OBJECT-ROLE MODELING FOR SPACE SYSTEMS

Christopher A. Kitts[*]

Space Systems Development Laboratory
Durand 453, Stanford University, Stanford CA 94305-4035
E-mail: kitts@leland.stanford.edu, Phone: (650) 725-6794, Fax: (650) 723-3377

ABSTRACT

Object Role Modeling (ORM) is a technique for formally modeling a domain at a conceptual level. By focusing on elementary facts within the domain, ORM organizes a system in terms of simple *objects* and the naturally expressed *roles* in which they participate. This modeling technique has been applied to a simple space system; the resulting conceptual schema is being used to explore new applications of model-based reasoning. This paper presents a simplified version of the developed ORM space system schema and describes how it has motivated the creation of new model-based reasoning approaches for managing anomalies and mission product processing within space systems. In addition, the paper describes how these approaches are being prototyped within a simple, real-world, global space system.

KEY WORDS

Object Role Modeling, Model-Based Reasoning, Anomaly Management, Mission Products Processing

1. INTRODUCTION

Since its inception, the space community has relied heavily upon experiential approaches to mission operations. In this strategy, reasoning is based upon a collection of heuristics, intuitions, and past experiences which is typically encoded in the form of procedures, diagrams, handbooks, manuals, and memorized information. This style of knowledge base represents the fundamental design and behavior of the space system in a very weak manner.

Widespread reports in the space operations literature, as well as years of the authors' own experience in operating a number of space systems, attest to the significant drawbacks of experiential systems. These include high training and staffing costs for human operators, the sensitivity of performance to personnel changes, the impacts of human error, the inability to reuse knowledge and procedures across missions and lifecycle phases, the latency in information feedback and analysis, the sensitivity of the knowledge base to small changes in the system, and a variety of other reasons. Together, these drawbacks can result in operations costs that constitute 25-60% of overall mission lifecycle costs [1]; yearly space system operations costs exceed $10 billion for the entire industry [2]. Declining federal budgets, the commercialization of space systems, and the future deployment of large-scale satellite constellations all provide significant impetus to improve the cost, quality, and timeliness of space operation systems.

Model-based reasoning is often cited as a technology that can ease some of these competitive pressures through its implementation within space systems. Model-based techniques use fundamental system description based upon component models in order to derive operational knowledge useful to

---

[*] Ph.D. Candidate, Design Division, Mechanical Engineering, Stanford University. Space Systems Engineer, Caelum Research Corporation, NASA Ames Research Center.

the system's controller. Typical elements in a system description include component connections, functions, and valid input/output values. The ideal benefits of deriving operational knowledge from such a representation of the system include the systematic evaluation of the problem space, the ease of maintaining system knowledge, the reusability of models and reasoning techniques, the ability to compose complex system descriptions from a set of simple component descriptions, etc.

The application of model-based reasoning strategies to space systems has been fairly recent and is typically limited in its scope of implementation to a small portion of a system’s components and/or tasks. Model-based operational techniques are currently under study and/or in development for ESA’s Advanced Technology Operations System (ATOS) Program and NASA's New Millennium Program Deep-Space 1 (NMP DS-1) spacecraft.

Researchers at Stanford University's Space Systems Development Laboratory (SSDL) are currently investigating additional space system applications of model-based reasoning. In doing this, a space system conceptual schema has been developed using the ORM technique. By identifying the existence of and relationships among fundamental elements within typical space systems, extensions to the modeling domain have been proposed. These, in turn, have motivated new model-based reasoning applications beyond fault detection and diagnosis.

2. THE OBJECT ROLE MODELING TECHNIQUE

A formal system description is a prerequisite for any model-based application. This description must span the complete domain under consideration and must specify the system's elements and their interactions in an unambiguous manner.

ORM is a technique for formally modeling a domain at such a conceptual level. By focusing on elementary facts within the domain, ORM organizes a system in terms of simple *objects* and the naturally expressed *roles* in which they participate. Originally developed in Europe in the mod-1970's, ORM has matured as a modeling technique and now has a mature drawing notation, a supporting design method, modeling languages, and CASE tools [3].

