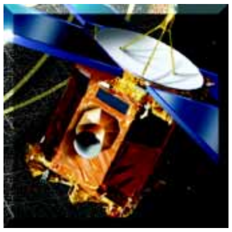

Efficient Spacecraft Test and Operations with the NEAR Ground System

Gary G. Whitworth, Anthony J. Somers, and William C. Stratton

The Near Earth Asteroid Rendezvous (NEAR) ground system was designed in keeping with the intent of the Discovery program to reduce development and mission operations costs for planetary missions. To do this, several nontraditional steps were taken in its development to increase efficiency: combining the ground system development projects for integration and test (I&T) and mission operations; committing to develop a single system to support both I&T and mission operations; competitively procuring a commercial off-the-shelf command and control system; and strategically networking the entire ground system to allow for reconfiguration and geographic dispersion. Efficiency was further enhanced by taking advantage of the state of the art in open system architectures, and also by using existing operations infrastructure when cost-effective.
(Keywords: Command and control systems, High-efficiency ground systems, NEAR ground system, Spacecraft ground systems.)

THE TYPICAL GROUND SYSTEM

A ground system is the network of facilities that support a spacecraft during the various phases of mission development. The functions provided by a ground system range from testing of spacecraft functionality during its integration, through prelaunch performance confirmation, to the following postlaunch operations tasks: planning and scheduling, spacecraft command and control, navigation, and data collection, analysis, assessment, and distribution. To perform these functions, ground systems typically use a wide array of items including generalized and specialized spacecraft test equipment, communications hardware and software, and computers with sophisticated specialized software for control and data gathering, analysis, and assessment.

Typically, a ground system is merely a collection of a few definable components, each featuring some mix of the several functions but having only a specialized part to play in the overall mission. For example, for the integration and test (I&T) of the spacecraft, a special suite of computers and test equipment is developed, which is commonly referred to as the Ground Support System (GSS). Its purpose is to send commands to the spacecraft, stimulate spacecraft subsystems and instruments, and receive and process spacecraft telemetry. This equipment interfaces directly with the spacecraft. It therefore resides in close proximity to the spacecraft, usually sitting in the same room or just outside the environmental test chambers, and travels with the spacecraft to all test and launch sites.

Other typical ground system entities are the ground stations, which provide the ground-to-space links with the on-orbit spacecraft. Further, there are various centers, consisting of computers, software, and people, which perform the many spacecraft support functions. These include the Mission Operations Center (MOC), where all postlaunch operations of the spacecraft are conducted, a science data center, and a mission navigation center. Another major component of a ground system is the communications infrastructure, which enables the various components to communicate and exchange data and data products.

All of these components may be located close together or widely distributed. Historically, they have been developed independently, with little concern for improving overall efficiency by developing commonly usable elements, by reusing existing or previously used software or computer systems, and with only secondary concern for agreeing on standards and mechanisms for the exchange and reuse of information. Further, the basic functions provided by a ground system at any point in time have varied with the successive mission phases of initial assembly, prelaunch testing, launch, and in-flight mission operations (MOPs). This situation has led to piecemeal ground system development, resulting in small and often almost independent "mini–ground systems" supporting the spacecraft during each of these phases. As the spacecraft development has progressed from one phase to the next, the spacecraft has been essentially "thrown over the wall" to the next ground system entity.

Since many functions and components are common to each of these mini–ground systems, this approach has used resources very inefficiently. The basic inefficiency has extended to independently developed and often not reusable software, protocols, and procedures. As a prime example, the team responsible for the early-on I&T of the spacecraft will develop different command and telemetry databases than will the MOPs planners. This situation occurs not only because the thrusts of the two tasks are considerably different, but also because MOPs development has generally been completed after I&T and is often called upon to compensate for any design shortcomings in the spacecraft.

THE NEAR GROUND SYSTEM

The Near Earth Asteroid Rendezvous (NEAR) ground system was designed in keeping with the intent of the Discovery program to reduce development and MOPs costs for planetary missions. It was recognized that the inefficiencies of the classical design approach could not be tolerated. At the beginning of the NEAR program, a ground system development policy was established that combined four elements:

1. A common architecture for both the I&T and MOPs activities to maximize efficiency
2. Commercial, off-the-shelf (COTS) resources when cost- and schedule-efficient
3. Open operating systems, networked workstations, and distributed processing
4. Existing support and operations infrastructure when cost-efficient

The first of these elements was a significant departure from the traditional approach because it required the I&T effort and MOPs development to be closely coordinated. This coordination made it possible to develop a common system to satisfy the different needs of these two activities, thereby leading to a more efficient, cheaper, and flexible system. It was desirable that this departure be evolutionary rather than revolutionary. Thus, for example, a COTS system was procured for spacecraft command and control functions, whereas MOPs software was written in-house because the commercial offerings in the this area were considerably immature.

The decision to use a single COTS command and control system for both I&T and MOPs provided not only improved efficiency by eliminating duplicative system procurements, but permitted a great deal of "fly as you test" capability. However, full concurrent engineering activities dictated by the 27-month development schedule required that the common command and control architecture had to permit I&T activity and MOPs development on a noninterferring, albeit coordinated, basis. This situation made it somewhat difficult to use a single system.

