passing is needed only at the beginning and end of the computation. The minimal message passing ensures the high speed of the proposed parallel implementation.

Trade analyses can be performed and optimization problems can be solved for space mission design with respect to relevant figures of merit. The mission merits/objectives/requirements are calculated with the set of design variables. The tradeoff space given by the design problem is visualized. SOAP is used for DAWN mission design, Cassini mission analyses, Team X mission concept studies, and DOD and Air Force projects.

This program was written by John M. Coggi and David Y. Stodden of Aerospace Corporation and Seungwon Lee and Robert C. Carnright of Caltech for NASA's Jet Propulsion Laboratory.

This software is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-44486.

Automated Sequence Generation Process and Software

NASA's Jet Propulsion Laboratory, Pasadena, California

"Automated sequence generation" (autogen) signifies both a process and software used to automatically generate sequences of commands to operate various spacecraft. Autogen requires fewer workers than are needed for older manual sequence-generation processes and reduces sequence-generation times from weeks to minutes. The autogen software comprises the autogen script plus the Activity Plan Generator (APGEN) program. APGEN can be used for planning missions and command sequences. APGEN includes a graphical user interface that facilitates scheduling of activities on a time line and affords a capability to automatically expand, decompose, and schedule activities. The autogen script performs the following tasks:

- Gathers needed data files from data repositories on the mission operations network.
- Builds other data files needed for the APGEN scheduling algorithms, based on inputs specified by the user.
- Sets up the environment to run APGEN, including scheduling instructions.
- Runs APGEN, which schedules activities and writes the corresponding sequences of commands to files.
- Manipulates the resultant sequence and other files, if needed.
- Initiates any automated sequence processors to prepare the sequence for uplink, if appropriate.

This work was done by Roy Gladden of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

This software is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-30746.

Periodic, On-Demand, and User-Specified Information Reconciliation

Ames Research Center, Moffett Field, California

POUR is a framework that accepts periodic information updates, collects information on demand, and accepts user-specified information while presenting a single unified view to the user. The primary functionality of POUR is through its query interface. A query consists of any number of XPaths, where each XPath returns a list of XML strings that satisfies the query. Depending on the XPaths specified and the contents of the POUR database, query processing may be as simple as a database lookup, or as complex as a series of queries down a POUR hierarchy to a set of POUR repositories that compute the requested information on the fly before the appropriate results are returned.

Users may use any valid XPath to retrieve results integrated from across the relevant periodic, on-demand, and user-specified sources. Periodic information comes from trusted sources at an unknown frequency such that any previous information from the same source can be completely overwritten. On-demand information is computed when needed using queries to other POUR instances or using scripts dynamically executed by the Globus GRAM service. Finally, users may add their own information into a POUR repository if it conforms to a site-defined XML schema.

This program was written by Paul Kolano of Advanced Management Technology for Ames Research Center. For further information, access http://opensource.arc.nasa.gov/ or contact the Ames Technology Partnerships Division at (650) 604-2954.
ARC-15468-1

Simulating Operations at a Spaceport

John F. Kennedy Space Center, Florida

SPACESIM is a computer program for detailed simulation of operations at a spaceport. SPACESIM is being developed to greatly improve existing spaceports and to aid in designing, building, and operating future spaceports, given that there is a worldwide trend in spaceport operations from very expensive, research-oriented launches to more frequent commercial launches. From an operational perspective, future spaceports are expected to resemble current airports and seaports, for which it is necessary to resolve issues of safety, security, efficient movement of machinery and

people, cost effectiveness, timeliness, and maximizing effectiveness in utilization of resources.

Simulations can be performed, for example, to (1) simultaneously analyze launches of reusable and expendable rockets and identify bottlenecks arising from competition for limited resources or (2) perform "what-if" scenario analyses to identify optimal scenarios prior to making large capital investments. SPACESIM includes an object-oriented discrete-event-simulation engine. (Discrete-event simulation has been used to assess processes at modern seaports.) The simulation engine is built upon the Java programming language for maximum portability. Extensible Markup Language (XML) is used for storage of data to enable industry-standard interchange of data with other software. A graphical user interface facilitates creation of scenarios and analysis of data.