It is interesting to note ORM’s relationship to other system modeling techniques such as Entity Relationship (ER) and Integrated Definition Language (IDEF) modeling. ER modeling, generally more popular than ORM, formalized a domain by recognizing *entities*, *attributes* of these entities, and *relationships* among entities. This technique, however, is less suitable for designing a schema due to ambiguity about attribute classification and the lack of expressibility in capturing constraints. In fact, the ER technique was originally used for this research study, but was replaced with ORM for these very reasons. The IDEF technique supports both functional and process modeling of a system. Comparing ORM to IDEF, ORM is generally more expressive, stable, and naturally specified, whereas IDEF is generally more compact and is a United States government standard. IDEF and related techniques have been used by the research team for modeling the detailed operation of portions of the space system. For the conceptual study discussed in this paper, however, ORM has been more valuable in stimulating concepts for new applications of model-based reasoning.

The ORM technique is implemented by formulating elementary facts that provide information about the system. Each fact consists of one or more objects that each play a role as prescribed through the use of a predicate. For example (where compound fact types are used for brevity): "a NOT gate has an (Input A, Output B)", “an OR gate has an (Input C, Input D, Output E)”, “a (NOT, OR) gate produces an output signal equivalent to the binary (NOT, OR) value of its input(s)", "Output B is connected to Input C", and so on for the entire system.

Once a significant quantity of elementary facts are specified, objects are then organized into sets, quality checks are enforced, and constraints are denoted. Figure 1 depicts a simplified ORM diagram for the example fact types (some of the more subtle information constraints are not displayed). The circles represent sets of objects. Roles are represented as links between object sets. These links are annotated with boxed phrases that naturally express the roles; annotations are made from the perspective of each object set participating in the role. The model shows that components have inputs, outputs, and behaviors. Component inputs and outputs may be connected, and a component's behavior constrains its output as a function of its input. Details of ORM diagramming are provided in [4, 5].

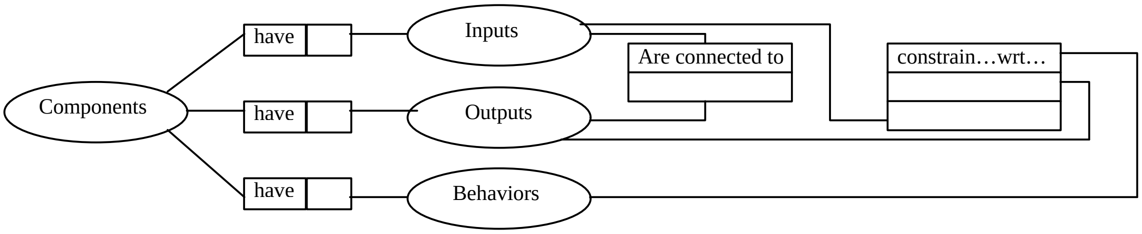

Figure 1 – *A simple example of an ORM system schema.*

3. EXTENDING THE SPACE SYSTEM SCHEMA

The object of this particular study is to consider ways of extending or modifying the conventional system schema in order to motivate new concepts for model-based reasoning applications. Work to date has focused primarily on two particular areas. The first is extending the conceptual framework of conventional fault detection and diagnosis to the broader task of anomaly management. The second is to incorporate mission product information into the schema to support model-based planning and service specification.

