Proven and Robust Ground Support Systems
- GSFC Success and Lessons Learned

Barbara Pfarr, NASA/GSFC, Barbara.B.Pfarr@nasa.gov, 301-286-7624
John Donohue, NASA/GSFC, John.T.Donohue@nasa.gov, 301-286-8731
Ben Lui, NASA/GSFC, Ben.Y.Lui@nasa.gov, 301-286-5984
Greg Greer, the Hammers Company, ggreer@hammers.com, 301-286-5999
Tom Green, Design America Inc., tgreen@www.dai-asist.com, 301-286-9915

NASA Goddard Space Flight Center
Greenbelt, MD 20771

Abstract—Over the past fifteen years, Goddard Space Flight Center has developed several successful science missions in-house: the Wilkinson Microwave Anisotropy Probe (WMAP), the Imager for Magnetopause-to-Aurora Global Exploration (IMAGE), the Earth Observing 1 (EO-1) [1], and the Space Technology 5 (ST-5) [2] missions, several Small Explorers, and several balloon missions. Currently in development are the Solar Dynamics Observatory (SDO) [3] and the Lunar Reconnaissance Orbiter (LRO) [4]. What is not well known is that these missions have been supported during spacecraft and/or instrument integration and test, flight software development, and mission operations by two in house satellite Telemetry and Command (T&C) systems, the Integrated Test and Operations System (ITOS) and the Advanced Spacecraft Integration and System Test (ASIST).

The advantages of an in-house satellite Telemetry and Command system are primarily in the flexibility of management and maintenance - the developers are considered a part of the mission team, get involved early in the development process of the spacecraft and mission operations control center, and provide on-site, on-call support that goes beyond Help Desk and simple software fixes. On the other hand, care must be taken to ensure that the system remains generic enough for cost effective re-use from one mission to the next. The software is designed such that many features are user-configurable. Where user-configurable options were impractical, features were designed so as to be easy for the development team to modify. Adding support for a new ground message header, for example, is a one-day effort because of the software framework on which that code rests.

This paper will discuss the many features of the Goddard satellite Telemetry and Command systems that have contributed to the success of the missions listed above. These features include flexible user interfaces, distributed parallel commanding and telemetry decommutation, a procedure language, the interfaces and tools needed for a high degree of automation, and instantly accessible archives of spacecraft telemetry. It will discuss some of the problems overcome during development, including secure commanding over networks or the Internet, constellation support for the three satellites that comprise the ST-5

mission, and geographically distributed telemetry end users. [1][2]

TABLE OF CONTENTS

1. INTRODUCTION

Since 1959, NASA Goddard Space Flight Center (GSFC) has been in the forefront of space explorations. Goddard's mission is to expand knowledge of the Earth and its environment, the Solar System, and the Universe through observations from Space. In order to accomplish its mission, Goddard has committed its diverse workforce and resources to design, develop, test, launch, and operate spacecrafts and instruments particularly for Earth and space science. Over the years, Goddard has successfully launched several dozen satellites into orbit. These missions, such as Solar Anomalous and Magnetospheric Particle Explorer (SAMPEX), Transition Region and Coronal Explorer (TRACE), Rossi X-ray Timing Explorer (RXTE), and Earth Observing 1 (EO-1), have provided valuable data to the science community.

The complexity of these missions is mind-boggling due to the level of technology utilized and the first-of-a-kind

science being acquired or investigated. The success of these missions relies heavily on highly sophisticated ground data systems to support their development, integration and test (I&T), as well as operations. The Telemetry and Command (T&C) system, one of the key components of the ground data system, is used to send commands to the spacecraft in real-time as well as receive and analyze telemetry data.

There are numerous commercial T&C systems available in the aerospace industry today. Although commercial T&C systems provide a technically sound solution, in house T&C systems are an effective solution for Goddard in-house end-to-end missions. This paper will discuss the two Goddard in-house T&C systems (ITOS and ASIST), the advantage of using them, the teams that support them, as well as some lessons learned.

