Proceeding of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space:
i-SAIRAS 2003, NARA, Japan, May 19-23, 2003

Recent JPL Results Supporting Automation of Ground and Flight Systems

Dr. David J. Atkinson
JPL, California Institute of Technology
MS 126-110, 4800 Oak Grove Dr.
Pasadena, CA 91109

David.J.Atkinson@jpl.nasa.gov

Introduction

This paper will survey three software systems in development at NASA's Jet Propulsion Laboratory (JPL) that are the product of many years of research and development in artificial intelligence and related disciplines. These systems each provide a framework for automation of ground data systems in the Deep Space Network and spacecraft autonomy. The Mission Data System (MDS) provides robust, reliable, and reusable software architecture and modules that implement the core functions found in many embedded real-time control systems. MDS is also a state-based system that provides an excellent software middleware layer for autonomous systems applications. Secondly, a fault detection and isolation system for NASA's Deep Space Network (DSN) is described. This technology validation project is built on years of research in diagnostics and prognostics including the development of the BEAM and SHINE systems. The third system described is an automated Deep Space Station Controller (DSSC), also for the Deep Space Network, which includes closed-loop error recovery as well as the BEAM and SHINE diagnostics technologies. Finally, we conclude with a discussion of technology infusion into the next generation ground and flight systems.

Mission Data System

To address the challenge of designing reliable software that will expand mission capabilities, support autonomy, and be reused in multiple flight and ground mission scenarios, JPL initiated the Mission Data System (MDS) project [1]. The objective of the project was to rethink the mission software life cycle and to develop software architectures that accommodate the complexities of future mission requirements.

The Jet Propulsion Laboratory, California Institute of Technology performed this work and the work described, under contract with the National Aeronautics and Space Administration.

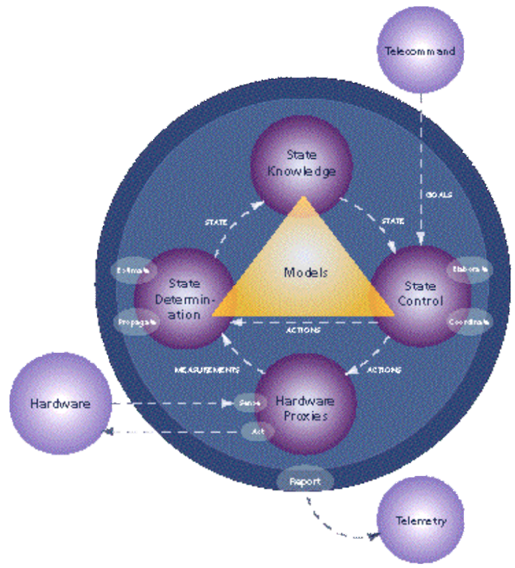

Figure 1: Generalized MDS Architecture

Figure 1 illustrates several MDS architectural elements: the central role of state knowledge and models, goal-directed operation, closed-loop control, and the separation of state determination from state control.

The project has produced a unified flight, ground, and test data system architecture that is revolutionary in scope and vision. It is a component-based, object-oriented design that assimilates and codifies years of JPL's domain knowledge in the areas of spacecraft and ground software.

The MDS core product is a unified architectural framework for building end-to-end flight and ground software systems. This framework include necessary elements for building goal-oriented systems, including the autonomous commanding; intelligent data management and transport, integrated guidance, navigation, and control, and most other capabilities needed for mission software. Design patterns provided with MDS enable adaptation of the framework for mission-specific software functions. MDS has been created with a focus on several key architectural considerations described in detail elsewhere (see earlier reference).

MDS Architectural Themes
Construct subsystems from architectural elements
Migrate capability from ground to flight to simplify ops
System state and models form the foundation for info processing
Express domain knowledge explicitly as models
Operate missions using constraints on desired state
Design for real-time reactions to changes in state
Make fault protection integral to the system
Authorize and monitor all resource usage
Separate state determination from state control
Determine state honestly from the evidence
Separate data management from data transport
Use a common mathematical base
Design interfaces for change

Table 1: **MDS Architectural Themes**

A key value of MDS is that it should enable customer missions to focus on mission-specific design and development without having to create and test a supporting infrastructure. Projects, both flight and ground, will receive a set of pre-integrated and pre-tested frameworks, complete with executable example uses of those frameworks running in an appropriate simulation environment.

