ther information, please contact Michael F. Abernathy, Rapid Imaging Software, Inc., 1318 Ridgecrest Place S.E., Albuquerque, NM 87108.
MSC-23096

CLARAty Functional-Layer Software

Functional-layer software for the Coupled Layer Architecture for Robotics Autonomy (CLARAty) is being developed. [CLARAty was described in "Coupled-Layer Architecture for Advanced Software for Robots" (NPO-21218), *NASA Tech Briefs*, Vol. 26, No. 12 (December 2002), page 48. To recapitulate: CLARAty was proposed to improve the modularity of robotic software while tightening the coupling between planning/execution and control subsystems. Whereas prior robotic software architectures have typically contained three levels, the CLARAty architecture contains two layers: a decision layer and a functional layer.] Just as an operating system provides abstraction from computational hardware, the CLARAty functional-layer software provides for abstraction for the different robotic systems. The functional-layer software establishes interrelated, object-oriented hierarchies that contain active and passive objects that represent the different levels of system abstrations and components. The functional-layer software is decomposed into a set of reusable core components and a set of extended components that adapt the reusable set to specific hardware implementations. The reusable components (a) provide behavior and interface definitions and implementations of basic functionality, (b) provide local executive capabilities, (c) manage local resources, and (d) support state and resource queries by the decision layer. Software for robotic systems can be built by use of these components.

This software was architected and written by Issa Nesnas, Richard Volpe, Hari Das, Darren Mutz, Richard Petras, and Tara Estlin of Caltech for **NASA's Jet Propulsion Laboratory***. For further information, access the Technical Support Package (TSP)* ***free on-line at www.nasatech.com****.*

This software is available for commercial licensing. Please contact Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-30132.

Java Library for Input and Output of Image Data and Metadata

A Java-language library supports input and output (I/O) of image data and metadata (label data) in the format of the Video Image Communication and Retrieval (VICAR) image-processing software and in several similar formats, including a subset of the Planetary Data System (PDS) image file format. The library does the following:

- It provides low-level, direct access layer, enabling an application subprogram to read and write specific image files, lines, or pixels, and manipulate metadata directly.
- Two coding/decoding subprograms ("codecs" for short) based on the Java Advanced Imaging (JAI) software provide access to VICAR and PDS images in a file-format-independent manner. The VICAR and PDS codecs enable any program that conforms to the specification of the JAI codec to use VICAR or PDS images automatically, without specific knowledge of the VICAR or PDS format.
- The library also includes Image I/O plug-in subprograms for VICAR and PDS formats. Application programs that conform to the Image I/O specification of Java version 1.4 can utilize any image format for which such a plug-in subprogram exists, without specific knowledge of the format itself. Like the aforementioned codecs, the VICAR and PDS Image I/O plug-in subprograms support reading and writing of metadata.

This program was written by Robert Deen and Steven Levoe of Caltech for **NASA's Jet Propulsion Laboratory***. For further information, access the Technical Support Package (TSP)* ***free on-line at www.nasatech.com****.*

This software is available for commercial licensing. Please contact Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-30470.

Software for Estimating Costs of Testing Rocket Engines

A high-level parametric mathematical model for estimating the costs of testing rocket engines and components at Stennis Space Center has been implemented as a Microsoft Excel program that generates multiple spreadsheets. The model and the program are both denoted, simply, the Cost Estimating Model (CEM). The inputs to the CEM are the parameters that describe particular tests, including test types (component or engine test), numbers and duration of tests, thrust levels, and other parameters. The CEM estimates anticipated total project costs for a specific test. Estimates are broken down into testing categories based on a work-breakdown structure and a cost-element structure. A notable historical assumption incorporated into the CEM is that total labor times depend mainly on thrust levels. As a result of a recent modification of the CEM to increase the accuracy of predicted labor times, the dependence of labor time on thrust level is now embodied in third- and fourth-order polynomials.

This program was developed by Merlon M. Hines of Lockheed Martin Space Operations 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 [see page 1]. Refer to SSC-00154.