N91-24044

Distributed, Cooperating Knowledge-Based Systems

Walt Truszkowski
Head, Automation Technology Section
Code 522.3
Goddard Space Flight Center
Greenbelt, MD 20771

Abstract

This paper addresses some current research in the development and application of distributed, cooperating knowledge-based systems technology. The focus of the current research is the spacecraft ground operations environment. The underlying hypothesis is that, because of the increasing size, complexity, and cost of planned systems, conventional procedural approaches to the architecture of automated systems will give way to a more comprehensive knowledge-based approach. A hallmark of these future systems will be the integration of multiple knowledge-based agents which "understand" the operational goals of the system and cooperate with each other and the humans in the loop to attain the goals. The current work includes the development of a reference model for knowledge base management, the development of a formal model of cooperating knowledge-based agents, the use of a testbed for prototyping and evaluating various knowledge-based concepts, and beginning work on the establishment of an object-oriented model of an intelligent end-to-end (spacecraft to user) system. The paper will present an introductory discussion of these activities, highlight the major concepts and principles being investigated and indicate their potential use in other application domains.

Situations

Before beginning a discussion of the specific R&D activities in the area of distributed knowledge-based systems which we have been pursuing, let us take a small digression and consider the following "situations". These constitute a small sample of problems whose solutions are or will be supported in the future by access to cooperating knowledge-based systems (the computerized variety). The purpose of this initial digression is to make clear the belief that the research which is being pursued in the area of distributed cooperating knowledge-based systems has broad applicability and will eventually touch many differing aspects of human life.

1. **Distributed Knowledge Base Management:** The inputs from several distributed knowledge sources, some with access to real time sensor data, need to be collected, analyzed, checked for consistency, checked for completeness, properly annotated, and archived for easy access by researchers.

2. **Spacecraft Control:** Elements of a ground system need to be monitored and controlled in support of a space mission.

3. **Traffic Control:** There is need for an automatic rerouting and adjustment of the timing of traffic lights through a very heavy traffic zone due to an unforeseen situation.

4. **Robot Operations:** There is need for the coordination of robots which are to be engaged in carrying out a potentially hazardous experiment in a highly automated laboratory.

5. **Medical Emergency:** A team of medical experts is needed to diagnose a mysterious ailment.

6. **Utilities Control:** Life sustaining resources need to be monitored and dynamically adjusted onboard a space station in response to changing crew needs.

7. **Software Engineering:** Within the context of a knowledge-based software engineering environment agents are needed to give advise to a designer on the best match between system requirements and performance criteria and the stock of reusable/tailorable system components available.

8. **Factory Operations:** The total operations cycle of a mining plant needs to be automated for this factory being built on the moon.

9. **Electronic Library:** The facilities of a library including access to bibliographic citations, article abstracts, electronically-stored books and journal, articles, need to be made available to scholars doing research.

Each of these situations deals with a different domain, involves different activities, has different goals and objectives, addresses different problems, and utilizes different resources in accomplishing desires results. However, at an appropriate level of abstraction, automation-assisted solutions to the problems arising in these different domains can be seen to involve very sophisticated knowledge management issues and could be viewed as a tailoring of a common framework for instantiating, activating, and using teams of distributed, cooperating, knowledge-based agents. This, at least, is the hypothesis which is being formulated and being put forth for evaluation.

The rest of this paper focuses on an overview of two major research activities, i.e., the development of a reference model for knowledge base management and a formal model of cooperating of knowledge-based systems and the prototype application of some of the research results to the specific area of concern to us at the Goddard Space Flight Center, namely control center operations systems for near-earth unmanned scientific spacecraft.

Current Research Activities

Automation of control center operational systems, related to situation 2 above, is currently realized through application of single expert systems to support individual subsystem functions. State-of-the-art research in artificial intelligence and the cognitive sciences now hold that this one-to-one mapping between system function and automating agent is an inappropriate paradigm, in the extreme, which will have limited usefulness as system complexities increase. What is needed to support higher levels of automation in such systems is the use of multiple autonomous agents cooperatively providing-for and supporting desired system behaviors. This is the long-term solution to the problem of providing operational knowledge-based spacecraft control centers of the future. To support this evolution to an "intelligent" control center, our work in this area has been concentrated in the areas of knowledge base management and formal models of cooperating knowledge-based agents. A testbed for demonstrating distributed knowledge-based technologies in a spacecraft command/control environment has been established.

