Trade Space Specification Tool (TSST) for Rapid Mission Architecture (Version 1.2)

NASA’s Jet Propulsion Laboratory, Pasadena, California

Trade Space Specification Tool (TSST) is designed to capture quickly ideas in the early spacecraft and mission architecture design and categorize them into trade space dimensions and options for later analysis. It is implemented as an Eclipse RCP Application, which can be run as a standalone program. Users rapidly create concept items with single clicks on a graphical canvas, and can organize and create linkages between the ideas using drag-and-drop actions within the same graphical view. Various views such as a trade view, rules view, and architecture view are provided to help users to visualize the trade space.

This software can identify, explore, and assess aspects of the mission trade space, as well as capture and organize linkages/dependencies between trade space components. The tool supports a user-in-the-loop preliminary logical examination and filtering of trade space options to help identify which paths in the trade space are feasible (and preferred) and what analyses need to be done later with executable models. This tool provides multiple user views of the trade space to guide the analyst/team to facilitate interpretation and communication of the trade space components and linkages, identify gaps in combining and selecting trade space options, and guide user decision-making for which combinations of architectural options should be pursued for further evaluation.

This software provides an environment to capture mission trade space elements rapidly and assist users for their architecture analysis. This is primarily focused on mission and spacecraft architecture design, rather than general-purpose design application. In addition, it provides more flexibility to create concepts and organize the ideas. The software is developed as an Eclipse plug-in and potentially can be integrated with other Eclipse-based tools.

This work was done by You-Yang Wang, Mitchell Schroock, Chester S. Borden, and Robert C. Moeller 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 Dan Broderick at Daniel.F.Broderick@jpl.nasa.gov. Refer to NPO-48158.
Memory-Efficient Onboard Rock Segmentation

NASA’s Jet Propulsion Laboratory, Pasadena, California

Rockster-MER is an autonomous perception capability that was uploaded to the Mars Exploration Rover Opportunity in December 2009. This software provides the vision front end for a larger software system known as AEGIS (Autonomous Exploration for Gathering Increased Science), which was recently named 2011 NASA Software of the Year. As the first step in AEGIS, Rockster-MER analyzes an image captured by the rover, and detects and automatically identifies the boundary contours of rocks and regions of outcrop present in the scene. This initial segmentation step reduces the data volume from millions of pixels into hundreds (or fewer) of rock contours. Subsequent stages of AEGIS then prioritize the best rocks according to scientist-defined preferences and take high-resolution, follow-up observations (see figure). Rockster-MER has performed robustly from the outset on the Mars surface under challenging conditions.

Rockster-MER is a specially adapted, embedded version of the original Rockster algorithm ("Rock Segmentation Through Edge Regrouping," (NPO-44417) Software Tech Briefs, September 2008, p. 25). Although the new version performs the same basic task as the original code, the software has been (1) significantly upgraded to overcome the severe onboard resource limitations (CPU, memory, power, time) and (2) bullet-proofed through code reviews and extensive testing and profiling to avoid the occurrence of faults. Because of the limited computational power of the RAD6000 flight processor on Opportunity (roughly two orders of magnitude slower than a modern workstation), the algorithm was heavily tuned to improve its speed. Several functional elements of the original algorithm were removed as a result of an extensive cost/benefit analysis conducted on a large set of archived rover images. The algorithm was also re-