AUTOMATION OF OPERATIONS AND GROUND SYSTEMS AT U.C. BERKELEY

Manfred Bester, Mark Lewis, Tim Quinn and Joseph Rauch-Leiba

Space Sciences Laboratory
University of California at Berkeley
Berkeley, CA 94720-7450
manfred@ssl.berkeley.edu, markl@ssl.berkeley.edu,
teq@ssl.berkeley.edu, rauch@ssl.berkeley.edu

ABSTRACT

In 1999, an operations center and a satellite ground station have been established at U.C. Berkeley's Space Sciences Laboratory (UCB/SSL) to support mission and science operations of multiple NASA funded explorer missions. The Mission and Science Operations Center (MOC/SOC) and the Berkeley Ground Station (BGS) now constitute a fully functional, highly integrated and automated facility that currently serves as the satellite command, control, tracking and data archiving center for three different missions, namely the *Fast Auroral SnapshoT* (FAST) explorer, the *Reuven Ramaty High-Energy Solar Spectroscopic Imager* (RHESSI), and the *Cosmic Hot Interstellar Plasma Spectrometer* (CHIPS) satellite. MOC/SOC and BGS systems include a secure network of computer workstations and servers, an 11-m parabolic reflector antenna and associated equipment, as well as software tools for flight dynamics, communications, command and telemetry data processing, archiving and storage. The scalable architecture of the Berkeley operations center allows straight forward expansion to support additional explorer missions.

1. INTRODUCTION

UCB/SSL has a long history of designing and building state-of-the-art scientific instrumentation that was flown on numerous space science missions. However, with the award of the RHESSI project NASA has for the first time given to a university full responsibility for a Small Explorer (SMEX) mission, including project management, satellite and instrument development, mission control, and telemetry recovery functions. This enabled UCB/SSL to design and implement a new operations center and a ground station. The focus of the RHESSI mission is the investigation of the physics of energy release in solar flares by means of high resolution imaging and spectroscopy at X-ray and gamma ray wavelengths [1].

During the early design phases of the RHESSI mission, the FAST mission, launched in August 1996, successfully completed the first three years of on-orbit operations. FAST's primary objective is to study the microphysics of space plasma and the accelerated particles that cause the aurora [2]. FAST mission operations were initially conducted at NASA/GSFC and science operations at UCB/SSL. To save costs during the extended mission phase, mission operations for FAST transitioned from NASA/GSFC to the newly established facility at UCB/SSL in October 1999. RHESSI launched in February 2002 and was operated from UCB/SSL from the first day on orbit.

In January 2003, CHIPS, a University-class Explorer (UNEX) studying the spectrum of the hot gas in the interstellar medium at extreme ultraviolet wavelengths, was launched [3]. Mission operations for CHIPS were initially conducted at SpaceDev, the spacecraft bus provider, and transitioned to UCB/SSL after four months.

While all three missions have unique operations requirements, they also share common ground system elements. The FAST ground system was initially adopted from NASA/GSFC and was further developed since then. The RHESSI ground system is based on similar elements, but was designed around an advanced client/server networking architecture. While FAST and RHESSI use the CCSDS protocol for spacecraft commanding and telemetry downlink [4], the CHIPS mission employs the fairly new Internet-in-space technology that is based on widely used and proven TCP/IP network protocols and software tools [5,6]. The following paragraphs describe the approach that was taken to support these three missions simultaneously out of one joint facility with single-shift staffing.

2. GROUND SYSTEMS ARCHITECTURE

An optimized satellite operations environment consists of robust ground systems architecture, solid hardware, and carefully designed and debugged software, operated by a well trained, versatile Flight Operations Team (FOT). The FOT needs to be able to handle a wide variety of situations, many times without assistance. The Berkeley FOT consists of a group of people who are part console operator, part spacecraft engineer, part programmer and part system administrator. A small FOT can operate multiple spacecraft efficiently and reliably only when routine functions are completely automated. To allow single shift operations, spacecraft passes must be taken without human intervention. Command and control systems, ground stations and data processing systems are required to run essentially flawlessly with every pass support. The key ideas behind the design, implementation and operation of the new facility are summarized below:

