Navigation and Mission Analysis Software for the Next Generation of JPL Missions

Steve Flanagan[1], Ted Drain[1], Todd Ely[1], Tomas Martin-Mur[2]

Abstract - MONTE (**M**ission analysis and **O**perational **N**avigation **T**oolkit **E**nvironment) is a new software system being developed to replace the navigation and trajectory analysis software currently in use in JPL's Navigation and Mission Design Section. MONTE will reproduce the existing functionality of the legacy systems, and also add significant new capabilities for the MONTE users - the mission and navigation analysts in the Section. Furthermore, MONTE will be developed as a single tightly integrated system, in contrast to the multiple disparate libraries currently in use.

MONTE is being designed to facilitate a variety of navigation and trajectory tasks in a broad range of contexts, including research and development, analysis and design, and operations. Several examples of how MONTE can be used include,

- Plan and design spacecraft trajectories that satisfy mission objectives,
- Estimate and control the actual spacecraft trajectories throughout the mission,
- Maintain and disseminate knowledge of the spacecraft trajectories,
- Provide related information for solar system bodies such as planetary orbits, pole orientations and rotation rates, etc.
- Develop new algorithms for trajectory targeting; conduct feasibility studies on new orbit determination techniques, etc.

Note that the MONTE system will be capable of supporting missions with multiple spacecraft as well as the more traditional interplanetary mission with only a single spacecraft. Currently, work on MONTE is focused primarily on addressing requirements for a ground-based mission design and navigation system; however, future adaptations of MONTE are planned that will provide navigation functions for embedded flight systems as well.

It is reasonable to ask why MONTE is needed at this time and how it is different from (and superior to) the navigation and mission design software tools currently in use. In addition to the new functionality that MONTE is delivering, MONTE is promoting a new software development and use paradigm that is intended to address the following key objectives.

- **Reduce life-cycle cost**

One of the strongest arguments in favor of pursuing this development effort is that by doing so, it is possible to reduce the overall lifecycle cost associated with maintaining the legacy software. Maintaining this software as robust, reliable software systems that meet the needs of their users is an expensive proposition, in part due to the fact that they are based on aging technology, and also because of the way these systems have evolved over time. Building new software from the ground up is expected to result in a system that is easier, and therefore less expensive, to maintain, without sacrificing functionality. One

[1]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109
[2]Raytheon ITSS, Pasadena, CA 91101

key element of this plan is to provide accurate and current documentation of software requirements, design, and code for all software components in the MONTE system.

- **Improve user-interaction**

Most of the legacy software currently in use provides Command-Line User Interfaces (CLUIs) exclusively. The appropriate application of Graphical User Interfaces (GUIs) will significantly enhance the user-interaction characteristics of the new software. Well-designed and constructed GUIs will make the software more accessible to the novice, without getting in the way of the expert. Standardized GUIs will also make it easier for the Section or for a Flight Project to provide officially approved approaches to using the software.

It would be inappropriate, however, to assume that GUIs are preferable to CLUIs in all applications. The prospective users of MONTE have repeatedly voiced an interest in having alternatives to GUIs. Therefore, another user-interface goal of MONTE is to provide a common interactive command-line and scripting capability to both Navigation and Mission Analysis users, similar to the Quick tool that is currently used in the Mission Analysis area.

- **Provide improved functionality and responsiveness to new requirements**

The domains of Navigation and Mission Analysis are dynamic, and it is reasonable to expect that MONTE's user requirements will change somewhat over time. As a result, it will never be possible to anticipate all potential uses of the software at the time of design and implementation. Many new requirements have come out of the demanding missions attempted by JPL in the years since the legacy software was originally developed. This situation has challenged the Section's ability to maintain and extend these software sets. MONTE must address the additional requirements expected based on our present knowledge of the missions in NASA's roadmap and be able to respond readily to new requirements that cannot be anticipated. This can only be accomplished through the careful design and documentation of an extensible and maintainable software system.

- **Integrate Mission Analysis and Navigation software**

In spite of the tremendous functional overlap between Navigation and Mission Analysis, these two areas have always maintained separate core software libraries. This has introduced a number of problems, with the two most important being maintenance cost and interoperability. Maintaining two software systems with very similar functional requirements is clearly more expensive than maintaining a single library, and it is difficult to find new people with the necessary FORTRAN skills. Also, it is very common for Mission Analysts to deliver solutions to Navigation Analysts. This is currently a complicated process and can contribute to errors being made. This situation is unnecessary. Building a single library that can be shared by these two areas will introduce economies and efficiencies that cannot be realized any other way.

This paper will describe in more detail the programmatic objectives of MONTE, the key functional requirements that the MONTE system must address, and relevant features of the high-level design that is being applied to the problem. We will also describe some elements of the approach to software development that is being followed on this project.

Acknowledgement - The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.