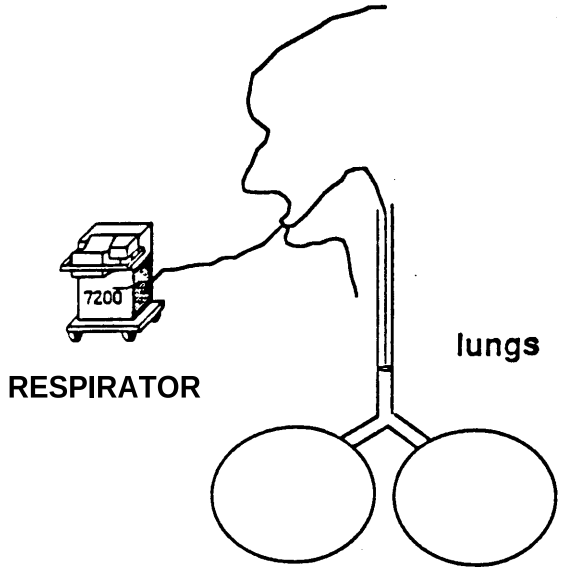


Figure 4-1: Stable Ventilator-Assisted Patient.

The scenario begins with a stable ventilator-assisted patient. (As shown in Figure 3-2, we simulate the patient and ventilator for development purposes.) As illustrated in Figure 4-1, the ventilator delivers a prescribed volume of air to the patient's two lungs on each breath. Two important measured parameters are the *peak pressure* applied by the ventilator and the *tidal* volume of air actually received by the patient on each breath. In the normal situation, these two parameters vary normally about the prescribed values.

As illustrated in Figure 4-2, Guardian's Mediator asynchronously receives every sensed data point for peak pressure and tidal volume. The Mediator applies "threshold filters" specified by Backlog and relays only data points that differ from their predecessors by the specified percentage. These data points are marked by vertical bars in Figure 4-2. The communications interface receives relayed data points and inserts them into appropriate logical input buffers, as illustrated in Figure 4-3, where they are available to Guardian's cognitive skills. Thus, Guardian's sensors, perceptual preprocessor, and communications interface function in parallel to provide selective perception of asynchronously occurring patient data.

Each new input data point triggers a Classify operation. When executed, these operations assign data points to value categories and to old or new temporal episodes of those value categories. Given its definition of threshold filters, Guardian interpolates between perceived

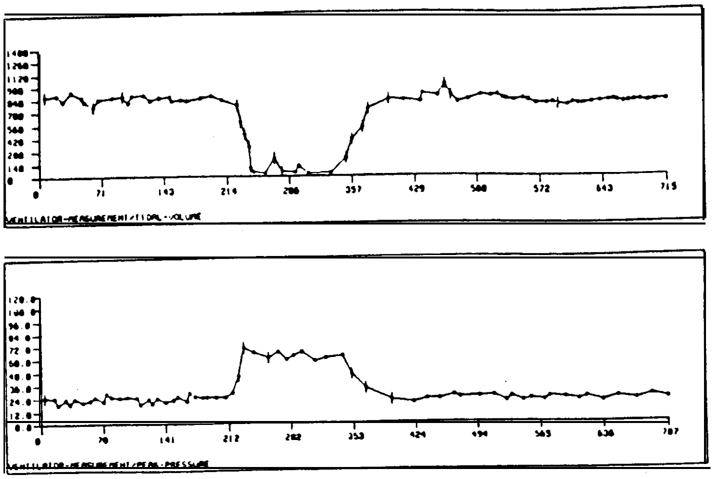
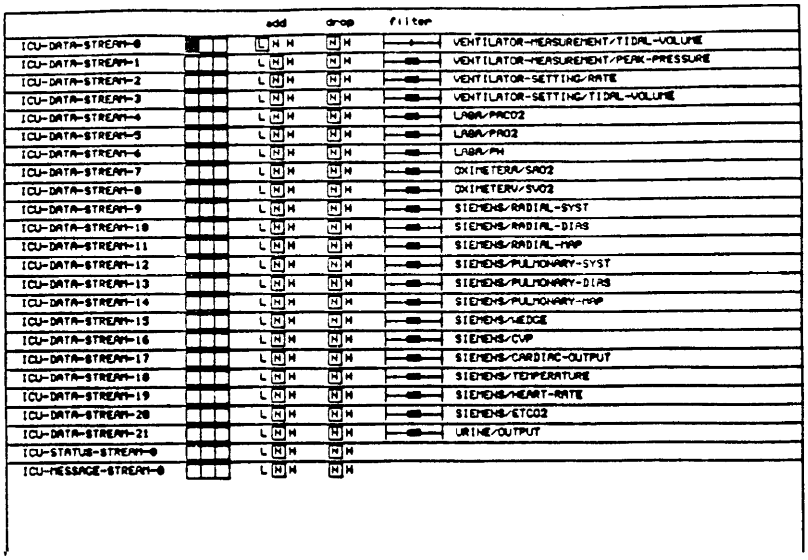


