modified recursive hierarchical segmentation (RHSEG) of data have been developed. While the current implementation is for two-dimensional data having spatial characteristics (e.g., image, spectral, or spectral-image data), the generalized algorithm also applies to three-dimensional or higher dimensional data and also to data with no spatial characteristics. The algorithm and software are modified versions of a prior RHSEG algorithm and software, the outputs of which often contain processing-window artifacts including, for example, spurious segmentation-image regions along the boundaries of processing-window edges. The modification consists of the addition of an efficient subroutine through which pairs of regions are identified that may contain pixels that are actually more similar to other regions in the pair. Once these pairs of regions are identified, pixels in one region that are more similar to pixels in the other region are reassigned to the other region. The subroutine is computationally efficient because it focuses only on those regions that could potentially contribute to the processing-window artifacts. In addition, any adverse effect of the subroutine on the computational efficiency of the algorithm is minimized by executing the subroutine at a point in the algorithm such that switching of pixels between regions that are subsequently merged is avoided.

"This program was written by Gary Moser of Goddard Space Flight Center, and Paul Stone and Christopher Hollery of Constellation Software Engineering Corp. Further information is contained in a TSP (see page 1)." GSC-14827-1

**Integrated Modeling Environment**
The Integrated Modeling Environment (IME) is a software system that establishes a centralized Web-based interface for integrating people (who may be geographically dispersed), processes, and data involved in a common engineering project. The IME includes software tools for life-cycle management, configuration management, visualization, and collaboration. It enables organized, efficient communication of engineering analyses and the statuses thereof. Key functions performed by use of the IME include creation, further development, and management of modeling analyses over their entire life cycles; publishing model and analysis information for availability and reuse throughout the user community; and managing legacy information without regard to original formats, database organizations, or computing platforms. The use of the IME creates an archive of analysis results, plus documentation that identifies the assumptions and data elements used for each analysis. This archive is configured to enable reuse of previous analysis results, and tracing of types and versions of software used for each step of each analysis. The IME utilizes a customized version of a commercial product-life-cycle-management application program that provides rich capabilities for managing configurations, workflows, data, and access through a single Web-based environment.

"This program was written by James C. Tilton of Goddard Space Flight Center. For further information, contact the Goddard Innovative Partnerships Office at (301) 286-5810. GSC-14681-1"

**Sizing Structures and Predicting Weight of a Spacecraft**
EZDESIT is a computer program for choosing the sizes of structural components and predicting the weight of a spacecraft, aircraft, or other vehicle. In designing a vehicle, EZDESIT is used in conjunction with a finite-element structural-analysis program: Each structural component is sized within EZDESIT to withstand the loads expected to be encountered during operation, then the weights of all the structural finite elements are added to obtain the structural weight of the vehicle. The sizing of the structural components elements also alters the stiffness properties of the finite-element model. The finite-element analysis and structural component sizing are iterated until the weight of the vehicle converges to a prescribed iterative difference. The results of the sizing can be reviewed in two ways:
1. An interactive session of the EZDESIT program enables review of the results in a table that shows component types, component weights, and failure modes; and
2. The results are read into a finite-element preprocess-and-postprocessing program and displayed on a graphical representation of the model.

"This program was written by Jeffrey Cerro and C. P. Shore of Langley Research Center. Further information is contained in a TSP (see page 1)." LAR-16878-1

**Stress Testing of Data-Communication Networks**
NetStress is a computer program that stress-tests a data-communication network and components thereof. NetStress comprises two components running, respectively, in a transmitting system and a receiving system connected to a network under test. The novelty of the program is that it has the capability to generate/receive varied network loading traffic profiles, which prior known programs were incapable of producing (i.e., various packet sizes and various packet rates all combined to make a pseudo-random traffic pattern). The transmitting-system component generates increasingly stressful data traffic for transmission via the network. The receiving-system component analyzes the resulting traffic arriving in the receiving system, generating such statistics as the number of data packets successfully received, the number of dropped packets, and the number of packets received out of order. The packet sizes must be configured before the transmitting-system component is started, but the packet frequencies, numbers of packets in bursts, and burst times can be configured during execution. Typically, a test begins with transmission of data at low sustained rates. Then the sustained rates are increased and burst rates are modified while monitoring to determine whether the receiving-system component reports any losses. When significant losses are reported, the user seeks to determine whether a malfunction or defi-
ciency has been found or normal net-
work saturation has been attained. Net-
Stress was written for execution in the
VxWorks real-time operating system, but
could easily be ported to other operat-
ing systems.