The solution was the separation of the command and control system into two identical segments. Since, in response to the third policy directive, the command and control system was based on networked workstations and distributed processing, the two segments could still be made equal parts of a common, fully integrated system. This approach permitted a full suite of cross operations. For example, operators sitting at workstations in either segment could receive telemetry from, and send commands to, the spacecraft no matter which segment was in actual contact with the spacecraft. The I&T segment is referred to as ITOGS (I&T operations ground segment). The identical MOPs segment is referred to as MOGS (MOPs ground segment). After launch, the two segments were combined to fill out a full and robust, and essentially redundant, computer system. Software developed in-house, such as MOPs software for command planning, higher-level telemetry analysis, and spacecraft performance and anomaly assessment, also runs on these workstations.

Another major advantage of this configuration is that the two segments are always joined by the common backbone network. This network, referred to as NEARnet, is an Ethernet-based network using transmission control protocol/internet protocol (TCP/IP). As the spacecraft development progressed, the I&T segment moved with the spacecraft from its integration site to environmental test sites and finally to the launch site, while the MOGS remained essentially in place in the MOC at APL during the entirety of spacecraft development, testing, and launch. At every spacecraft location, however, members of the in-training MOPs team could have full access, including participation in and even conduct of, I&T procedures from their workstations in the fledgling MOC. Further, MOPs personnel cross-trained as on-line I&T team members while training on the system they would subsequently use to control the spacecraft after launch.

As the major computing system for NEAR, the ITOGS and MOGS fulfill the policy element for distributed processing on networked computers with open operating systems. The word "segment" in the names ITOGS and MOGS is literally intended to imply a part of the NEARnet structure. As identical halves of the common command and control system, the ITOGS and MOGS have identical "front ends" for interfacing with the spacecraft for telemetry and commands. These interfaces are made through either umbilical or RF ground support equipment (GSE) or through the Deep Space Network (DSN) via NASA Communications (NASCOM).

It has been noted that the ground system development approach taken on NEAR was a significant departure from the more familiar approach of virtually independent I&T and MOPs development efforts. This shift was jump-started at the beginning of the program by defining both the I&T team and the MOPs team as the customers for the NEAR ground system. As such, the essential requirements for the ground system were levied from both I&T and MOPs as a coordinated effort. This coordination yielded requirements that were sufficiently melded to permit satisfaction of the often disparate needs of the two camps. Because spacecraft design was concurrent with the development activity for I&T, MOPs, and the ground system, this was an on-going effort and led to some development beyond the COTS vendor's initial package. Although the NEAR experience did not totally integrate the different viewpoints and goals of the two camps, it did pave the way for their closer integration on future spacecraft.

The final policy statement called for using existing infrastructure where possible. This requirement was implemented fully by using the DSN for all ground-to-spacecraft contacts and using NASCOM for all significant communications. This procedure was followed during development and also for the postlaunch operational system configuration.

The essentials of the NEAR ground system configurations during development and in final operational form are shown in Fig. 1. This illustration has changed little since the NEAR Conceptual Review. As shown, the basic elements are a mix of existing entities and components developed specifically for NEAR. The former include NASCOM and the DSN. The latter include the MOC, the I&T-oriented GSS, the Science Data Center at APL, and the Mission Navigation Center at the Jet Propulsion Laboratory (JPL). Also among the latter is the Mission Design Team, which has no specific facility since the team is spread over a large geographic area, but it does have access to NEARnet via a security router. The development of the NEAR ground system focused on properly incorporating the existing infrastructure and providing the needed communications and data services to these NEAR-specific entities. The MOGS and ITOGS were resident in the MOC and GSS, respectively. This article will not discuss the makeup or operation of the various centers.

The balance of this article will first discuss the aspects of NASCOM and the DSN services that are significant to NEAR. We will then focus on the two remaining major components of the NEAR ground system: (1) the distributed and highly versatile command and control system and (2) NEARnet. The command and control system provides all monitoring and control of the spacecraft. NEARnet provides the communications backbone for the command and control system, as well as the integrated pathway for all operations-related spacecraft and ground system monitoring and control, science data collection, and operational and mission-oriented planning, analysis, and assessment.

DEEP SPACE NETWORK

Since launch, and throughout the mission, all communications with the NEAR spacecraft are via the three Deep Space Complexes of the DSN. These are located in Goldstone, California; Madrid, Spain; and Canberra, Australia. The DSN provides both store-and-forward and throughput spacecraft commanding. All commands are packaged into CCSDS (Consultative Committee for Space Data Systems) standard protocol structures by the command and control system at the NEAR MOC before transmission to the DSN. For telemetry, the DSN provides convolutional decoding, frame synchronization, and time-of-receipt stamping, and sends telemetry in real time to the MOC in 4800-bit NASCOM blocks. (NASCOM service to the NEAR program, as for all programs being served by NASCOM using the 4800-bit blocking protocol, is