Figure 4-2: Sensed Data as Filtered by the Mediator



	add	drop	filter	
ICU-DATA-STREAM-0	L N H	N H		VENTILATOR-MEASUREMENT/TIDAL-VOLUME
ICU-DATA-STREAM-1	L N H	N H		VENTILATOR-MEASUREMENT/PEAK-PRESSURE
ICU-DATA-STREAM-2	L N H	N H		VENTILATOR-SETTING/RATE
ICU-DATA-STREAM-3	L N H	N H		VENTILATOR-SETTING/TIDAL-VOLUME
ICU-DATA-STREAM-4	L N H	N H		LABA/PACO2
ICU-DATA-STREAM-5	L N H	N H		LABA/PAO2
ICU-DATA-STREAM-6	L N H	N H		LABA/PH
ICU-DATA-STREAM-7	L N H	N H		OXIMETERA/SAO2
ICU-DATA-STREAM-8	L N H	N H		OXIMETERV/SVO2
ICU-DATA-STREAM-9	L N H	N H		SIEMENS/RADIAL-SYST
ICU-DATA-STREAM-10	L N H	N H		SIEMENS/RADIAL-DIAS
ICU-DATA-STREAM-11	L N H	N H		SIEMENS/RADIAL-MAP
ICU-DATA-STREAM-12	L N H	N H		SIEMENS/PULMONARY-SYST
ICU-DATA-STREAM-13	L N H	N H		SIEMENS/PULMONARY-DIAS
ICU-DATA-STREAM-14	L N H	N H		SIEMENS/PULMONARY-MAP
ICU-DATA-STREAM-15	L N H	N H		SIEMENS/WEDGE
ICU-DATA-STREAM-16	L N H	N H		SIEMENS/CVP
ICU-DATA-STREAM-17	L N H	N H		SIEMENS/CARDIAC-OUTPUT
ICU-DATA-STREAM-18	L N H	N H		SIEMENS/TEMPERATURE
ICU-DATA-STREAM-19	L N H	N H		SIEMENS/HEART-RATE
ICU-DATA-STREAM-20	L N H	N H		SIEMENS/ETCO2
ICU-DATA-STREAM-21	L N H	N H		URINE/OUTPUT
ICU-STATUS-STREAM-0	L N H	N H		
ICU-MESSAGE-STREAM-0	L N H	N H		

Figure 4-3: Input Buffers and Associated Filters Managed by Backlog

data points of a given value category to model continuous temporal episodes (see Figure 4-4). Thus, Guardian incrementally builds a history of asynchronously perceived patient data.