1. Maintain spacecraft health and safety at all times
2. Maximize data quality and quantity while minimizing systems downtime and operations costs
3. Implement a scalable architecture to allow for expansion towards future missions
4. Minimize the number of interfaces to external ground system elements
5. Implement a topology that allows support of both CCSDS and TCP/IP missions
6. Use modern technologies such as TCP/IP networking, scripting and web based tools
7. Include heritage where practical, but avoid inefficient legacy systems
8. Pay attention to lessons learned from previous projects
9. Utilize COTS and GOTS products whenever possible
10. Limit in-house software development to mission unique programs and scripts
11. Provide a high degree of automation and integration
12. Include tools for remote monitoring and error messaging
13. Emphasize user friendliness and clean functionality
14. Implement fault-tolerant features
15. Install back-up power systems
16. Provide a secure network interface for access to NASA ground stations
17. Provide physical access control and firewall protection for all mission critical elements
18. Pay attention to adequate workmanship with all hardware systems
19. Carefully test and debug all software systems
20. Maintain and upgrade ground systems periodically

The Berkeley ground data system, shown in Fig. 1, was designed and implemented following the above ideas and guidelines. The left side of the schematic diagram represents the Berkeley Ground Station and external ground stations that provide communications with the three supported spacecraft. Mission planning, mission control and flight dynamics functions are performed within the MOC. Operational software comprises various COTS and GOTS products, namely the SatTrack Suite for flight dynamics [7], the Integrated Test and Operations System (ITOS) for spacecraft command and control [8], the Mission Planning System (MPS), formerly known as the SMEX Command Management System (CMS) for command load generation [9], the Spacecraft Emergency Response System (SERS) [10], and a number of auxiliary programs and scripts developed in-house at SSL. The schematic shown in Fig. 1 is representative of the topology used with RHESSI, FAST and other CCSDS type missions that may be supported in the future. For CHIPS mission support, the TCP/IP based control system discussed further below takes the place of ITOS and MPS. The SOC handles all data processing, archiving and distribution functions. While many of the ground data system elements are shared between missions, there are also differences in how each mission handles its automation functions, as will be explained in following sections.

3. BERKELEY FLIGHT DYNAMICS SYSTEM

The Berkeley Flight Dynamics System (BFDS), shown as part of the MOC in Fig. 1, is based on the SatTrack Suite and provides complete flight dynamics support for all three missions. The entire process execution is completely automated and controlled by the SatTrack Gateway Server (SatTrack/GS), running on two redundant LINUX systems with 2.8 GHz Intel Pentium 4 processors. All ephemeris and mission planning products for FAST, RHESSI and CHIPS are based on two-line element (TLE) sets that are downloaded twice per day from the Orbital Information Group (OIG) web server at NASA/GSFC, and are verified and archived locally. Pass scheduling is based on a dynamically modeled link margin of 6 dB, calculated from spacecraft attitude and antenna pattern, range, ground station Figure of Merit (G/T) and station mask for each ground station, rather than pure line-of-sight geometry. This approach optimizes utilization of ground assets while at the same time providing very high data quality. SatTrack also generates products in specialized formats such as Predicted Site Acquisition Tables (PSAT) and Improved Interrange Vector (IIRV) files for all missions supported at the MOC/SOC. These files are not used locally for tracking or scheduling, but are required for some of NASA's Ground Network (GN) systems.

Monitoring and controlling spacecraft autonomously from the MOC/SOC involves the coordination of multiple independent processes running on various hosts. In addition to controlling ephemeris production, SatTrack/GS also provides a wide variety of networking services for the ground system, allowing centralized pass scheduling, system monitoring, fault detection and messaging. Pass support requests are distributed to all required ground system elements in real-time via client/server network socket connections. A variety of client applications connect to the server and sign on with a particular client type that allows information exchange in support of specific automation functions. ITOS systems, acting as the *user interface* to the spacecraft, connect to the server as *ITOSLINK* clients. An instance of ITOS running on a