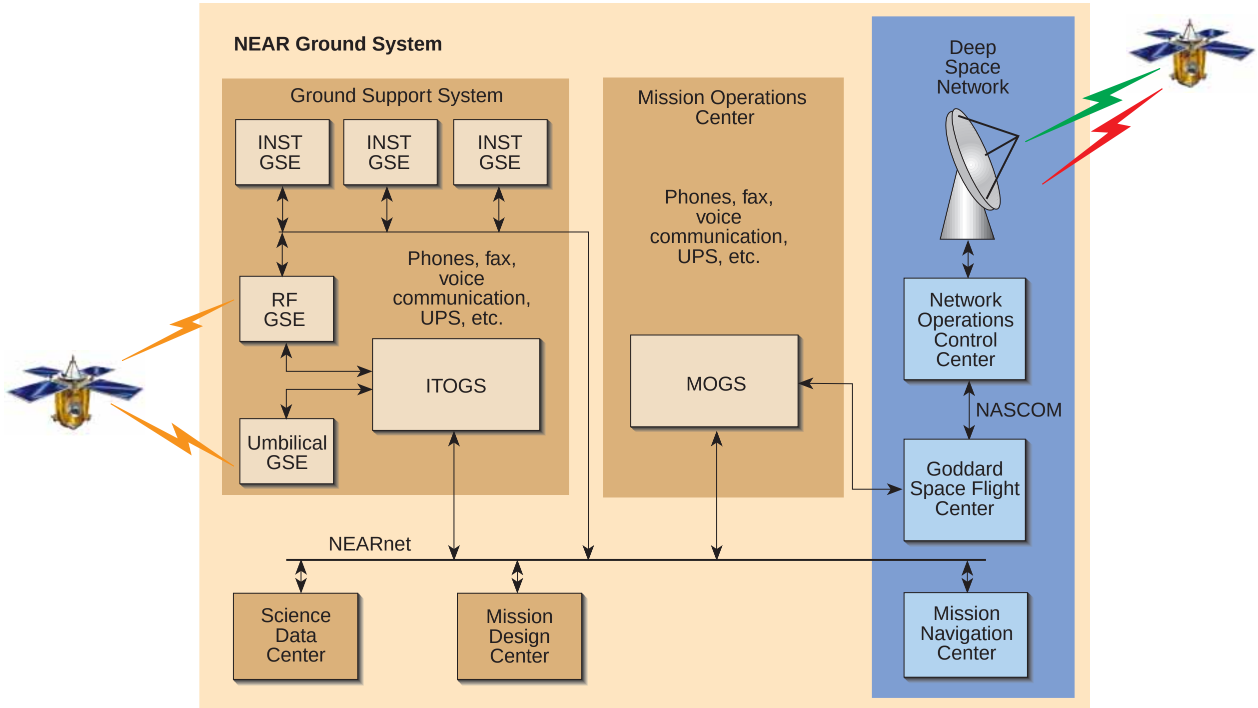

Figure 1. The NEAR ground system. (INST = instrument, GSE = ground support equipment, UPS = uninterruptible power supply, ITOGS = integration and test operations ground segment, MOGS = mission operations ground segment, MOC = Mission Operations Center, NASCOM = NASA Communications.)

to be moved to a TCP/IP service by 1998 as part of the NASCOM IP transition program.) All Reed-Solomon decoding, as well as virtual channel separation and depacketization, is done in the NEAR MOC by the command and control system.

During development and off-site test and launch activities, two other services of the DSN were used. The DSN compatibility test trailer (CTT) was used for testing the spacecraft at APL and during environmental testing at Goddard Space Flight Center (GSFC). In both locations, the CTT, which emulates a DSN station, had a real RF interface with the spacecraft and provided the full and on-line DSN command and telemetry links with the NEAR MOC. Additionally, the DSN compatibility testing and launch support facility known as MIL-71 was used extensively during launch operations. This fixed-antenna DSN station emulator also provided the full DSN linkup.

GSFC AND NASCOM

The essentials of the NEAR ground system communications connectivity are shown in Fig. 2. All significant NEAR data and voice communications with ground system entities external to APL are carried by NASCOM. A node on the T1 digital frame-based NASCOM 2000 system exists in the APL NASCOM interface facility. This node provides direct access to the switching center at GSFC. Of particular interest is that during off-site I&T and launch operations, this NASCOM service provided the pipeline for the extensions of NEARnet to the roving ITOGS. The NASCOM interface facility also provided DSN connectivity for the CTT while at APL.

NEARNET

From a networking viewpoint, the NEAR MOGS and ITOGS are a group of UNIX computer systems. These computers are interconnected by NEARnet as shown in Fig. 3. Rather than using proprietary hardware and software components to interconnect the ITOGS and MOGS computer systems, efficiency is gained by using not only standard hardware networking components, but also standard software networking components and services. The ITOGS and the MOGS subsystems are further divided into two Ethernet subnets. The so-called user datagram protocol subnet contains the subsystem front-end processor (FEP) and mission-critical operations workstations. The X.11 subnet contains additional telemetry viewing workstations.

