FILE-BASED DATA PROCESSING ON MESSENGER

Christopher J. Krupiarz, David A. Artis, Andrew B. Calloway, Constantine M. Frangos, Brian K. Heggestad, Douglas B. Holland, William C. Stratton

The Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road, Laurel, MD USA 20723-6099
E-mail: Christopher.Krupiarz@jhuapl.edu

ABSTRACT

As part of the system software, MESSENGER will be using a file-based system for the transfer and processing of instrument and spacecraft data. The flow of files within the MESSENGER software architecture begins with the receipt of science data by the main processor and the creation of files containing these data by the flight software. The files are then autonomously selected for downlink via a priority-based algorithm, packaged for transmittal via the CCSDS File Delivery Protocol (CFDP), and radiated to the ground. The ground software reconstructs the files via its implementation of CFDP, performs further processing on the file, and sends it to the operations data archive and the Science Operations Center. The MESSENGER spacecraft operations team manages the overall handling of these files through interaction with both the flight and ground systems.

1. INTRODUCTION

MErcury Surface, Space ENvironment, GEochemistry, and Ranging (MESSENGER) is a NASA Discovery mission to study the planet Mercury. It is scheduled to launch in March 2004 and will include two flybys each of Venus and Mercury, followed by Mercury orbit insertion in April 2009 for a one-year science-gathering mission. MESSENGER will orbit Mercury in 2009 following two reconnaissance flybys each of Venus and Mercury. MESSENGER will investigate key scientific questions regarding Mercury's characteristics and environment [1]. Data are provided by an optimized set of miniaturized space instruments [2] and the spacecraft telecommunications system [3].

MESSENGER is a unique mission for The Johns Hopkins University Applied Physics Laboratory (JHU/APL)in that it is the first spacecraft developed by the laboratory to use an on-board file system for data collection. Previous missions such as the Near Earth Asteroid Rendezvous (NEAR) and the Comet Nucleus Tour (CONTOUR) have used a raw device model consisting of sequentially storing data to a Random Access Memory (RAM) based Solid State Recorder (SSR) and then downlinking blocks of the memory using read/write pointers. MESSENGER, however, has unique constraints that resulted in the selection of a file-based SSR for science data storage, driven by two key points in particular.

First, MESSENGER is a severely downlink-constrained deep space mission because of the combination of available downlink rates and long round-trip light times. To maximize science return, substantial amounts of contingency engineering data will be recorded that would be routinely deleted without downlink unless an anomaly occurs. A file system is the most expedient way to record contingency data and allow mission operations the option of downlinking or deleting such data in a pre-defined and automated fashion. Furthermore the file system readily supports recording of science data with various levels of priority; some low priority files may not be downlinked unless sufficient bandwidth allows it. To fit into the raw partition model, an elaborate partitioning scheme with significant custom software development would be needed to provide the same flexibility that a file system offers.

The second driving constraint is that the majority of science data on the SSR will be Mercury Dual Imaging System (MDIS) instrument images. Raw images will be collected and recorded by the Main Processor (MP), then at a later time read back into processor memory, compressed using several image-compression algorithm options, and stored back to the SSR. This requirement to have flight software logically read back in variable-length data previously written to the SSR is best implemented by using a file system.

With these factors in mind, a software architecture was developed to operate a file system to meet the mission objectives. It was designed to fit within the current JHU/APL flight and ground systems and, where possible, to use proven technology. The flight, ground, and mission operations teams worked closely to ensure the SSR could be managed at great distances.

2. MESSENGER ARCHITECTURE

The primary flow of data within the MESSENGER architecture begins at the instruments as shown in

Figure 1. They feed data to the flight computer where the data are placed into files via an operations-controlled filter table. A file is either downlinked automatically or a mission operations team member manually selects a file to transmit. Upon receipt by the ground system, the file contents are archived in the Mission Operations Center (MOC) and delivered to the Science Operations Center (SOC).

will compress any available images previously stored on the SSR. MESSENGER allows for Integer Wavelet Transform (IWT) compression, "jailbarring" of the data, and subframing. IWT is similar to the common JPEG compression and allows for selectable levels of compression, both lossless and lossy. Jailbar compression reduces an image by removing columns of data. Subframing is performed prior to either IWT or jailbar compression and it selects only certain parts of the image for compression.

