Telemetry Routing, Archive and Retrieval: A Flexible, Re-Usable Architecture

Johns Hopkins University
Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723

Priscilla Mckerracher
240-228-4474
priscilla_mckerracher@jhuapl.edu

Kenneth Heeres
240-228-5853
kenneth_heeres@jhuapl.edu

Paul Lafferty
240-228-5285
paul_lafferty@jhuapl.edu

Kevin Lyons
240-228-7099
kevin_lyons@jhuapl.edu

Keywords: Mission Operation, Telemetry Archive, Telemetry Retrieval, Telemetry Playback, Integration and Test

Abstract. In most Mission Operation Centers the routing, archival and retrieval of spacecraft telemetry are routine, required activities. Across different spacecraft programs variations in the requirements for these activities can be identified. This knowledge of how the requirements vary can then be used to develop generic software processes to support these functions. The development of a fully re-usable architecture can be an evolving process. An existing design from a prior program is often retro-fitted to meet the new requirements. After some iteration it is a useful exercise to review the modifications that have been made across different programs and to plan how to generalize the architecture further for use in future programs. The goal should be to accomplish re-use in the most cost efficient manner. This is the goal of this paper.

The requirement variations of concern are found in the following areas: telemetry rates supported by routing and playback, the quantity of data archived, the criteria used for ingest to the archive, the selection criteria used for routing and retrieval, and the formats supported for routing, archive, and retrieval. In addition, while the general requirements for routing, archival and retrieval are present at all stages of spacecraft development, each development stage may also have variability in the detailed requirements. For example, the data rates and data volumes requiring support during spacecraft integration and test may be significantly higher than the rates and volumes supported during flight operations. A set of processes to support both flight operations and integration and test can be a cost-saver to the overall mission.

In addition, other status information from ground systems is routinely collected and archived by Mission Operations Centers (MOCs). By packaging this information in "telemetry packets" it is possible to use the telemetry archive and retrieval system to support archive and retrieval of ground status from hardware and software systems in the MOC. Again, the general purpose use of a core set of processes supporting telemetry can be a cost-saver.

In this paper we will review the architecture and planned re-use of the TIMED Mission Data Center Telemetry Router, Archive Spooler, Archive Ingest and Archive Playback components. A version of the Telemetry Router was used on the FUSE program. A version of all of these components is planned for the CONTOUR, STEREO, and MESSENGER programs.

Introduction

Upon receipt by the ground station, the telemetry data received from a spacecraft RF system is transferred to a standard computing system. Now it is technically no longer telemetry. It resides in a computer system as data. However, the data "flow" does not end in a one-stop operation. Remote monitoring or archiving systems "down-stream" have requirements for receiving the one-time "telemetry" data as quickly as possible. Remote mission operation centers require telemetry receipt for display and monitoring purposes, or sometimes for quick-look processing of science data. Another use includes processing data from the telemetry for feedback into a navigation system for a satellite maneuvering its way to an encounter with an asteroid or comet.

In order to transfer the telemetry data from one machine to another, or from one process to another, a telemetry router application may be used. What is meant in this case by telemetry router is a computer application, which receives a unit of telemetry data via a TCP/IP socket connection, then transfers it to another TCP/IP socket connection. The port designations represent a particular system configuration. The unit of data transferred is a CCSDS formatted transfer frame wrapped with additional header information.

The router may transfer the data to another router, to a telemetry processing system, or to an archival system. For real-time processing appropriate buffers or queues must be established to keep up with the maximum telemetry delivery rate. Optionally, some processes may elect to drop data if they cannot keep up with the incoming rates. This is an appropriate option for real-time display and monitor systems; however, the overall system must ensure complete data archive. Once the data is resident in the data archive, it may be retrieved in the same format as stored, or different formats may be made available. In addition, criteria may be applied to mask out telemetry data that is not of interest to the end user, or to present the data in a different chronological order from that in which it was received. The users of the data may be mission operations center flight controllers, who are interested in spacecraft health & safety or attitude control, or science payload teams, who wish to monitor and process data from their particular instrument located onboard the spacecraft.

The Applied Physics Laboratory Space Department has developed a telemetry routing, archival and retrieval system which has passed through extensive Integration & Test and Mission Simulation use on the TIMED ground system. It is ready for post-launch use on TIMED and is being adapted for future use on three other NASA deep space missions: CONTOUR, STEREO and MESSENGER. This paper will review the requirements for each of these missions, the Telemetry Router and Archival architecture, then present re-use plans and lessons learned.

Requirements

Table 1. presents a comparison of the approximate processing rates and archive requirements for telemetry systems supporting the TIMED, CONTOUR, STEREO and MESSENGER missions. The most stable requirements exist for the most mature mission, the TIMED project.