One of the key virtues of MDS is its approach to technology infusion. The MDS architecture and framework software provide a standard software platform for technology plug-ins. By design, MDS does *not* prescribe a single algorithm for capabilities such as planning & scheduling, fault diagnosis, image processing, etc; many techniques exist having different strengths and weaknesses. Rather, MDS prescribes interfaces for technology plug-ins. MDS customers benefit from choices among a growing set of technology plug-ins, and technology providers benefit from a *single* technology infusion platform that applies to a growing set of mission customers. This benefit applies to all technology providers in NASA, other government agencies, academia, and the private sector.

As a project, MDS is balancing a long-term architectural vision against a near-term commitment to its first customer mission, Mars Science Laboratory project (MSL). Such commitments help focus MDS design efforts on pragmatic, well-understood mechanisms for supporting the architectural themes. The structure and framework MDS is providing to MSL also are providing perhaps the best opportunity to date for rapid infusion of advanced automation and autonomy component technologies into both flight and ground systems at JPL.

Fault Diagnostics and Prognostics for the Deep Space Network

An on-going objective of research and technology development at JPL is the creation of a framework of automated tools and techniques for reducing operational and maintenance costs in the NASA's Deep Space Network (DSN). A recent product of this research has been a technology validation system demonstrating fault diagnostics and prognostics for ground systems during DSN tracking operations [2]. The specific system targeted for the systems demonstration was the new DSN Full Spectrum Processing Array configuration located at the Goldstone Deep Space Communications Complex (GDSCC) (Figure 2).

The DSN FDI system, demonstrated first in 1999, is based on a Fault Detection and Isolation (FDI) framework developed at JPL that provides a DSN-compatible infrastructure for seamless integration of heterogeneous, intelligent tools for the purpose of DSN FDI analysis. For this demonstration, the framework was integrated with the DSN's new Network Monitor and Control Subsystem.

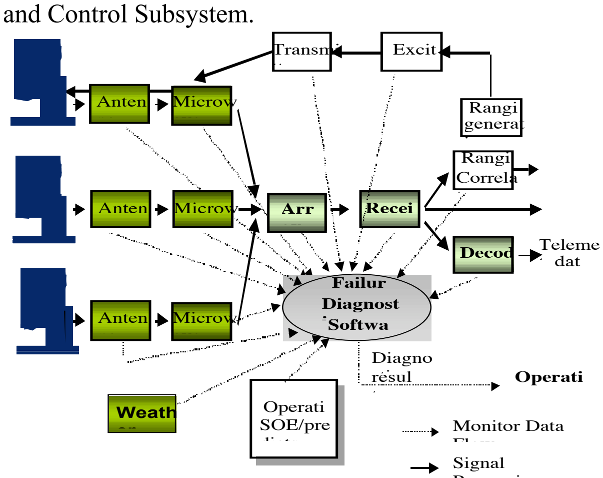

Figure 2: **Block Diagram of DSN Hardware**

Configuration

The resulting system integrated two mature JPL-developed tools, Beacon-based Exception Analysis for Multi-missions (BEAM) and Spacecraft Health Inference Engine (SHINE). BEAM is used as a highly advanced prognostic state estimator and SHINE is being used for hard real-time diagnostics and interpretation of the system state output by BEAM. These technologies provide new insights into system visibility that were not previously possible using channel-based diagnostics techniques. The integrated systems enable raw sensor data and software-derived data to be simultaneously fused in real-time. This automatically abstracts system physics and information invariants (constants). This enables the system to be ultra-sensitive to degradation and changes, and to isolate significant events in both time and space to specific sensors.