2. GODDARD IN-HOUSE TELEMETRY AND COMMAND SYSTEMS

Work on the software and system that would become ITOS began in 1990 by a small combined civil servant and contractor team. The original development team was charged with creating a telemetry and command system to support development, integration, and testing for the Small Explorer Program, beginning with the SAMPEX mission. This "Test Conductor's Workstation" software moved into mission operations and became ITOS during development of the second set of SMEX missions which were launched in 1998 and 1999. Subsequently, ITOS became the mission operations telemetry and command system for all Goddard-developed SMEX missions, and for the RHESSI SMEX spacecraft developed by the University of California, Berkeley and the Spectrum Astro Corp.

ITOS also began supporting larger missions with the Triana and Swift projects, and currently supports very large projects/spacecraft such as the Gamma Ray Large Area Space Telescope (GLAST) and the Lunar Reconnaissance Orbiter (LRO). At the same time, ITOS supports smaller, lower-cost projects such as Space Environment Testbed (SET-1) and the Ultra-Long Duration Balloon/Cosmic Ray Energetics And Mass experiment (ULDB/CREAM). The ITOS team currently includes five on-site contractor engineers and two to three off-site contractor engineers with part-time support from civil service engineers.

ITOS's full mission history is given in the following table:

Mission	Role	Launch
SAMPEX (SMEX)	Bus development, spacecraft (S/C) I&T, mission operations	1992
FAST (SMEX)	Bus development, S/C I&T, mission operations	1996
SWAS (SMEX)	Bus development, S/C I&T, mission operations	1998

Mission	Role	Launch
Spartan 201-05	Bus development, S/C I&T, mission operations	1998
TRACE (SMEX)	Bus development, S/C I&T, mission operations	1998
WIRE (SMEX)	Bus development, S/C I&T, mission operations	1999
CATSAT (STEDI)	Bus development, S/C I&T, mission operations	Cancelled
Spartan 250 / 400	Bus development	Cancelled
RHESSI (SMEX)	Bus development, S/C I&T, mission operations	2002
Triana	Bus and instrument development, S/C I&T, mission operations	Mothballed
Swift (MIDEX)	Bus and instrument development, S/C I&T, mission operations	2004
CREAM (ULDB)	Bus development, ballooncraft I&T, mission operations	2004, 2005, 2007, 2008
SET-1	Bus development, S/C I&T, mission operations	TBD
GLAST	Mission operations, high-rate telemetry front-end processors	2008
LRO	Bus and instrument development, S/C I&T, mission operations, high-rate data collection.	2008

Table 1

Mission operations for NASA's WIND and ACE missions are transitioning to ITOS over the next two years.

ITOS was commercialized by the Hammers Company approximately eight years ago. Missions supported by the Hammers Company with their commercialized version of ITOS include:

Mission	Customer	Role	Launch
SCISAT-1	Canadian Space Agency	Bus development, S/C I&T, mission operations	2003
SECCHI instrument	NRL	Instrument development, mission operations	2006
DRS instrument	JPL	Instrument flight software development	2010
THEMIS	Swales Aerospace / UC Berkeley	Bus development, S/C I&T, mission operations	2007
Cassiope	Canadian Space Agency	Bus development, S/C I&T, mission operations	2008
NEOSAT	Canadian Space Agency	Bus development, S/C I&T, mission operations	TBD

Table 2

In a similar way, the Advanced Spacecraft Integration and System Test (ASIST) ground system was first developed in the early 1990s by a small combined civil servant and contractor team to support the Rossi X-ray Timing Explorer (RXTE) and Tropical Rainfall Measuring Mission (TRMM) missions at GSFC. In fact, ASIST origins were in human factors studies done in the Flight Telerobotic Servicer lab with the first emphasis on developing an effective user interface. ASIST proved to be a key component in the integration and test ground system for all phases of satellite development, including flight hardware development, individual instrument integration and testing, and full observatory mission integration and testing.

ASIST's user interface, extended functionality, and proven track record historically have made it very successful for GSFC in-house projects. Such projects have included (in addition to RXTE and TRMM) EO-1, WMAP, IMAGE, ST-5 and SDO.

Several Goddard engineering organizations in various disciplines are responsible for in-house spacecraft development. These organizations consist of the Flight Software Systems Branch, Ground Software Systems Branch, as well as the Flight System Integration and Test (I&T) Branch. Teams from these organizations work closely together throughout the design, development, and testing phases.