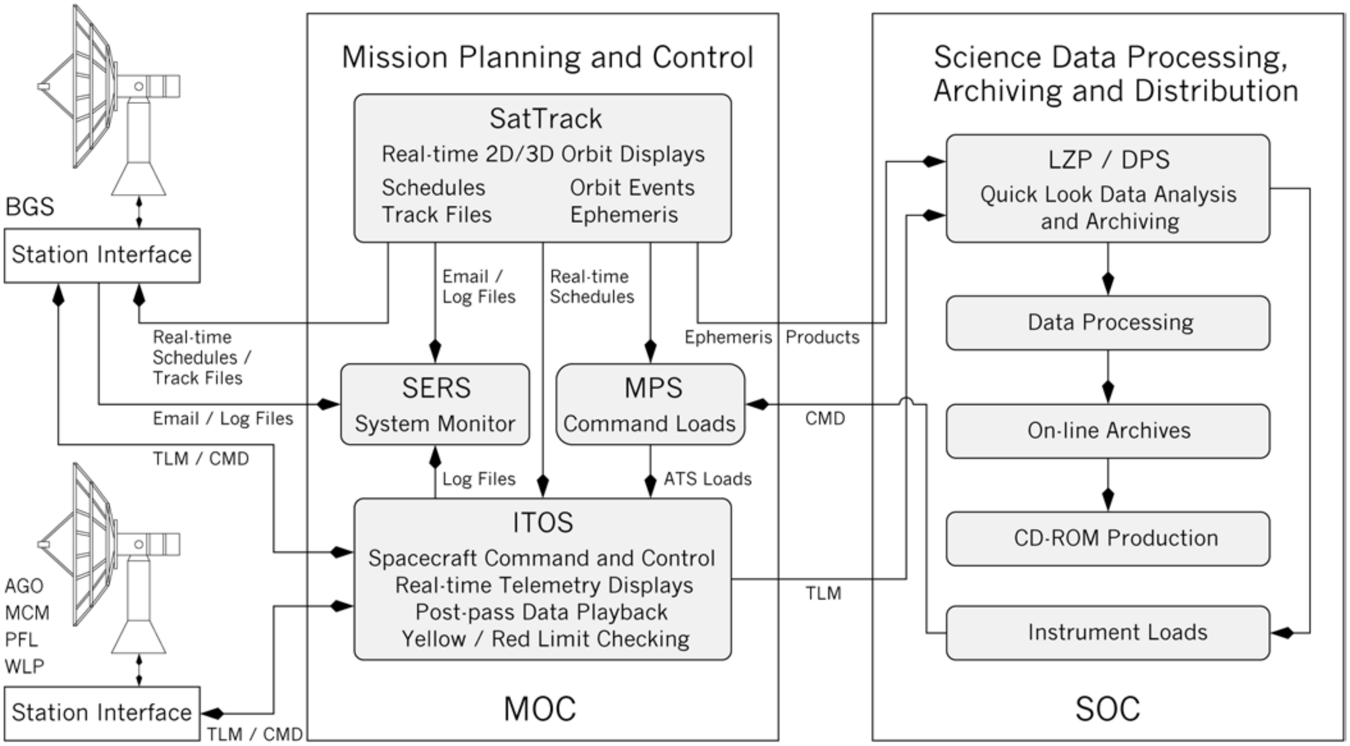

Figure 1. Block Diagram of the Berkeley Ground Data System.

workstation that is designated as the primary command and control system for RHESSI pass supports at the Berkeley Ground Station submits a request *SENDPASSES FACILITY BERKELEY OBJECT RHESSI*. SatTrack/GS then provides the corresponding timeline of events as well as configuration parameters to ITOS at pre-pass begin, typically 10 minutes prior to Acquisition of Signal (AOS). ITOS in turn starts to configure itself for the upcoming pass. Each timeline event triggers execution of an operational task by ITOS as well as the underlying UNIX operating system at specified times relative to AOS. Like ITOS, the Monitor & Control System (SatTrack/MCS) [7] of the Berkeley Ground Station connects to SatTrack/GS, but logs on as a *FACILITY* client and submits a request *SENDPASSES FACILITY BERKELEY*, assuming *objects* are *any*, thereby allowing supports of multiple spacecraft. The ground station does not know the entire pass schedule in advance, but receives a request for an individual pass support at the moment of pre-pass begin. SatTrack/MCS then reads a mission specific configuration file containing the settings of all subsystems required to support the pass.

While ITOS prepares itself for the upcoming pass support, the ground station proceeds with its pre-pass configuration and system set-up. At AOS minus 5 minutes, the primary frontend processor in the ground station is ready to accept command and telemetry TCP/IP network connections from ITOS, and in turn ITOS establishes these connections. At AOS the antenna waits at the intercept point where the spacecraft breaks the station mask, the spacecraft transmitter is turned on following an on-board Absolute Time Sequence (ATS) load, and telemetry data begin to flow. The ground station then performs an acquisition sweep for the command link and subsequently enables the command modulation. ITOS in turn sends commands to the spacecraft. At the moment of Loss of Signal (LOS) the de-configuration sequence commences both on the ITOS side and on the ground station side. ITOS then starts to play back spacecraft engineering data while the ground station transfers recorded telemetry files from the frontend processor to the data processing systems at the SOC. At post-pass end which usually occurs at LOS plus 3 minutes, the ground station is ready again to receive the next pass support request from the Gateway Server.

As illustrated in Fig. 2, different client types have been implemented for a variety of applications. Telemetry and command processing systems, ground stations, countdown clocks and monitoring systems that are connected to the server are programmed remotely, based on the current, deconflicted pass schedule. SatTrack/GS allows up to 100 clients to connect simultaneously in order to receive pass support requests or to exchange related information. All clients except for web browsers remain connected continually. Keep-alive messages are sent across the links to allow for network or subsystem fault detection, and to prevent firewalls from terminating the connections in case no other traffic is exchanged for a period of time.

