Phoenix mission. The products generated by this program were part of the RA commanding process, as well as the SSI, RAC, OM, and MECA image and science analysis process. Its output products were used to advance science of the near polar regions of Mars, and were used to prove that water is found in abundance there.

Phxtelemproc is part of the MIPL (Multi-mission Image Processing Laboratory) system. This software produced Level 1 products used to analyze images returned by *in situ* spacecraft. It ultimately assisted in operations, planning, commanding, science, and outreach.

This work was done by Alice Stanboli of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov.

This software is available for commercial licensing. Please contact Dan Broderick at Daniel.F.Broderick@jpl.nasa.gov. Refer to NPO-47801.

Contact Graph Routing Enhancements Developed in ION for DTN

NASA's Jet Propulsion Laboratory, Pasadena, California

The Interplanetary Overlay Network (ION) software suite is an open-source, flight-ready implementation of networking protocols including the Delay/Disruption Tolerant Networking (DTN) Bundle Protocol (BP), the CCSDS (Consultative Committee for Space Data Systems) File Delivery Protocol (CFDP), and many others including the Contact Graph Routing (CGR) DTN routing system. While DTN offers the capability to tolerate disruption and long signal propagation delays in transmission, without an appropriate routing protocol, no data can be delivered.

CGR was built for space exploration networks with scheduled communication opportunities (typically based on trajectories and orbits), represented as a contact-graph. Since CGR uses knowledge of future connectivity, the contact graph can grow rather large, and so efficient processing is desired. These enhancements allow CGR to scale to predicted NASA space network complexities and beyond.

This software improves upon CGR by adopting an earliest-arrival-time cost metric and using the Dijkstra path selection algorithm. Moving to Dijkstra path selection also enables construction of an earliest-arrival-time tree for multicast routing. The enhancements have been rolled into ION 3.0 available on sourceforge.net.

This work was done by John S. Segui and Scott Burleigh of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov.

This software is available for commercial licensing. Please contact Dan Broderick at Daniel.F.Broderick@jpl.nasa.gov. Refer to NPO-48186.

GFEChutes Lo-Fi

Lyndon B. Johnson Space Center, Houston, Texas

NASA needed to provide a software model of a parachute system for a manned re-entry vehicle. NASA has parachute codes, e.g., the Descent Simulation System (DSS), that date back to the Apollo Program. Since the space shuttle did not rely on parachutes as its primary descent control mechanism, DSS has not been maintained or incorporated into modern simulation architectures such as Osiris and Antares, which are used for new mission simulations. GFEChutes Lo-Fi is an object-oriented implementation of conventional parachute codes designed for use in modern simulation environments.

The GFE (Government Furnished Equipment), low-fidelity (Lo-Fi) parachute model (GFEChutes Lo-Fi) is a software package capable of modeling the effects of multiple parachutes, deployed concurrently and/or sequentially, on a vehicle during the subsonic phase of reentry into planetary atmosphere. The term "low-fidelity" distinguishes models that represent the parachutes as simple forces acting on the vehicle, as opposed to independent aerodynamic bodies. GFEChutes Lo-Fi was created from these existing models to be clean, modular, certified as NASA Class C software, and portable, or "plug and play."

The GFE Lo-Fi Chutes Model provides basic modeling capability of a sequential series of parachute activities. Actions include deploying the parachute, changing the reefing on the parachute, and cutting away the parachute. Multiple chutes can be deployed at any given time, but all chutes in that case are assumed to behave as individually isolated chutes; there is no modeling of any interactions between deployed chutes. Drag characteristics of a deployed chute are based on a coefficient of drag, the face area of the chute, and the local dynamic pressure only. The orientation of the chute is approximately modeled for purposes of obtaining torques on the vehicle, but the dynamic state of the chute as a separate entity is not integrated — the treatment is simply an approximation.

The innovation in GFEChutes Lo-Fi is to use an object design that closely followed the mechanical characteristics and structure of a physical system of parachutes and their deployment mechanisms. Software objects represent the components of the system, and use of an object hierarchy allows a progression from general component outlines to specific implementations. These extra chutes were not part of the baseline deceleration sequence of drogues and mains, but still had to be simulated. The major innovation in GFEChutes Lo-Fi is the software design and architecture.

This work was done by Emily Gist, Gary Turner, Robert Shelton, Mana Vautier, and Ashraf Shaikh of Odyssey Space Research. LLC for Johnson Space Center. Further information is contained in a TSP (see page 1). MSC-25004-1