N88-17256

# BEYOND RULES
# THE NEXT GENERATION OF EXPERT SYSTEMS


Jay C. Ferguson/Robert E. Wagner
Ford Aerospace & Communications Corporation
1260 Crossman Avenue
Sunnyvale, California 94089-1198



## ABSTRACT

This paper introduces the PARAGON Representation, Management and Manipulation system. The concepts of knowledge representation, knowledge management, and knowledge manipulation are combined in a comprehensive system for solving real-world problems requiring high levels of expertise in a real-time environment. In most applications the complexity of the problem and the representation used to describe the domain knowledge tend to obscure the information from which solutions are derived. This inhibits the acquisition of domain knowledge, the capability to perform knowledge verification/validation, places severe constraints on the ability to extend and maintain a knowledge base while making generic problem solving strategies difficult to develop. Ford Aerospace has pioneered a unique hybrid system to overcome these traditional limitations.


## INTRODUCTION


This paper introduces the PARAGON Representation, Management and Manipulation system. The concepts of knowledge representation, knowledge management, and knowledge manipulation are combined in a comprehensive system for solving real-world problems requiring high levels of expertise in a real-time environment. In most applications the complexity of the problem and the representation used to describe the domain knowledge tend to obscure the information from which solutions are derived. This inhibits the acquisition of domain knowledge, the capability to perform knowledge verification/validation, places severe constraints on the ability to extend and maintain a knowledge base while making generic problem solving strategies difficult to develop. Ford Aerospace has pioneered a unique hybrid system to overcome these traditional limitations.


To address these problems, Ford Aerospace has developed a model-based paradigm which is realized in a system called PARAGON. PARAGON consists of three major areas: Knowledge Representation, Knowledge Management, and Knowledge Manipulation. Knowledge Representation is the foundation of PARAGON. The knowledge representation determines how you manage and manipulate the knowledge of the domain. A hybrid representation scheme was chosen that integrates frames [Minsky], semantic networks [Quillian], classification hierarchies [Quillian], blackboards [Hayes-Roth], demons [Waterman], transition networks [Petri, Woods], and rules [Shortliffe]. Knowledge Management encompasses the acquisition of the domain models the translation of the models into the knowledge representation, and the verification and validation of the domain models. This is accomplished through graphic interfaces that provide a framework to develop, maintain and access the knowledge base. Knowledge Manipulation provides a modular set of generic problem solving modules [Clancey) that can be combined to solve different types of problems within a domain.

## KNOWLEDGE REPRESENTATION

Ford Aerospace has designed a formalism in which to express knowledge about the world by modeling the structure and function of the concepts that are part of the problem we are trying to solve. This formalism evolves the classification/rule based representations to a more explicit, consistent and robust structured conceptual network representation [Sowa, Mylopoulos]. Rule based systems lack the explicit structure for expression of descriptive hierarchical and temporal knowledge. Classification systems attempt to rectify this situation by providing a hierarchical structuring mechanism to express the knowledge. These systems have had limited success because they lack a methodology to direct and dictate the structure of the knowledge. This has resulted in a mixing of completely different types of knowledge within the same structure. Both types of systems, rule and classification, have been combined in new representation systems [Fikes, Stefik]. These hybrid systems solved some problems, created new problems and left many problems unaddressed.

Although current tools incorporate frame-based classification techniques with rule-based knowledge representation mechanisms, they fail to provide generic problem solving techniques. These mechanisms allow more generic algorithms but lack the structure and information required to solve problems generically. Although consistent, these representations lack an underlying principle from which a description of the domain can be derived. In a rule-based system there is no structure to dictate what may be used in the context (antecedent) or the action (consequent) of the rules. This flexibility allows a completely ad hoc development to occur.

The same situation exists for frame-based classification systems. There are no guiding principles or constraints on what may be an object, what it may contain, the structure of the classification hierarchy, or how objects communicate with each other. Conceptually this level of freedom is appealing; however, without the required structure and constraints large systems quickly become unextendable, impossible to validate and difficult to maintain [McDermott]. Because these areas are handled in an ad hoc manner there is not enough consistency to apply generic problem solving techniques to these representational systems. This inconsistency also causes problems with the ability to acquire knowledge through generic tools.