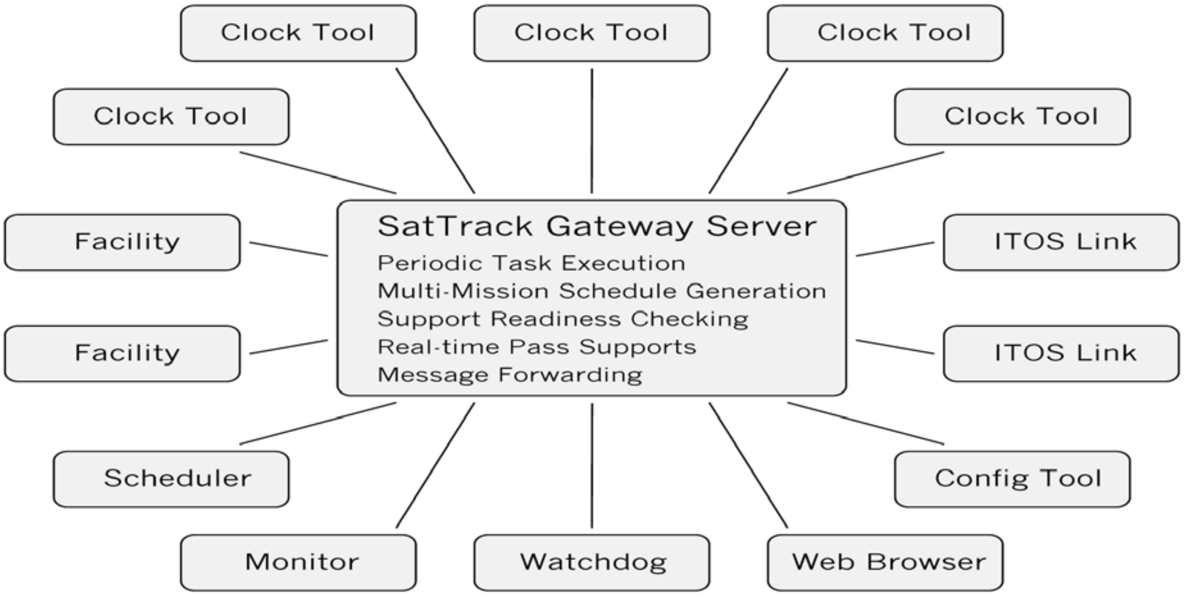

Figure 2. The SatTrack Gateway Server (SatTrack/GS) handles many simultaneous client connections and also acts as a process control and message forwarding system.

4. BERKELEY GROUND STATION

The Berkeley Ground Station, shown in Fig. 3, presently serves as the primary ground station for RHESSI and CHIPS. The facility is located adjacent to SSL and consists of a pedestal with an 11-m parabolic reflector antenna [11]. A three-axis drive system eliminates the key hole at the zenith. The antenna is equipped with a full-duplex S-band telemetry and command system. The dual-channel receiving system comprises circular polarized prime-focus feed and multi-mode receivers with diversity combination [12]. The Figure of Merit (G/T) in either channel is 24.0 dB/K for clear-sky elevations above 5°. A conical scan feed system provides autotrack capabilities with a typical accuracy of 0.1°. Redundant exciters generate the command uplink signals [13,14]. The transmit RF power is 100 W (63.0 dBW) with selectable polarization (RHCP and LHCP).

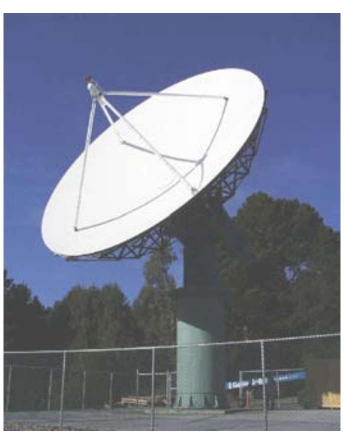

Figure 3. BGS 11-m antenna system located at UCB/SSL.

The LINUX-based SatTrack Monitor & Control System (SatTrack/MCS) for the ground station is a multi-threaded application [7]. System and mission specific configuration files are in plain ASCII format. All subsystems are controlled via RS-232, IEEE 488 or TCP/IP network socket interfaces. Station equipment configuration and control is completely automated, but can also be achieved manually. Task execution follows the principle of relative time sequencing based on the absolute times of pre-pass begin, AOS, LOS and post-pass end. Pre-pass station set-up begins 10 minutes prior to AOS, and post-pass de-configuration is completed 3 minutes after LOS. Hence each pass includes an overhead of 13 minutes in addition to the pure track time. Antenna motions are controlled with track files generated by BFDS in the same daily production run that also generates the multi-mission schedules and all mission planning products used to build spacecraft command loads. Therefore, task execution onboard the spacecraft is very well synchronized with corresponding ground station control functions.