Knowledge Base Management

In our research, a Knowledge Base Management System (KBMS) is defined as a utility for supporting the life cycle of acquiring, refining, using, and maintaining large-scale distributed knowledge bases [Ref. 1,2,3]. The basic drivers for the KBMS research are the expectations that future autonomous systems, used in operational control centers and ground systems, will be: (1) both knowledge- and data-based, (2) distributed yet cooperating and integrated, (3) potentially large-scaled , and (4) long-lived and needing to be maintained and updated regularly. A KBMS is intended to support these drivers. Our analysis of the functional requirements for a KBMS have not been driven by general domain-independent considerations. Rather our guidance has come from what we readily expect the future (long range) control center architectures will require for operations. Some of these operational requirements are 1. a framework is needed to support the addition, connection, and operation of distributed cooperating knowledge-based systems in a rapid reliable and asynchronous fashion: 2. rules, plans, schedules, and knowledge must be continually acquired and updated from a broad array of sources with a minimum amount of human intervention; 3. knowledge verification is essential; 4. intelligent interfaces between knowledge bases and realtime sources of sensor data need to be supported; 5. a wide variety of human-factored human/machine interfaces, interaction techniques, tutoring aids, and utilities must be readily available for supporting the range of personnel which will be developing, using, or maintaining the system. To help give us an overview of the major issues to be addressed in the engineering of a comprehensive KBMS work was begun on the development of a KBMS Reference Model. In our usage of the phrase, a "reference model" refers to a map of the activities, functions to support the activities, data/information flows, interfaces/interactions which need to be supported for the general case of any KBMS. Figure 1. illustrates the current version of the KBMS Reference Model in development. The purpose of the model is to aid in identifying the major concepts associated with knowledge base management and to put these concepts in proper perspective.

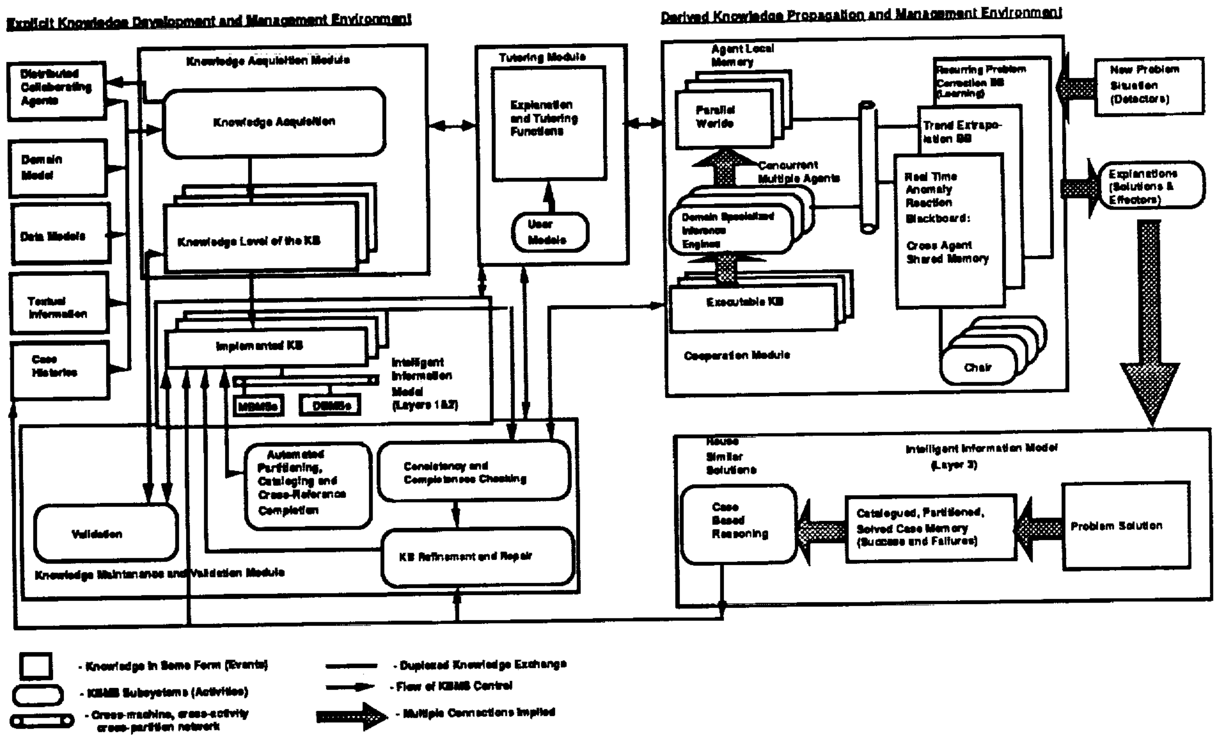


Figure 1. Reference Model for a KBMS

This reference model identifies five major functional components . These are the knowledge acquisition, knowledge maintenance and validation, tutoring, cooperation framework, and intelligent information model.

