Cost Reductions From Multi-Mission Sequencing Software

Laura Needels
Jet Propulsion Laboratory/California Institute of Technology
Pasadena, California

For the Fifth International Symposium on Reducing the Cost of Spacecraft Ground Systems and Operations in Pasadena, CA
July 8-12, 2003

Sequencing software for deep space missions has historically been one of the most critical parts of the ground software used to communicate with and control the spacecraft. At JPL, the sequencing software is responsible for planning and creation of science and engineering activities, checking command syntax, checking mission and flight rules, and translating the commands into packets which can be uplinked to the spacecraft. Significant effort has been spent by earlier missions to ensure the integrity of this software since errors in this area could cause the spacecraft to enter fault protection or cause the loss of the spacecraft.

Over the last several years, the Advanced Multi-Mission Operations System (AMMOS), part of the Interplanetary Network Directorate, has spent considerable effort to reduce the costs associated with sequencing software in two ways. The first way is by developing a multi-mission form of sequencing software. The sequencing software is now developed as two separate components. The multi-mission "core" software provides, in a generic sense, the capability to perform the functions needed in the sequencing software: planning and scheduling events, checking flight rules, and packetizing commands. The "core" software is then "adapted" to a project specific mission. The "adaptation" part of the software task involves providing the models for activities needed for planning and scheduling, converting the Command List for the project into models that can be used for sequence checking, coding project and mission flight rules and the modeling needed to support them, and developing project blocks for repetitive activities. The second way is by developing an Automatic Sequence Processor (ASP) that automates the generation and uplink of most of the sequences sent to missions that use the ASP.

These two efforts have reduced the development and operations costs for JPL flight projects. By partitioning the sequencing software into a "core" component and an "adaptation" component, software verification costs have been reduced. The core component of the software is verified once, by a central group, rather than by multiple projects. Projects can then focus on verification of just the adaptation part of the software. By implementing an ASP, operations costs are reduced because staff is no longer needed process the sequences. Both of these efforts have produced additional cost savings because both adaptation and operation teams are multi-mission, which allows personnel to move from one project to another with minimal start-up and training time.

This paper gives a brief overview of the multi-mission software described above and includes data on the cost savings obtained by using this methodology.