These teams rely greatly on the in-house T&C systems for spacecraft development. For example, the flight software (FSW) team spends approximately 35 to 40% of their effort on testing their software. The testing effort depends heavily on the T&C system to provide capabilities for managing the command and telemetry database, flexibility of the System Test and Operations Language (STOL), ease of display pages creation, and flexibility to run STOL procedures. All the effort and capabilities used by FSW testing will also be used later by the FSW maintenance team.

Both ITOS and ASIST were developed specifically to support Goddard's approach to FSW test as well as spacecraft I&T. Using either of these systems for Goddard in-house missions allows for rapid responses in meeting the needs and deadlines of the tight development schedules that usually transpire in the aerospace industry. In addition, these systems are highly compatible with the Goddard flight software design and thus making them the preferred choice of T&C systems for Goddard in-house missions.

3. INTEGRATED TEST AND OPERATIONS SYSTEM (ITOS)

The ITOS is a user-configured, integrated collection of software comprising a tool for processing, analyzing, and displaying spacecraft and spacecraft component telemetry; for generating telecommands; and for performing automated operations and testing [5]. ITOS runs on UNIX systems, and is supported on Red Hat Enterprise Linux, Solaris, FreeBSD, and Mac OS X. The software was designed for missions using the Consultative Committee for Space Data Systems (CCSDS) [6] telemetry and command recommendations, and also handles non-CCSDS formats such as Time-Division Multiplexed (TDM) telemetry.

ITOS is configured principally through its telemetry and command database. The database defines telemetry data points, and provides information required for extracting those data values from the telemetry stream; and it defines telecommand mnemonic syntax and its translation into a binary telecommand stream. The database also provides for the definition of telemetry alarm limits and conversion to engineering units or state values.

ITOS can perform telemetry frame synchronization, Reed-Solomon decoding, CRC checking, and Pseudorandom Noise (PN) decoding in software for rates demonstrated at more than 40 Mbps on commodity PC personal computer hardware. ITOS handles CCSDS version 1 and 2 frames, CCSDS packets, performs packet extraction from the frame layer, and can process TDM and other telemetry formats. It provides full CCSDS Command Operations Procedure One (COP-1) command verification and Command Link Transmission Unit (CLTU) creation, and includes tools for managing absolute and relative time sequences and table loads. Also, ITOS supports file transfers to and from the spacecraft using the CCSDS File Delivery Protocol (CFDP) and other protocols.

ITOS can ingest telemetry data and send commands on multiple simultaneous telemetry streams, and supports a wide variety of transport protocols including TCP/IP, Space Link Extension (SLE), GSFC Mission Services Evolution Center (GMSEC), serial (232, 422, ECL, LVDS), MIL-STD-1553, and SpaceWire. Some of these interfaces -- the last two in particular -- make ITOS especially well suited for use in spacecraft instrument and component development because ITOS can communicate directly with the component under test, acting in the role of a medium fidelity spacecraft simulator. ITOS is compatible with a wide variety of ground stations, front-end processors, and ground message wrappers including NASA's Ground Network, Space Network, Deep Space Network; Universal Space Network; and others.

Telemetry displays are user-defined using a simple text language, or a drag-and-drop editor. Most displays may be viewed using a web browser with proper set-up of a web server on an ITOS computer. Display features include objects such as X-Y plots, strip charts, bar and dial gauges, and numeric displays.

ITOS provides a STOL interpreter for executing closed-loop test and operations procedures. STOL provides simple programming constructs such as variables, loops, and "if" tests. STOL procedures have access to all telemetry values and may send spacecraft commands and execute memory loads. STOL also provides operating system access, control over IEEE-488 interfaces and other ground instrumentation, access to the UNIX file system, and to the computer communication network.

The ITOS team has many years of experience with the product, and with spacecraft development at Goddard and elsewhere. Three of the original developers currently are part of the ITOS group, and two other members have more than 10 years experience with the product in the Goddard environment. Given the missions that have used the ITOS software, the team also has a breadth of experience that encompasses large and small missions, and all phases of development from board- and box-level through on-orbit operations.

ITOS has evolved much since its creation as the developers have adapted to evolving mission requirements. But because the on-site ITOS team have had the opportunity to work closely with mission developers, ITOS has evolved to incorporate many of those users' suggestions and desires, as well as their "hard" requirements. ITOS has, in a real sense, "grown up" beside the evolving capabilities of the in-house spacecraft builders so that ground and flight development teams can work symbiotically together.