Each of these components requires a set of capabilities to carry out its role in the overall KBMS structure. An indication of these capabilities follows:

• **Knowledge Acquisition** - learn/induce, optimize, analogize, associate, specialize, generalize, experiment, ask,attach principles, annotate;

• **Knowledge Maintenance** - criticize, reward, test, trap, influence, debias, clarify, repair, refine, correct, update;

• **Tutoring** - model, display, advise, quiz, hint, explain, adapt, help, teach, monitor, oversee;

• **Cooperation** - distribute control, instantiate work breakdown structures, negotiate, broadcast, direct cast, poll, plan, dispatch, control access, propagate, search, synthesize, communicate;

• **Information Model** - cluster/abstract, subsume, access/store, sample, meta-reason, index, catalogue, model management.

This is not a complete list. Other activities may be gleaned from the reference model diagram. What is clear, even at this simplified level of description, is that the overall concept of a KBMS is a complicated and complex one involving technologies from software engineering, artificial intelligence, and the cognitive sciences.

ORIGINAL PAGE IS
OF POOR QUALITY

Work is continuing on the further refinement of the reference model. It is helping us keep a good perspective on the state-of-the-art of KBMS research. One specific problem which we are pursuing is that associated with accessing and using knowledge from heterogeneous knowledge sources. The ability to fuse knowledge from these sources into an integrated presentation to human operators is a requirement for nearer term automated ground systems operations.

Formal Modelling of Distributed Cooperating Knowledge-Based Systems

Concurrent with our work on the KBMS model we have been studying some of the mechanisms required to support cooperation among knowledge-based systems (agents) [Ref. 4,5]. Currently the use of knowledge-based components, i.e. expert systems, is rapidly becoming state-of-practice in NASA's operational ground systems. Currently ground/space network operations and the monitoring of spacecraft status data are supported in this manner. As their number and usage increases in support of operational systems it is apparent that a new system architectural concept will evolve. This new architecture will be such so as to fully exploit the computational power of these knowledge-based components and fully integrate these components into an efficient and effective operational system. The hypothesis which we share with a growing number of researchers and developers is that the new architectures will need to support distributed , cooperating knowledge-based components. As a starting point for our investigations into this area we began on the development of a logical model of cooperating knowledge-based systems. We saw the role of the model as a tool to facilitate technological research and system planning. Development of this model is continuing today. What follows is an introduction to our early work in this area.

In our analysis of what it would take to develop a team of cooperating computer-based agents to support a highly automated ground system for satellite control, several general characteristics were identified. These are:

1. There should be a logical and physical distribution of the total knowledge of the team among the agents which comprise it. **(Knowledge Partition)**

2. Each agent should have some internal model of some of the other agents in the team. **(Agent Awareness)**

3. Cooperation among agents requires communication among agents. **(Inter-agent Communication)**

4. Proper interpretation of information shared between agents may require that the agent receiving the information have access to the information context in the sender agent. **(Shared Contexts)**

5. The cooperating agents should be able to partition the problem to be solved. **(Problem Partitioning)**

6. The agents should not only be able to communicate, they should also be able to coordinate the team's activity. **(Agent Coordination)**

7. The agents in the team and the team as a whole should be capable of adjusting performance in response to environmental changes. **(Performance Adjustment)**

8. A mechanism should exist for integrating new knowledge-based agents into the team. **(Integration)**

These general characteristics will be briefly discussed in the context of the ultimate paradigm of cooperative activity among knowledge-based agents: a team of humans jointly working on a problem. This is done to help clarify the ideas and to help contribute to their justification.