While classifying newly perceived input data, Guardian continues to perceive new data and monitor its input data rates. If new data of a given type arrive too quickly, they will overflow the input buffer and Guardian will build an incomplete patient history. If new data arrive too slowly, Guardian will build the patient history at an unecessarily low precision. In an effort to perceive sensed data at the maximum rate Guardian can handle, Backlog monitors activity in all input buffers and adjusts the filter thresholds used by the Mediator as necessary. The right

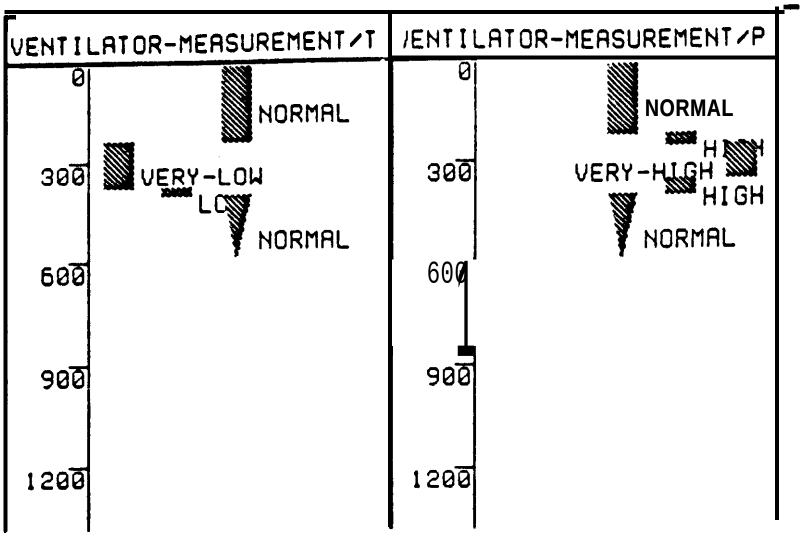


Figure 4-4: History of Patient Data Constructed by Classify

side of Figure 4-3 indicates a filter threshold for each input buffer, as established by Backlog.

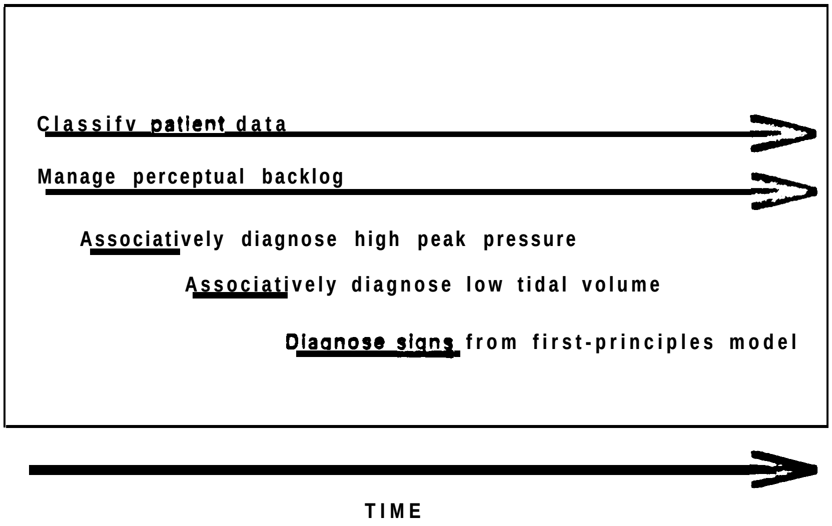


Figure 4-5: Illustrative Control Plan

Guardian performs the Classify task and the Backlog task concurrently. As illustrated in Figure 4-5, it makes separate control decisions for each task and additional control decisions for allocating computational resources between them. Guardian interleaves component operations for the two tasks on successive BB1 reasoning cycles according to these control strategies.

4.2. **Diagnosing and Explaining** Time-Varying **Signs and Symptoms**

After a period of monitoring a stable patient, Guardian notices that something has gone wrong. Classify notices abnormally high value; for the parameter peak pressure (see the right side of Figure 4-4), triggering both Assoc and ICE. **To** diagnose this sign quickly, however, Guardian makes a control decision to apply **ASSOC'S** highly efficient (but less explicit and less complete) associative reasoning, in favor of ICE's computationally expensive model-based reasoning.