Table 1. Telemetry Processing Requirements for different NASA Missions
Kpbs - Kilo bits per second
Gbytes = Giga Bytes

Mission	**TIMED**	**CONTOUR**	**STEREO**	**MESSENGER**
Maximum Real-time channel telemetry rate	~20 Kbps	~70 Kbps	~40 Kbps	10's Kbps
Maximum SSR channel Telemetry Rates	~4 Mbps	~70 Kbps	~350 to 700 Kbps	~125 Kbps
Maxiumum I&T Real-time Telemetry Rates	~100 Kbps	~70 Kbps	~350 to 700 Kbps	~125 Kpbs
Archive Ingest Rates	300 Mbytes in 10 minutes			
Raw Telemetry Volume	~800 Gbytes/mission ~1 Gbyte/day - mission ~120 Gbytes/I&T ~2 Gbyte/day - I&T	~70 Gbytes/mission	~1000 Gbytes/mission	~10 Gbytes/mission
Archive Criteria **Ground Receipt Time = GRT** **Spacecraft Time = SCT** **Mission Elapsed Time = MET**	GRT UTC seconds since Jan 6, 1980 epoch SCT (GPS based) Data Quality	GRT UTC seconds since Jan 1, 1970 epoch SCT MET seconds	GRT UTC day of year, milliseconds of day SCT UTC seconds	GRT UTC day of year, milliseconds of day SCT MET seconds Must handle variable length telemetry packets.
Transfer Frame Playback Criteria	Source Front End Virtual Channel Frame Quality GRT	Minor changes to Source & Front End Definition, plus Deep Space Antenna additions. GRT format changes.	Minor changes to Source & Front End Definition, plus Deep Space Antenna additions. GRT format changes.	
Packet Playback Criteria	Virtual Channel Application ID Sequence count Frame Quality SC Time	SCT format changes. Frame Quality definition changes.	SCT format changes. Frame Quality definition changes.	

Architecture

The collection of processes which provide routing, archive and retrieval services for the TIMED ground operations is known collectively as the "Telemetry Server" (TS) subsystem. The Telemetry Server provides required functionality to the TIMED Mission Operation Center and the Mission Data Center (MDC). The Telemetry Server is a TCP/IP based client/server system. Its purpose is to take telemetry from many sources and route the data to external clients or to internal clients for archival and later playback. It consists of multiple components, all of which communicate with each other using TCP/IP. It also routes data to a Raw Archive Tape Server (RATS) for back-up storage to off-line media.

Data are supplied to the system in one of two ways. Real-time data are delivered using TCP/IP sockets and batch data is delivered to the system, using FTP. Real-time and playback data are obtained from the system using TCP/IP sockets. Files of data can be obtained using an HTTP interface but even in this case, making a TCP/IP socket connection and replaying the result to form the file.