Knowledge Partition: There are several reasons for supporting the partitioning of knowledge. These include the observations that partitioning supports: 1. increased performance through parallelism, 2. enhanced extensibility and maintainability of the knowledge base, and 3. well defined mappings of the knowledge to the various internal knowledge structures. When forming a team of humans it is usually desirable that each bring a special knowledge to the group and not merely know what everyone else knows. We note that knowledge overlap is however necessary to support meaningful communication.

Agent Awareness: In order to work together agents must be aware of each other's existence and attributes, i.e., some model of some of the other agents. The working of a group of humans is greatly facilitated if each team member knows something of the knowledge and capabilities of the other members. For one thing it greatly facilitates the appropriate assigning of subproblems to be solved.

Inter-agent Communication: Cooperation among agents presumes some form of communication. Both synchronous and asynchronous communication capabilities are needed for effective and efficient operation of the team. A quick unscheduled message passing among human team members is quite a common occurrence in addition to regularly scheduled status reports.

Shared Contexts: Among communicating agents each agent must be able to interpret the meaning of the information it receives. In order to ensure this the agents must have the capability to ask for the context, including underlying assumptions, of a piece of information it has received from the sending agent. Each cooperating agent needs criteria for evaluating information it receives from other agents in terms of importance, certainty, and timeliness. Irrelevant, uncertain, or out of date information may distract an agent from promising lines of reasoning. Failure t sent or receive context information may cause an agent to overlook potentially important lines of reasoning. In team meetings of humans it is quite often the case that in analyzing a new piece of information the generator of the new information is asked how it came about and how should it be interpreted in the proper context.

Problem Partitioning: In order for a team of cooperating agents to work effectively, if at all, it is not only necessary for there to be some partitioning of knowledge but also the ability to partition the work to be done. As a problem is presented to the team there need to be mechanisms available to break the problem into smaller and smaller subproblems until there are good matches between subproblem size/complexity and agent capability. Mechanisms must also exist for the synthesis of partial results, from the subproblem solutions, into a solution of the original problem. The concept of breaking a task into subtasks and making assignments to various members is a normal method of operation for teams of humans.

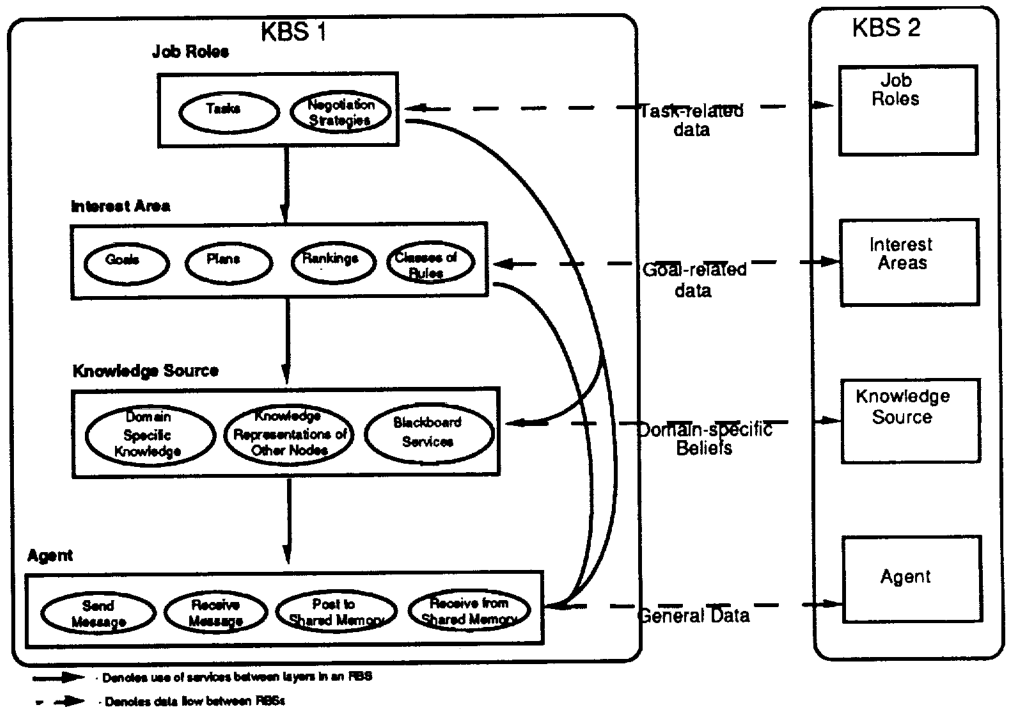
Agent Coordination: Partitioning of work ensures that each involved agent knows what it is supposed to do within the overall system. Models of other agents and communication among them ensure that the agents can interact in performing their tasks. However, synthesis of the subproblem solutions to achieve a global objective, the solution of the problem originally presented,, requires more. It requires an ability for the agents to coordinate their activities. Scheduling and control mechanisms are required to ensure the necessary degree of coordination is realized and maintained. Even very loosely-coupled teams of humans agree on the milestone schedule to be followed and may even identify one team member as the lead for a certain portion of the work.