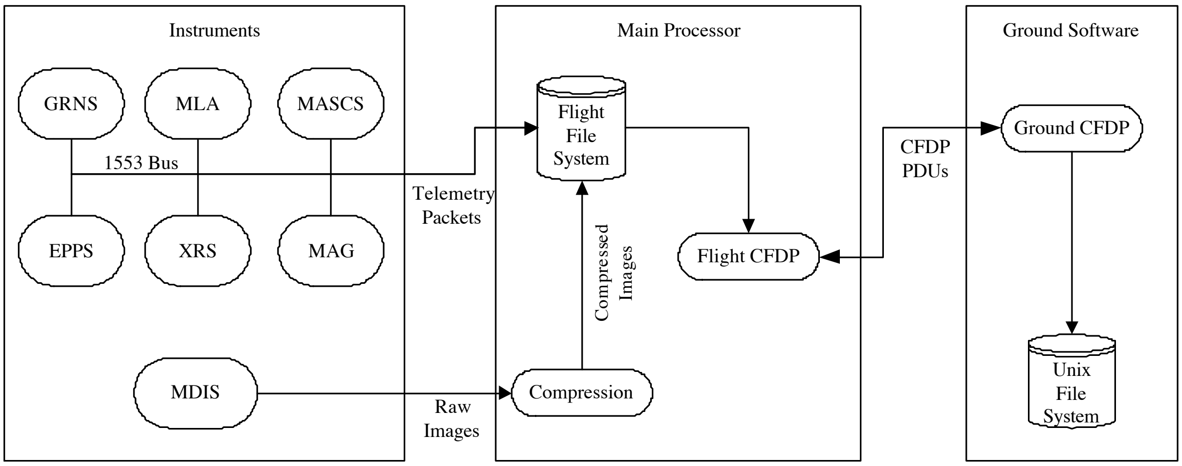

Figure 1: MESSENGER File Data Flow

2.1. Flight Software

The MESSENGER flight computer consists of an Integrated Electronics Module (IEM) with a Main Processor (MP) running Command and Data Handling (C&DH) and Guidance & Control (G&C) along with a Fault Protection Processor (FPP). A backup IEM is normally powered off except for software updates or anomalies. Files are stored on the MP. This module consists of a RAD6000 computer running at 25 MHz with 8 Megabytes of RAM. The MP communicates via a Peripheral Component Interconnect (PCI) backplane to an SSR with 8 Gigabits of RAM. The VxWorks operating system developed by Wind River Systems, Inc., was selected on the basis of its proven performance in various space missions including Mars Pathfinder and the Mars Exploration Rovers. The science data to be recorded include MDIS Images, packetized instrument data, housekeeping data, and ASCII directory listings.

The MDIS image data are transmitted to the MP via a high-speed interface in 64-kilobyte blocks. Upon receipt of the first block of data, a file is opened on the SSR and the data placed into the file via a Direct Memory Access (DMA) transfer. At a point determined by operations personnel, the flight software

The other instruments on MESSENGER include the Gamma-Ray and Neutron Spectrometer (GRNS), the Magnetometer (MAG), the Mercury Laser Altimeter (MLA), the Mercury Atmospheric and Surface Composition Spectrometer (MASCS), the Energetic Particle and Plasma Spectrometer (EPPS), and the X-Ray Spectrometer (XRS). All of these instruments transmit their data in packets via a 1553 data bus back to the MP. The packets are then placed sequentially into files previously opened for those instruments by operational command loads.

The final type of data is engineering data from the MP or the FPPs. Individual telemetry packets can be selected for storage in a file for playback. This process is similar to storage of instrument data.

In the event the mission operations team requires a directory listing, the MP can be commanded to list the contents of a directory in a file. Two types of directory listings are provided. One lists the file names and pertinent information such as file sizes, mission elapsed time (MET), and VxWorks attribute information, while the other hierarchically lists how many files are in a given directory and the space allocated for that directory.