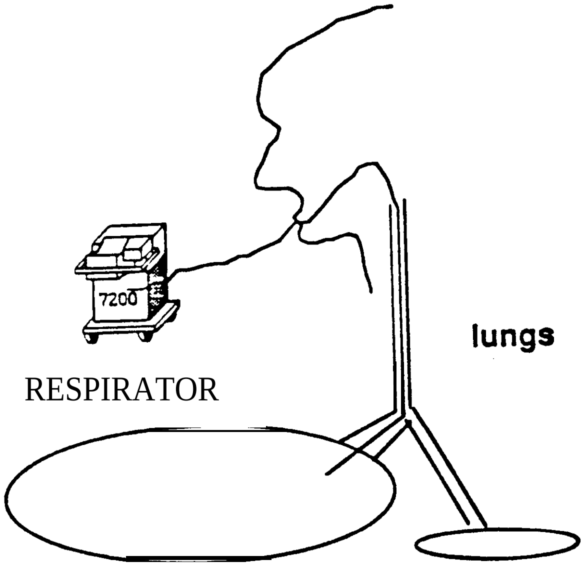


Figure 4-6: Increased Peak Pressure Caused by One-Sided Intubation

Assoc diagnoses "one-sided in tubation." As illustrated in Figure 4-6, when the respirator tube slips into one of the patient's bronchi, the ventilator delivers the prescribed volume of air to only one lung, causing peak pressure to rise.

Because SICU monitoring is a dynamic situation, Guardian must continue to monitor new patient data and keep the patient model up to date while it performs its diagnosis. In fact, it must-be prepared to revise its diagnosis in light of new patient data. Accordingly, as illustrated in Figure 4-5, Guardian decides to perform Classify and Assoc tasks concurrently. It so happens that, while Assoc is diagnosing "one-sided intubation," Classify records a new sign, low tidal volume. This new sign triggers both Assoc and ICE and, again, Guardian prefers the more efficient Assoc method.

Taking into account the new sign, Assoc revises its diagnosis in favor of "kinked tube." As illustrated in Figure 4-7, a kinked tube prevents the ventilator from delivering air past the point of the kink. As a result, peak pressure rises dramatically and tidal volume drops to zero.

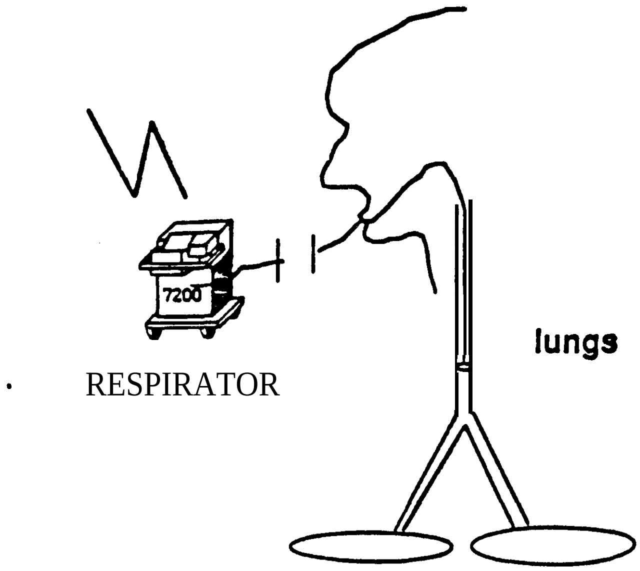


Figure 4-7: Increased Peak Pressure and Decreased Tidal Volume Caused by a Kinked Tube