PARAGON is a modeling paradigm used as the conceptual basis for describing the domain. This paradigm is realized in a hierarchical knowledge representation that allows concepts to be defined, their inter-relationships specified and their behavior described within a dynamic conceptual network. The network consists of two types of entities, conceptual entities (the nodes) and relational entities (the links). Both entities are defined in a classification hierarchy. Each class of entities is characterized by a specific definition that describes the behavior (semantics) of the entity.

This approach allows the structure in the domain to be directly represented within the computer in a generic and consistent manner that corresponds to the way people perceive the domain. The domain is structured along five dimensions (figure 1); definition, composition, functional relationships, structural relationships, and sequential behavior. Each relationship has a well-defined behavior that describes how information is propagated between associated concepts. This allows knowledge to be described in a modular and well defined manner at varying levels of granularity.

Figure 2 is a slice of the representation that displays the major concept and relationship types within the system. This structure makes the efficient management of large amounts of complex knowledge possible through a set of knowledge acquisition interface tools. The concepts of generalization, abstraction, and cause/effect are combined in a uniform and consistent manner. Many existing representation techniques are integrated to provide consistency for both the representation and manipulation of the domain information.

Concepts are the building block of the system. Each concept can have a set of attributes that describe it as an individual entity. Primitive concepts organize the local attributes, inter-relationships and behavior of instances [Quillian]. Abstract concepts are defined through the aggregation of other concepts [Sowa]. Abstract concepts hide the internal details of their components while maintaining the external relationships to other concepts. Definitional concepts are a covering set for the specializations that belong to the concept [Brachmann]. Definition concepts are used to store generic information and to create new specializations based on the definition of their current specializations. State concepts specify the context in which particular facts are currently true. Event concepts specify how to determine the attribute values in a state and the actions that may occur. Concepts are inter-related with other concepts through functional, structural and temporal relationships.

In order to describe behavior we have developed a representational methodology to model temporal knowledge within a declarative framework. Sequential descriptions are used to represent processes and to model the dynamic behavior of concepts within a domain. When this information is represented in code it is not accessible for explanation or reasoning. Similar to the process of moving situation/action knowledge from code into rules, this methodology is opening the black-box of code and making declarative knowledge available to solve the problems within a domain. By formalizing the representation of procedural knowledge, generic algorithms can be developed to manipulate,