Upon opening the file for data collection, a directory is chosen in which to place the file. The MESSENGER team developed a priority-based directory structure as shown in Figure 2. The five main directories defined for the SSRs are /REC, /DNL, /TRASH, /OPNAV and /IMG, and the downlink directory in turn has ten priority sub-directories. The /REC directory contains open files of actively recording packets. The /TRASH directory contains files that are candidates for deletion. Successfully downlinked files can be configured for automatic deletion or for placement into /TRASH. The /IMG directory contains science raw images unprocessed by the MP, and the /OPNAV directory contains unprocessed raw images that are used for navigation support.

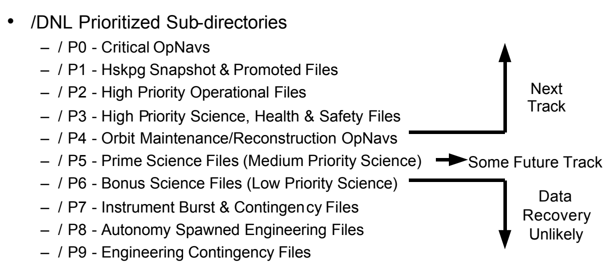

Figure 2: MESSENGER Directory Structure

Once the data have been stored, the next step is to transmit the information to the ground. A file downlink can be initiated in one of two ways: a file selected by mission operations or autonomously by the file software. The auto-playback mode may be enabled or disabled by mission operations command. The first method is the most straightforward. A command was defined to select a given file by name and downlink it. This is particularly useful when a directory listing is created since the filenames are listed in the directory files. However, for the majority of flight operations, an autonomous file downlink capability was required. This capability alleviates the burden on mission operations to select all the files individually to be downlinked. The algorithm works with the directory structure to determine what file should be downlinked next. It constantly scans from P0-P9 to determine if a file exists in any of the directories. If so, it creates a playback list for that directory from oldest to newest and initiates the downlink process for those files. Mission operations can select the frequency of these scans and can also move files up and down the directory structure, thereby changing their downlink priority. The highest priority files can also automatically interrupt the current playback list or even an individual transaction depending on the circumstances. This reduces the need for operations to constantly monitor individual transaction status in order to ensure the downlink of the most critical high-priority files.

2.2. Flight and Ground Communication

For the actual file transmission, MESSENGER will be one of the first deep space missions to use CFDP. CFDP is an FTP-like standard that was developed specifically for missions where there is a long communication delay and an asymmetric bandwidth [4]. The protocol consists of two entities, one in space and one on the ground, which communicate via a series of data transfers that ensure file delivery. These data transfers consist of various Protocol Data Units (PDUs).

Following a typical transaction shown in Figure 3, the first step in sending a file to the ground is the transmission of a Metadata PDU that contains information about the file such as the name and size. The protocol can handle a dropout of this information by requesting its retransmission and assigning temporary information until it is received. The file is then broken into smaller pieces called File Data Units (FDUs), and each of these is transmitted to the ground. Finally, an End-of-File (EOF) is sent to indicate to the ground that all the pieces of the file have been transmitted. At this point a timer is started on the flight side and the flight software awaits an Acknowledge (ACK) PDU from the ground. Receipt of this PDU by the flight software indicates that the ground software successfully received the EOF. If EOF ACK is not received in the specified time, the flight software re-transmits the EOF PDU. The length of the timeout is dependent upon the round trip light time (RTLT) between the Earth and the spacecraft. Mission operations has command control over the various CFDP timers and settings in order to optimize the system for various mission phases.

