AIAA/AAS Astrodynamics Specialist Conference and Exhibit
18 - 21 August 2008, Honolulu, Hawaii

AIAA 2008-7371

The Use of GMSEC Architecture for Multi-Mission Flight Dynamics Orbit and Attitude Support

Jennifer Sager Cosgrove [1]
Honeywell Technology Solutions Inc., Greenbelt, MD 20771

Timothy Esposito [2]
Emergent Space Technologies, Inc. (EST), Greenbelt, MD 20771

and

Patrick Crouse [3]
NASA Goddard Spaceflight Center, Greenbelt, MD 20771

Abstract

This paper provides an overview of the use of the Goddard Space Flight Center (GSFC) Mission Services Evolution Center (GMSEC) ground system architecture for re-engineering Flight Dynamics (FD) orbit and attitude services for multi-satellite applications. The first GMSEC FD implementation was completed in early 2004 to provide orbit-related services for Transition Region and Coronal Explorer (TRACE), Solar, Anomalous, and Magnetospheric Particle Explorer (SAMPEX), Submillimeter Wave Astronomy Satellite (SWAS), and Wide-field Infrared Explorer (WIRE). Currently, there are plans to re-engineer attitude related functions for Advanced Composition Explorer (ACE), WIND, and Solar and Heliospheric Observatory (SOHO). This paper details these GMSEC FD implementations and the use of the GMSEC ground system architecture for re-engineering flight dynamics orbit and attitude services for supporting multiple missions in the Multi-Mission Operations Center (MMOC) at GSFC.

I. Introduction

This paper provides an overview of the use of the NASA Goddard Space Flight Center (GSFC) Mission Services Evolution Center (GMSEC) ground system architecture for re-engineering Flight Dynamics (FD) orbit and attitude services for multi-satellite applications. The first GMSEC FD implementation ever built was completed in early 2004. The goal of the first re-engineering phase was to provide orbit-related services for Transition Region and Coronal Explorer (TRACE), Solar, Anomalous, and Magnetospheric Particle Explorer (SAMPEX), Submillimeter Wave Astronomy Satellite (SWAS), and Wide-field Infrared Explorer (WIRE) in the Small Explorers (SMEX) Mission Operations Center (MOC) located at GSFC. The GMSEC re-engineered and consolidated MOC being built at GSFC is now referred to as the Multi-Mission Operations Center (MMOC). The MMOC consolidates operations for the SMEX MOC, the Advanced Composition Explorer (ACE) MOC, the WIND MOC, and the Solar and Heliospheric Observatory (SOHO) MOC into one control center. Currently, there are plans to re-engineer attitude

[1] Systems Engineer, NASA Goddard Space Flight Center, NASA GSFC Mailstop 452.4, AIAA Professional Member
[2] Systems Engineer, NASA Goddard Space Flight Center, NASA GSFC Mailstop 452.1, AIAA Professional Member
[3] Space Sciences Mission Operations Project Manager, NASA Goddard Space Flight Center, NASA GSFC Mailstop 444, AIAA Professional Member

Copyright © 2008 by the American Institute of Aeronautics and Astronautics, Inc.
The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes.
All other rights are reserved by the copyright owner.

related functions for ACE, WIND, and SOHO by adding those capabilities to the MMOC GMSEC bus. This paper will discuss the GMSEC architecture as it relates to these FD implementations in the MMOC.

The GMSEC architecture provides a scalable and evolving ground systems development and integration approach for supporting missions. Key features of the architecture include standardized message definitions and Application Programming Interfaces (APIs) for use in a middleware-based system. With these architectural elements in place, it enables all ground system components, and even the middleware itself, to be treated as a plug-and-play component. The GMSEC API supports the use of multiple Message-Oriented Middleware (MOM) systems, which grant missions the flexibility to select different vendors for their messaging infrastructure. This architecture facilitates integration of components, as well as improved capabilities for adding, deleting, or modifying functional components on the middleware to meet evolving mission requirements. The GMSEC architecture also enables technology infusion and supports varying mission operations concepts. With all ground system components treated as plug-and-play components, various Government-Off-The-Shelf (GOTS) and Commercial-Off-The-Shelf (COTS) software packages have been integrated onto the GMSEC middleware, including navigation and attitude packages associated with flight dynamics functionality.

The first phase of the MMOC FD implementation integrated GOTS and COTS software packages into the GMSEC system, including Analytical Graphics Incorporated's (AGI) Satellite Tool Kit (STK), which was operational in use within the SMEX MOC, and NASA GSFC's XFDS. Integration of these products into the SMEX MOC enabled automated orbit product generation, quality assurance (QA) and delivery to science, mission planning, antenna network, and scheduling customers. In the second phase of the MMOC GMSEC FD implementation, developers plan to utilize current GMSEC interfaces with Matlab and the ITOS telemetry system to implement the attitude operations, including attitude determination, prediction, calibration, sensor monitoring, and slew planning. The GSFC-provided Multi-Mission Three-Axis Stabilized Spacecraft (MTASS) and the Multi-Mission Spin Axis Stabilized (MSASS) systems will retrieve telemetry directly from the GMSEC middleware, perform attitude functions, and interact with other elements of the ground system through the GMSEC interface to support operations and automation.