Performance Adjustment: Depending on the nature of the problem presented, or the status of available resources, or the availability of expertise a team of agents may be required to modify its performance criteria in order to meet some overall performance objectives. Among human teams the discovery of unsuspected complexity in a portion of a problem may require regrouping and reassignments.

Integration: As new expertise is needed in both human and computer-based teams there needs to be a mechanism whereby new agents, or team members, are made a part of the team.

The reasonableness and operational semantics of these characteristics as they manifest themselves in our implementations are being evaluated.

The work on the formal modelling of autonomous agents is intended to assist in the planning, specification, development, and verification of control centers involving distributed cooperating knowledge-based systems. The current model describes a community of cooperating rule-based systems at four layers of increasing capability: (1) communicating agents - with no assumption of intelligence or rule-based capability, (2) belief-sharing knowledge sources - where a knowledge source is an agent specializing in a specific domain and a belief is any data arrived at through an inference process, (3) goal sharing interest areas - where the first indications of goal-directed reasoning appears, and (4) task sharing job roles - the fourth level of the model where overall system goals are decomposed into tasks which are allocated among various job roles. At this final level, cooperation is most fully achieved. The following figure gives an overview of the model as it is now conceived.



The CRBS Logical Model identifies: layers of abstraction needed in KBSs to support cooperation, services and information available at each layer, and information flow between layers.

Figure 2. Logical Model

Implementation of this model is proceeding. The implementation is in C^{++} on a Macintosh II computer. An adaptation of the Contract Net protocol is being used to support the implementation. A network of a number of agents, only limited by the amount of finite memory, can be simulated in this proof-of-concept development. In the current implementation agent behaviors are determined by the protocol defined by the logical model, the Contract Net protocol, and by scripts read in and executed by the agents themselves. Various functional capabilities are being currently implemented at each of the four levels of the logical model [Ref. 6]. These are as follows:

- Level 4 - identification and representation of the dynamic attributes of other agents including their capability and availability
  - negotiated and non-negotiated assignment of work across the network based on each agent's capabilities (functional decomposition of tasks) and availability (load balancing)
  - establishing a basis for fault tolerance through a policing of contractual obligations between agents and through flexibility of inter-agent associations
  - establishing a basis for fault tolerance through flexible and dynamic association of functionality to hardware

- Level 3 - a canonical representation of goals, plans, and priorities of an agent to the outside community as a method of asserting passive influence on the behavior of outside agents
  - channels and protocols for actively influencing the goals, plans and priorities of other agents
  - evaluation of proposed work and of bids to accomplish proposed work to aid in the optimal
  - assignment of tasks to agents

- Level 2 - creation of a set of input and output daemons which accomplish the trading of beliefs between agents without the need to modify the agent's knowledge base

- Level 1 - network communication services and testing routines.

ORIGINAL PAGE IS
OF POOR QUALITY

The development of an implementation of the proposed logical model is affording us the opportunity to critically evaluate the model concepts, to firmly establish the operational semantics which the model embodies, and to change and refine the model as required.

Now that you have been introduced to two research activities dealing with knowledge base management and cooperating knowledge-based agents the "Intelligent Ground System" (IGS) will be introduced. The IGS is the testbed in which we are prototyping, demonstrating, and evaluating the application of our research concepts in an operational setting.

