



## ➤ Analysis, Simulation, and Verification of Knowledge-Based, Rule-Based, and Expert Systems

**This method allows valid updates to be made quickly, efficiently, and without corruption of the existing rule base.**

*Goddard Space Flight Center, Greenbelt, Maryland*

Mathematically sound techniques are used to view a knowledge-based system (KBS) as a set of processes executing in parallel and being enabled in response to specific rules being fired. The set of processes can be manipulated, examined, analyzed, and used in a simulation. The tool that embodies this technology may warn developers of errors in their rules, but may also highlight rules (or sets of rules) in the system that are underspecified (or overspecified) and need to be corrected for the KBS to operate as intended.

The rules embodied in a KBS specify the allowed situations, events, and/or results of the system they describe. In that sense, they provide a very abstract specification of a system. The system is implemented through the combination of the system specification together with an appropriate inference engine, independent of the algorithm used in that inference engine. Viewing the rule base as a major component of the specification, and choosing an appropriate specification notation to represent it, reveals how additional power can be derived from an approach to the knowledge-base system

that involves analysis, simulation, and verification.

However, in a complex rule base that may have taken years, if not decades, to build, expecting users to have in-depth understanding of the rules that make up the system is not practical. This innovative approach requires no special knowledge of the rules, and allows a general approach where standardized analysis, verification, simulation, and model checking techniques can be applied to the KBS.

The rules of the system are likely written in a particular syntax, the possibilities for which include a language or grammar used by a particular inference engine, logic rules (written in Prolog or another logic programming or declarative programming language), propositional or predicate calculus, Horn clauses, or some form of structured English. A translator is required to translate these into the grammar of a tool that is used (and for which there is a prototype) to convert to a formal language that is process-based: that is, that recognizes that processes (or units of computation) are key components of a system.

All systems, regardless of how trivial, involve at least two processes, one being the environment in which the system is executing. Processes being enabled are analogous to rules firing; “deadlock” is equivalent to contradictions or internal inconsistencies existing in the rule base; “livelock” is equivalent to having rules that are unspecified; “top” is equivalent to overspecification; and “bottom” is equivalent to rules being underspecified. The system having been translated to the appropriate formal language, tools that already exist for that language may be applied to the analysis of the system. For example, if CSP were used as the formal language, the laws of CSP could be used to prove the absence, or otherwise, of contradictions, to pinpoint unimplemented rules, or to transform rules into a more efficient form. Then other available tools could be used for simulation and model checking.

*This work was performed by Mike Hinchey, James Rash, John Erickson, and Denis Gracanin of Goddard Space Flight Center and Chris Rouff of SAIC. Further information is contained in a TSP (see page 1). GSC-14942-1*

## ➤ Core and Off-Core Processes in Systems Engineering

**This methodology can reduce the difficulty of coordinating multiple systems-engineering activities.**

*NASA's Jet Propulsion Laboratory, Pasadena, California*

An emerging methodology of organizing systems-engineering plans is based on a concept of core and off-core processes or activities. This concept has emerged as a result of recognition of a risk in the traditional representation of systems-engineering plans by a Vee model alone, according to which a large system is decomposed into levels of smaller subsystems, then integrated through levels of increasing scope until

the full system is constructed. Actual systems-engineering activity is more complicated, raising the possibility that the staff will become confused in the absence of plans which explain the nature and ordering of work beyond the traditional Vee model.

Core activities are those that produce a top-down decomposition and bottom-up integration of a system in order of increasing time. Examples of core activi-

ties are definition of requirements, design, acquisition, and integration. Because of ordering according to time, these activities are often readily understood and depicted by use of such elementary graphical aids such as timelines and Gantt charts.

Off-core activities are other systems-engineering activities that add desirable qualities to a system solution, but are not directly involved in decomposition and

integration. Examples of off-core activities are management of risk and opportunity, verification, validation, and troubleshooting. Because these activities are usually repeated many times and may not inherently be ordered in the same way as the core processes, they often cannot be represented by use of simple graphical aids. The complexity and difficulty of the task of representing off-core activities is increased by the fact that the

timing and type of work involved in these activities are more unpredictable than are those of core activities.