When complete, critical mission operations functions will be implemented and automated for multiple missions via a single installation of GOTS and COTS software. This paper details these GMSEC FD implementations and the use of the GMSEC ground system architecture for re-engineering flight dynamics orbit and attitude services for supporting multiple missions in the MMOC at GSFC.

This paper is organized into five sections. This section is the introduction that provides a general overview and structure for the paper. Section II presents an overview of GMSEC, including a discussion detailing GMSEC features, components, and architecture. Section III details the Multi-Mission GMSEC FD Orbit Re-Engineering effort, which summarizes the first GMSEC FD implementation completed in early 2004 to provide orbit-related services for TRACE, SAMPEX, SWAS, and WIRE. Section IV describes the Multi-Mission GMSEC FD Attitude Re-Engineering effort, which discusses the implementation plans to re-engineer attitude related functions for ACE, WIND, and SOHO using the GMSEC architecture. The final section summarizes the GMSEC FD implementations and the use of the GMSEC ground system architecture for re-engineering flight dynamics orbit and attitude services for supporting multiple missions in the MMOC at GSFC.

II. GMSEC Overview

A. GMSEC Architecture

There are three primary artifacts of the GMSEC architecture: the GMSEC Application Programming Interface (API), the GMSEC Message Specification, and the ground system components themselves. The GMSEC architecture was first implemented in the MMOC in early 2004. The architecture was developed by a team of engineers who well understood how difficult and costly it was to integrate existing ground system components into new missions. Their experience was that even though many of the ground system components to be used were already developed and operational in other implementations, many custom interfaces had to be developed because that exact combination of ground system components in a MOC had not been used before. To greatly simplify the integration of ground system components, the GMSEC team chose to implement a message-oriented middleware (MOM) system to interface with all ground system components. Since all components will utilize the same protocols to communicate, the interface issues are solved.

A significant cost achievement of the GMSEC team was the avoidance of vendor lock-in. Some middleware systems can be very costly and the mission requirements and subsequent budgets range drastically, from small missions like SMEX to very large projects with considerably more resources like James Webb Space Telescope (JWST). With potentially long mission life cycles, product obsolescence and support needs to be considered. To

solve some of these issues, the GMSEC team developed their own API that wraps around third-party middleware systems. By doing this, component developers can implement the GMSEC API and be compatible with multiple middleware systems. It also allows missions to swap middleware systems during the life of a mission as necessary, and it facilitates access to a representative middleware for GMSEC developers. Using MOM and the GMSEC API solves the common communication protocol problem with a messaging infrastructure, but the GMSEC Message Specification solves the content of messages problem. Messages in their rudimentary form are composed of subject and field-value pairs. The message specification defines various types of messages, subject naming conventions, field types, and the field names for all the various types of messages. The types of messages include logs, telemetry requests and responses, directive requests and responses, product requests and responses, heartbeats, and various others. In an ideal implementation, a component performing a specific task can be swapped with another component performing the same task without any interruption in operations. In reality, just like the various implementations of the Structure Query Language (SQL) for relational databases, components can each have their small enhancements to the specification or slight differences from the actual specifications that require a small level of effort to exchange a ground system component.

The GMSEC architecture provides a scalable, extensible ground and flight system approach for future missions (see Figure 1). The architecture enables quick and easy integration of functional components that are selected to meet the unique needs of a particular mission. The architecture enables the addition, deletion, and exchange of components to meet the changing requirements of missions as they progress through their lifecycles and provides a rapid, flexible, and cost-effective means to meet a wide variety of evolving mission concepts and challenges.