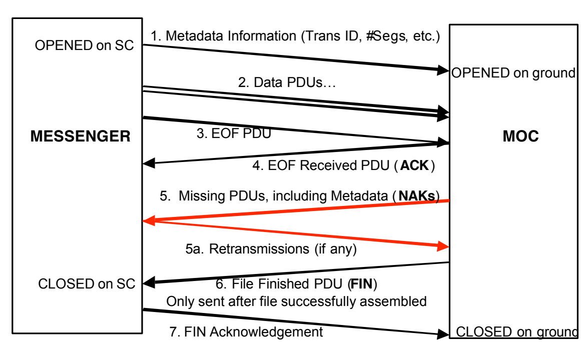

Figure 3: CFDP File Transmissions

Once the ground software receives an EOF PDU, it examines the file and determines if any pieces are missing or corrupt. If so, it transmits a Negative Acknowledgement (NAK) to the flight software listing the pieces of the file that need to be resent. The flight software responds by retransmitting those pieces of the file. The ground software uses a timer to ensure the flight side is responding. When the ground receives the resent data and confirms the file is complete, it transmits a Finished Indicator (FIN) to instruct the flight software that the file has been successfully received or, if there was a failure to reconstruct the file, unsuccessfully received. A FIN would also be generated if operations cancels a transaction in order to keep the flight and ground segments synchronized, within the corresponding latency.

CFDP provides for various configurations depending upon mission requirements. In the example just discussed and for the MESSENGER mission, Acknowledged Mode and Deferred NAKs were selected. Other options include Unacknowledged Mode, where the flight side would just send the PDUs without replies from the ground, and Immediate NAKs, where the NAK is sent when it is first determined data are missing or corrupted as opposed to waiting until an EOF is received. The latter may be more appropriate for a low Earth-orbiting mission in which communications are essentially immediate. CFDP can also transmit from ground-to-flight, although only flight-to-ground was used on MESSENGER.

The flight version of CFDP was developed in-house at JHU/APL [5]. This decision was driven by the resource constraints of the MP including both RAM usage and processor usage. It is a simplified version of a CFDP implementation although it follows the CFDP specification in the pieces that it does implement. The ground-based CFDP is an implementation developed by the NASA Jet Propulsion Laboratory (JPL). It was adapted for use with the JHU/APL ground architecture and greatly reduced the amount of development time needed to add this capability.

2.3. Ground Software

Figure 4 illustrates the JHU/APL ground software architecture, common to all of JHU/APL's NASA space missions. The common ground system is comprised of four functional areas: (1) telemetry, (2) commanding, (3) planning and (4) assessment. The commanding and telemetry functional areas work with the EPOCH 2000 system to provide control, monitoring and display capabilities for the spacecraft and for the Ground Support Equipment (GSE). The planning area provides offline commanding and spacecraft management planning. The assessment area is responsible for data archiving and production of data products used by the operations personnel for spacecraft assessment and by other data users [6].

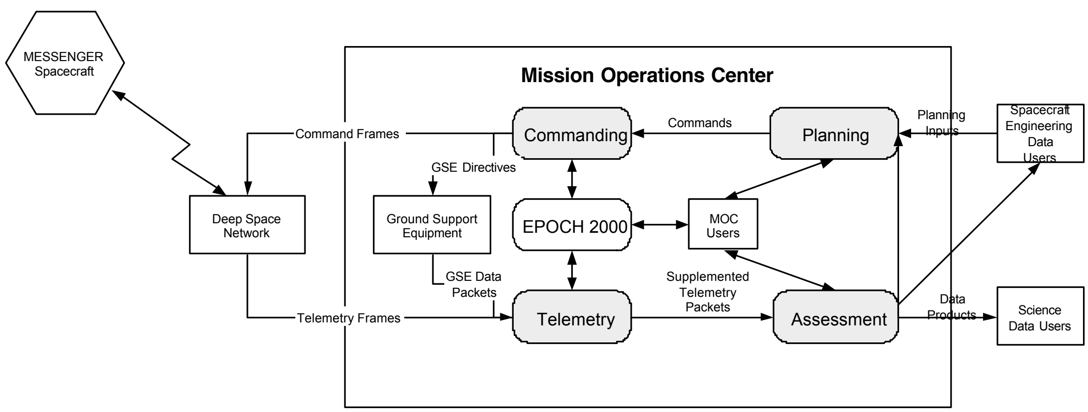

Figure 4: JHU/APL's Common Ground System Architecture

A priority in introducing file system operations to the ground software architecture was to minimize and isolate the changes. This is especially true since other missions currently in development at JHU/APL such as the Solar-Terrestrial Relations Observatory (STEREO) and New Horizons will not be using a file system. This concern was primarily addressed by integrating a CFDP implementation licensed from JPL.