In the present methodology, as applied to the development of a given system, the systems-engineering plan is organized to explicitly treat core and off-core activities separately. This approach to organization provides a conceptual framework that can facilitate and accelerate understanding, by mem-

bers of the systems-engineering staff, of the relationships among many parallel activities. In so doing, this approach can reduce the difficulty of coordinating those activities.

*This work was done by Julian C. Breidenthal of Caltech and Kevin Forsberg of the Center for Systems Management for NASA's Jet Propulsion Laboratory. For more information, contact Julian Breidenthal at julian.breidenthal@jpl.nasa.gov. NPO-45745*

---

## ➤ Digital Reconstruction Supporting Investigation of Mishaps

*Lyndon B. Johnson Space Center, Houston, Texas*

In support of investigations of mishaps like the crash of the space shuttle *Columbia*, a process based on digital reconstruction from recovered components has been developed. The process is expected to reduce the need for physical reconstruction from recovered parts, reduce the time and cost of determining the cause of a mishap, and provide information useful in redesigning to prevent future mishaps.

The process involves utilization of pre-existing techniques, hardware, and software to capture sizes and shapes of recovered parts in sets of digital data. The data are manipulated to enable rendering of captured geometric information by use of computer-aided design (CAD) and viewing software. The digitization of a part and study of its

spatial relationship with other parts is taken to one of three levels of successively greater detail, depending on its importance to the investigation. The process includes a trajectory-analysis subprocess in which information from the digital reconstruction is combined with locations of recovered parts to reduce the area that must be searched to find other specified parts that have not yet been recovered. The digital product of the process is compatible with pre-existing CAD and solid-model-rendering software.

*This work was done by William D. Macy and Robert B. Luecking of The Boeing Co. for Johnson Space Center. For further information, contact the JSC Innovation Partnerships Office at (281) 483-3809.*

*Title to this invention has been waived under the provisions of the National Aeronautics and Space Act {42 U.S.C. 2457(f)}, to The Boeing Co. Inquiries concerning licenses for its commercial development should be addressed to:*

*Terrance Mason,  
Boeing Patent Licensing Professional  
Mail Code 1650-7002*

*Boeing Management Co.  
15460 Laguna Canyon Road  
Irvine CA 92618*

*Phone No. (949) 790-1331*

*E-mail: terrance.mason@boeing.com*

*Reference: Boeing ID 03-0354*

*Refer to MSC-23783-1, volume and number of this NASA Tech Briefs issue, and the page number.*

---

## ➤ Template Matching Approach to Signal Prediction

**An improvement is made in accurate prediction of future behavior and early detection of system problems.**

*NASA's Jet Propulsion Laboratory, Pasadena, California*

A new approach to signal prediction and prognostic assessment of spacecraft health resolves an inherent difficulty in fusing sensor data with simulated data. This technique builds upon previous work that demonstrated the importance of physics-based transient models to accurate prediction of signal dynamics and system performance. While models can greatly improve predictive accuracy, they are difficult to apply in general because of variations in model type, accuracy, or intended purpose. However, virtually any flight project will have at least some modeling capability at its disposal, whether a full-blown simulation, partial physics models, dy-

namic look-up tables, a brassboard analogue system, or simple hand-driven calculation by a team of experts.

Many models can be used to develop a "predict," or an estimate of the next day's or next cycle's behavior, which is typically used for planning purposes. The fidelity of a predict varies from one project to another, depending on the complexity of the simulation (i.e. linearized or full differential equations) and the level of detail in anticipated system operation, but typically any predict cannot be adapted to changing conditions or adjusted spacecraft command execution. Applying a predict blindly, without adapting the predict to current conditions, produces

mixed results at best, primarily due to mismatches between assumed execution of spacecraft activities and actual times of execution. This results in the predict becoming useless during periods of complicated behavior, exactly when the predict would be most valuable. Each spacecraft operation tends to show up as a transient in the data, and if the transients are misaligned, using the predict can actually harm forecasting performance.

To address this problem, the approach here expresses the predict in terms of a baseline function superposed with one or more transient functions. These transients serve as signal templates, which can be relocated in time and space against the