Multiport local-area network (LAN) and wide-area network (WAN) routers were used to implement NEARnet's local and remote communications. Critical network protection features, such as packet filtering,

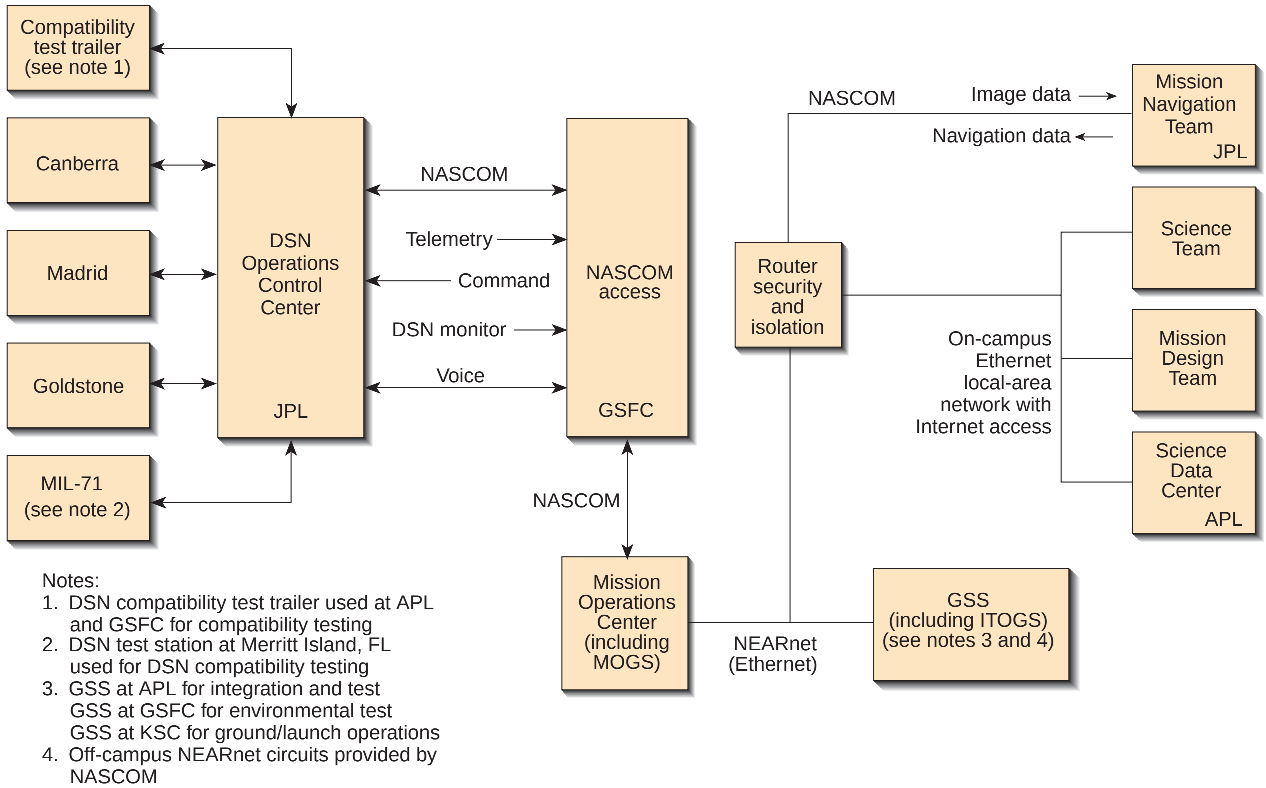

Figure 2. NEAR ground system communications. (DSN = Deep Space Network, JPL = Jet Propulsion Laboratory, NASCOM = NASA Communications, GSFC = Goddard Space Flight Center, MOGS = mission operations ground segment, ITOGS = integration and test operations ground segment, KSC = Kennedy Space Center, GSS = Ground Support System.)

