



2 Software for Collaborative Engineering of Launch Rockets

The **R**ocket **E**valuation and **C**ost **I**ntegration for **P**ropulsion and **E**ngineering software enables collaborative computing with automated exchange of information in the design and analysis of launch rockets and other complex systems. RECIPE can interact with and incorporate a variety of programs, including legacy codes, that model aspects of a system from the perspectives of different technological disciplines (e.g., aerodynamics, structures, propulsion, trajectory, aeroheating, controls, and operations) and that are used by different engineers on different computers running different operating systems. RECIPE consists mainly of (1) ISCRM — a file-transfer subprogram that makes it possible for legacy codes executed in their original operating systems on their original computers to exchange data and (2) CONES — an easy-to-use file-wrapper subprogram that enables the integration of legacy codes. RECIPE provides a tightly integrated conceptual framework that emphasizes connectivity among the programs used by the collaborators, linking these programs in a manner that provides some configuration control while facilitating collaborative engineering tradeoff studies, including “design to cost” studies. In comparison with prior collaborative-engineering schemes, one based on the use of RECIPE enables fewer engineers to do more in less time.

This program was written by Thomas Troy Stanley of International Space Systems, Inc., for Marshall Space Flight Center. Further information is contained in a TSP (see page 1). MFS-31692

2 Software Assists in Extensive Environmental Auditing

The Base Environmental Management System (BEMS) is a Web-based application program for managing and tracking audits by the Environmental Office of Stennis Space Center in conformity with standard 14001 of the International Organization for Standardization (ISO 14001). (This standard specifies requirements for an environmental-management system.) BEMS

saves time by partly automating what were previously manual processes for creating audit checklists; recording and tracking audit results; issuing, tracking, and implementing corrective-action requests (CARs); tracking continuous improvements (CIs); and tracking audit results and statistics. BEMS consists of an administration module and an auditor module. As its name suggests, the administration module is used to administer the audit. It helps administrators to edit the list of audit questions; edit the list of audit locations; assign mandatory questions to locations; track, approve, and edit CARs; and edit completed audits. The auditor module is used by auditors to perform audits and record audit results: it helps the auditors to create audit checklists, complete audits, view completed audits, create CARs, record and acknowledge CIs, and generate reports from audit results.

This program was written by Christopher Callac and Charlie Matherne of Lockheed Martin Corp. for Stennis Space Center.

Inquiries concerning rights for the commercial use of this invention should be addressed to the Intellectual Property Manager, Stennis Space Center, (228) 688-1929. Refer to SSC-00180.

2 Software Supports Distributed Operations via the Internet

Multi-mission Encrypted Communication System (MECS) is a computer program that enables authorized, geographically dispersed users to gain secure access to a common set of data files via the Internet. MECS is compatible with legacy application programs and a variety of operating systems. The MECS architecture is centered around maintaining consistent replicas of data files cached on remote computers. MECS monitors these files and, whenever one is changed, the changed file is committed to a master database as soon as network connectivity makes it possible to do so. MECS provides subscriptions for remote users to automatically receive new data as they are generated. Remote users can be producers as well as consumers of data. Whereas a prior program that provides some of the same services treats disconnection of a user from the network of users as an error from which re-

covery must be effected, MECS treats disconnection as a nominal state of the network: This leads to a different design that is more efficient for serving many users, each of whom typically connects and disconnects frequently and wants only a small fraction of the data at any given time.

This program was written by Jeffrey Norris, Paul Backes, and Robert Steinke of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

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

Intellectual Assets Office

JPL

Mail Stop 202-233

4800 Oak Grove Drive

Pasadena, CA 91109

(818) 354-2240

E-mail: ipgroup@jpl.nasa.gov

Refer to NPO-30448, volume and number of this NASA Tech Briefs issue, and the page number.

2 Software Estimates Costs of Testing Rocket Engines

Simulation-Based Cost Model (SiCM), a discrete event simulation developed in Extend™, simulates pertinent aspects of the testing of rocket propulsion test articles for the purpose of estimating the costs of such testing during time intervals specified by its users. A user enters input data for control of simulations; information on the nature of, and activity in, a given testing project; and information on resources. Simulation objects are created on the basis of this input. Costs of the engineering-design, construction, and testing phases of a given project are estimated from numbers and labor rates of engineers and technicians employed in each phase, the duration of each phase; costs of materials used in each phase; and, for the testing phase, the rate of maintenance of the testing facility. The three main outputs of SiCM are (1) a curve, updated at each iteration of the simulation, that shows overall expenditures vs. time during the interval specified by the user; (2) a histogram of the total costs from all iterations of the simulation; and (3) table displaying means and variances of cumulative costs

for each phase from all iterations. Other outputs include spending curves for each phase.

This program was written by C. L. Smith of Lockheed Martin Corp. for Stennis Space Center.

Inquiries concerning rights for the commercial use of this invention should be addressed to the Intellectual Property Manager, Stennis Space Center, (228) 688-1929. Refer to SSC-00168.

yourSky: Custom Sky-Image Mosaics via the Internet

yourSky (<http://yourSky.jpl.nasa.gov>) is a computer program that supplies custom astronomical image mosaics of sky regions specified by requesters using client computers connected to the Internet. [yourSky is an upgraded version of the software reported in "Software for Generating Mosaics of Astronomical Images" (NPO-21121), *NASA Tech Briefs*, Vol. 25, No. 4 (April 2001), page 16a.] A requester no longer has to engage in the tedious process of determining what subset of images is needed, nor even to know how the images are indexed in image archives. Instead, in response to a requester's specification of the size and location of the sky area, (and optionally of the desired set and type of data, resolution, coordinate system, projection, and image format), yourSky automatically retrieves the component image

data from archives totaling tens of terabytes stored on computer tape and disk drives at multiple sites and assembles the component images into a mosaic image by use of a high-performance parallel code. yourSky runs on the server computer where the mosaics are assembled. Because yourSky includes a Web-interface component, no special client software is needed: ordinary Web-browser software is sufficient.

This program was written by Joseph Jacob 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 Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-30556.

Software for Managing Inventory of Flight Hardware

The Flight Hardware Support Request System (FHSRS) is a computer program that relieves engineers at Marshall Space Flight Center (MSFC) of most of the non-engineering administrative burden of managing an inventory of flight hardware. The FHSRS can also be adapted to perform similar functions for other organizations. The FHSRS affords a combination of capabilities, including those formerly provided by three separate pro-

grams in purchasing, inventorying, and inspecting hardware. The FHSRS provides a Web-based interface with a server computer that supports a relational database of inventory; electronic routing of requests and approvals; and electronic documentation from initial request through implementation of quality criteria, acquisition, receipt, inspection, storage, and final issue of flight materials and components. The database lists both hardware acquired for current projects and residual hardware from previous projects. The increased visibility of residual flight components provided by the FHSRS has dramatically improved the re-utilization of materials in lieu of new procurements, resulting in a cost savings of over \$1.7 million. The FHSRS includes sub-programs for manipulating the data in the database, informing of the status of a request or an item of hardware, and searching the database on any physical or other technical characteristic of a component or material. The software structure forces normalization of the data to facilitate inquiries and searches for which users have entered mixed or inconsistent values.

This program was designed and written by John Salisbury, Scott Savage, and Shirman Thomas of Cortez III for Marshall Space Flight Center. Further information is contained in a TSP (see page 1). MFS-31661