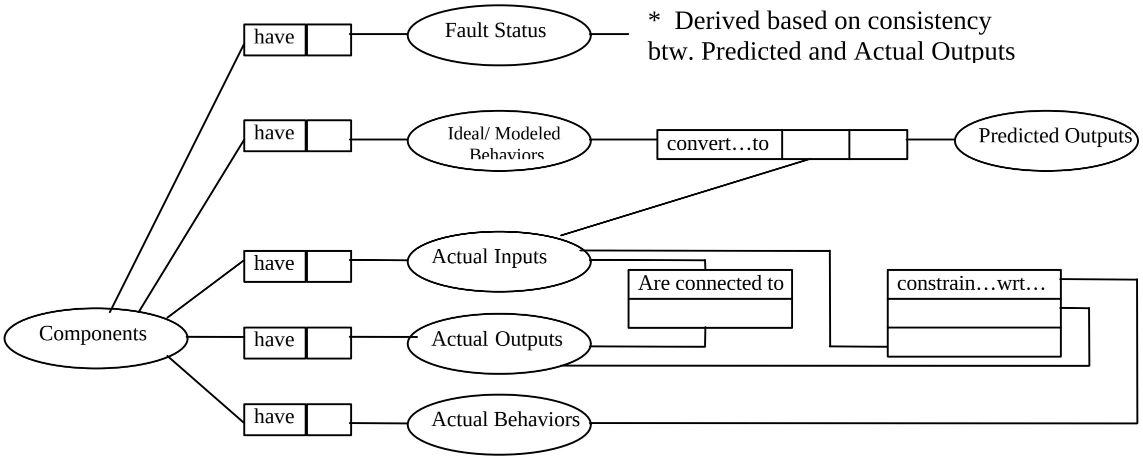

Figure 2 – *A simple example of an ORM system schema that supports fault detection.*

Extension 1 – Anomaly Management - Figure 2 expands Figure 1 in order to depict a simplified schema for the contemporary theory and practice of model-based fault detection and diagnosis. In addition to the system elements already discussed, the theoretical foundation of this field distinguishes between actual and ideal behaviors. A model of the ideal behavior can be used with observations of component inputs in order to predict the outputs. These predicted outputs are compared to observed outputs. If consistent, the component's fault status is considered normal; otherwise, it is abnormal [6]. In addition, abnormal behavior is usually defined as a malfunction within the component. This strategy is generally referred to as reasoning from the first principles of structure and behavior.

Consideration of this schema, however, brings to light a number of potential deficiencies [7]. First, a component can operate exactly as expected given its specified physical model while still causing problems within the system. Consider a transistor that saturates but that is *intended* to be used in its linear range. The transistor has no internal malfunction and operates precisely as specified. Yet this is a situation that can threaten the proper operation of the overall system. For this reason, domain elements are being added to the fault schema to incorporate teleological constraints upon the state and input variables for each component. Inconsistency between these constraints and observed telemetry results in a new component designation called a *hazard* status. Thus, a component may experience faults and/or hazards each of which is precisely defined. The result is an extension of the fault domain to a more general domain covering *anomalies*. Additional innovations are being introduced into the schema in order to alleviate modeling requirements over uncertain environments, support anomaly reconfiguration operations, etc.

Extension 2 – Product Specification and Representation - The second extension under investigation is to incorporate knowledge about mission products and services into the schema. For instance, the system offers services to clients, these services involve the production and delivery of products, and the production and delivery tasks are accomplished by components within the space system. This track of reasoning highlights the relationship between a system's services and the use of its components. It is therefore of interest to model these relationships and to explore ways of exploiting this fundamental knowledge during the operation of the system.

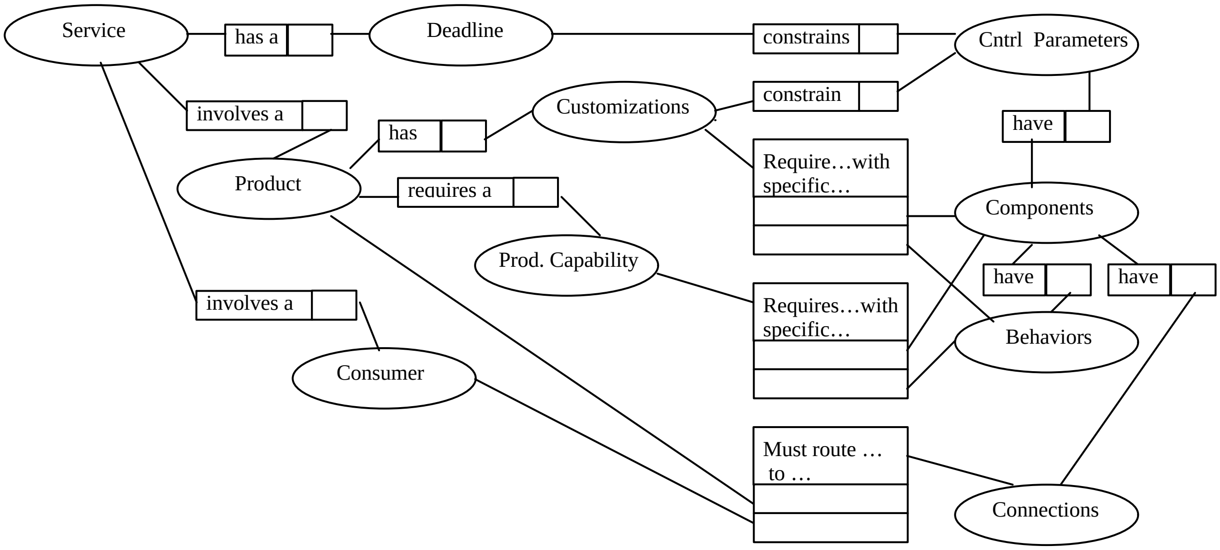

Figure 3 – *A simple example of an ORM system schema that supports transformations between service level specification and low level control parameter specification.*