The Intelligent Ground System (IGS)

As stated previously the major goal of our work is to influence the evolution of the systems which support the operations of space-related missions. Current ground operations systems are very complex. Though aspects of them are highly automated they nevertheless require many manually intensive operations. In some instances the cognitive workload on the human operators is reaching critical limits and the bandwidth of data and information needed to be processed by the operators is far exceeding human capability. The engineering of some of the human operator's expertise into the computer systems, i.e. the development of knowledge-based components, seems to be a viable alternative for the implementation and operation of systems with the real potential of outstripping human capability to manage. The intent of the IGS effort is to help crystalize and clarify what a highly knowledge-based ground system of the future could be like. Work on the development of the IGS is proceeding along two lines. First, a preliminary testbed, the Intelligent Control Center (ICC), has been developed. This testbed, to be discussed shortly, addresses a major component of the ground system. Experience in developing and demonstrating the ICC testbed has helped us to determine the best way to define and develop the IGS testbed. As the ICC is expanded and refined to support a higher-fidelity simulation of a portion of ground system operations work will begin on the development of an object-oriented model of the IGS. The intent is to have the ICC transform into the IGS with restructuring and the addition of several more knowledge-based components. The next sections of this paper give an overview of the ground system configuration, the portion of the ground system on which we have been concentrating our testbed efforts, and a brief overview of our current testbed configuration.

Our Current Focus

Before proceeding with a discussion of how we are focusing our research activities in a testbed environment let us briefly examine the operational domain which is intended to host the results of our work. We have been investigating the nature and effective use of cooperating knowledge-based systems in the context of ground systems needed to support the successful operation of near-earth unmanned scientific spacecraft (situation 2 above). The following Figure 3. gives a high level overview of the existing ground system.

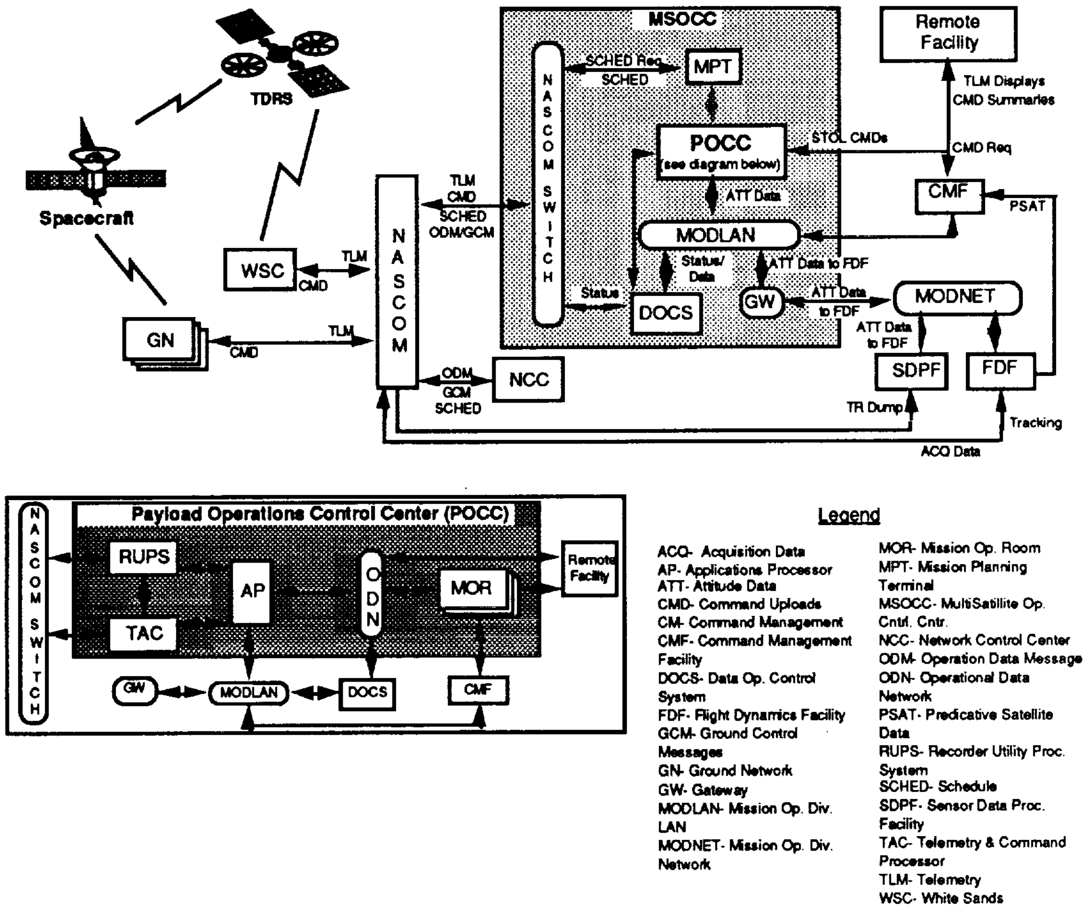


Figure 3. Functional View of the NASA End-to End Ground System

The insert box detailing the functional components in the Payload Operations Control Center (POCC) indicates that portion of the overall ground system that has been our focus for an initial analysis. Within that component we have simulated portions of the Mission Operations Room (MOR) to aid in demonstrating the use of distributed knowledge-based systems to support operations.