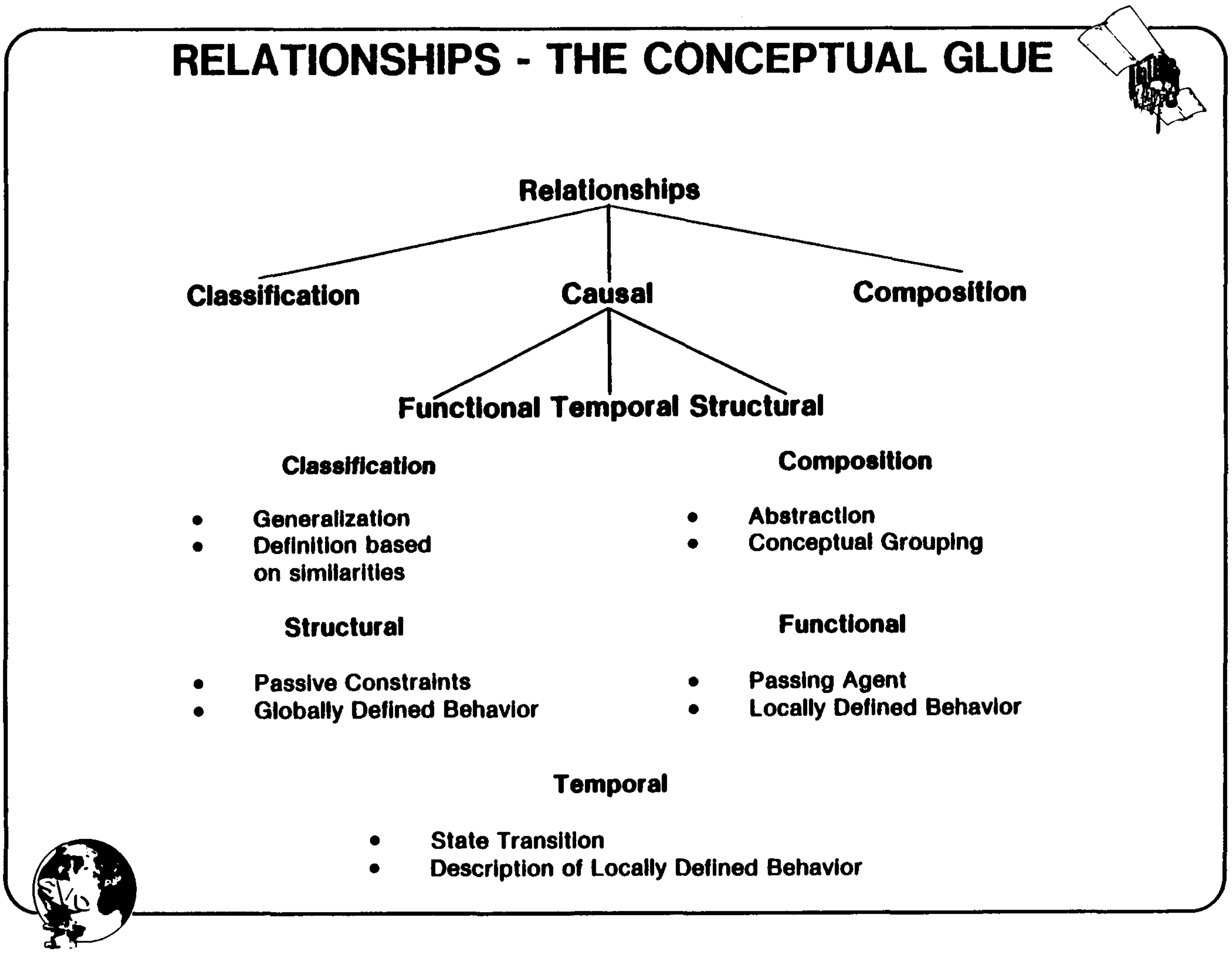


Figure 1. Relationships in PARAGON

reason about, and explain temporal events.

A process is described by a set of states, the conditions in which a state transition may occur and tne events that take place in each state. An event in a process is used to compute the value of attributes local to the concept that tne process describes. PARAGON uses a theory of LOCALITY that defines the inter-communication of independent parallel processes and constrains access to information by requiring explicit causal relationships to be defined between the communicating processes [Hoare]. Events can only affect local attributes. Events can access external information in their computations through the explicit causal relationships associated with the concept. The conditions in which a state transition can occur are constrained by the same mechanism. Processes maintain temporal histories based on the