When visiting development labs or the I&T area for a mission, ITOS engineers often are approached by ITOS users with questions and suggestions. The users would not bother to call or e-mail these items, but they are eager to communicate with them face-to-face when the opportunity arises. This is another advantage in having a continuous on-site presence during the development of the space mission.

But in spite of this close relationship with spacecraft developers, the ITOS team has consistently embraced the most general approach practical for solving specific mission problems rather than writing custom, project-specific solutions. When mission-specific code must be added

to ITOS, it usually is limited to areas designed to accommodate code extensions such as those needed to handle new ground message wrappers, or special command checksums. These extensions then remain in ITOS to be re-used by or adapted for other missions.

While it is not necessary for a telemetry and command product team to be "in-house", it is of great benefit to both the customer and the product itself to have dedicated on-site engineering support. This on-site staff is part of the overall mission development team. It can provide rapid responses to help diagnose problems in the space / ground interface, and quicker turn-around of fixes for ground system problems or software bugs. Additionally, the team is a ready resource for developing the small ground software tools that often are needed to assist in developing and testing spacecraft and their components, and which usually fit under the umbrella of the telemetry and command system.

4. ADVANCED SPACECRAFT INTEGRATION AND SYSTEM TEST (ASIST)

ASIST is a scalable real-time command and control system for spacecraft development, integration and test, and operations [7]. Designed as a database driven command, monitoring and control system, ASIST can support real-time operator commands or autonomous script based commanding which allows a broad range of missions to be supported.

ASIST's unique architecture (see Figure 1.0) supports parallel commanding, distributed decommutation, structured language databases, a compiled procedure language, powerful derived telemetry, and a fully integrated archival system. And importantly, ASIST executes on commodity PCs running LINUX. Projects in this way get very rich and
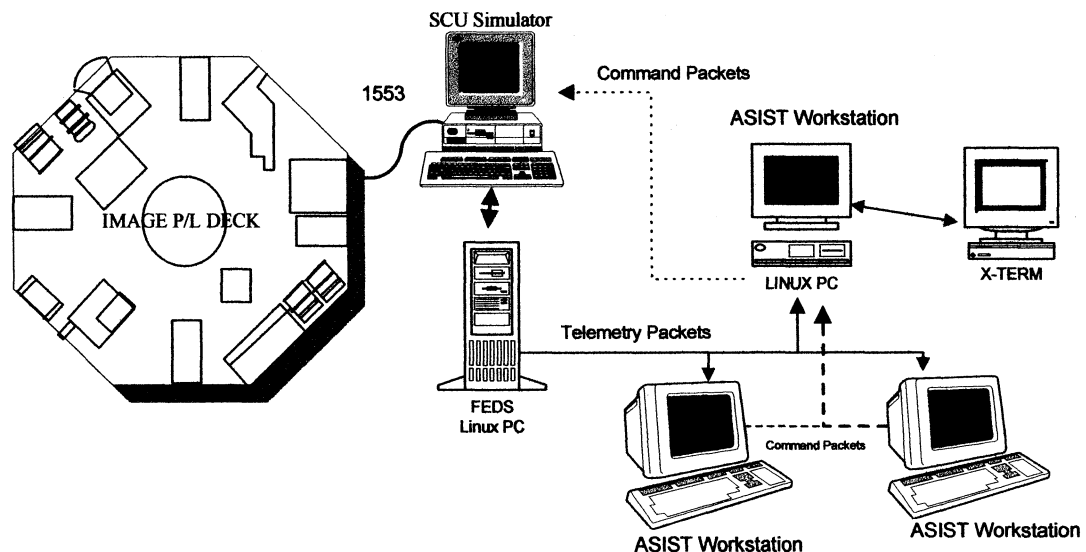

Figure 1. IMAGE Instrument Payload Deck I&T Configuration

4

deep functionality (the strength of ASIST) with an easy to use interface for both users and programmers on an inexpensive hardware platform.