Intelligent Control Center (ICC) Testbed

Results from our two major research activities have been combined to support the development and operation of a distributed knowledge base testbed. This testbed, depicted in Figure 4, currently incorporates three expert systems, a spacecraft simulator, and a user interface module. This initial testbed is designed to demonstrate and test out some preliminary ideas and concepts that derive from the KBMS and modelling work and which are felt to be important for supporting advanced knowledge-based automation in future control centers. These include: communication among agents, information fusion, knowledge acquisition and refinement, information synthesis and presentation to external agents, model-based reasoning, and various levels of cooperative activity. This initial testbed is currently running in the Data Systems Technology Lab/520 at the NASA/Goddard Space Flight Center.

ORIGINAL PAGE IS
OF POOR QUALITY

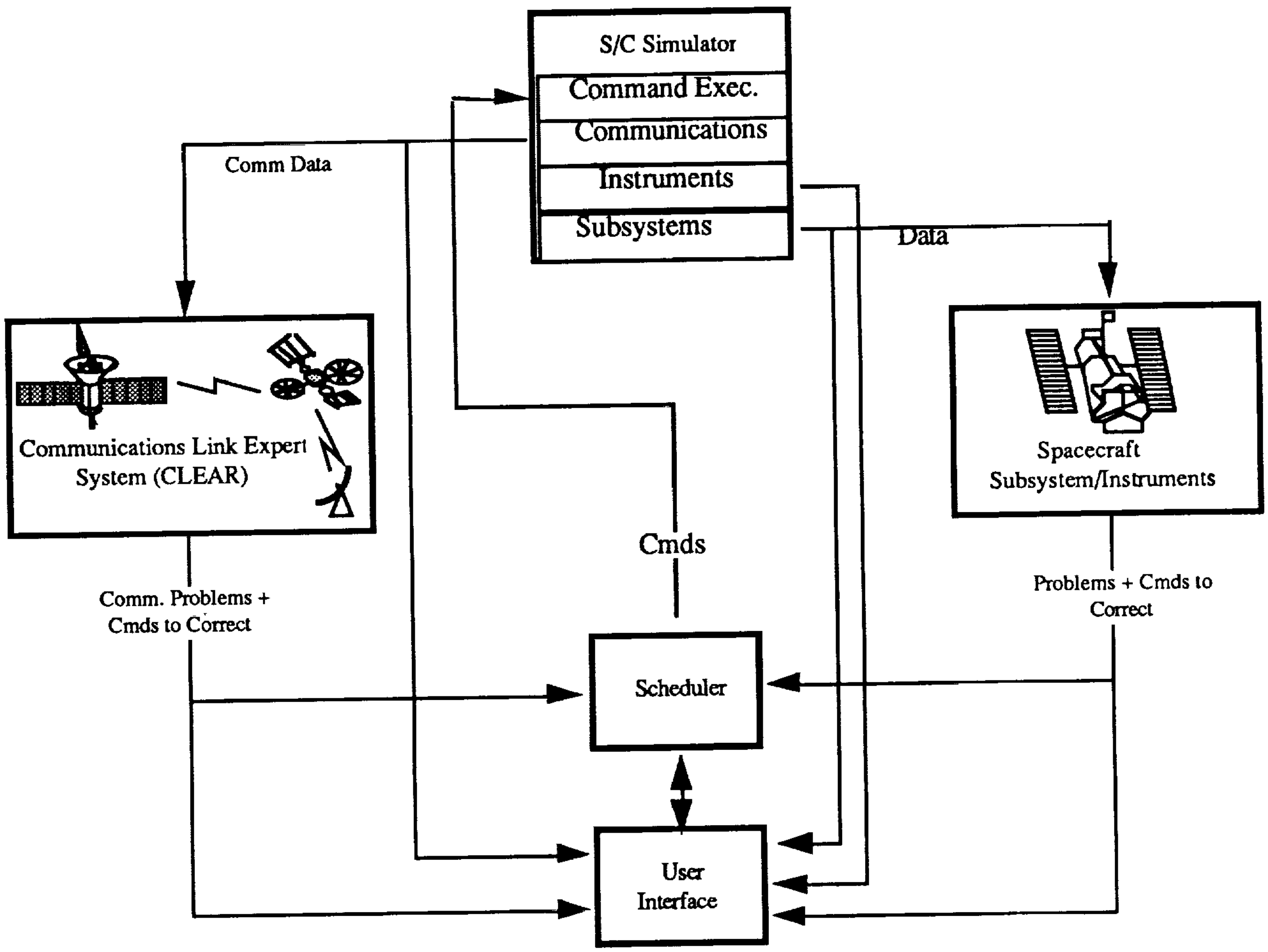


Figure 4. Current Intelligent Control Center (ICC) Testbed Configuration

The purpose of this initial prototype is to demonstrate the general concept of cooperating knowledge-based systems and specifically to demonstrate the possibility of integrating existing standalone knowledge-based systems in a cooperative framework. This particular version of the testbed only addresses the functionalities associated with levels 1 and 2 of the logical model. The following is a brief description of the testbed The testbed consists of five main components: Operator's Station (User Interface), Scheduler, Spacecraft Simulator, Communications Link Expert System (CLEAR), and the Subsystem Expert System (in this case a Power subsystem). The operator uses the operator station to issue commands to the spacecraft and monitor the spacecraft's status. The scheduler schedules the commands issued by the operator for transmission to the spacecraft. The spacecraft simulator simulates the execution of the commands by the spacecraft, and it models the dynamic performance characteristics of the spacecraft's components. CLEAR diagnoses communications problems between the spacecraft and other systems, and it recommends fixes to the problems it detects. The Power expert system monitors the spacecraft's power system telemetry to detect and diagnose power system problems. The testbed components interact via a publications/subscription mechanism. The publications for a components are the output it produces, and the subscriptions for a component are the publications it receives from other components. These terms are used to reflect the fact that the information flows in the testbed are not pre-defined.

Conclusion