## A SLICE OF THE MODEL

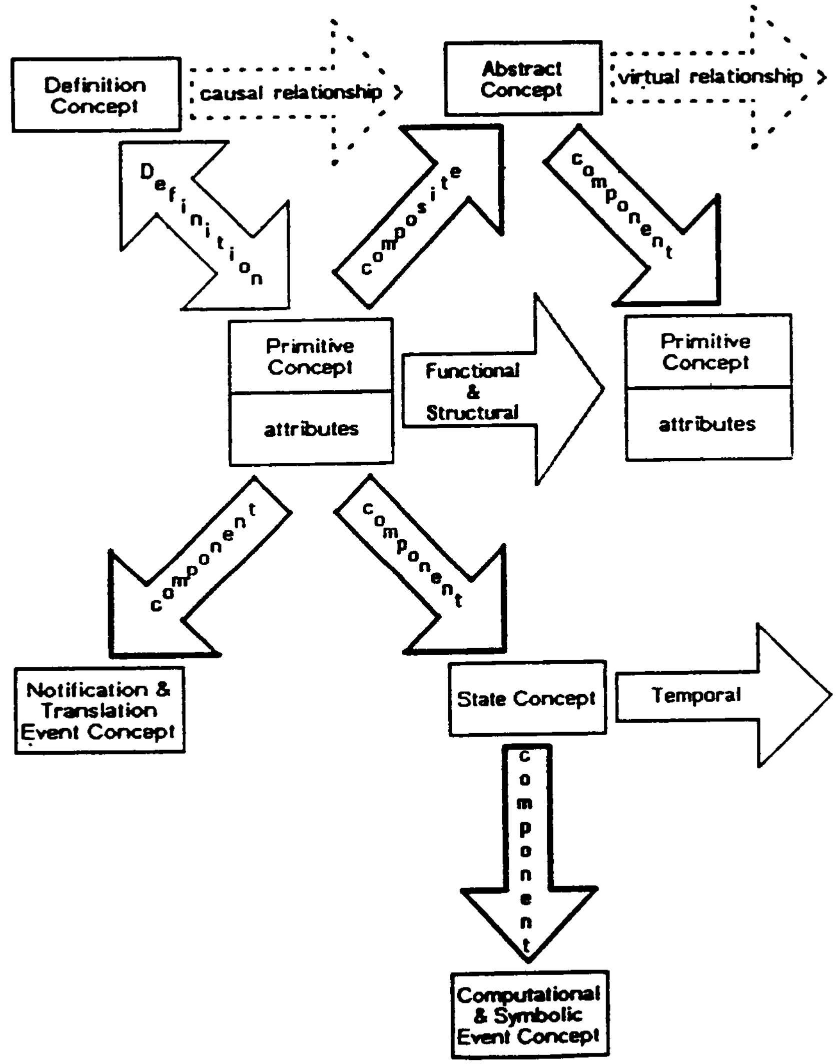


Figure 2. PARAGON Representation

hierarchical structure of the process [Kahn]. This data base can be used for various forms of temporal reasoning. An outcome of the theory of LOCALITY is that each process is a self-contained description that has no side-effects. Information is passed between concepts on an access only basis. These properties lead to a highly programmable parallel processing system.

This methodology combines both discrete and continuous processes to create a hybrid model-based representation system. Techniques have been developed to simplify complex processes and events through equation segmentation and linear approximation. This amounts to reverse engineering knowledge from mathematical equations. Both code-based procedure descriptions and equation-based computations are explicitly represented in a consistent descriptive representation.

## KNOWLEDGE MANAGEMENT

We have also developed a set of graphical interfaces that are used to map the domain concepts, structure, inter-concept and intra-concept relationships, and temporal behavior into the PARAGON knowledge representation system. The knowledge acquisition tools are a reflection of the underlying model-based knowledge representation. This allows direct translation of the acquired data into the knowledge base through generic algorithms while maintaining a high degree of correspondence between the data and the world view of the knowledge engineer (cognitive resonance).

The purpose of these tools is to allow experts with minimal training to enter knowledge into the system while capturing enough information to solve problems that occur within the domain. Each tool is a graphical interface that provides the expert with a framework in which he can enter the various types of knowledge. This framework is obtained through generalization, abstraction and the inter-relationship of concepts. Users are allowed to define specializations of relationship classes to specify specific semantic and communication information in a particular domain. Various methods have been developed to efficiently organize, index and access the large amounts of information contained in the knowledge base.

Knowledge management consists of the acquisition, validation and maintenance of the information that is used to solve the problems in the domain. In most expert systems there is no way to acquire the information through a generic mechanism. By using a modeling approach to expert systems we are able to acquire the information through graphic interfaces from knowledge engineers untrained in artificial intelligence or programming. This knowledge can then be validated based on the consistency of the representation and tested through simulations using the model description. The knowledge engineer has the ability to define complex concepts by grouping primitives through composition.

Input through the graphic interface provides a consistent and well organized knowledge base that can be extended and maintained. This is possible because PARAGON is representing information in a well defined and structured methodology that emphasizes the semantic instead of syntactic level. This is sometimes referred to as deep vs. shallow models [Genesereth, De Kleer]. Most expert systems use information in the form of rules and frames. The frames are used to store the facts and operations that can be performed on the facts (LISP programs). The rules are used to represent contextually dependent information, control sequences and contextually dependent behavior. All of this information is stored at a syntactic level. There is no well-defined methodology used to process the data. Programs are used to give the words in the rules and frames some meaning. The meaning is based on syntactic patterns of characters that match other patterns of characters. Most of the time the meaning in rule based expert systems is derived by the person looking at the rules and not the rules themselves.