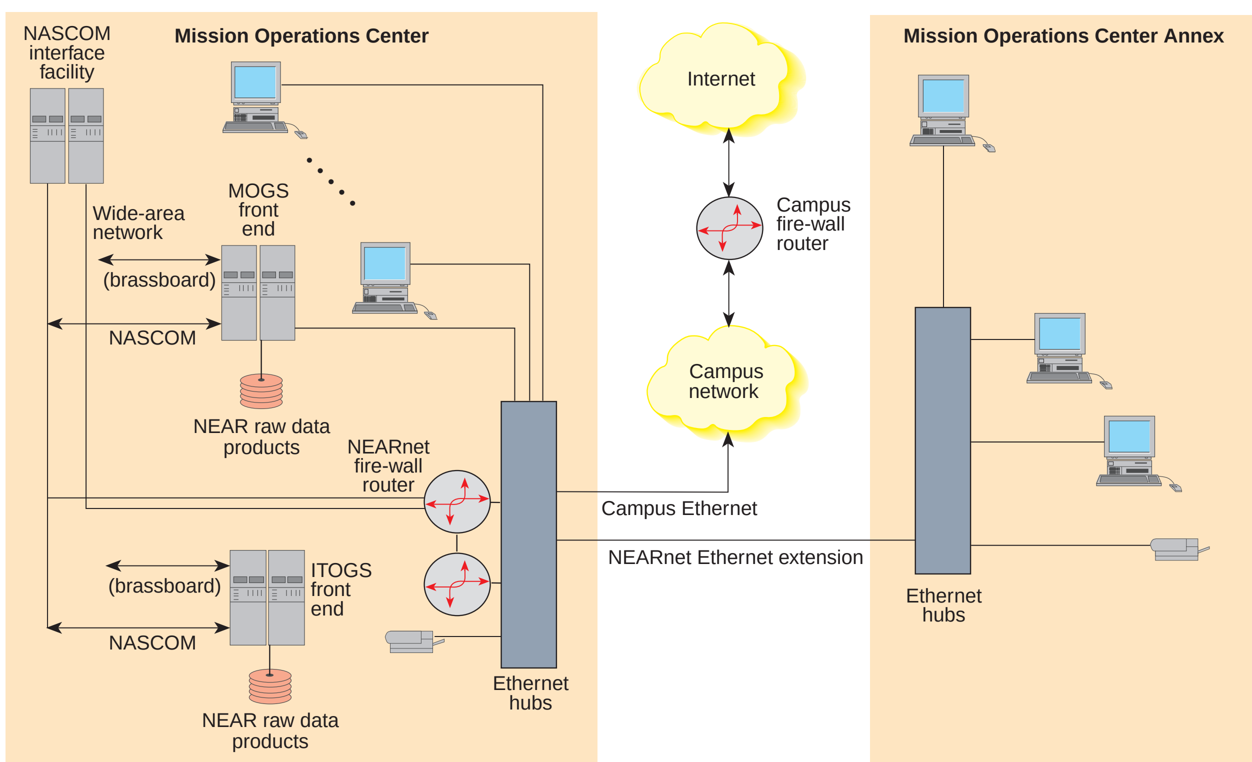

Figure 3. The NEARnet network. (NASCOM = NASA Communications, MOGS = mission operations ground segment, ITOGS = integration and test operations ground segment.)

are built-in using COTS NEARnet routers. The NEARnet network provides a communications channel for both UNIX system network traffic (i.e., remote file system mounting, remote workstation login) and spacecraft commanding and telemetry receipt. Partitioning each subsystem network into two subnets ensures that critical spacecraft commanding and telemetry receipt activities will occur with no interference from other UNIX network workstation activities.

Industry-standard network hardware and software are used in the design and implementation of NEARnet. The multivendor TCP/IP communications protocol is used extensively, both for spacecraft command and control and for workstation network activities. For example, the built-in UNIX network file system file-sharing capability is used to share the ground system database file between the FEP and the client workstations. Two different computer systems vendors are used to reduce single system supplier dependencies.

The NEAR FEP provides satellite telemetry to client workstations. Network configuration options provide for both point-to-point and broadcast methods. By using the built-in TCP/IP network communications, no application-level software needed changing to support telemetry delivery over the LAN link or the WAN configuration network link.

The development of the NEAR ground system occurred in distinct phases. The early phases supported the NEAR spacecraft bench testing; the later phases (and final phase) focused on MOPs of the satellite. The architecture of NEARnet was designed to support and facilitate each of the development phases. For example, spacecraft telemetry flowed between the ITOGS FEP and a MOGS workstation using 10-Mbps Ethernet in the I&T development phase. To support spacecraft launch, the same telemetry flowed over a 224-kbps WAN. The I&T phases of the NEAR command and control system are shown in Fig. 4. The MOGS and ITOGS were interconnected by both an Ethernet LAN and a 224-kbps WAN.

As shown in Fig. 4, NEARnet is connected to the APL network via the NEARnet fire wall, which performs IP packet filtering of incoming and outgoing TCP/IP network packets. Internal NEAR computers are on the closed side of NEARnet. Computers not on the closed side cannot command the spacecraft.

NEAR COMMAND AND CONTROL SYSTEM

The NEAR command and control system consists of four major components: (1) workstations, (2) front-end computers, (3) network hardware, and (4) satellite control center software. These components are all COTS and follow industry standards. This situation minimizes initial system cost and allows easy upgrade and/or replacement of components as necessary during the extended NEAR mission.