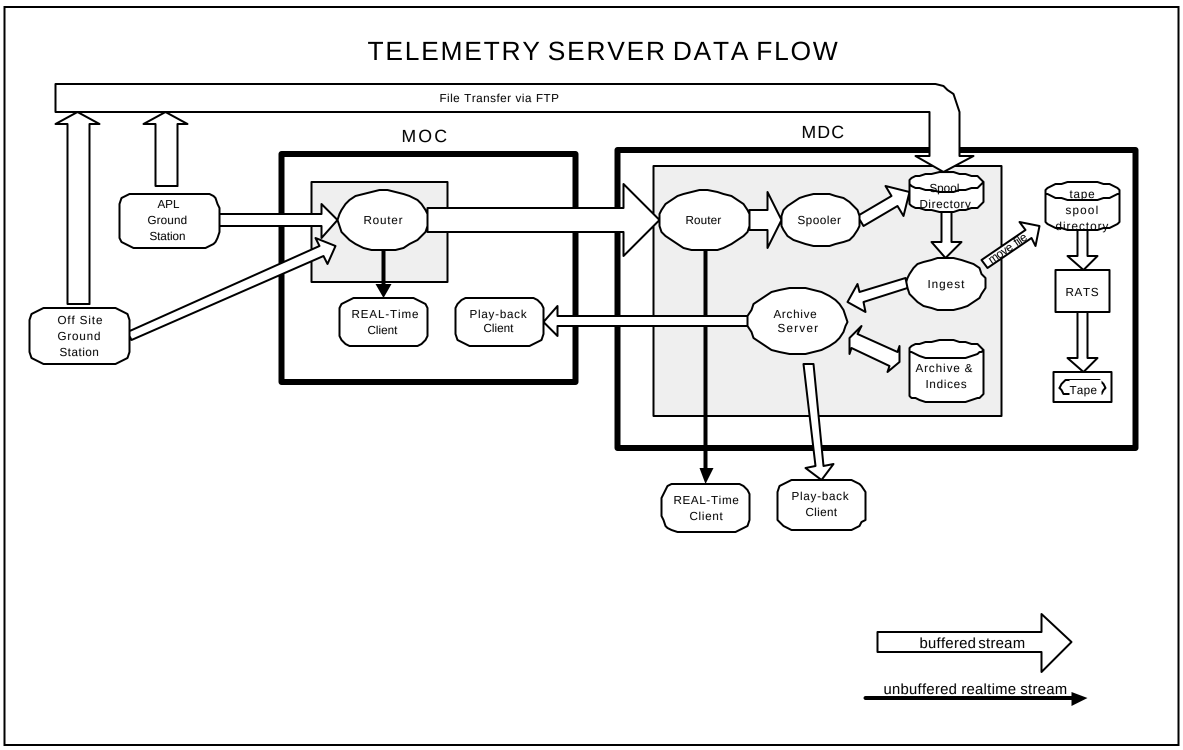

Figure 1. Telemetry Server Data Flow

The format for data within the TS (Telemetry Server) system is based on CCSDS compliant transfer frames and packets. CCSDS transfer frames are augmented with information by a ground receiving station, adding a GRH (Ground Receipt Header), or by TS itself to form a STF (Supplemented Transfer Frame). Other data types include: STP – Supplemented Telemetry Packet and PTP – POC (Payload Operations Center) Telemetry Packet.

The TS (Telemetry Server) consists of several independent cooperative processes. The key processes are the Router, Archive Server, Spooler and Ingest processes. The first process describe is the Router.

Router

The purpose of Router is to move real-time telemetry from one client process to another. The client processes can be local or remote. A sending client connects to socket using a defined port number. A receiving client connects to a defined port and then issues configuration commands to the Router. A receiving client process can configure its connection to Router so that data is filtered by APID (APplication IDentifier) or other information in the GRH and it can set the type of data it is to receive. The Application Identifier normally maps directly to the subsystem, to a specific onboard instrument system, or to the ground support equipment, which originally produced the telemetry packet. After the client has configured the connection, data that arrives at the Router is sent to the client.

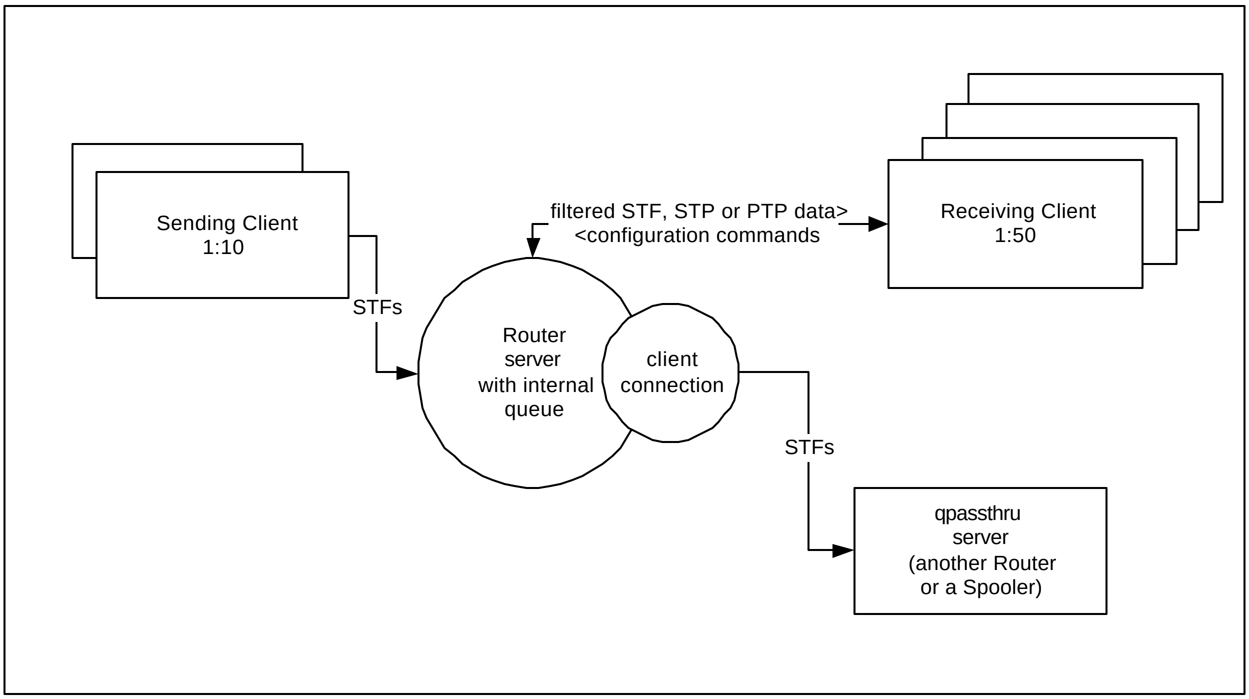

Figure 2. Router TCP/IP connections

All but one of the receiving clients is set up to use a non-blocking TCP connection. If the client is not ready to receive the data then the client is skipped. A special port, called the passthru port, uses a normal (i.e. blocking) TCP connection. For the passthru port, Router act as a client and attempts to connect to a server. The server can be another Router, or a Spooler Process, which supports the archiving function. For the passthru port connection, Router uses an internal dynamic queue to hold data. The maximum size of the queue is configurable but its default size is set to handle an expected burst of data. If the queue fills up then Router will be blocked which will in turn possibly block the sending clients. The standard sending clients are designed to detect when they are being blocked and to reduce the amount of low priority data they are sending to help clear the blockage.