Redundant control systems and fault-tolerant design features such as protection against false autotrack lock on interfering objects, i.e. the Sun or another spacecraft contribute to a very reliable operation. A large effort went into implementing software features to work around various *subsystem peculiarities*. Extensive warning and error messaging as well as system logging and generation of detailed pass support summaries round out the control system. A variety of ground station control functions such as raising or lowering of the RF carrier can be initiated remotely. A 16x16-port solid-state matrix switch allows data routing between transmitters, receivers, frontend processors, Internet routers and test equipment. The matrix switch is also used to enable or disable main RF carrier and subcarrier modulation during command link acquisition sweeps. System and mission specific matrix switch settings are defined in the configuration files.

The Berkeley Ground Station employs two identical frontend processors for bit synchronization, Viterbi decoding, frame synchronization, Reed-Solomon decoding, and CCSDS frame routing [15]. Data streams that carry real-time spacecraft, instrument and sensor data are typically routed directly into the ITOS workstations for state-of-health monitoring and control functions. In addition, all received telemetry data are stored locally on the ground station control computers in separate files for each virtual channel, and are automatically transferred to their final destinations via *ftp*, once a pass support has been completed. For support of CHIPS, a LINUX-based router is interfaced directly to the receiver baseband output and the exciter modulation input by means of the matrix switch, thus bypassing the CCSDS frontend processors [6].

Approximately every six hours, the ground station performs self-tests that are dynamically inserted into gaps in the tracking schedule. SatTrack/MCS treats these self-tests as regular pass supports and exercises most critical subsystems such as the S-band feed system, transmitters, receivers, matrix switches, fiber-optic links and dual frontend processors. RF signal levels in these long-loop telemetry playback tests are adjusted such that any degradation of the system's nominal performance causes correctable and possibly uncorrectable Reed-Solomon errors that are detected separately in each of the two frontend processors. The engineer on duty is paged in turn to investigate and resolve the system anomaly. With this automated procedure, any system faults are typically detected and corrected in time to prevent loss of telemetry data during real spacecraft supports. Various other test modes allow for automated generation of the station mask based on a 1 dB increase in antenna temperature, or for rigorous, automated testing of individual subsystems.

5. RHESSI OPERATIONS

ITOS is the command and control system for RHESSI. Running on a UNIX workstation, it connects directly to the frontend processors in each ground station to receive telemetry and send telecommands. The primary user interface to ITOS is the Spacecraft Test and Operations Language (STOL) [8]. At the console the operator enters spacecraft telecommands and program control commands known as STOL directives that may also be organized into procedures. Once the telemetry stream from the spacecraft becomes active, ITOS scans state-of-health mnemonics for out-of-limit conditions and anomalies in both real-time or post-pass playback mode. All activities, operator interactions, telemetry violations, spacecraft events, and internal program faults and warnings are logged to a file.

Crucial for automation, ITOS has a built-in remote-control facility that permits it to receive STOL directives from client applications through a TCP/IP connection. Theoretically, any task that can be executed by an operator at the console can also be executed remotely by an automated process. ITOS is designed to run on a cluster of workstations. When properly configured, only a few directives are necessary to share the telemetry stream between any number of machines. Each workstation can then invoke its own telemetry displays, allowing different flight controllers to focus on particular subsystems of the spacecraft. In the MOC, shown in Fig. 4, all ITOS workstations are set up automatically and seemlessly for contacts involving the Berkeley Ground Station or other remote NASA ground stations. Each workstation is controlled by an application that knows, based on information received from SatTrack/GS in real-time, how and when to perform the set-up for that machine. The interface program then reads a mission configuration file, and subsequently builds a list of tasks that are executed at specific times during the contact. These tasks are either ITOS STOL directives or UNIX system commands. Real-time distribution of contact schedule timelines via TCP/IP network connections by SatTrack/GS, and the remote control capabilities of ITOS are the main design features for supporting automated RHESSI spacecraft operations.