4.3. Falling Back on First Principles

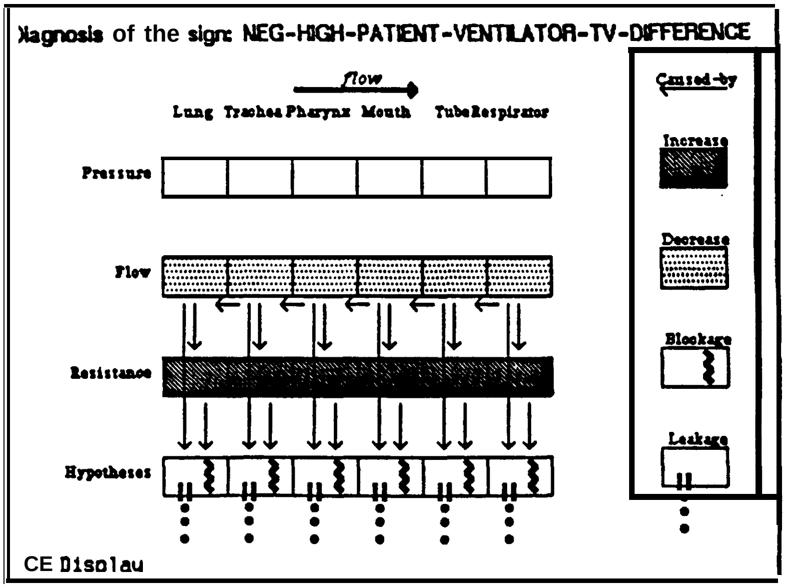


Figure 4-8: ICE Hypothesizes Plausible Problems underlying the Decrease in Tidal Volume

Having hypothesized “kinked tube” with a stable, high probability, Guardian now learns (presumably from the nurse) that, in fact, there is no kink in the respirator tube. Without additional relevant patient data, Assoc cannot suggest alternative diagnoses. However, ICE can apply its knowledge of potential faults in generic flow systems, along with its knowledge of the anatomy and physiology of the respiratory system, to hypothesize plausible problems underlying the observed patient data.

As illustrated in Figure 4-5, Guardian decides to perform the ICE task concurrently with other tasks. Classify continues to integrate new input data into the patient model and Backlog continues to monitor input data rates. In addition, Backlog notices the decision to perform an ICE task, anticipates ICE's high demand for computational resources, and instructs the Mediator to increase its filtering threshold to conserve resources. Figure 4-8 illustrates ICE's hypotheses regarding the observed decrease in tidal volume.

5. Preliminary Conclusions

We have shown how our design of Guardian instantiates the single individual model. In the scenario presented above, Guardian exploits all seven of the design objectives discussed above:

1. To adapt to a dyanamic environment, Guardian exploits locally controlled agents to achieve asynchronous and concurrent: (a) perception, to learn about the patient's changing condition; (b) cognition, to interpret patient data, build a dynamic model of the patient, and decide what actions to take; and (c) action, to inform critical care staff of its observations, reasoning, and conclusions.

2. To perform loosely coupled reasoning tasks, Guardian exploits sets of functionally independent reasoning agents for the following tasks: focus of attention, data classification, associative diagnosis, and model-based diagnosis and explanation.

3. To exploit a range of reasoning strategies, Guardian dynamically controls its application of task-specific reasoning agents in accordance with the changing situation. For example, it typically relies upon the more efficient Assoc method of diagnosis, but falls back on the model-based ICE method when its clinical knowledge fails.

4. To concurrently perform multiple reasoning tasks, Guardian constructs control plans for each of them and interleaves their respective operations. For example, it almost always interleaves Classify and Backlog tasks with whatever other tasks it may be performing.

5. To insure an orderly and explainable reasoning process, Guardian constructs higher-level control plans to allocate computational resources among the reasoning agents involved in different concurrent reasoning tasks. For example, under time stress,

Guardian may allocate most of its resources to diagnosis and action planning.

6. To coordinate loosely coupled reasoning tasks, Guardian places all intermediate reasoning results in a globally accessible knowledge base. For example, both Assoc and ICE are triggered by any perceived data classified as abnormal.

7. To achieve global coherence of system behavior, Guardian imposes centrally determined control parameters on perception and action agents, as well as on its many cognition agents. For example, Backlog adjusts the Mediator's perceptual filters whenever input buffers overflow or underflow and whenever Guardian undertakes or completes a computation-intensive reasoning task.

In addition to the scenario presented in this paper, Guardian exploits these capabilities to handle several other SICU scenarios involving other respiratory and circulatory problems.