Figure 3 shows a simple ORM schema relating "high level" service attributes to "low level" implementation variables within the space system. As can be seen, a service consists of a product, a consumer of that product, and a deadline for product delivery. The product may have customizations and will generally require a particular type of production capability. For example, a client might request a photo by tomorrow. The photo may be customized in terms of the subject, wavelengths, filters, etc. In addition, a photograph product implies a capability for recording light from a scene.

These service level attributes, shown on the left side of Figure 3, constrain the use of equipment within the space system. For instance, a component with a behavior supporting photographic capability is required.

These model extensions can be used to explicitly relate “high level” product/service level attributes to “low level” operational variables. In contemporary experiential space systems, the transformation from a client’s demand to its representation as a request within the system can be highly inefficient due to inadvertent over-specification, inadequate request techniques, simplistic conversion heuristics, request-time system state biases, and other factors [8]. Model-based conversion, however, enables efficient conceptual level direction of the space system since all possible operational implementations can be computed. For example, a client's request for a photograph implies the use of a spacecraft containing a component with a behavior that produces such a photo. It also implies constraints, such as the existence of a line of sight view, that must hold between the selected component and the subject of the photograph; these ultimately constrain the low level operational parameters such as time of execution. Use of a model-based problem solver to compute the valid range of operational variables for a specific client request conserves flexibility within the system.

While conceptual direction of the system offers many benefits, sophisticated clients often require a more refined level of control. Because the ORM model captures the constraint relationships between the levels of specification for each operational parameter, it is possible to use this knowledge to create an adaptable user interface that offers a range of specification options, from high to low level, for each such parameter. The client may choose to completely specify a request at the conceptual level: “Take a photo of California and deliver it by tomorrow”. Or the client may choose to completely specify a request at the operational level: “At 1200 GMT, contact the Sapphire microsatellite, transmit the command ’photo snap’, and then transmit the command ‘photo download’”. Or the client may choose to mix and match the level of specification: “Snap a photo at 1200 GMT and send me the result”. The result is a system that effectively integrates the needs of the client by offering the conveniences of high level direction with the precision of low level control.

4. EXPERIMENTAL PROTOTYPING OF NEW TECHNIQUES

The model-based techniques described in this paper are being integrated into ASSET, SSDL’s experimental real-world space system. The ASSET system is a simple yet comprehensive space operations network that is being developed for the purpose of operating university microsatellites as well as for conducting research in advanced space operations strategies.

The ASSET System - Figure 4 shows a high level view of the ASSET mission architecture. The basic components include the user interface, a central mission control center, globally distributed groundstations, Internet and amateur radio based communication links, and a collection of spacecraft [9]. Command and control tasks include 1) having clients specify the mission products they desire, 2) performing the task planning and resource scheduling to convert these client requests into contact plans, 3) executing these contact plans through real-time interaction with system equipment, 4) formatting mission products and distributing them to clients and system archives, and 5) performing system-wide health and anomaly management tasks.

Four university microspacecraft are currently being integrated into the ASSET system: SSDL’s Sapphire and Opal satellites (ready for launch and expecting launch in 1999, respectively), Weber State University’s WeberSat satellite (operational in orbit), and Santa Clara University’s Barnacle spacecraft (manifested for a sounding rocket launch in the summer of 1998). Groundstation integration has been demonstrated with SSDL’s OSCAR-class station; remote control of this station via the Internet is operational. In addition, integration with OSCAR stations and special beacon receivers is ongoing at other universities California, Utah, Alabama, Montana, Sweden, Japan, and

Italy. A number of additional universities throughout the world have also expressed interest in becoming a part of this system in the future.