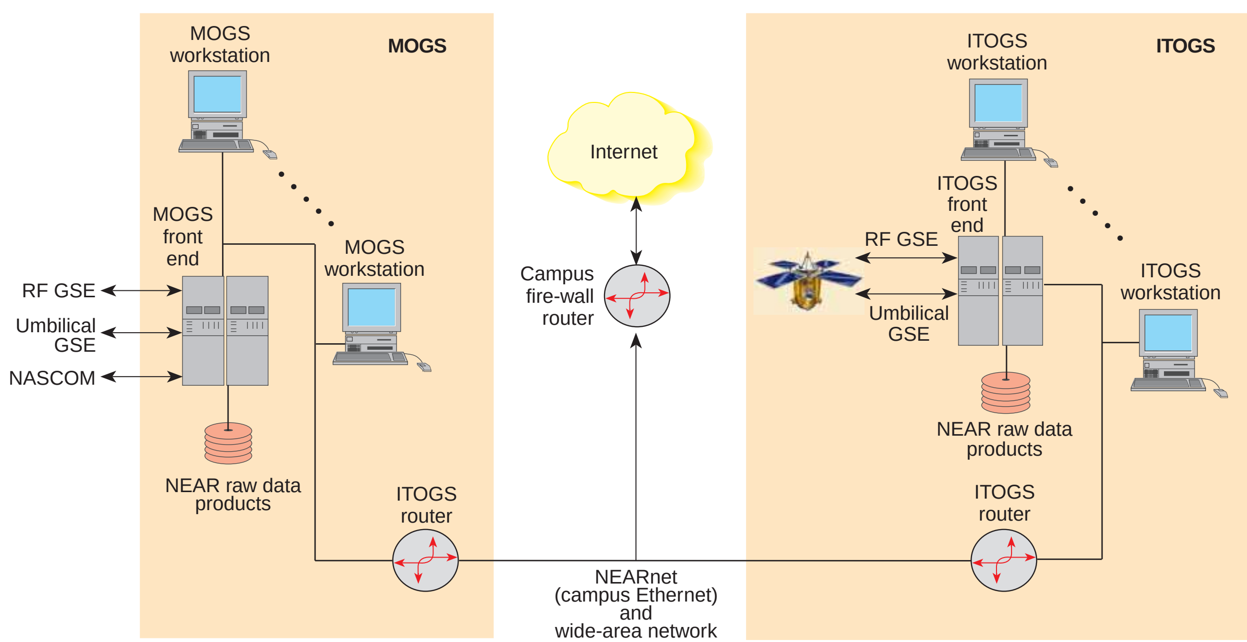

Figure 4. The integration and test phases of NEARnet. (MOGS = mission operations ground segment, ITOGS = integration and test operations ground segment, GSE = ground support equipment, NASCOM = NASA Communications.)

The workstations provide a high-performance graphical interface for each operator. Through this interface, an operator can control and monitor all aspects of the spacecraft and the command and control system. Each workstation runs the UNIX operating system with its usual complement of user interface and networking software, including X.11/Motif user interface tool kits, the Common Desktop Environment, TCP/IP network protocol, distributed Network File System, and Open Network Computing Remote Procedure Calls. Most vendors supply these tools on their workstations at no additional cost, providing a powerful, inexpensive foundation for the satellite control center software.

The front-end computers perform the real-time telemetry and command processing. They are based on the VMEbus standard, making it easy to integrate the COTS hardware needed for NEAR baseband and NASCOM communications.

The front ends run commercial real-time UNIX and the standard networking software listed previously. UNIX is a time-share operating system, so commercial real-time extensions to UNIX are used on the front ends to ensure that the throughput and timing requirements of the NEAR mission are met.

The satellite control center is based on EPOCH 2000 COTS software. It integrates the workstations, front ends, and network into a unified system for NEAR operations. By using a COTS control center package, new software development is limited to NEAR-specific requirements. Most of the NEAR mission requirements have been met just by populating the database. The NEAR EPOCH 2000 software comprises the major components described in the following sections.

System Database

The system database stores the spacecraft and ground system data that configure the commercial satellite control center software for the NEAR mission. It is built on the COTS Oracle relational database platform, which provides lots of tools for loading and maintaining the data. An easy-to-use forms interface facilitates data entry, validation, display, and reporting.

The spacecraft portion of the system database stores all the information necessary for processing telemetry and generating commands. The telemetry database defines the CCSDS packet formats produced by NEAR, as well as conversion coefficients, limits bands, and state ranges for each telemetry point. The command database defines the CCSDS command frame formats and validation sequences for each NEAR command.

The ground portion of the system database contains configuration and monitoring data definitions for the front-end VMEbus hardware. It also includes data definitions for the DSN station equipment used by the command and control system for NEAR communications.

Because commercial relational database packages are notoriously slow, the run-time software does not access the database directly. Instead, ASCII reports containing all configuration data from the database are generated off-line. The run-time software parses the ASCII reports at start-up and stores the parsed reports on disk. Subsequent runs only parse an ASCII report file if it has been updated.

User Interface

NEAR operators monitor and control the spacecraft and the command and control system using COTS user interface software running on each workstation. Operators use tiled and overlapping window layouts specified in EPOCH Display Language (EDL) for monitoring. The System Test and Operations Language (STOL) is used for control.

EDL is a proprietary screen and page layout definition language supported by the COTS user interface software. It is designed for the control center environment and hides as many of the details of X.11 and Motif as possible. The NEAR control center users have developed an extensive library of window layouts using both the user interface's built-in display editing capabilities and EDL directly. Figure 5 depicts a typical NEAR user interface layout, designed and implemented by the NEAR MOPs team.

STOL is a procedural language for spacecraft command and control applications. The COTS user interface supports both STOL directives (one-line STOL commands) and STOL procedures (files of STOL commands, including argument passing and flow control). STOL procedures can be nested, supporting structured, modular automation of all NEAR operations. STOL is easy to use—a library of STOL procedures has been developed by the NEAR spacecraft engineers, I&T team, and operators for their specific needs. The library continues to evolve with the mission.

The COTS user interface supports stand-alone pass playbacks, so operators can review spacecraft and command and control system activity on their workstations without affecting ongoing NEAR support. This capability has been very useful in evaluating NEAR subsystem performance.

The user interface also includes a spacecraft data simulator, used primarily for checking out new database ASCII report files and STOL procedures before they are put on-line.