At the heart of ASIST are its command/telemetry databases, and the ability to run procedures or 'procs' as they are known. Since the ASIST based ground system is scalable, as many ASIST workstations can be added as needed – aiding ground system growth as the mission transitions from component testing to on-orbit operations. Additional ASIST workstations can easily be added to the ground system because of the ASIST's distributed workstation architecture. The primary workstation controls the forward link and can enable and disable any workstation. Up to 31 workstations can command through a single primary workstation. This flexibility of ASIST allows the users (test conductors, spacecraft operators) to customize procedures, databases, display pages, and pseudo-telemetry.

Within ASIST are two main databases: the command database and the telemetry database. The command database in ASIST links user-defined command mnemonics to command packets defined in the database. All commands are defined in the single command database. These command mnemonics can be manually typed into the STOL command line, or called by procedures. Once a command is called within STOL and the spacecraft command packet definition is verified, the packet is then sent to the Front End Data System (FEDS) for packaging and distribution. Command packets are defined in CCSDS format. ASIST also supports "command backsolving", where a bit pattern can be analyzed to determine the possible command mnemonics that could have created the resultant command. The other database category on ASIST is the telemetry

database. Finalized prior to launch, the ASIST telemetry database matches the telemetry packets output by the spacecraft. Once ASIST receives a telemetry packet, ASIST matches the incoming data to its own packet definition based on the packet's Application Identification (APID). ASIST then decommutates the packet data into the individual telemetry points present in the data packet. Each telemetry point is tied to a user-defined mnemonic for easy reference.

Key to the internal design is the Current Value Table (CVT) (see Figure 2) where the most recent telemetry point values are stored. The user can then make graphical display pages that call specific telemetry mnemonics from the CVT for real-time display on the workstation.

An important feature designed into ASIST is the ability to create pseudo-telemetry points – new data points which are based on other existing telemetry points. A user can define a pseudo-telemetry point, which takes a CVT value and performs a math operation (in the form of a mathematical equation) on the CVT value.

Two types of derived telemetry are provided: periodic and event driven. The definitions are stored in the telemetry database and equations can be added, deleted or modified at runtime.

Very important for in-house projects is that ASIST has proven to be very robust and reliable. Building on hundreds of thousands of on-orbit mission hours, ASIST has proven to be a viable choice for the next in-house project thereby maximizing reuse and reducing costs while providing unparalleled functionality to the users.
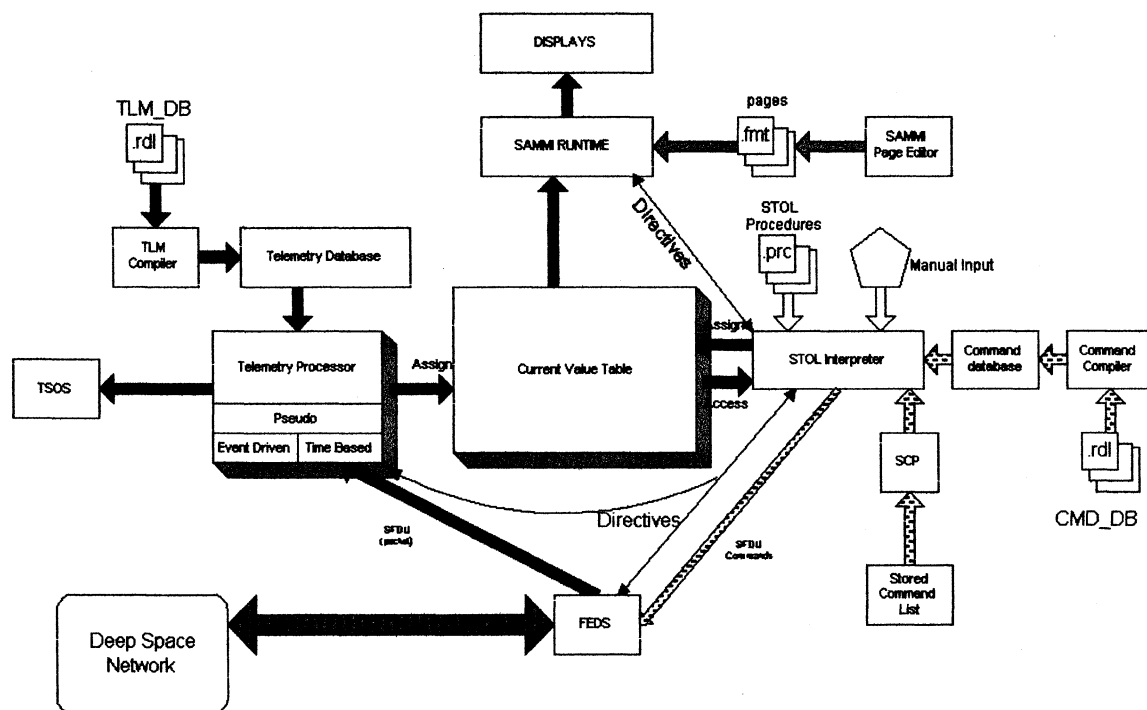