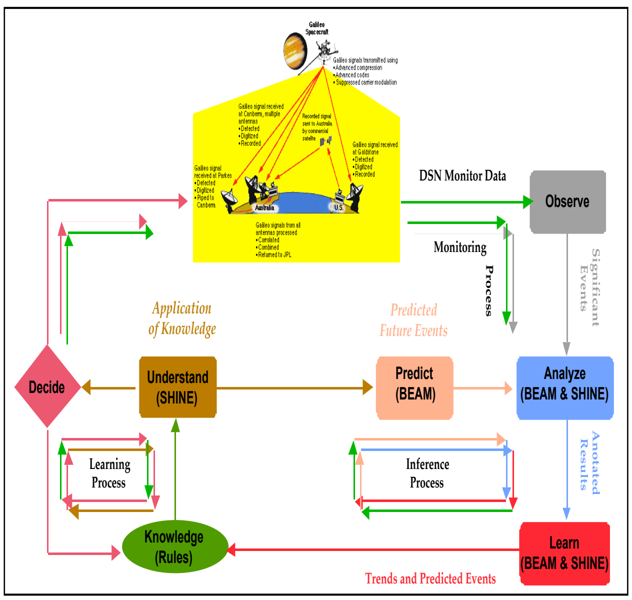

Figure 3: **FDIR BEAM and SHINE Interactions**

The key innovation of the DSN FDI system arises from the synergy of BEAM's ability to provide complex DSN system analysis using stochastic modeling, nonlinear information filtering, temporal channel analysis and adaptive wavelet theory, with SHINE's high speed inferencing capability in a real-time continuous environment. Figure 3 illustrates some of the key interactions between BEAM and SHINE. We have packaged this technology as a DSN subsystem that is compatible with the new DSN monitor and control protocols and systems.

These components are integrated into a tightly coupled architecture composed of a number of analysis

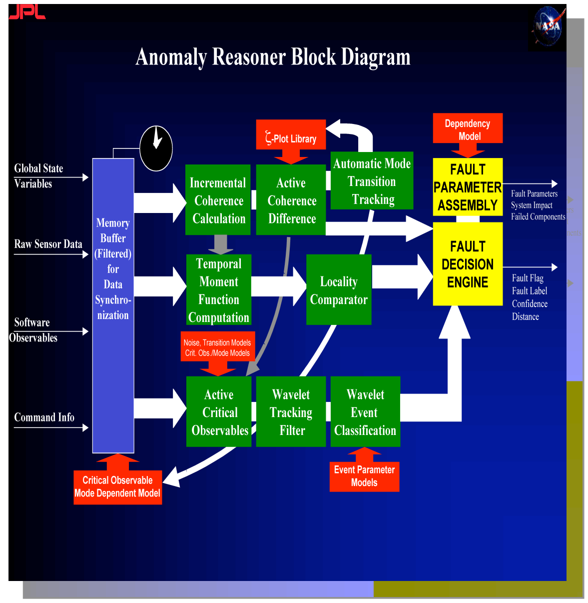

Figure 4: **Overview of Anomaly Reasoner Based on BEAM and SHINE Systems**

components. Figure 4 illustrates how these components work together in reasoning about anomalies. The DSN FDI system has the ability to continuously process all monitor data and quantify monitor data contribution to system health. By identification of data signatures associated with nominal conditions and comparison with actual received monitor data signatures, the system can detect and isolate anomalous events in an autonomous manner.

The DSN FDI Subsystem includes three components: FDI Server (or Router); SHINE-based Expert System; and, FDI-BEAM client application.

The FDI Server software adheres to the DSN's standards and protocols for data exchange. The purpose of the FDI Server is to isolate tools such as SHINE and BEAM from the intricacies of DSN interface requirements. It accomplishes this using c-based wrapper techniques and dynamic linking, all in conformance with DSN standard communications protocols.

SHINE is a powerful inference engine for expert-system type applications. It is a direct descendent of the Star*Tool [3,4], and SHARP systems [5,6,7], originally developed by this author and Mark James.

Knowledge-based systems for automated task planning, monitoring, diagnosis and other applications require a variety of software modules based on artificial intelligence concepts and advanced programming techniques. The design and implementation of the modules require considerable programming talent and time and background in theoretical artificial intelligence. Sophisticated software development tools that can speed the research and development of new artificial intelligence applications are highly desirable. The SHINE system was developed for that purpose. Included in the system are facilities for developing reasoning processes, memory-data structures and knowledge bases, blackboard systems and spontaneous computation daemons.