To date a great deal of progress has been made in understanding the role that knowledge base management and cooperating knowledge-based agents will play in the ground systems of the future. Though the level of cooperative activity and knowledge base management that has been actually achieved in the current ICC testbed is minimal, it is a good start. The expansion of the ICC to the IGS will provide a more fertile environment in which to prototype and evaluate the ideas which are being developed in the various modeling activities. One additional feature that will be added in the near future is the ability to monitor and analyze the operational performance and behavior of the testbed. Reconsidering the very brief digression at the start of this paper it seems reasonable that the framework in which we are working need not be specific to the ground operations environment but could be used to discuss the engineering of highly automated systems involving teams of computer-based cooperating agents for all the other situations identified. Exploring other applications of the concepts discussed in this paper will only help clarify our understanding of the potential benefits which will be realized through the use of cooperating knowledge-based systems.

Acknowledgements

This report has freely drawn on the works of a very talented group of people. Dr. Barry Silverman with his colleagues at the AI Lab at George Washington University and at Intellitek Inc. is the major architect of the knowledge base management system reference model. Dr. Sidney Bailin, Mike Moore, and Scott Henderson from CTA are chiefly responsible for the development of the logical model of cooperating agents. Robert Dominy and Troy Ames from Goddard/Code 522.3 are chiefly responsible for the development and demonstration of the ICC testbed. Dr. Mel Montemerlo from NASA Headquarters and Dr. Peter Friedland from NASA Ames Research Center have provided funds and guidance to us.

References

1. Truszkowski, W. F. , Silverman, B. G., Hexmore, H.: A Design Methodology for Knowledge Base Management Systems. Artificial Intelligence in NASA, AIAA Press 1989

2. Silverman, B.G., Truszkowski, W. F., et.al,: White paper Part I:Knowledge Base Management System Reference Model, Intellitek Technical Report, 1988

3. Silverman, B. G. : A Knowledge Base Management System. Institute for Artificial Intelligence, The George Washington University, 1989

4. Bailin, Sidney, et.al.: A Logical Model of Cooperating Rule-Based Systems. CTA , 1988

5. Bailin, Sidney, et.al.: Practical Issues in the Construction of Cooperative Rule-Based Systems CTA, 1989

6. Henderson, S., Moore M.: Status of Formal Model Implementation. Private Communication, 1990