Internally the router uses the `select` statement to process TCP/IP socket requests. Router uses TCP/IP connections to do real-time reconfiguration of the router and to provide status information. Currently Router supports 10 sending clients and 50 receiving clients. Clients are organized into groups and groups have a limited number of connections.

The normal use of the Router is to act as a router for telemetry data to a subnet. A ground station will connect to it and begin sending telemetry. Telemetry processing and display stations in the subnet will connect to the Router and start receiving telemetry. If configured the Router will attempt to connect to a server to pass data through to the server. The server is normally another Router in another subnet or another component in TS called the Spooler. Spooler lies between the Router and the AS (Archive Server).

Archive Server

The purpose of the Archive Server (AS) is to store the data in its permanent online storage location, to index the data, and to provide playback services to clients. The data is indexed by the time it is received on the ground (GRT, Ground Receipt Time) and, optionally, by the spacecraft time (SCT). (The details of the indexing are described elsewhere in this paper.) All of the telemetry received during a single day is stored in a single file. The indexing keeps track of what file and what byte offset a particular record is in. Playback services are included as part of the AS so that we can provide "immediate" playback services without incurring the overhead of excessive file locking. Otherwise, playback clients might be continuously trying to access data from archive files while they were still be written. This design was driven by a requirement to provide playback services of the indexed data within a few minutes of receipt.

Spooler & Ingest

Two other minor applications are also part of the system, Spooler and Ingest. The purpose of Spooler is to quickly take the data from a socket connection and write it to a disk. This is to prevent a Router from blocking because the system is busy indexing. The purpose of Ingest is to read files from the spool directory and feed them into AS. The batch processes that FTP files may also put files in the spool directory, so that Ingest can read them up and feeds them to AS.