SHINE provides a novel paradigm for ultra-fast inferencing that goes well beyond traditional forward and backward chaining methodology. A sophisticated mathematical transformation based on graph-theoretic data flow-analysis reduces the complexity of conflict-resolution during the match cycle from $O(n^2)$ to $O(n)$ for many kinds of pattern matching operations. Computational overhead is further minimized by a built-in source-to-source transformational system for the optimization of code generated from the rules through data flow reduction (See figure 5).

SHINE, as an operational system, has contributed to reduced operations cost, improved reliability and safety in eight NASA deep space missions that include Voyager, Galileo, Magellan, Cassini and Extreme Ultraviolet Explorer (EUVE). SHINE has also been delivered to NASA's X-33 project as a flight software system component.

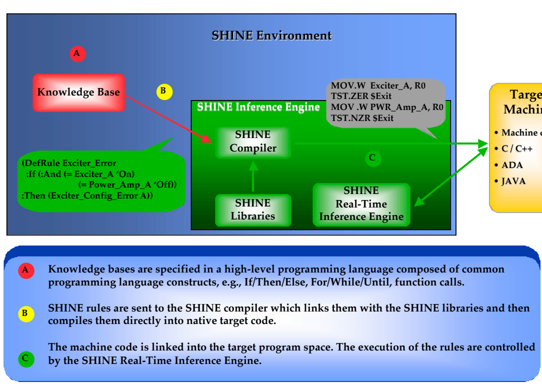

Figure 5: **SHINE Architecture**

Beacon-based Exception Analysis for Multi-missions (BEAM) is a unique new technology for complex system health assessment, prognostics and increased predictability in autonomic logistics. The technology is grounded in a radical approach to complex systems analysis that combines recent advances in adaptive wavelet theory, nonlinear information filtering (NIF), neuro-fuzzy system identification and stochastic modeling. BEAM is based on a new real-time high-fidelity algorithm that robustly and compactly analyzes complete system state from an information-theoretic standpoint, thus enabling unattended long duration operations. The NASA relevance of BEAM lies in the ability to provide accurate system health and wear assessment on demand.

The essential benefits of BEAM include:

1. Fuse and simultaneously analyze all system observables such as sensor data, executing software, commands, etc.
2. Automatically abstracts system physics and information invariants notwithstanding model availability and fidelity.
3. Ultra-sensitive to degradation/changes in the system detect and isolate changes in both space and time.
4. Highly scaleable and runs on conventional processors.

The theoretical foundation in which BEAM is based is beyond the scope of this paper and is discussed elsewhere [8]. The DSN FDI system for diagnostics and prognostics of the DSN antenna array successfully demonstrated: 1) Autonomous monitoring of downlink signal processing using DSN predicts and telemetry channel data; 2) Detection of anomalies based on SNR channel data analysis; and 3) Identification of DSN channel contribution to significant system behavior changes. The SHINE-based expert system and BEAM detected anomalies simultaneously.

Figure 6 shows a sample of the real-time output of the system, including channels automatically determines to involved in state transition of the antenna array and detection of an anomaly during a Galileo spacecraft pass.

The system validation demonstration showed that these innovative software systems are well suited for the monitoring and diagnosis of ground systems. Both BEAM and SHINE executed well in an environment where system resources such as processor cycles and memory are at a premium. The system demonstrated capability as fault detection and isolation component of an embedded system as well as effective advisory system for human operators.

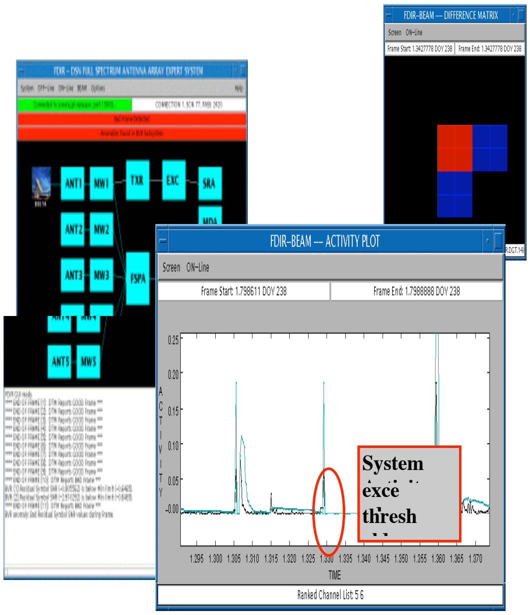

Figure 6: **BEAM and SHINE Detecting a Real-Time Anomaly During a Galileo Pass**

The ultimate goal of the DSN automation effort is "lights out" operations in order to achieve significant cost reduction. The lights, however, must be dimmed gradually due to technical, business, and pragmatic concerns. The DSN FDI automation work described here has taken the first steps toward the dimming process and is now influencing the design of the next-generation monitor and control system for the Deep Space Network.