Knowledge acquisition is the process of entering knowledge about a domain into the computer and translating the knowledge into an internal representation [Musen, Tsuji]. This area has traditionally been a bottleneck for building expert systems. The problem has been that most application systems are not understood or understandable at the level of heuristic rules. Rules do not offer any framework in which a knowledge engineer can work. The knowledge engineer must build the framework himself by trial and error. Because rules offer no structuring mechanism this task becomes increasingly difficult as the system grows.

By approaching the task from a modeling perspective the knowledge engineering becomes feasible with generic tools. PARAGON allows the knowledge engineer to define the concepts in the domain based on actual examples. Because the knowledge engineer has samples of the concepts in the domain available, this method provides a framework to develop and extend the definition of the model. The examples are then generalized through a process of shared attributes to create definitions for the engineer to reuse in defining other similar concepts in the system. This significantly reduces the knowledge engineering task of entering redundant information, and provides a framework that the knowledge engineer can use to more easily define concepts. This methodology of definition by example also provides a generic knowledge base that can be used for other systems.

## KNOWLEDGE MANIPULATION

The ultimate goal of modeling a domain is to allow questions to be asked and problems to be solved within the domain. Application domains frequently require a wide variety of problem solving categories [Clancey]. In Diagnostics and Repair domains [Ferguson, Siemens] the problem solutions require determining the current state of the system (classification, situation assessment and interpretation), determining what caused the system to be in the current state (diagnostics) determining what state(s) the system should/could be in for it to function correctly (goal determination), how to get more data about the system (testing) how to actually fix the system (planning & command), and what the future state of the system might be (prediction). Each of these areas is extremely complex by itself. To be able to deal with all of these areas within the same consistent representation is beyond any of the individual representation techniques or expert system shells available today.

PARAGON uses the dynamic memory model [Schank] applied to the domain model for event recognition (expectation failure) and causal diagnosis (explanation through accountability). Spreading activation [Quillian] on the domain model is used for interpretation. The behavioral specification and causal relationships are used for goal determination, prediction, planning and command. By having an internal deep model of the structure, behavior and causal inter-relationships we have found that we can solve many of the difficult problems within the modeling framework.

PARAGON has the advantage of access to a model of the system that it is reasoning about. This allows the knowledge manipulation algorithms to access the causal pathways and the behavioral aspects of the system. Because of the consistent and formalized nature of our representation, we can apply generic problem solving algorithms to the model independently of the domain. Another major advantage is the reduction of search through indexing techniques. Because the model is used to focus on small, localized sets of components, the search eliminates most of the model immediately. This allows more time to be spent dealing within a restricted environment to solve the problem. This modeling paradigm can be used as the basis for complex, real-time expert systems. Because of the capabilities of the knowledge representation language to model the different dimension of a domain, many types of reasoning can be performed using generic algorithms.

## CONCLUSION

We have described a new paradigm to solve problems using a domain model and generic acquisition, representation and manipulation methods. The domain independent capabilities of PARAGON are derived from the use of robust and consistently defined behavioral and structural descriptions to model knowledge at the semantic level. Defining how information is propagated through various relationships and constraining how concepts can be related achieves a uniform and consistent set of semantics. PARAGON also has the unique ability to represent temporal and behavioral knowledge which can be used for all classes of problem solving. By making this information available in a declarative formalism, the level of reasoning able to be performed by the system moves from a shallow level to a deep level. At the same time the models that are constructed are natural to the domain engineer. This makes the task of definition, validation, and maintenance of large & complex domain knowledge base feasible. The most important feature of PARAGON however, is the ability to develop a specifiable methodology for the construction and testing of expert systems. Without such a methodology, construction of expert systems would only be applicable to small problems by a limited supply of highly trained knowledge engineers. All these features of PARAGON have combined to make building large expert systems a reasonable task.

## ACKNOWLEDGEMENTS

This work is the result of a combined effort of the Artificial Intelligence Group at the Sunnyvale Division of the Ford Aerospace & Communications Corporation (Marilyn Golden, Alain Rostain, Ron Siemens, Dennis Heher, Tihmer Toth-fejel, Vince Contreras, Paul Pownall) and the progressive management that allowed it to happen.