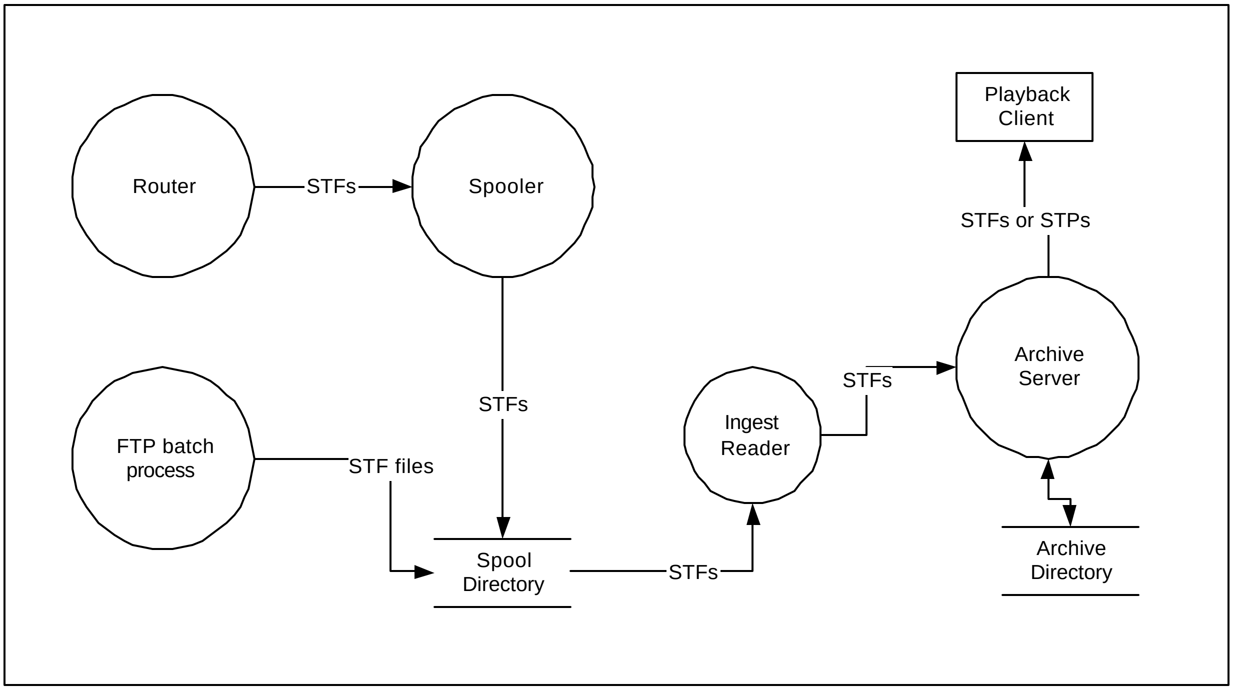

Figure 3. Spooler, Ingest and Archive Server Data Flow

Indexing

The purpose of the indices built by the AS is to perform a virtual cleaning and merging of the data. Telemetry data is concatenated in the order received into one file (the *tlm* file) for each operational day. Simultaneously two other files are created and/or updated by the archive server indexing process. One of the other two files is the *gri* file, containing information that allows the user to retrieve data in the order, which it was received upon the ground. The *gri* file is always updated upon receipt of a new, non-duplicate telemetry record. The other file is the *sci* file. This file is only updated if the indexing by Spacecraft Time option is turned on. The *sci* file contains information, which allows a user to retrieve data in the order in which it was created onboard the spacecraft. In addition both the *gri* and *sci* files contain other data qualifiers, which allow the user to filter the type of data retrieved.

One GRT or SCT index record is created for each new telemetry frame added to the *tlm* file. The index records each contain: a) data needed for record sorting and playback filtering, b) link pointers and c) location of the data in the *tlm* file. The GRT index records contain: previous and next pointers, byte offset of the STF in the *tlm* file, source ID (e.g., spacecraft), Front End ID, Virtual Channel, and GRT. The SCT index is similar to the GRT index, but is an index of the telemetry packets structures found within the telemetry frame. The SCT index contains previous and next pointers, byte offset of the STF in the *tlm* file, spacecraft time, sequence number (i.e., CCSDS packet sequence number for packets with that APID), APID, GRT, packet number (i.e., number of packet within STF), and frame quality flag. The GRT in the SCT index is required to determine the *tlm* file where the raw data is located. The first index record in the each index file is an index object whose only populated members are the previous and next indices. These are set to indicate the first and last index records in the file.

The indexing process was also designed to avoid archiving duplicate telemetry frames. Duplicate telemetry frames can occur in the system when data is received from different frame routing mechanisms. For example multiple front ends or ground stations could be receiving spacecraft data simultaneously. In addition, when duplicates are encountered, the indexing process will attempt to replace a transmission frame of poor transmission quality with one of better quality.