Autonomous Ground Station Controller

The Deep Space Station Controller (DSSC) is a state of the art ground station control architecture being developed at the Jet Propulsion Laboratory (JPL) [9]. 1) The DSSC has been designed for robust closed loop control of ground communication stations utilized for communications with and commanding of NASA's deep space exploration missions.

The operation of the DSN is a very difficult task due to the extreme sensitivity of the equipment, the volume of data collected, the number of missions operated, and the frequency of service that must be provided. In an attempt to reduce cost and increase operations reliability, the DSN has looked towards automation.

The new Deep Space Station Controller (DSSC) architecture under research and development is designed to be modular and extendable. While initially this architecture is being considered for station controller, it has been designed so that the same architecture and much of the same code can be used as a complex controller and as a sub-system controller. This general-purpose solution is being referred to as the Common Automation Engine (CAE).

The architecture is scalable to provide control functionality of a DSN complex, the DSN network (collection of three complexes), and down to the level of DSN sub-systems.

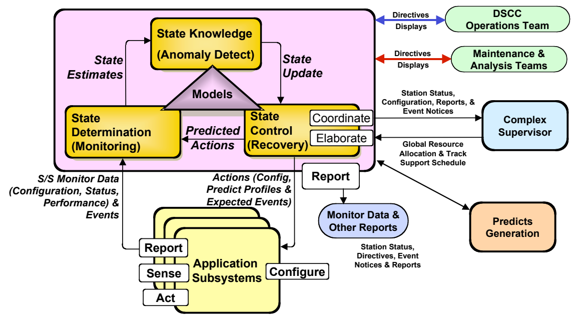

Figure 7: DSSC Architecture Diagram

The DSSC/CAE is built around two powerful state of the art technologies developed at JPL. CLEaR (Closed Loop Execution and Recovery) is a reasoning and controlling component used for selecting, executing and monitoring commands as well as re-planning recovery scenarios. The second is the fault detection and isolation (FDI) framework presented earlier in the discussion of diagnostics and prognostics demonstration for the DSN Full Spectrum Processing Array: the integrated the Beacon-based Exception Analysis for Multi-missions (BEAM) and Spacecraft Health Inference Engine (SHINE).

The DSSC/CAE was developed concurrently with the Mission Data System, and considerable cross-fertilization occurred, as a cursory comparison of Figure 1 and Figure 7 will attest. A point of major similarity is the use of a common control cycle. This cycle has been described as the Sense-Act-Plan or SPA cycle [10]. In the DSSC/CAE instantiation of this architecture there is overlap between the different tools utilized to perform each of these functionalities. At one level the planning and execution component performs all of these tasks, but because of limitations to how planners perform monitor and diagnostics functions we also utilize specific fault detection and isolation (FDI) techniques to provide greater data understanding.

Planning and execution functionality in the system is based on a continuous planning paradigm provided by the Continuous Activity Scheduling Planning

Execution and Replanning (CASPER) system [11], combined with a task level control system execution functionality provided by Task Description Language (TDL) [12]. These two components have been combined into a single system for providing a framework for Closed Loop Execution and Recovery (CLEaR) [13]. Through the combining of CLEaR, BEAM and SHINE into a single control architecture, the system is knowledgeable of both its intentions and its well-being.

Figure 8: Deep Space Station

The DSSC/CAE system is an advance beyond the current command sequence scripting used in the current DSN Network Monitor and Control (NMC) system. The static scripting method, called Temporal Dependency Networks (TDNs) in the NMC, must deal with a large number of types of services available to missions and the variation of context that events can occur. The problem faced by static TDNs is combinatorially complex. The CLEAR system resolves this by dynamically instantiating the command script from smaller script-lets, which are treated as self-contained atomic actions, to produce a command sequence. Through the use of planning technology the interactions between the script-lets are resolved.