Contact timelines contain events like Acquisition of Signal (AOS) and Loss of Signal (LOS), and more specific events such as spacecraft link access or solar interference entry and exit times. The tasks associated with these events may be categorized as pre-pass, real-time, or post-pass activities. Pre-pass activities include tasks such as opening log files, connecting to the respective ground station, configuring command and telemetry paths, opening telemetry archives and sequential prints, and enabling checking of telemetry limit violations and spacecraft configuration. Real-time activities are kept to a minimum to avoid interference with commands entered by operators from the console. Post-pass activities mirror the pre-pass

Figure 4. Operations Consoles in the MOC at UCB/SSL.

activities, moving log files, telemetry archives, and sequential prints off the network for further processing and storage.

Personnel must be notified when a problem with the spacecraft occurs. For this purpose, SERS scans the log files produced by ITOS for faults such as telemetry limit violations or commanding errors. The log files are transferred to SERS' input directory at the end of each contact by a UNIX shell script triggered off a post-contact event. Any faults discovered are transmitted to the pager of an on-call operator, who must accept responsibility or defer to back-up personnel.

A few other processes and tools allow operations personnel to monitor the spacecraft during off-hours. The first is the Spacecraft Status Page, generated by a SatTrack utility that connects to ITOS' Data Point Server (DPS) [8] and queries the values of important telemetry mnemonics once every 20 seconds. During pass supports, when the telemetry stream becomes alive, the utility generates an HTML document which in turn is transferred to a web server outside of the secure operations network once per minute. The utility also performs limit checking, independent of SERS and sends out email and paging messages, in case any limit violations are detected.

Another aid for remote monitoring is derived from sequential prints recorded by ITOS. Each print contains time-ordered values of related telemetry mnemonics. During the post-pass period of a contact, the prints are moved to a processing queue where the data are plotted by an Interactive Data Language (IDL) program [16] and posted on a web server for remote viewing. Operators can browse recent trends in telemetry, grouped by subsystem and compare these against long-term plots of the same telemetry points averaged daily over the entire mission.

Since most personnel alerts originate from spacecraft conditions detected by ITOS, an independent process must ensure that ITOS itself is running properly. A script started periodically by the UNIX *cron* utility accomplishes this task by checking the ITOS process list and restarting ITOS if necessary. In addition, SatTrack/GS *locks* the client connections from all ground systems required to be online for a given pass support at AOS minus 30 minutes. Clients are unlocked at post-pass end. If a locked client loses connectivity within this time interval, a paging message is sent out. Furthermore, the interface program between SatTrack/GS and ITOS sends a paging message if it loses its connection to either SatTrack/GS or ITOS.

Upon completion of a RHESSI pass at any of the supporting ground stations, data are transferred from the respective ground station to the SOC. RHESSI utilizes four ground stations for telemetry recovery: U.C. Berkeley, CA; NASA Wallops Flight Facility, VA; University of Chile, Santiago; and DLR Weilheim, Germany. Data received at the Berkeley and Santiago ground stations are pushed to the SOC via *ftp*. The Wallops ground station delivers data files to the Standard Autonomous File Server (SAFS) system at GSFC. SAFS in turn sends an email message to the SOC with details about the files which are ready for retrieval. A local process then wakes up, reads the email and downloads the files using *ftp* via the open Internet. Upon completion and verification, a return email is sent to SAFS informing the system that the data can be removed from the SAFS server. Telemetry data recovered at the Weilheim ground station are transferred from a file server at the German Space Operations Center (GSOC). If a problem is encountered with any data transfer, an email alert is sent out to relevant operations personnel.

6. FAST OPERATIONS

Like RHESSI, FAST uses ITOS for command and control of the spacecraft. However, scheduling of passes at the NASA/GN stations that FAST uses, namely Poker Flat, AK, Wallops Flight Facility, VA, and McMurdo Base, Antarctica, is handled by the White Sands Scheduling Office. Corresponding schedule files are sent to the MOC several times per week. These schedule files are processed by an ITOS procedure which executes STOL commands and UNIX shell scripts. The procedure sets up ITOS for each FAST pass, including establishing connections to the appropriate ground station, setting the telemetry data rate on the spacecraft, and preparing sequential prints and configuration monitors. The procedure also sends NO-OP commands to the spacecraft during each contact to reset the onboard watchdog timer. Upon completion of a pass, the procedure cleans up and then sleeps until the next contact begins. A second FAST workstation also runs ITOS and continuously executes another procedure which replays stored spacecraft housekeeping (HK) files and generates sequential prints suitable for creating trending plots of important HK mnemonics. In addition, certain data are extracted that are subsequently used for ground-based attitude determination purposes.

ITOS saves spacecraft limit and configuration monitor violations in log files which are pre-processed by shell scripts and then sent to SERS. SERS in turn creates problem alert files and sends out email and pager messages. The system rolls over to the next team member if the on-call engineer does not respond. Operators respond by returning a paging message to SERS or access the system via a web interface. Additional information about each alert as well as a description of the resolution for each problem can be viewed via the web interface. SERS runs on two redundant systems that monitor each other. The FOT is paged if any one of the automated systems exhibits an anomaly. There is also a watchdog system in place to ensure the FAST ITOS systems are running.