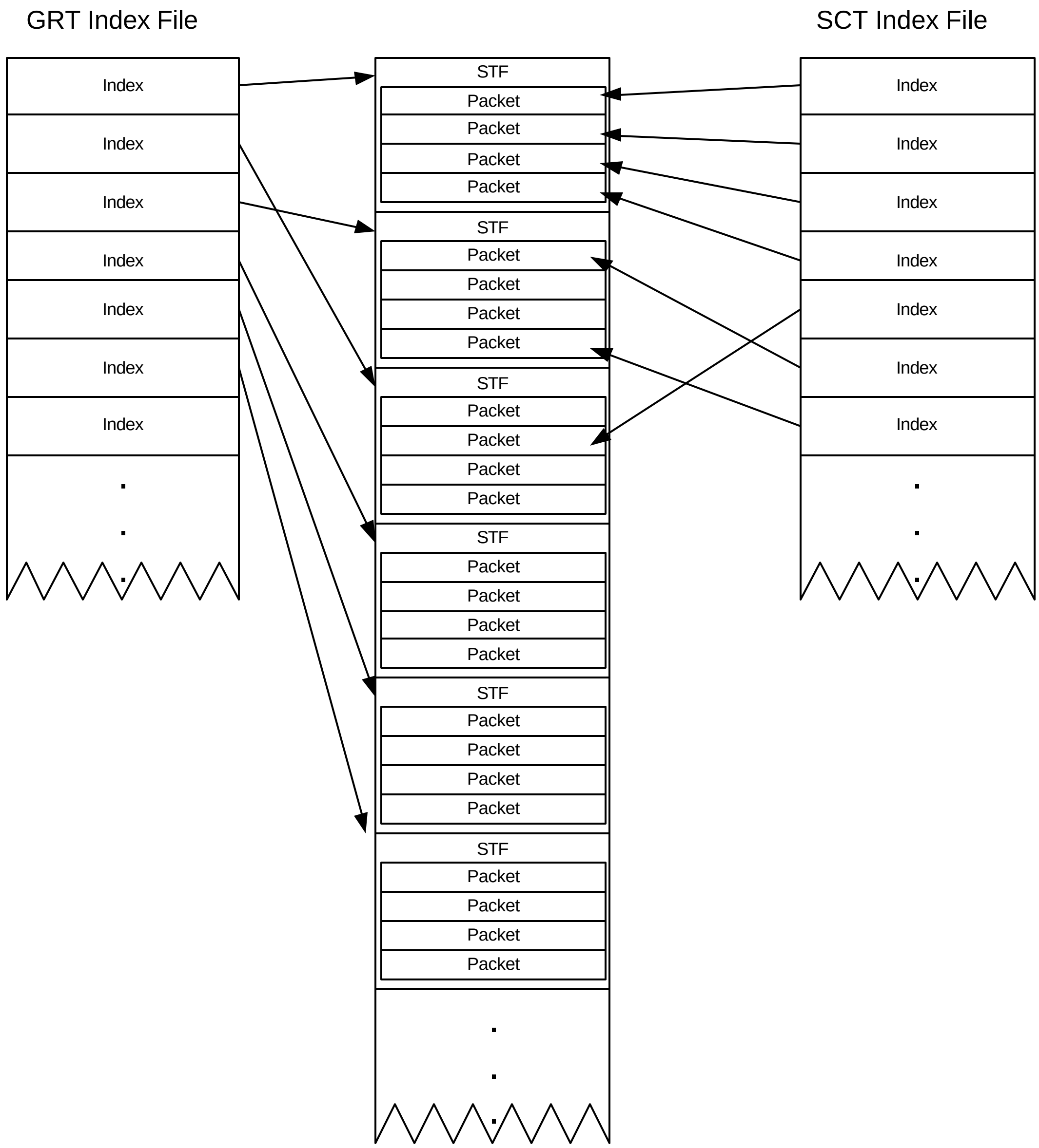

Figure 4. Telemetry Server Indexing Schema

Data Formats

The router supports several different output data format. The output most useful formats are:

1. STF (Supplemented Telemetry Frame) – normally a user of data would not use STFs. These are used for router to router communication. STF's consist of CCSDS telemetry frames with an associated Ground Receipt Header.

2. STP (Supplemented Telemetry Packet) this format contains all of the information of a STF but the data portion consists of only a CCSDS packet, rather than a CCSDS frame.

3. PTP (POC Telemetry Packet) the format consists of a GRH and a CCSDS packet.

A client can specify the desired output format using directives sent to the router when the connection is first made.

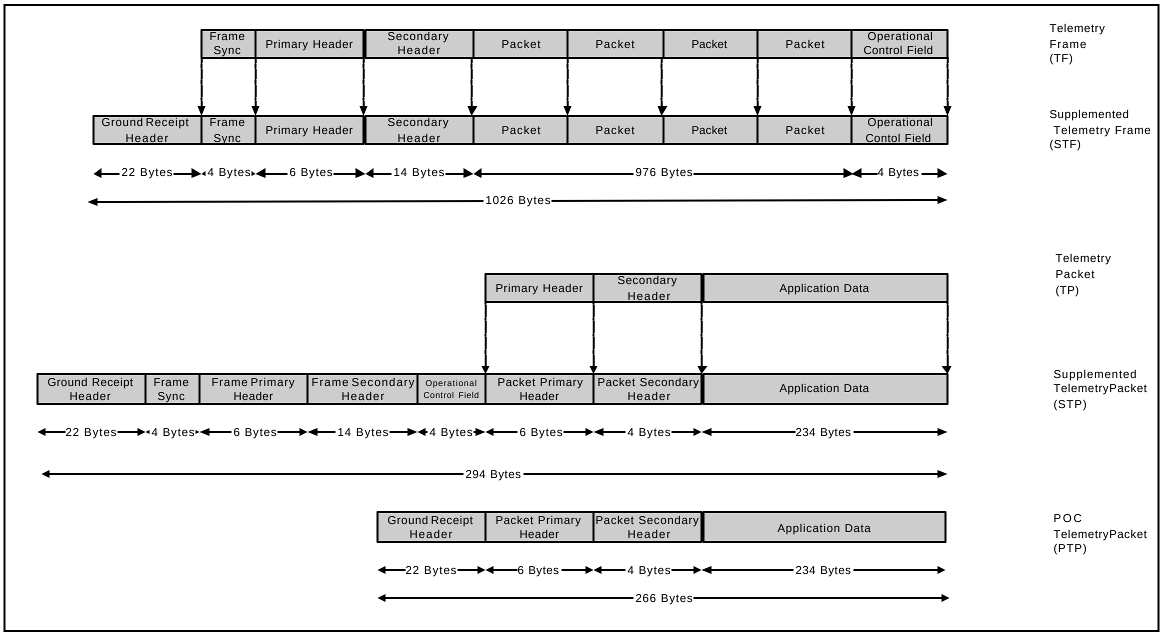

Figure 5. Telemetry Data Structures

Ground Telemetry

By collecting ground status information and packaging it within data structures of the same format as that used for spacecraft telemetry data, the TIMED ground system was able to use the Telemetry Server archive and playback services to handle ground information also. This information, while not strictly telemetry data, is referred to as ground Telemetry. Data was collected from systems such as the Blockhouse Control Unit, the Front End, the testbed systems, simulation systems, and ground software processes. Use of the system in this way enhanced the overall usefulness and cost effectiveness of the Telemetry Server system.