Although simple in design, the ASSET system offers a large range of mature services that includes Earth photography, data and synthesized voice broadcasting, sensor characterization, and telemetry capture and processing. The operational framework for conducting these operations is nearly identical to that of more complex space systems; in addition, these services and the components that enable them have provided excellent coverage of the space system domain thereby allowing the development of a sound ORM model of space systems. This is a fundamental requirement for future attempts at extending this work to more complex space systems.

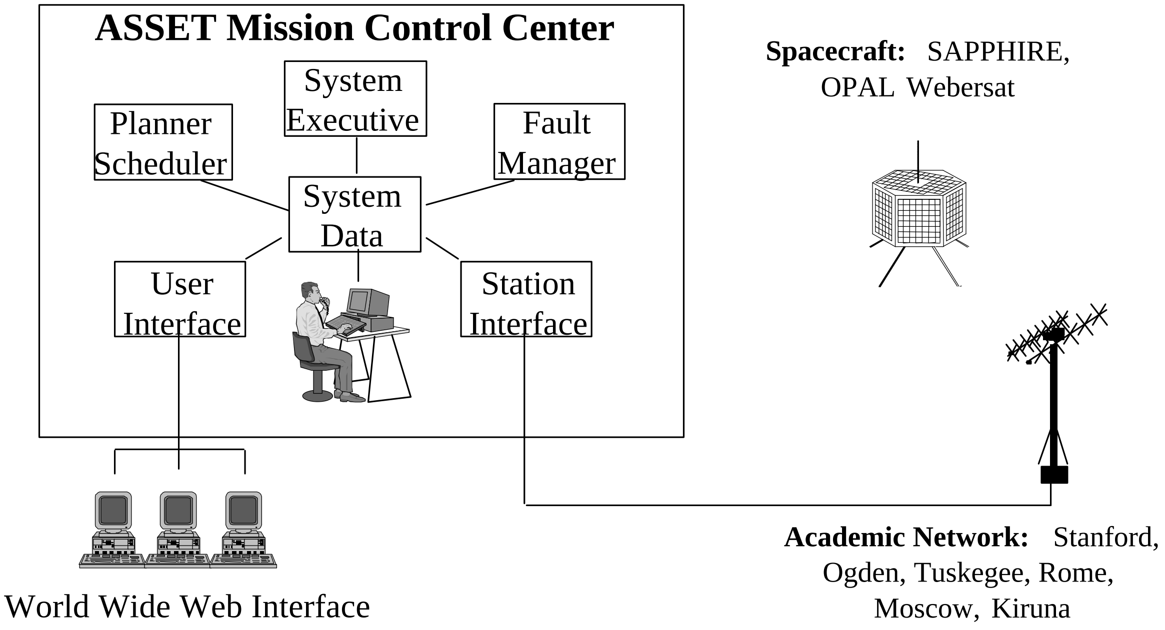

Figure 4 – *The ASSET space system architecture.*

Preliminary Prototyping Results - The model-based strategies described in Section 3 are currently being prototyped within the ASSET space system.

The anomaly management theory is still in a developmental stage where the precise detection, diagnosis, and recovery algorithms are being refined. Application of the conceptual framework and non-automated demonstrations of the reasoning methodologies have been demonstrated for simple anomaly test cases, both real and simulated. In addition, the framework has been found to be quite useful in reasoning about design bugs encountered during the development of the Sapphire and Opal microspacecraft. Ultimately, the anomaly management algorithms will be incorporated into both a real-time contact control executive program as well as into an off-line engineering analysis tool for proposing diagnosis conjectures and reconfiguration options.

The product/service modeling has been extensively used to design the client interface as well as the request representation scheme for the product planning and scheduling system. The ASSET client interface has been implemented as a Web site through the use of Dynamic HTML and JavaScript. This interface supports varying levels of specification ranging from defining desired product attributes to explicitly controlling the choice of spacecraft, component, commandable parameters, etc. Preliminary results have shown that the interface dramatically simplifies the use of the space system by novice users. In addition, requests can be made asynchronously and remotely with respect to the system's mission control center. Automation of the request process also reduces the cost and time of processing the request. Finally, significant flexibility is regained since the model-based transformation of the request conserves the full range of operational options; this leads to opportunities to increase system throughput [8].