This program was written by Michael R. Nevins of Nevins Software, Inc. for Kennedy Space Center.

In accordance with Public Law 96-517, the contractor has elected to retain title to this invention. Inquiries concerning rights for its commercial use should be addressed to:

Mr. Mike Nevins
Nevins Software, Inc.
P.O. Box 308
Morris, IL 60450
Phone No.: (815) 941-2406
E-mail: mnevins@nevinssoftware.com

Refer to KSC-12943, volume and number of this NASA Tech Briefs *issue, and the page number.*

Web-Based Real-Time Emergency Monitoring

Stennis Space Center, Mississippi

The Web-based Real-Time Asset Monitoring (RAM) module for emergency operations and facility management enables emergency personnel in federal agencies and local and state governments to monitor and analyze data in the event of a natural disaster or other crisis that threatens a large number of people and property. The software can manage many disparate sources of data within a facility, city, or county. It was developed on industry-standard GeoSpatial software and is compliant with open GIS standards.

RAM View can function as a standalone system, or as an integrated plug-in module to Emergency Operations Center (EOC) software suites such as REACT (Real-time Emergency Action Coordination Tool), thus ensuring the widest possible distribution among potential users. RAM has the ability to monitor various data sources, including streaming data. Many disparate systems are included in the initial suite of supported hardware systems, such as mobile GPS units, ambient measurements of temperature, moisture and chemical agents, flow meters, air quality, asset location, and meteorological conditions.

RAM View displays real-time data streams such as gauge heights from the U.S. Geological Survey gauging stations, flood crests from the National Weather Service, and meteorological data from numerous sources. Data points are clearly visible on the map interface, and attributes as specified in the user requirements can be viewed and queried.

This program was written by Craig A. Harvey and Joel Lawhead of NVision Solutions, Inc. for Stennis Space Center.

In accordance with Public Law 96-517, the contractor has elected to retain title to this invention. Inquiries concerning rights for its commercial use should be addressed to:

NVision Solutions, Inc.
NASA Stennis Space Center
Building 1103, Suite 147C
Stennis Space Center, MS 39529
Phone No.: (228) 688-2205
E-mail: Charvey@nvs-inc.com

Refer to SSC-00244, volume and number of this NASA Tech Briefs *issue, and the page number.*

Real-Time Data Display

Marshall Space Flight Center, Alabama

RT-Display is a MATLAB-based data acquisition environment designed to use a variety of commercial off-the-shelf (COTS) hardware to digitize analog signals to a standard data format usable by other post-acquisition data analysis tools. This software presents the acquired data in real time using a variety of signal-processing algorithms. The acquired data is stored in a standard Operator Interactive Signal Processing Software (OISPS) data-formatted file.

RT-Display is primarily configured to use the Agilent VXI (or equivalent) data acquisition boards used in such systems as MIDDAS (Multi-channel Integrated Dynamic Data Acquisition System). The software is generalized and deployable in almost any testing environment, without limitations or proprietary configuration for a specific test program or project. With the Agilent hardware configured and in place, users can start the program and, in one step, immediately begin digitizing multiple channels of data. Once the acquisition is completed, data is converted into a common binary format that also can be translated to specific formats used by external analysis software, such as OISPS and PC-Signal (product of AI Signal Research Inc.).

RT-Display at the time of this reporting was certified on Agilent hardware capable of acquisition up to 196,608 samples per second. Data signals are presented to the user on-screen simultaneously for 16 channels. Each channel can be viewed individually, with a maximum capability of 160 signal channels (depending on hardware configuration). Current signal presentations include: time data, fast Fourier transforms (FFT), and power spectral density plots (PSD). Additional processing algorithms can be easily incorporated into this environment.

This program was written by Marc Pedings of Optical Sciences Corporation for Marshall Space Flight Center. Further information is contained in a TSP (see page 1). MFS-32325-1