Computer Programs

Electronic Systems

Software for a GPS-Reflection Remote-Sensing System

A special-purpose software Global Positioning System (GPS) receiver designed for remote sensing with reflected GPS signals is described in "Delay/Doppler-Mapping GPS-Reflection Remote-Sensing System" (NPO-30385), which appears elsewhere in this issue of *NASA Tech Briefs*. The input accepted by this program comprises raw (open-loop) digitized GPS signals sampled at a rate of about 20 MHz. The program processes the data samples to perform the following functions: detection of signals; tracking of phases and delays; mapping of delay, Doppler, and delay/Doppler waveforms; dual-frequency processing; coherent integrations as short as 125 μs; decoding of navigation messages; and precise time tagging of observable quantities. The software can perform these functions on all detectable satellite signals without dead time. Open-loop data collected over water, land, or ice and processed by this software can be further processed to extract geophysical information. Possible examples include mean sea height, wind speed and direction, and significant wave height (for observations over the ocean); bistatic-radar terrain images and measures of soil moisture and biomass (for observations over land); and estimates of ice age, thickness, and surface density (for observations over ice).

This program was written by Stephen Lowe 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-30386.

Mathematics and Information Sciences

Software for Building Models of 3D Objects via the Internet

The Virtual EDF Builder (where "EDF" signifies Electronic Development Fixture) is a computer program that facilitates the use of the Internet for building and displaying digital models of three-dimensional (3D) objects that ordinarily comprise assemblies of solid models created previously by use of computer-aided-design (CAD) programs. The Virtual EDF Builder resides on a Unix-based server computer. It is used in conjunction with a commercially available Web-based plug-in viewer program that runs on a client computer. The Virtual EDF Builder acts as a translator between the viewer program and a database stored on the server. The translation function includes the provision of uniform resource locator (URL) links to other Web-based computer systems and databases. The Virtual EDF builder can be used in two ways: (1) If the client computer is Unix-based, then it can assemble a model locally; the computational load is transferred from the server to the client computer. (2) Alternatively, the server can be made to build the model, in which case the server bears the computational load and the results are downloaded to the client computer or workstation upon completion.

This program was written by Tim Schramer and Jeff Jensen of Boeing North American, Inc., for **Johnson Space Center***. For further information, access the Technical Support Package (TSP)* ***free on-line at www.nasatech.com****.*
MSC-22988

"Virtual Cockpit Window" for a Windowless Aerospacecraft

A software system processes navigational and sensory information in real time to generate a three-dimensional-appearing image of the external environment for viewing by crewmembers of a windowless aerospacecraft. The design of the particular aerospacecraft (the X-38) is such that the addition of a real transparent cockpit window to the airframe would have resulted in unacceptably large increases in weight and cost.

When exerting manual control, an aircrew needs to see terrain, obstructions, and other features around the aircraft in order to land safely. The X-38 is capable of automated landing, but even when this capability is utilized, the crew still needs to view the external environment: From the very beginning of the United States space program, crews have expressed profound dislike for windowless vehicles. The well-being of an aircrew is considerably promoted by a three-dimensional view of terrain and obstructions. The present software system was developed to satisfy the need for such a view. In conjunction with a computer and display equipment that weigh less than would a real transparent window, this software system thus provides a "virtual cockpit window."

The key problem in the development of this software system was to create a realistic three-dimensional perspective view that is updated in real time. The problem was solved by building upon a pre-existing commercial program — LandForm C3 — that combines the speed of flight-simulator software with the power of geographic-information-system software to generate real-time, three-dimensional-appearing displays of terrain and other features of flight environments. In the development of the present software, the pre-existing program was modified to enable it to utilize real-time information on the position and attitude of the aerospacecraft to generate a view of the external world as it would appear to a person looking out through a window in the aerospacecraft. The development included innovations in realistic horizon-limit modeling, three-dimensional stereographic display, and interfaces for utilization of data from inertial-navigation devices, Global Positioning System receivers, and laser rangefinders. Map and satellite imagery from the National Imagery and Mapping Agency can also be incorporated into displays.

After further development, the present software system and the associated display equipment would be capable of providing a data-enriched view: In addition to terrain and obstacles as they would be seen through a cockpit window, the view could include flight paths, landing zones, aircraft in the vicinity, and unobstructed views of portions of the terrain that might otherwise be hidden from view. Hence, the system could also contribute to safety of flight and landing at night or under conditions of poor visibility.

In recent tests, so precise was the software modeling that during the initial phases of the flight the software running on a monitor beside the video camera produced nearly identical views.

This work was done by Michael F. Abernathy of Rapid Imaging Software, Inc., for **Johnson Space Center***. For fur-*

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.