Ground Receipt Header

The purpose of the ground receipt header is to carry along information on the success and time of the spacecraft to earth antenna transmission process. Telemetry data, which carries along the ground receipt information is referred to as "supplemented" telemetry frame or packet data. The types of information found in the Ground receipt header are:

- Spacecraft Identifier
- Ground Receipt Time
- source ID (S/C, simulator, etc.)
- Path (Front End, DSMS antenna, etc.)
- Reed Solomon Decoding Success/Failure Information
- Other Types of Decoding Success/Failure Information
- Master Sequence Number
- Frame Sync Fields
- Archive Flag
- SSR Playback Error Information

The Ground Receipt headers for each mission have required some customization due to differences in preferred Ground Receipt Time formats, and they types of encoding processes used for RF transmission.

Re-Use

While the flexibility required to meet TIMED Integration & Test as well as flight operations requirements contributed to some of the complexity of the indexing schema and drove some performance requirements, this same flexibility makes the Telemetry Server a good candidate for re-use on future projects with different performance and different retrieval requirements. A version of the TIMED telemetry server is currently supporting CONTOUR development. There are plans to re-use it on at least two other NASA projects currently under development at the JHU Applied Physics Laboratory, the STEREO and MESSENGER programs. The TIMED satellite will operate in a Low Earth Orbit (LEO); the other three missions are deep space missions and must interface to the Deep Space Management System (DSMS). CONTOUR has a planned encounter with a comet. STEREO will consist of two satellites in orbit about the sun. MESSENGER will travel to Mercury.

A portion of the telemetry server already has a re-use history, a pre-cursor to the TIMED Telemetry Router was developed for the FUSE project during spacecraft Integration and Test. The FUSE Router ran on WinNT platform developed under Visual C++. The code base for this Router was the same as that used for TIMED except for the object-oriented software classes that defined the input data structure. The input data to the FUSE Router were Telemetry Frames formatted as CCSDC Standard Formatted Data Units (SFDU), which is the structure currently supported by the Jet Propulsion Laboratory for telemetry frame delivery from the Deep Space Management System (DSMS).

The TIMED Telemetry Router was ported to Sun Solaris and adapted to a Telemetry Frame formatted as a custom Supplemented Telemetry Frame (STF). Although the CONTOUR, STEREO and MESSENGER programs will be receiving SFDU structures from the Deep Space Network, the plan is not to convert back to using the SFDU structures, but instead to convert the SFDU structures to STF's prior to routing and archive. This approach was chosen to simplify re-use and because much of the overhead in the SFDU structure is not required by the end user. This approach also allows us to use the system during ground subsystem testing, and during I&T, when SFDU structures from the Deep Space Network are not yet available. During this time STF structures are built by a hardware interface known locally as the "Front End," which provides communication to the spacecraft command uplink and telemetry downlink systems.

The other issues requiring attention for efficient re-use of the Telemetry Server are the time formats used, different archive management requirements and regression testing improvement. The TIMED system used Global Positioning System (GPS) supplied time for SCT, and used UTC seconds since January 6, 1980 for GRT. The CONTOUR, STEREO and MESSENGER systems do not have GPS systems onboard their spacecraft. CONTOUR is using an MET (Mission Elapsed Time) counter for SCT, and is using UTC seconds since January 1, 1970 (Unix based epoch) for GRT. STEREO and MESSENGER system have not yet finalized their time-keeping formats. Therefore the indexing and playback structures and comparisons within the telemetry server code will require updating in these areas.

Another requirement that cannot be satisfied by the current TIMED telemetry server, is the requirement to selectively delete portions of the archive using SCT, GRT, or APID criteria. An efficient way to meet this requirement is currently under review. In addition, it should be noted that the Telemetry Server currently only handles telemetry data structures which have fixed size telemetry packets, and therefore an integral number of telemetry packets per telemetry frame. Each of the missions with the exception of MESSENGER are using fixed size telemetry packets. However, the fixed size used varies in each case.

Despite these differences in requirements, the follow-on missions to TIMED are indeed re-using not only the architectural design of the ground system, but also most of the actual code. This has allowed the developers of the CONTOUR ground system to devote additional time to reviewing and documenting the existing code. In the course of this review work they have added additional error checking to the code, especially in the areas dealing with socket connections and maintenance. They are also developing an enhanced test procedure for use in regression testing. This work is being shared across the other missions, including feedback to the TIMED project, thereby enhancing the overall quality of each team's product.

Lessons Learned

The TIMED Telemetry Server is a major component of the TIMED Mission Data Center (MDC), which was developed using an iterative development approach. There were three builds that supported three phases of the spacecraft and ground system development: the mini-MOC (miniature Mission Operations Center) build, the Integration & Test (I&T) build, and the flight operations build.

Requirements were imposed for each of these builds. It was discovered that the I&T phase placed the greatest stress on the system in terms of performance and data volume requirements. It was noted that trying to meet the needs of all three builds tends to increase the complexity of the system and can diminish its capability to satisfy some requirements. Therefore, a line should be drawn between satisfying a diverse set of needs and minimizing complexity.