REFERENCES


1. Brachmann, R.J., On the Epistemological Status of Semantic Networks, in Associative Networks: Representation and Use of Knowledge by Computers, pp. 3-50, edited by N.V. Findler, Academic Press, 1979.

2. Clancey, W.J., Heuristic Classification, AI Journal, Vol. k27 Number 3, Dec., 1985, pp. 289-350, North-Holland, Amsterdam.

3. Contreras, V., Ferguson, J., Knowledge Management: An Introduction to the Concept of Knowledge Management as a System for Controlling the AI Devėlopment Environment for an Expert System which Performs Satellite Anomaly Resolution, Mil Com 86, 1986

4. De Kleer, J., Brown, J.S., A Qualitative Physics Based on Confluences. Qualitative Reasoning About Physical Systems. MIT Press. 1985, pp. 7-83.

5. Ferguson, J.C., Siemens, R.W., Wagner, R.E., STAR-PLAN: A Satellite Anomaly Resolution and Planning System, Proceedings of AAAI Workshop on Coupling Symbolic and Numerical Computing in Expert Systems, August, 1985.

6. Fikes, R., Kehler, T., The Role of Frame-Based Representation in Reasoning, Communications of the ACM, Vol. 28 Number 29, Sept. 1986.

7. Genesereth, M.R., The Use of Design Descriptions in Automaed Diagnosis, Qualitative Reasoning about Physical Systems, D.G. Bobrow, pp. 411-436, MIT Press, Cambridge, Mass.

8. Golden, M., Siemens, R., Ferguson, J., Whats Wrong With Rules?, Westex 86.

9. Hayes-Roth, B., The Blackboard ARchitecture: A General Framework for Problem Solving?, Stanford University HPP Report HPP-83-30, 1983.

10. Hoare, C.A.R., - Communicating Sequential Processes. 1985, Prentice-Hall.

11. Kahy, M.G., Ferguson, J.C., Shortliffe, E.H., & Fagan, L.M., An Approach for Structuring Temporal Information in the ONCOCIN System, Proceedings of the Symposium on Computer Applications in Medical Care, 1985.

12. Minsky, M., Matter, Mind & Models in Semantic Information Proceedings, Edited by Marvin Minsky, MIT Press, Cambridge, Mass., 1968.

13. Musen, M.A., Fagan, L.M., Combs, D.M,., Shortliff;e, E.H., Facilitating Knowledge Entry for an Oncology Advisor Using a Model of the Application Area, Stanford University Technical Memo KSL-86-1.

14. Mylopoulos, J., Brodie, M., Schmidt, J., On Conceptual Modeling. Springer-Verlag, 1984.

15. Petri, C.A., Fundamentals of ;a Theory of Asynchronous Information Flow. Information Proceedings 1962, Proceedings of the IFIP Congress 62, Munich. North Holland Publishing Company, Amsterdam, pp. 386-390.

16. Quillian, M.R., Semantic Memory, edited by M. Minsky, Semantic Information Processing, Cambridge, Mass., MIT Press, pp. 216-270.

17. Schank, R.C., Dynamic Memory: A Theory of Reminding and Learning in Computers and People, Cambridge University Press, 1982.

18. Shortliffe, E.J., Computer-based Medical Consultation: MYCIN. American Elsevier, New York, 1976.

19. Siemens, R., Golden, M., Ferguson, J., StarPlan II: Evolution of an Expert System, The National Conference on Artificial Intelligence, 1986.

20. Sowa, J.F., Conceptual Structures. Addison-Wesley, 1984.

21. Tsuji, S., Shortliffe, E.H., Graphics for Knowledge Engineers: A Window on Knowledge Base Management. Stanford University Memo KSL-85-11.

22. Waterman, D.A., Hayes-Roth, F., Pattern-Directed Inference Systems, Academic Press, 1978.

23. Woods, W., Whats in a Link?, Foundations for Semantic Networks. In Representations and Understanding. Academic Press, pp. 35-82.