This program was written by Kurt Leucht
and Guy Bedette of Kennedy Space
Center. For further information, contact the
Kennedy Innovative Partnerships Office at
(321) 861-7158.

KSC-12589

Framework for Flexible
Security in Group
Communications

The Antigone software system defines
a framework for the flexible definition
and implementation of security policies
in group communication systems.
Antigone does not dictate the available
security policies, but provides high-level
mechanisms for implementing them. A
central element of the Antigone archi-
tecture is a suite of such mechanisms
comprising micro-protocols that provide
the basic services needed by secure
groups. Policies are implemented
through the composition and configura-
tion of these mechanisms. Mechanisms
are composed in different ways to ad-
dress new requirements and environ-
mental constraints. The Antigone frame-
work provides an easy-to-use application
programming interface (API), from
which secure group application pro-
grams can be built. Written entirely in
the C++ programming language, the sys-
tem consists of over 18,000 lines of
source code and has been ported to sev-
eral versions of Linux, FreeBSD, and
SunOS. Information for accessing re-
cent versions of the source code and re-
lated documentation is available at http://antigone.eecs.umich.edu.

This program was written by Patrick Mc-
Daniel and Atul Prakash of the University of
Michigan for Kennedy Space Center.

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:
Electrical Engineering and
Computer Sciences Department
University of Michigan
3115 EECS
1301 Beal Ave.
Ann Arbor, MI 48109
Refer to KSC-12207, volume and number
of this NASA Tech Briefs issue, and the
page number.

Software for Collaborative
Use of Large Interactive
Displays

The MERBoard Collaborative Work-
space, which is currently being deployed
to support the Mars Exploration Rover
(MER) Missions, is the first instantiation
of a new computing architecture de-
signed to support collaborative and
group computing using computing de-
vices situated in NASA mission opera-
tions rooms. It is a software system for
generation of large-screen interactive
displays by multiple users. The architec-
ture provides a platform and applica-
tions programming interface (API) for
the development of collaborative appli-
cations for NASA mission operations.
The standard deployment configuration
provides an integrated whiteboard, Web
browser, remote viewing and control for
collaboration over distance, and per-
sonal and group storage spaces that pro-
vide ubiquitous access and sharing of
data. Customization for specific domains
is provided through plugins. For the
MER mission, plugins include a flow-
charting tool for strategic rover oper-
a tions and mission planning, 3D visualiza-
tion of the Martian terrain, a data
navigator to navigate the mission data-
base, and situational awareness tools.
The MERBoard software is designed to
run on large plasma displays with touch-
screen overlays, thus providing an im-
mersive and interactive environment for
teams to view, annotate, and share data.
The MERBoard overcomes the obstacles
to communication, retention, and col-
laborative modification of information
in diverse forms that can include text,
data (including images) from scientific
instruments, handwritten notes, hand
drawings, and computer graphics. The
MERBoard provides a unifying interface
for the integration of heterogeneous ap-
plications, and provides those applica-
tions with a consistent model for saving
and retrieving data. All applications may
be viewed and controlled from any loca-
tion that has a MERBoard. A personal
client provides integration of a user’s
personal computing environment with
the MERBoard environment.

This program was written by Jay Trimble,
Thodore Shah, Roxana Wales, Alonso Vera,
Irene Tollinger, Michael McCurdy, and
Dmitry Lyubimov at Ames Research Cen-
ter. For further information, contact the
Ames Technology Partnerships Division at
(650) 604-2954.

ARC-14951-1