The JPL CFDP software is delivered as a complete, self-contained system. It provides both a core implementation of the complete CFDP standard as well as a test suite for verifying operation in a stand-alone system. In order to integrate the software into the JHU/APL architecture, "glue" programs needed to be developed to handle the role of the end-user of CFDP and to supply PDUs to the CFDP core through the transport layer.

JPL's CFDP implementation assumes the presence of a "user program" that issues directives and handles indications for CFDP control and monitoring purposes. JPL provides an example program to assist in the development of a compatible program for the end user that can then be modified to suit the needs of a particular architecture. JHU/APL developed its own implementation of this process which wraps JPL's CFDP software. It implements the standard ground systems communication protocol with sub-modules that allows the JPL CFDP to appear as another process to the rest of the JHU/APL ground system, as if it were another piece of ground support equipment capable of receiving telemetry or sending commands.

To reconstruct PDUs from incoming telemetry, an existing ground system process was extended to take telemetry frame data and produce PDUs for input into the JPL CFDP process. A selected stream of MESSENGER telemetry frames is dedicated to transporting CFDP PDUs. The PDUs are not aligned in the transfer frame data areas, so the software reconstructs PDUs from data segments spanning frames. Upon receipt of all elements of a PDU, it is submitted to the JPL CFDP software.

The other primary addition to the MESSENGER ground system is that it had to perform post-processing of three different types of files downlinked via CFDP: (1) files of telemetry packets, (2) files containing ASCII spacecraft directory information, and (3) compressed image files. The file types are distinguished by special naming conventions provided by the MESSENGER flight software. Post-processing begins when the CFDP core determines that file has been completely downlinked. For files of telemetry packets, additional headers are added so that other ground software tools can process them at a later time. Directory files are simply formatted and displayed in a window on the screen. Image files that are needed immediately for other processing in the MOC are decompressed. After the files are processed, they are distributed to the appropriate location.

3. MISSION OPERATIONS

The concerns of mission operations fall primarily into two areas: (1) management of the SSR and (2) management of downlinking files through the use of CFDP.

SSR management primarily concerns the deletion of files. Although files can be individually chosen for downlinking, the mission operations team will rely on the autonomous file selection on-board the spacecraft. Therefore, the primary concern for SSR management is avoiding SSR saturation and reprioritization of files within the directory structure.

In order to avoid recorder space saturation, files will be periodically deleted from the SSR. For example, a pre-defined set of spacecraft and instrument contingency files will be generated by command load each day, in case information in addition to the standard housekeeping packet is desired. These files will be maintained for at least seven days in low-priority directories. A day of contingency files from the previous week will be routinely deleted each day in a rotating fashion to make room for the next day, unless operations decide that the information is needed. During the cruise phase of the mission, closed files will automatically be moved to the /Trash directory, but they will remain there indefinitely since recorder space is not as critical. Mission operations will delete ranges of these files via command load once successful product delivery from the MOC to the SOC is confirmed. Files will be automatically deleted upon transaction closure, rather than moved to the /Trash directory, once in-flight experience and confidence with the system is achieved. It may also become necessary for operations to delete low priority science files to make room for new medium priority science if recorder space approaches saturation levels.

The movement of files to higher-priority directories for earlier downlink may occur somewhat regularly during Mercury orbit operations, and may also occasionally require the re-evaluation command if those files are desired prior to the next planned track. For example, operators will have directory size telemetry to infer that autonomy-spawned engineering files in /P8, or XRS flare or MAG burst files in /P7 have been created. It will also be possible to downlink a file directly by name regardless of the resident directory, and it would become the next file to be inserted into the downlink stream. This may become the preferred method of early downlink when only one or two files are desired.

During Mercury orbit operations, one 8-hour Deep Space Mission System (DSMS) track will be supported each Earth day. The RTLT during that time will vary from 10 to 25 minutes. Downlink bandwidth will vary by pass.

