This program was written by George V. Khazanov of **Marshall Space Flight Center** *and Mei-Ching Fok of Goddard Space Flight Center. Further information is contained in a TSP (see page 1).MFS-32128-1*

JMS Proxy and C/C++ Client SDK

JMS Proxy and C/C++ Client SDK ("JMS" signifies "Java messaging service" and "SDK" signifies "software development kit") is a software package for developing interfaces that enable legacy programs (here denoted "clients") written in the C and C++ languages to communicate with each other via a JMS broker. This package consists of two main components: the JMS proxy server component and the client C library SDK component. The JMS proxy server component implements a native Java process that receives and responds to requests from clients. This component can run on any computer that supports Java and a JMS client. The client C library SDK component is used to develop a JMS client program running in each affected C or C++ environment, without need for running a Java virtual machine in the affected computer. A C client program developed by use of this SDK has most of the quality-of-service characteristics of standard Java-based client programs, including the following:

- Durable subscriptions;
- Asynchronous message receipt;
- Such standard JMS message qualities as "TimeToLive," "Message Properties," and "DeliveryMode" (as the quoted terms are defined in previously published JMS documentation); and
- Automatic reconnection of a JMS proxy to a restarted JMS broker.

This program was written by Paul Wolgast and Paul Pechkam 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-42527.

XML Flight/Ground Data Dictionary Management

A computer program generates Extensible Markup Language (XML) files that effect coupling between the command- and telemetry-handling software running aboard a spacecraft and the corresponding software running in ground support systems. The XML files are produced by use of information from the flight software and from flight-system engineering. The XML files are converted to legacy ground-system data formats for command and telemetry, transformed into Web-based and printed documentation, and used in developing new ground-system data-handling software. Previously, the information about telemetry and command was scattered in various paper documents that were not synchronized. The process of searching and reading the documents was time-consuming and introduced errors. In contrast, the XML files contain all of the information in one place. XML structures can evolve in such a manner as to enable the addition, to the XML files, of the metadata necessary to track the changes and the associated documentation. The use of this software has reduced the extent of manual operations in developing a ground data system, thereby saving considerable time and removing errors that previously arose in the translation and transcription of software information from the flight to the ground system.

This program was written by Jesse Wright and Colette Wiklow 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-42291.

Cross-Compiler for Modeling Space-Flight Systems

Ripples is a computer program that makes it possible to specify arbitrarily complex space-flight systems in an easy-to-learn, high-level programming language and to have the specification automatically translated into LibSim, which is a text-based computing language in which such simulations are implemented. LibSim is a very powerful simulation language, but learning it takes considerable time, and it requires that models of systems and their components be described at a very low level of abstraction. To construct a model in LibSim, it is necessary to go through a time-consuming process that includes modeling each subsystem, including defining its fault-injection states, input and output conditions, and the topology of its connections to other subsystems. Ripples makes it possible to describe the same models at a much higher level of abstraction, thereby enabling the user to build models faster and with fewer errors. Ripples can be executed in a variety of computers and operating systems, and can be supplied in either source code or binary form. It must be run in conjunction with a Lisp compiler.

This program was written by Mark James 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-42532.