Issues involving performance degradation and data retrieval revealed themselves when I&T and pre-launch operations testing imposed additional requirements on the telemetry system. These issues include

Performance degradation occurred when running multiple playback clients while concurrently ingesting a recorder dump file,

Performance degradation was caused by the receipt of input streams where the packet time stamping was not recent or was randomly ordered,

Data retrieval problems occurred for data created by the repetition of spacecraft mission simulation tests where the spacecraft time was repeatedly reset to the same time sequence.

Performance degradation occurred when running multiple playback clients while concurrently ingesting a recorder dump file. This effect was discovered when the I&T operational scenario varied from the post-launch scenario. The flight operations requirement was that recorder dump files (up to 300 MB each) shall be ingested into the system and available for playback within 10 minutes of receiving the file via FTP. At the time the requirements were established, the focus was directed towards the flight operations scenario. In this scenario, the spacecraft passes over the ground station, during which time a real-time telemetry stream is fed to the MDC which in turn feeds the real-time stream to various users and simultaneously archives the data. After the pass, the ground station FTP's a file containing telemetry stored on the spacecraft recorder and downlinked during the pass. The MDC ingests the dump file within the 10 minute requirement. Significant playback of the data would occur thereafter but not during the ingest operation. However, during flight operations testing, more playback clients than expected were running concurrently with the ingest operation. It was confirmed that with 10 playback clients running concurrently with the ingest operation the ingest process required 15 to 20 minutes, instead of the expected 10 minutes.

The second issue deals with the performance degradation which occurred in the presence of some input streams where the packets were not recent or were randomly ordered The expected scenario was that the telemetry would generally be in time increasing order. This expectation that packets would be time ordered (established originally for flight operations) did not hold true in I&T and flight operations testing. The result was that the ingestion and archival functions slowed considerably, because the wide range of times ingested caused an excessive number of indices files to be opened and closed. The problem was not a result of the spacecraft itself producing unordered data but that there were multiple inputs to the MDC including various test beds, each of which had their own clocks, which were not synchronized with all the other inputs. The telemetry server was not designed to accept such time varied inputs. A solution was found that involved adjusting the method by which the inputs for spacecraft time indexing would be accepted.

The third issue deals with the data retrieval problems found when repeating spacecraft mission simulation tests and while resetting the spacecraft time to the same time period as used in earlier tests. During I&T and flight operations, specific tests such as "a-day-in-the-life" of the spacecraft test are often repeated as system capabilities are enhanced and problems are fixed. In order to make the tests repeatable so that results can be compared from one test to the next, the tests need to be run with the same spacecraft environment (position over the earth, location of the sun, etc.). To produce the same environment, the environment simulators' clocks and the spacecraft clock (used to stamp the packets' spacecraft time) are set to the same simulated time period for each test. The ground station time (used to stamp the packets' ground receipt time), however, remains at current time. This makes the ground receipt times different for each test run which in turn allows playback of data specific to the test of interest (by playing back in ground receipt time order.) The problem arises when there is a desire to playback the telemetry from a particular test in spacecraft time order. Since there are multiple tests that have been run using the same spacecraft time period, it was not possible to retrieve the results of only one of those tests. The telemetry from all the tests has been "archived" with the same spacecraft time, so telemetry from all the tests was delivered when requesting packet data for that simulation time range. A possible fix to this problem is to allow filtering of packet databased on GRT. Currently filtering on GRT is only allowed when retrieving transfer frames, not packet data.

Conclusion

The TIMED Telemetry server provides solutions to the general requirements of telemetry routing, archival and retrieval. It provides support across the entire spacecraft development cycle from bench testing to Integration and Test to Flight Operations. Its flexibility and capabilities made it an excellent candidate for re-use on future missions. Differences in the requirements among the other missions have been identified and solutions planned. It appears that solutions will be developed that do not require any major redesigns of the system. This proved to be the case when unexpected requirements or issues surfaced during TIMED Integration and Test. Through efficient re-use of architecture and code across several missions, an overall improved product for an overall lower cost per program is achievable. This success is being achieved in an evolutionary method, where improvements and enhancements from one program are passed on from one mission to the next.

References:

1. Herrera, Leeha, "CONTOUR Telemetry Server Users Guide", Version 1.0, November 9, 2000

2. Consultative Committee for Space Data System, CCSDS 102.0-B-4: Packet Telemetry. Blue Book. Issue 4. November 1995. http://www.ccsds.org.

3. Consultative Committee for Space Data Systems CCSDS 620.0-B-2. Standard Formatted Data Units - Structure and Construction Rules with Technical Corrigendum 1, Blue Book. Issue 2. November 1996. http://www.ccsds.org.

4. Stevens, W. Richard, UNIX Network Programming, Networking APIs: Sockets and XTI, Volume 1, Prentice Hall, Upper Saddle River, New Jersey, 1998.