Figure 2 – Current Value Table (CVT) Interface

5. TELEMETRY AND COMMAND SYSTEM TEAM MEMBERS (THE TRUE ASSETS)

One of the key reasons for the success of Goddard's in house T&C systems is the make-up of the development teams. Both systems are being developed and maintained by small teams of approximately 5-8 people. Each team includes a couple of civil servants, and a few contractors from more than one company. Team members are defined by their software engineering skills, not their badges. Each team has its own lab in which all team members are co-located. There is turnover in the members of the team – that's unavoidable today. However, being collocated allows new team members to quickly come up to speed on the design and operations of these systems. The breakdown of barriers to teamwork, both physical and bureaucratic, has been crucial to the successful development of these systems.

Another reason for the success of Goddard's in house T&C systems in that each development team is also part of the overall mission team. T&C developers interact on a daily basis with the designers of the flight hardware and software systems, and therefore identify early any unique command or telemetry needs. They work closely with the flight software teams to ensure that their systems communicate with the flight software from the first flight software release. Flight software teams have requirements that differ from the mission operations team requirements – it's the T&C team's job to accommodate both. They participate in spacecraft integration and test, both to quickly identify errors in the T&C system as well as to suggest novel ways to track and identify potential spacecraft problems. A key to helping to identify problems early in development is on-site availability and rapid response to problems, which often involves a visit to a lab or the I&T facility. During integration and test, the T&C development team provides support via the Help Desk and as requested, 24 hours a day, seven days a week. Not only are their products integrated into the mission – they themselves are integrated. This recognition that hardware and software are interdependent, and to a lesser extent the ground and flight systems as well, means that simultaneously testing by integrated teams of hardware and software developers is critical to the successful development of modern satellites.

6. LESSONS LEARNED AND CONCLUSION

Telemetry and Command (T&C) systems are critical subsystems of a mission. These systems have broad functionality, perform real-time support, and some portions of these systems are more complex than spacecraft and instrument flight software. When implementing T&C systems for missions, utilizing an experienced technical team with a (repeatedly) proven solution increases the probability of success significantly.

This paper has reviewed the benefits of having an in-house T&C system to support GSFC/NASA missions. Choosing to implement an in-house system has proven successful for GSFC. However, given budget constraints and changes in the NASA vision, GSFC continues to seek innovative solutions for command, control and telemetry processing, including commercial solutions. Staying abreast of new functionality and cost/pricing models of commercial T&C systems is very important when trading in-house versus commercial solutions. For critical mission support, "system control" and "timely support" are key terms/concepts when discussing T&C implementations. Developing and maintaining in-house systems allows full control over enhancements, bug fixes, source code and maintenance in a timely fashion. Commercial systems can provide the same level of support when purchasing engineering services directly from the vendor. Vendors tend to provide high quality support to "good" customers, where the term "good" implies the expenditure of a significant amount of funds. Since NASA is (usually) not a high-dollar customer of a given vendor, timely support from a commercial vendor may be expensive, especially if the company is located out of state.

GSFC has recorded lessons learned regarding T&C implementations over the past fifteen years. Some of the lessons learned at GSFC on implementing in-house T&C solutions include:

1. Developing a plan for the long-term use and maintenance of in-house systems is beneficial for identifying funds for long-term sustaining engineering of these systems.

2. For commercial systems, ensure that the vendor has a well-documented pricing model for their system. When purchasing a system from a commercial vendor, negotiating a cost-escalation clause in the contract will help maintain costs of the system over the long-term. Once a system is "embedded" into an agency or organization, the agency/organization becomes tightly coupled to and dependent on that solution.