5. FUTURE WORK

Space system modeling is an ongoing track of research within SSDL. The current focus of exploiting fundamental system models in order to improve the competitiveness of spacecraft operations is a primary element of several research initiatives and spacecraft design projects. This work is concentrating on advanced model-based techniques for anomaly management, goal-level direction, planning, scheduling, and engineering data summarization. A significant component of this work will be to conduct controlled experimental verification and validation of developed innovations in order to prove their functionality and to measure their contribution to improving system operations.

Longer term research objectives include exploring related modeling issues such as design capture, modeling architectures for integrated lifecycle management systems, simulation-based design, human operator modeling, etc. In addition, to generate greater challenges, the ASSET space system is being expanded to include additional microsatellites, ground stations, communication links, services, and clients. Finally, to ensure the applicability of this research work, SSDL will continue to collaborate with both industry and governmental space organizations. Ultimately, a specific objective of this research program is to develop innovations capable of being introduced into more complex space systems within academia, industry, and the government.

6. CONCLUSIONS

The ORM technique is a simple and natural process for developing conceptual models of systems. Researchers within SSDL have used ORM to develop a conceptual space system schema. This schema has fostered exploration of new model-based reasoning techniques for managing space system operations tasks. This has resulted in the development of model-based methodologies and application software for both anomaly management and mission products processing. The ultimate research objective of these innovations is to improve the performance of real-time spacecraft contact operations. Although not a panacea, model-based reasoning promotes systematic evaluation of a problem, compact representation of a system, and reusable algorithms and modeling constructs.

The ASSET system is proving to be a valuable, low-inertia experimental prototype for iteratively developing and validating operational innovations such as those described in this paper. The benefit to the spacecraft industry is clear: as a comprehensive, low inertia, flexible, real world validation testbed, the ASSET system is provides an unparalleled opportunity for experimentation with novel and high risk operational technologies. Furthermore, the academic validation process will assist in supplanting the anecdotal analysis commonly performed within the space community with standard evaluation practices aimed at assessing overall system competitiveness. These initiatives provide SSDL students with exciting engineering problems; collaborators, in turn, receive fresh, experimentally tested innovations in operational strategies. It is SSDL's hope that this unique research option will significantly accelerate the development of more cost-effective end-to-end space system operations.

ACKNOWLEDGEMENTS

The author wishes to thank the members of the ASSET research team, especially Mike Swartwout and Brian Engberg, for their excellent feedback concerning this research activity. Special thanks are extended to Professor Robert Twiggs, the SSDL Director, for his support and guidance in the administration of this research program. Finally, the NASA Ames Research Center is acknowledged for its interest, support, and assistance with this research endeavor. This work has been performed in partial satisfaction of graduate studies at Stanford University.

REFERENCES

[1] Ely, N., and T. O'Brien, "Space Logistics and Reliability", in Space Mission Analysis and Design, ed. James R. Wertz and Wiley J. Larson, 633-656. London: Kluwer Academic Publishers, 1991.

[2] *1997 Outlook: State of the Space Industry*, Joint Publication by Space VEST, KPMG Peat Marwick, Space Publications, and the Center for Wireless Telecommunications Publication, 1997.

[3] Halpin, T., “Object Role Modeling: An Overview”, Visio/InfoModelers White Paper, 1997.

[4] "ORM FAQ", Visio/InfoModelers Document, 1998.

[5] Nijssen, G., and T. Halpin, *Conceptual Schema and Relational Database Design*, New York: Prentice Hall, 1989.

[6] Reiter, R., “A Theory of Diagnosis from First Principles", *Artificial Intelligence* **32** (1987) 57-95.

[7] Kitts, C., *Theory and Experiments in Model-Based Space System Operations*, Draft Research Report, Space Systems Development Laboratory, Stanford University, 1998.

[8] Kitts, C., “The ASSET Client Interface: Balancing High Level Direction with Low Level Control”, In *Proceedings of the 1998 IEEE Aerospace Conference, Snowmass, CO*, March 21-28, 1998.

[9] Kitts, C., "A Global Spacecraft Control Network for Spacecraft Autonomy Research", In *Proceedings of SpaceOps '96: The Fourth International Symposium on Space Mission Operations and Ground Data Systems, Munich, Germany*, September 16-20, 1996.