FAST ATS loads are generated by the FOT using MPS. This system ingests the schedule and orbit information, and folds them together with instrument commands. The instrument commands are generated by a planning system which is unique to the FAST mission and decides when to activate particular instruments based on memory capacity, orbit position and predicted power budget. Precession maneuvers and other attitude control commands are prepared using a MATLAB utility [17] while the power budget is generated using the Power Management Tool. Both tools were developed at NASA/GSFC.

All ground stations supporting FAST record telemetry in data files that are transferred post-pass via *ftp* to the Data Processing System (DPS) at the SOC. Once the data transfer for a given pass is completed, the DPS automatically performs level zero processing functions. Processed data are subsequently transferred to another host at the SOC for additional science processing and archival.

7. CHIPS OPERATIONS

For command and control, CHIPS uses the Mission Control Center (MCC) software developed by SpaceDev [6,18]. The MCC software includes server and client programs running on distributed hosts connected via a Virtual Private Network (VPN). This network includes dedicated CHIPS routers at each of the ground stations supporting CHIPS, located at U.C. Berkeley, CA, NASA Wallops Flight Facility, VA, and the University of South Australia at Adelaide. The routers provide firewall protection for each segment of the CHIPS network, thus facilitating secure communications over the open Internet. While in contact with a ground station, the CHIPS spacecraft acts as another node on this network, and utilizes *telnet* and *ftp* for data and file transfer.

As with FAST and RHESSI, CHIPS passes are normally taken in a fully automated fashion. The prime MCC program used for pass support automation is *PassExec*. This program sets up the appropriate routers before each pass and then begins sending two commands repeatedly until command verification is received from the spacecraft. The first command turns the CHIPS transmitter on, and the second command sets the telemetry data rate. Once two-way communications are established, *PassExec* waits for a specified period of time and then uploads any files waiting in the upload directory on the MCC system. These files can be target pointing files, batch files or a new flight software patch. Once files are uploaded, *PassExec* initiates the transmission of stored housekeeping and instrument science data files from the spacecraft to the ground. All of these file transfers use *ftp*, so packets dropped during transmission are simply retransmitted. A pass with poor link margin will result in a lower throughput, but no data are lost.

Unlike RHESSI and FAST, the CHIPS spacecraft does not know about upcoming passes in advance. Instead, the pass schedule is maintained on the ground and the spacecraft transmitter is enabled for each pass by ground commands, as described above. The schedule is directly derived from SatTrack products for the Berkeley Ground Station. For pass supports at Wallops and Adelaide, a confirmation from the respective scheduling offices is received, and confirmed passes are included in a configuration file used by *PassExec.* Upon completion of a pass support, *PassExec* spawns the *TlmMonitor* application which transfers HK and science data to another host for processing and archiving. *TlmMonitor* also processes real-time and stored HK files, and scans for yellow and red limit violations. HK data are further processed after each pass, and trending plots are placed on a web server. This quick turn-around combined with web access allows the FOT to respond to alerts very efficiently. As with FAST and RHESSI systems, there are watchdogs in place to alert the FOT about any problems with the MCC or the routers at the remote ground stations. Typically, the latter can be resolved via secure access through the Internet.

8. DATA PROCESSING AND ARCHIVING

Like the MOC, the SOC is a highly automated facility for science data processing, archival, distribution, access, and retrieval. A variety of COTS and GOTS products plus locally developed software have been integrated with a number of commercial hardware systems, such as Sun Microsystems and LINUX-based workstations, servers and RAID storage systems, and a fully automated CD-ROM production system with a label printer.

UNIX shell scripts running on the Sun Microsystems workstations serve as the backbone of the automated SOC data processing system. Primary scripts, initiated by the UNIX *cron* function, scan for incoming data and perform a series of processing steps, depending upon data type. These scripts also spawn additional shell and IDL scripts and programs for further processing, for instance to produce summary and key parameter data, storing data sets on disk arrays, updating databases for local and web access, transferring data to external mirror sites, and starting the CD-ROM processing system. Operational scripts in the SOC also generate numerous processing log files for review by operations personnel, and send out warning messages via email or pager in case processing errors occur, or if data gaps in the telemetry stream are detected. In the latter case, missing data are typically replayed within 12-96 hours, depending upon where the data loss was encountered. The overall system performance of the Berkeley ground data system is very high. For all three missions currently supported, the number of missed passes is very small, and the completeness of the data recovery falls between 99 and

100% on average. Day-to-day operations entail mainly monitoring of system functions, reviewing of data products, and maintenance of the CD-ROM production system.