3. In selecting a T&C system, the ground system engineer must define and analyze T&C requirements for every phase of the mission when trading solutions. T&C support requirements for flight software development, spacecraft/instrument integration and test, and operations can vary dramatically. Solutions for operations generally do not meet the requirements for flight software development and spacecraft/instrument system integration and test. Substantial cost savings can be realized when using the same T&C system for all missions.

4. When the spacecraft development organization uses their own T&C solution, and the mission operations organization selects a different T&C system, the project must allocate funding and schedule for converting test procedures and T&C databases from one system to the other. In addition,

the plan for flight software maintenance may require the purchasing of an instance of the spacecraft vendors T&C system for testing purposes. This could add substantial cost to the operations/maintenance phase of the mission.

Recording and applying lessons learned are a vital part of any "learning" engineering organization. The application of these lessons has had a direct impact on the success of GSFC missions.

REFERENCES

[1] Earth Observing – 1 (EO-1) web site
http://earthobservatory.nasa.gov/Library/EO1/

[2] Space Technology 5 (ST-5) web site
http://nmp.nasa.gov/st5/

[3] Solar Dynamics Observatory (SDO) web site
http://sdo.gsfc.nasa.gov/

[4] Lunar Reconnaissance Orbiter (LRO) web site
http://lunar.gsfc.nasa.gov/

[5] Mike Prior et al. "Autonomous Operations of the WIRE Spacecraft," 2nd International Symposium on Spacecraft Ground Control and Data Systems,
Foz de Iguacu, Brazil, February 1999, p 4-5

[6] The Consultative Committee on Space Data Systems (CCSDS). Data Standards web site http://www.ccsds.org/ Internet.

[7] Julia Breed et al. "The MAP Autonomous Mission Control System," Space Ops 2000, Toulouse France, June 2000, p 3-4.

BIOGRAPHY

Barbara Pfarr is currently the Program Systems Engineer for the Geostationary Operational Environmental Satellites (GOES) R Program. Previously, she was an Associate Chief of the Information Systems Division of the Goddard Space Flight Center, head the Earth Sciences Missions Branch, which provided end-to-end systems engineering for NASA programs, missions and projects and head of the Real-Time Software Engineering Branch, which develops Command and Control systems and simulators. She served as the Program Chair for the AAS Goddard Memorial Symposium in March 2005. She chaired the INCOSE Systems Engineering Management Working Group during 2003. She received a B.A. in Mathematics and Astronomy from Smith College in 1981, a M.S in Computer Science (concentration: Artificial Intelligence) from Johns Hopkins in 1991, and a M.S. in Computer Science (concentration: Graphics) from George Washington University in 1998.

John Donohue is the Associate Division Chief of the Software Engineering Division at the Goddard Space Flight Center (GSFC). Prior to his Division management position, he served as the Branch Head of the Real-Time Software Engineering (RTSE) Branch at GSFC. The RTSE Branch is responsible for the implementation of both in-house and commercial telemetry and command systems at GSFC. Prior to management, Mr. Donohue led the development of embedded hardware and software systems for NASA's communication networks. He holds a BSEE from the University of Maryland and a Masters of Engineering Management from the George Washington University.

Ben Lui is the Associate Branch Head of the Ground Software Systems Branch (GSSB) at NASA/GSFC. GSSB manages both in-house and commercial Telemetry and Command Systems at Goddard to support in-house missions. Prior to his current position, he served as the IT Systems Manager of the Hitchhiker Project at Goddard. Mr. Lui graduated from University of Maryland with a B.S. in Electrical Engineering and Johns Hopkins University of a M.S. in Electrical Engineering.

Greg Greer is the senior on-site contractor in the ITOS Development and Support Group within Code 583 at NASA/GSFC. He also is the ITOS Product Lead for the Hammers Company, Inc., which offers ITOS commercially. He is one of the original ITOS developers.

Thomas Green *is Senior Engineer and President of DesignAmerica, Inc., a small business specializing in Advanced Satellite Ground Systems. Mr. Green was one of the original architects of ASIST for the contractor team at NASA GSFC. Prior to his work on ASIST, he was Principle Engineer for Digital Equipment Corp. specializing in Automation Systems and Robotics, including work in the Flight Telerobotic Servicer lab at NASA GSFC. . For any questions about ASIST, Mr. Green can be reached at the DesignAmerica website: www.dai-asist.com*