To expand the set of problems Guardian can handle, we must increase its range of perceptual inputs, its repertoire of facts and skills, and its capabilities for therapeutic and communications actions. We expect these developments to increase Guardian's dependence on our model of the DAI individual. The more complex and variable the environment, the more essential it is that Guardian perceive, reason about, and act upon that environment asynchronously and concurrently so as to adapt to it in a timely fashion. The broader the set of problems Guardian must handle, the more different skills it must have and the more flexibly it must determine its strategic approach to a given problem. The more knowledge and skills Guardian has, the more important it is to apply variable subsets of those skills concurrently in order to exploit them fully and respond promptly to aynchronous events. The more tasks Guardian performs, the more carefully it must control its reasoning to insure coherence and explainability and the more information it must provide'in a globally accessible form. Finally, the more demands and opportunities Guardian has for perception, cognition, and action, the more effectively it must coordinate all three of these functions in order to insure global coherence.

Perhaps the most important distinguishing feature of the DAI individual is its emphasis on central coordination of a hierarchy of locally controlled agents. The architecture we have adopted for Guardian is designed to provide a foundation for central coordination. However, efforts to extend Guardian will provide essential empirical evidence regarding the adequacy of the architecture and the achievability of its design objectives for significant tasks.

References

[1] Cammarata, S., McArthur, D., and Steeb, R.
Strategies of coopration in distributed problem solving.
Proceedings of the Eighth International Joint Conference on Artificial Intelligence , 1983.

[2] Corkill, D.D., Lesser, V.R., and Hudlicka, E.
Unifying data-directed and goal-directed control: An example and experiments.
Proceedings of the AAAI :143-147, 1982.

[3] Davis, R., and Smith, R.G.
Negotiation as a metaphor for distributed problem solving.
Artificial Intelligence 20:63-109, 1983.

[4] Durfee, E.H., Corkill, D.D., and Lesser, V.R.
Cooperation through communication in a distributed problem solving network.
In Huhns, M. N. (editor), *Distributed Artificial Intelligence,* . London: Pitman, 1987.

[5] 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.

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

[7] Hayes-Roth, B.
A multi-processor interrupt-driven architecture for adaptive intelligent systems.
Technical Report KSL-87-31, Stanford, Ca.: Stanford University, 1987.

[8] Hayes-Roth, B.
Dynamic control planning in adaptive intelligent systems.
Proceedings of the DARPA Knowledge-Based Planning Workshop , 1987.

[9] Hayes-Roth, B.
Making intelligent systems adaptive.
In K. VanLehn (editor), *Architectures for Intelligence,* . Lawrence Erlbaum, 1988.

[10] Hayes-Roth, B. and Hewett, M.
Building systems in the BB1 architecture.
In R. Engelmore and A. Morgan (editor), *Blackboard Systems,* . London: Addison-Wesley, 1988.

[11] Hayes-Roth, B., Buchanan, B.G., Lichtarge, O., Hewett, M., Altman, R., Brinkley, J., Cornelius, C., Duncan, B., and Jardetzky, 0.
PROTEAN: Deriving protein structure from constraints.
Proceedings of the AAAI , 1986.

[12] Hayes-Roth, B., Garvey, A., Johnson, M.V., and Hewett, M.
A layered environment for reasoning about action.
Technical Report KSL-86-38, Stanford, Ca.: Stanford University, 1986.

[13] 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.

[14] Hayes-Roth, B.
A blackboard architecture for control.
Artificial Intelligence Journal 26:251-321, 1985.

[15] Hewett, M., and Hayes-Roth, B.
Real-Time I/O in Knowledge-Based Systems.
In *Proceedings of the AAAI88 Workshop on Blackboard Systems.* 1988.

[16] Hewitt, C.
Viewing control structures as patterns of passing messages.
Artificial Intelligence :323-364, 1077.

[17] Knaus, W.A., Draper, E.A., Wagner, D.P., and Zimmerman, J.E.
An evaluation of outcome from intensive care in major medical centers.
, .

[18] Rosenschein, J.S., and Genesereth, M.R.
Deals among rational agents.
Proceedings of the Ninth International Joint Conference on Artificial Intelligence , , 1985.

[19] Shastri, L.
A connectionist encoding of semantic networks.
In Huhns, M. N. (editor), *Distributed Artificial Intelligence*, . London: Pitman, 1987.

[20] 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.