Telemetry and Command Processing

Real-time spacecraft T&C processing is performed on the front end. T&C processing functions include

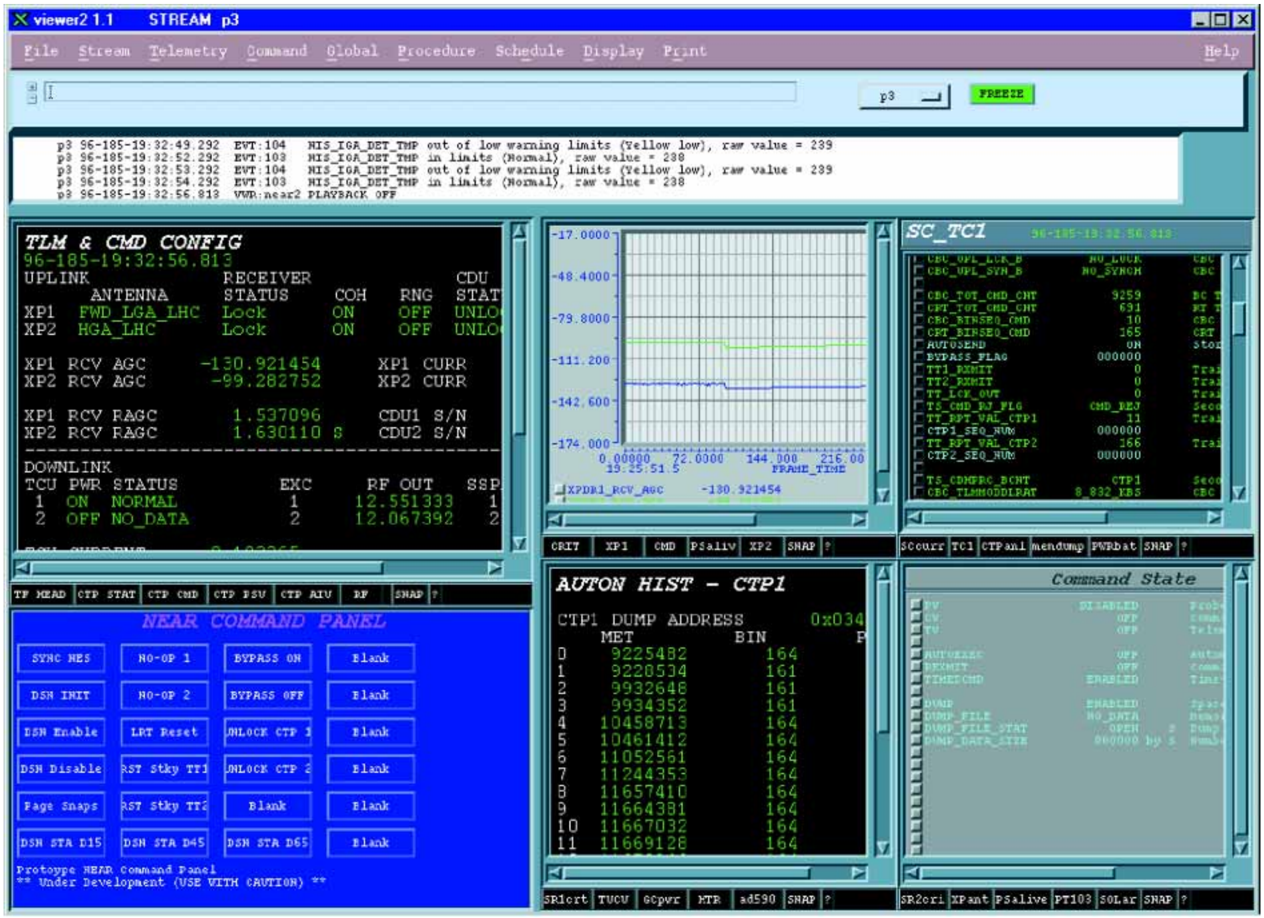

Figure 5. Typical NEAR user interface layout. All aspects of the spacecraft and the command and control system can be controlled and monitored through user-defined windows, such as this top-level display created by the NEAR mission operations team.

telemetry packet decommutation, point- and subsystem-level status checking, dump collection, command frame generation, STOL schedule execution, data archival, and T&C data service.

The COTS control center software supports multiple T&C communications streams in parallel. This capability has been exploited in NEAR testing (simultaneous spacecraft communications via NASCOM and ground support equipment interfaces) and in NEAR operations (simultaneous spacecraft communications through two DSN stations during station hand-off).

All T&C processing is performed by front-end software modules. No hardware is used for telemetry decommutation or command frame construction. NEAR has quite complex T&C formats, so the flexibility of software T&C processing is a benefit—the NEAR requirements are met with COTS software configured by database entries.

Processing in each T&C communications stream is controlled by a schedule, which is just a STOL procedure that runs on the front end and is visible at all workstations. Planned NEAR control activities are coded in STOL and executed as schedules so that all operators can see what is going on. An example schedule as seen at a command and control workstation is shown in Fig. 6.

Each active T&C communications stream performs its own data archival to hard disk, where it is available for post-pass analysis. Archived data include all telemetry, relevant ground equipment parameters, and system events. Each stream also provides data service to client programs. This capability has been used with NEAR to drive a three-dimensional spacecraft display program and a real-time Internet site.