9. STAFFING

The Berkeley operations facility is staffed by an experienced and well trained multi-mission operations team whose members have accumulated close to 60 person-years with spacecraft and ground systems operations. Team members are cross-trained on flight dynamics, mission planning, spacecraft command and control, ground station operations, and data processing and archiving functions. The university environment also allows involvement of students with non-critical tasks.

10. SUMMARY

The Berkeley operations concept has proven very successful and reliable. Over the last three years we demonstrated that one coherent, integrated facility with a high degree of automation, staffed by a small, but highly skilled, dedicated and well trained operations team can accomplish all mission critical functions to support multiple spacecraft simultaneously, while maximizing data quality and minimizing risk and operations costs. Lights-out operations have been successfully demonstrated with FAST, RHESSI, and CHIPS. Planned system upgrades for the operations center and the Berkeley Ground Station include advanced scheduling functions as well as autonomous orbit determination capabilities to meet the requirements of more complex future missions.

The Berkeley team spends part of the time saved with automating routine functions on further improving systems reliability by refining hardware, software and operations procedures. Of course, there are a number of tasks that are not automated. These include command load generation and uplink to the spacecraft (FAST, RHESSI), attitude determination and power management functions (FAST), generation of observing target files (CHIPS), and special instrument commanding with all spacecraft. Other non-automated tasks include daily systems monitoring, review of trending data and log files, and anomaly resolution.

11. REFERENCES

[1] R. P. Lin, B. R. Dennis and A. O. Benz (Eds.), “The Reuven Ramaty High-Energy Solar Spectroscopic Imager (RHESSI) − Mission Description and Early Results”, Reprinted from *Solar Physics*, vol. 210, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002. ISBN 1-4020-1107-5.

[2] R. F. Pfaff, Jr. (Ed.), “The FAST Mission”, Reprinted from *Space Science Reviews,* vol. 98, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001. ISBN 0-7923-7046-5.

[3] M. Hurwitz and M. Sholl, “The CHIPS University-Class Explorer”, *Bulletin of the American Astronomical Society*, vol. 31, p. 1505, 1999.

[4] Consultative Committee for Space Data Systems (CCSDS) (http://www.ccsds.org).

[5] W. Marchant, M. Hurwitz, M. Sholl and E. Riddle Taylor, “Status of CHIPS: A NASA University Explorer Astronomy Mission”, *Proceedings of the Fifteenth Annual Small Satellite Conference,* AIAA / Utah State University, paper SSC01−V−8, August 13−16, 2001.

[6] K. Rubio, J. Janicik and J. Szielenski, “CHIPSat’s TCP/IP Mission Operations Architecture − Elegantly Simple”, *Proceedings of the Sixteenth Annual Small Satellite Conference,* AIAA / Utah State University, paper SSC02−IV−4, August 12−15, 2002.

[7] Bester Tracking Systems, Inc., “SatTrack Suite V4.5”, Suisun City, CA, USA, 2003.

[8] the Hammers Company, “Integrated Test and Operations System (ITOS)”, Greenbelt, MD, USA, 2002.

[9] Hughes Applied Information Systems, “Command Management Prototype Study Phase Results Report for the ECS Project”, Landover, MD, USA, 1994.

[10] mobileFOUNDATIONS, Inc., “Spacecraft Emergency Response System (SERS)”, Falls Church, VA, USA, 2002.

[11] L-3 Communications, EMP Systems, “Operation and Maintenance Manual for Model 300-11m S-Band Ground Based Telemetry Tracking System”, Simi Valley, CA, USA, 2002.

[12] L-3 Communications, Microdyne, “Model 700-MR Receiver and Model 1620-PC Diversity Combiner Instruction Manuals”, Ocala, FL, USA, 1999.

[13] L-3 Communications, Microdyne, “Model TSS-2000 Signal Simulator Instruction Manual”, Ocala, FL, USA, 1992.

[14] Agilent Technologies, “ESG Family Signal Generators User’s Guide”, Palo Alto, CA, USA, 2001.

[15] Avtec Systems, Inc., “Programmable Telemetry Processor for Windows NT (PTP NT) User’s Manual”, Fairfax, VA, USA, 1998.

[16] Research Systems, Inc., “The Interactive Data Language (IDL)”, Boulder, CO, USA, 2003.

[17] The MathWorks, Inc., “MATLAB”, Natick, MA, USA, 2003.

[18] SpaceDev, Inc., “CHIPSat User’s Manual”, Poway, CA, USA, 2003.