Conclusions

This paper has been a survey of several important developments at JPL towards automation of ground and flight systems. Although the example projects were presented in terms of ground applications, the applicability of each to a flight application is equally important. This is an important realization: that real-time command and control architectures, if designed with reuse and generality in mind, can be an important bridge between ground and flight software systems. The MDS system in particular has been designed from the beginning to enable a unified flight ground architecture. The relationship between the DSSC/CAE, part of the Autonomous Ground Station Controller, and MDS illustrates furthermore the potential for rapid adoption of mature artificial intelligence technology for increasing systems autonomy.

The overall research and development program in these and other information technology and computer science areas remains robust at JPL. If anything, the Laboratory now realizes that a major portion of future flight and ground system capability will be delivered, or enabled by complex software systems, many of which would heretofore have been deemed too risky, such as those promising increased automation. At this inflection point in JPL's approach to space exploration systems, both flight and ground, we are now seeing the beginning of what promises to be a period of strong demand for the mature products of our last two decades of research and development in artificial intelligence and autonomous systems.

Acknowledgments

The author wishes to thank Mark James, Forrest Fisher and Dan Dvorak for assistance in preparation of this paper and the opportunity to summarize and quote from existing publications. Please consult the original sources listed below for more details on these and other related projects.

References

[1] D. Dvorak, R. Rasmussen, G. Reeves, A. Sacks. "Software Architecture Themes In JPL's Mission Data System", Proceedings of the IEEE Aerospace Conference (IAC), Big Sky, MT, March 2000

[2] James, Mark L., Dubon, Lydia P. "An Autonomous Diagnostic and Prognostic Monitoring System for NASA's Deep Space Network". Proceedings of the IEEE Aerospace Conference (IAC), Big Sky, MT, March 2000

[3] James, Mark, and Atkinson, David, "STAR*TOOL — An Environment and Language for Expert System Implementation", *Jet Propulsion Laboratory Report NTR C-17536*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, August 19, 1988

[4] James, Mark L. and Atkinson, David J. "*Software for Development of Expert Systems*," NASA Tech Brief Vol. 14, No. 6, Item #8 JPL Invention Report NPO-17536/7049 June 1990

[5] Atkinson, David J., et al. "Automated Spacecraft Monitoring," *Proceedings of the 1990 IEEE International Conference on Systems, Man and Cybernetics*. Institute of Electrical and Electronics Engineers. Los Angeles, CA. November 1990.

[6] Atkinson, David, and James, Mark. "Applications of AI for Automated Monitoring: The SHARP System," *Proceedings of the Second International Symposium on Space Information Systems*. National Aeronautics and Space Administration and American Institute of Aeronautics and Astronautics. Pasadena, CA. September 1990.

[7] Atkinson, David. "Artificial Intelligence for Monitoring and Diagnosis of Robotic Spacecraft," Chalmers University of Technology, Sweden, Technical report 237, 1992.

[8] Ryan Mackey, Mark L. James, Han Park, Michail Zak "BEAM: Technology for Autonomous Self-Analysis," IEEE Aerospace Conference 2001.

[9] F. Fisher, M. James, L. Paal, B. Engelhardt, "An Architecture for an Autonomous Ground Station Controller," Proceedings of the IEEE Aerospace Conference (IAC), Big Sky, MT, March 2001.

[10] E. Gat, "On Three-Layer Architectures," Artificial Intelligence and Mobile Robots – Case Studies of Successful Robot Systems, Kortenkamp, Bonasso and Murphy, AAAI Press, 1998

[11] S. Chien, R. Knight, A. Stechert, R. Sherwood, and G. Rabideau, "Integrated Planning and Execution for Autonomous Spacecraft," Proceedings of the IEEE Aerospace Conference (IAC), Aspen, CO, March, 1999.

[12] R. Simmons and D. Apfelbaum, "A Task Description Language for Robot Control," Proceedings of the International Conference on Intelligent Robots and Systems, Vancouver, Canada, October 1998.

[13] F. Fisher, R. Knight, B. Engelhardt, S. Chien, N. Alejandre, "A Planning Approach to Monitor and Control For Deep Space Communications," Proceedings of the 1998 IEEE Aerospace Conference, Big Sky, MT, March 1998.