Prior to the end of each track, playback will be stopped by a stored command and the overall CFDP engine will key off that and freeze as well. The time period for stopping the playback is chosen to correspond to the one-way light time (OWLT) between Earth and the spacecraft. It is likely that some MP autonomy rules will also be developed that would stop the playback, and therefore freeze CFDP, whenever a loss of downlink is detected, to reduce unnecessary retransmissions. Although the overall CFDP protocol will freeze whenever the playback is stopped on-board, suspension of individual transactions is not planned for MESSENGER due to the complexity that is introduced by the large RTLTs. Cancellation of individual transactions will still be maintained, however, since certain operational scenarios may require this. Operations will not relinquish spacecraft attitude control to science operations until at least an OWLT after the scheduled end of track (EOT). This will ensure that the spacecraft can still receive commands or PDUs that are radiated from the ground up until the very end of the track to maximize bandwidth. For example, FINs, NAKs and EOF Acks will continue to flow out until just before command modulation is turned off, about five minutes before the end of each track. Even though CFDP will already be frozen by time-tag command by the time they reach the spacecraft, the FINs will still be processed and those transactions will be closed to help avoid saturation. The NAKs, EOF Acks and FIN Acks, however, will all be buffered on-board and will not be processed until CFDP resumes on the following track. The FIN ACKs and NAK retransmissions are buffered once CFDP is frozen since they would not be able to reach the ground before the EOT, and the EOF ACKs cannot be processed while the CFDP timers are all frozen, so they are buffered.

When CFDP is resumed for each new track via a stored command, any buffered NAKs or ACKs will have first priority for processing and downlink. Then the standard automatic re-evaluation of the file priorities occurs for the downlink directories. CFDP will be timed to start just under an OWLT from start of track, so that the data will begin arriving at the station a few minutes after the change to coherent mode several minutes into each track. The ground timers will be frozen at the end of each track to ensure secondary FINs are not sent erroneously to the spacecraft between tracks.

Real-time tracks will be scheduled to center around the Mercury orbit periherm as much as possible, occurring every other 12-hour Mercury orbit. As a result, most tracking events will require a DSMS station handover at some point during the track in order to match the orbit geometry. Some handovers will probably occur periodically during the cruise phase of the mission, as well. The ground segment of CFDP discards duplicate PDU data, so although the same data may briefly arrive from two different stations simultaneously on two different telemetry streams, it is not believed that handovers will cause any problems with accounting or unnecessary retransmission of data. This scenario will be thoroughly tested in simulations.

The operations crew will be provided with a report prior to each track that will contain the expected files to be downlinked for that track, based on the modeled state of the recorder and the expected downlink rates. They will also be provided with a report of files that were started but not yet completed, based on status information pulled from the CFDP log files and tabulated into report format. The operations crew will also be responsible for periodically verifying that the ground system is properly archiving data and routing data to the SOC.

Once the software was developed sufficiently for operational interaction, it became apparent that early operational requirements needed considerable modifications that evolved in an iterative fashion. This section shows the types of compromises that were required from all parties as development proceeded in order to implement a realistic operational system.

- Operations needed a Set Interleave Ratio to Zero directive in order to pause uplink PDUs during critical load activities (and corresponding status telemetry).

- Operations needed a workstation failover capability that resulted in a parallel instantiation implementation.

- Operations needed a file-based memory storage capability for the ground data storage to support workstation reboots (also required memory sizing).

- Operations needed a directory listing capability (and corresponding user-friendly output conversion) in order to assist with modelling the recorder contents.

- Operations requested a second CFDP diagnostic packet in order to reduce bandwidth by removing less important information from the primary downlink packet.

- Operations requested an increase in the number of transactions from 200 to 600.

- Operations requested that the uplink PDUs would always be uplinked in CCSDS bypass mode automatically.

- Operations requested that if a file transaction is cancelled, it will not linger in /DNL but will be returned to the previous directory.

- Operations requested a delete and move by MET range capability.

- Operations requested the ground software maintain a Transaction ID to filename mapping to assist with identifying IDs for realtime Cancel commands.

- Operations accommodated flight software by shortening directory names and filenames, using standard naming conventions, and using a 32-bit number in filenames instead of text strings

- Max PDU size was defined to be 1024 bytes so that a PDU would at most span only two transfer frames if dropouts occur.

- Operations accommodated software by taking responsibility for not duplicating filenames so the flight software could eliminate the search for duplicate names to optimize processor utilization.