Ground Equipment Processing

The COTS ground equipment processing software provides the connection between the T&C processing and the VMEbus hardware for the configured NEAR communications path (DSN station 1, DSN station 2, or baseband hardware). Functions performed by this software include VMEbus hardware interface, DSN

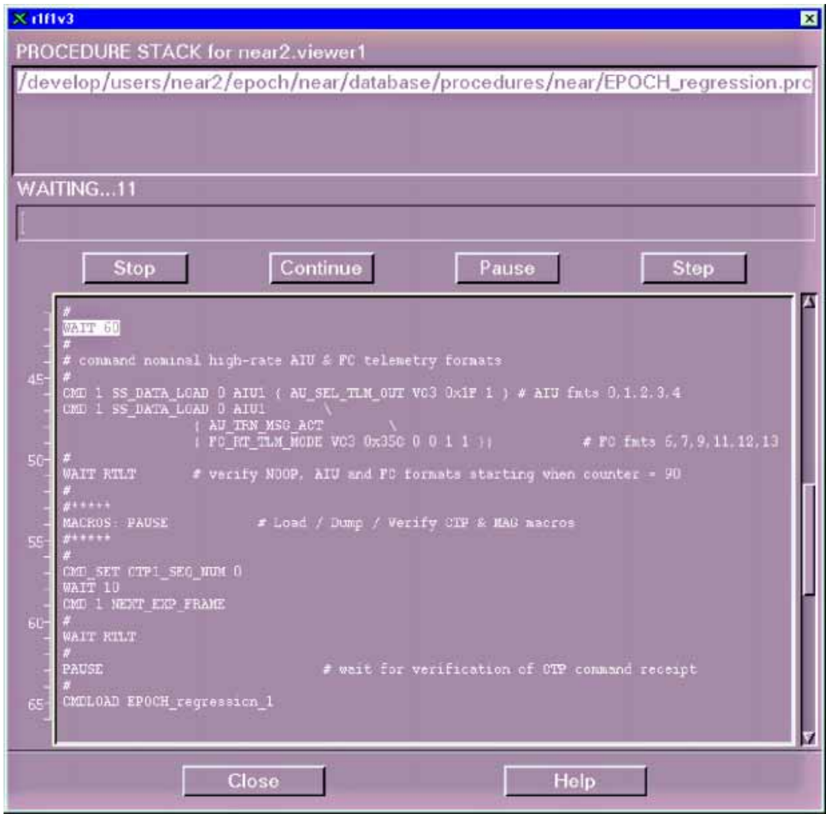

Figure 6. Typical telemetry and command schedule as seen at a command and control workstation. Planned NEAR control activities are coded in STOL and executed as schedules so that all operators can see what is going on in a telemetry and command stream.

protocol and simulation, Reed Solomon error correction, Command Link Transmission Unit encoding, embedded transfer frame synchronization, NASCOM block and transfer frame archival, and Science Data Center interface. The major benefit of this software is that it encapsulates NEAR-specific processing so that the commercial control center software can be used without modification.

Off-Line Utilities

Off-line processing software is provided for command load generation and archive data processing. The command load generation program allows an operator to generate a binary load file of spacecraft commands from an ASCII specification. Archive data processing includes extraction of selected telemetry points and conversions between various file formats.

SUMMARY

The NEAR ground system provides efficient command and control for the NEAR spacecraft. It achieves this by maximizing the use of a COTS command and control system and an existing communications infrastructure. Additionally, it employs an open and networked computer system architecture, which is used jointly for spacecraft integration and test and for MOPs. This system will remain flexible and effective for the 4-year life of the NEAR mission.

ACKNOWLEDGMENT: This work was supported under contract N00039-95-C-0002 with the U.S. Navy.

THE AUTHORS

GARY G. WHITWORTH recently retired from APL, where he had been employed since receiving his B.S.E.E. from the University of Tennessee in 1966. He spent his career in ground system and space-to-ground communications for several programs, including Apollo, the space shuttle, the Tracking and Data Relay Satellite, and several Navy programs. He is the author of a chapter on ground systems in the textbook *Space Mission Analysis and Design*, published by Kluwer Academic Publishers. Mr. Whitworth was the Ground System Lead Engineer for the NEAR program.

ANTHONY J. SOMERS received his bachelor's degree in computer science from the University of Maryland in 1977. After working as a systems engineer at the GE Space Division, he co-founded Alexis Logic Corporation in 1982. At Alexis, Mr. Somers performs both day-to-day company management and client consulting activities. He provides computer systems' administration and network management for the NEAR Mission Operations Center. His e-mail address is anthony_somers@alexis.com.

WILLIAM C. STRATTON received his Sc.B. in applied mathematics from Brown University in 1977. He is a senior member of the technical staff at Integral Systems, Inc. For the past 6 years, he has served as one of the lead systems programmers for Integral Systems' EPOCH 2000 control center product, the foundation of the NEAR command and control system. Prior to EPOCH, Mr. Stratton led the R&D effort behind NASA Goddard Space Flight Center's TPOCC system, which is now supporting most of NASA's scientific satellite operations. His e-mail address is stratton@integ.com.