- It was mutually agreed that individual transaction Suspend capability was no longer required.

- Operations worked with flight software to define what telemetry is reset with counter clear commands.

- Operations accepted flight software implementation of flight CFDP parameter information storage changes and how they affect open transactions.

4. SUMMARY

MESSENGER is the first JHU/APL spacecraft to use an on-board file system. Whereas previous missions used a raw partition approach to managing the spacecraft's SSR, MESSENGER's unique characteristics such as its severe downlink constraints and its image management required an alternative method for data storage. This change to the JHU/APL software system architecture required modifications to both the flight and ground systems, as well as the methods used by mission operations to control data flow. Flight systems were added to collect data into files and to select these files autonomously for downlinking. To ensure transmission of the data, CFDP was chosen as an FTP-like protocol for transferring the files. Software was added to the ground to receive the files and to post-process the data depending upon file content. Finally, mission operations developed strategies for managing the files as well as the downlinking of the data.

6. REFERENCES

1. Solomon, S. C., R. L. McNutt, Jr., R. E. Gold, M. H. Acuña, D. N. Baker, W. V. Boynton, C. R. Chapman, A. F. Cheng, G. Gloeckler, J. W. Head, III, S. M. Krimigis, W. E. McClintock, S. L. Murchie, S. J. Peale, R. J. Phillips, M. S. Robinson, J. A. Slavin, D. E. Smith, R. G. Strom, J. I. Trombka, and M. T. Zuber, The MESSENGER mission to Mercury: Scientific objectives and implementation, *Planet. Space Sci., 49,* 1445-1465, 2001.

2. Gold, R. E., S. C. Solomon, R. L. McNutt, Jr., A. G. Santo, J. B. Abshire, M. H. Acuña, R. S. Afzal, B. J. Anderson, G. B. Andrews, P. D. Bedini, J. Cain, A. F. Cheng, L. G. Evans, W. C. Feldman, R. B. Follas, G. Gloeckler, J. O. Goldsten, S. E. Hawkins, III, N. R. Izenberg, S. E. Jaskulek, E. A. Ketchum, M. R. Lankton, D. A. Lohr, B. H. Mauk, W. E. McClintock, S. L. Murchie, C. E. Schlemm, II, D. E. Smith, R. D. Starr, and T. H. Zurbuchen, The MESSENGER mission to Mercury: Scientific payload, *Planet. Space Sci., 49,* 1467-1479, 2001.

3. Santo, A. G., R. E. Gold, R. L. McNutt, Jr., S. C. Solomon, C. J. Ercol, R. W. Farquhar, T. J. Hartka, J. E. Jenkins, J. V. McAdams, L. E. Mosher, D. F. Persons, D. A. Artis, R. S. Bokulic, R. F. Conde, G. Dakermanji, M. E. Goss, Jr., D. R. Haley, K. J. Heeres, R. H. Maurer, R. C. Moore, E. H. Rodberg, T. G. Stern, S. R. Wiley, B. G. Williams, C. L. Yen, and M. R. Peterson, The MESSENGER mission to Mercury: Spacecraft and mission design, *Planet. Space Sci., 49,* 1481-1500, 2001.

4. *CCSDS File Delivery Protocol (CFDP).* Recommendation for Space Data System Standards, CCSDS 727.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, January 2002.

5. Krupiarz, C. J., S. C. Burleigh, C. M. Frangos, B. K. Heggestad, D. B. Holland, K. M. Lyons, and W. C. Stratton, The use of the CCSDS file delivery protocol on MESSENGER, *Space Operations 2002 Conference, American Institute of Aeronautics and Astronautics (AIAA),* abstract T5-35, Houston, TX, October 9, 2002. (CD-ROM)

6. Stratton, W. C., C. M. Frangos, J. J. Harrison, and D. B. Holland, Reuse of the JPL CFDP software in the APL Common Ground System, *Proceedings of the 5th*

International Symposium on Reducing the Cost of Spacecraft Ground Systems and Operations (RCSGSO), Paper 103-A0036, 8 pp., Pasadena, CA, July 8-12, 2003.