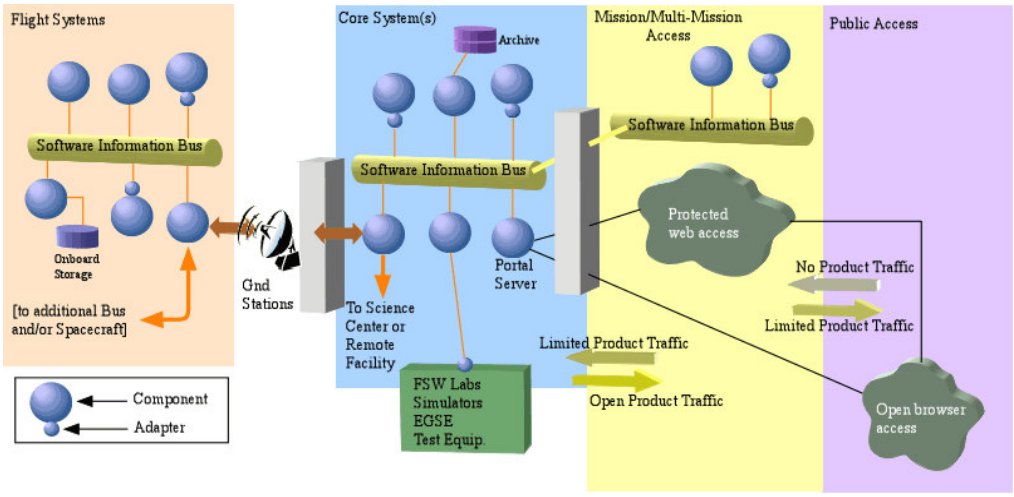

Figure 1. GMSEC Reference Architecture

Additional capabilities can be added to the system without impacting existing system/design, including security (remote and shared access), shared operations with another site (university, commercial provider), mission critical system risk improvement, simplified collaboration across missions with interface commonality, and anticipation of movement of some ground functions onboard.

B. GMSEC Components

Standardization of components allows for a high level of reuse as multiple missions utilize the same tools and creates an environment where new tools can be added over time, obsolete components can be replaced, and new missions are not bound by component selection decisions made many years earlier. Technology advancement can be made incrementally and the impacts of product vendor failures or new requirements are minimized.

The Software Information Bus provides the communications pipe used to carry information between GMSEC components (refer to Figure 2). GMSEC can swap different middleware products in and out, providing additional flexibility. Legacy components interface to the Software Information Bus using an Adapter. The Adapter is custom

software that translates Legacy Component messages and/or message protocols to those that are GMSEC-compliant. Additionally, legacy components may, in combination with GMSEC, continue to use legacy interconnections.

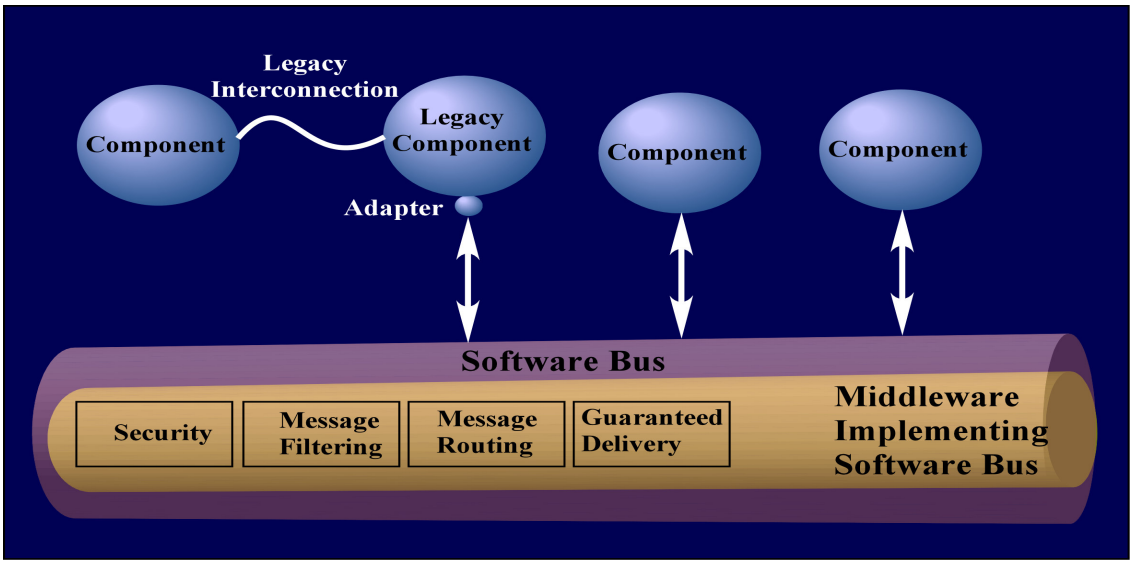

Figure 2. Components Interface to GMSEC Software Bus

III. Multi-Mission GMSEC FD Orbit Re-engineering

A. Summary of MMOC GMSEC Orbit Re-engineering

The first phase of the MMOC FD implementation integrated Government Off-the-Shelf (GOTS) and Commercial Off-the-Shelf (COTS) software, including Analytical Graphics Incorporated's (AGI) Satellite Tool Kit (STK) and National Aeronautics and Space Administration (NASA) Goddard Space Flight Center's (GSFC) XFDS, with the GMSEC architecture and components. By using these products, the flight dynamics system that produces orbit product generation, quality assurance (QA), and delivery to both internal and external customers, including science, mission planning, antenna network, and scheduling customers, can be more fully automated and connected to the larger MMOC system.

SMEX orbital products generated in the MMOC include mission planning, acquisition data, scheduling, and science products. Orbital products for the SMEX missions are generated in two phases. The Flight Dynamics Facility (FDF) is responsible for generating, validating, and delivering the ephemerides for all SMEX missions, including SAMPEX, SWAS, TRACE, and WIRE (SWAS and WIRE have recently been decommissioned). The ephemeris file for each mission is a prerequisite for generating all other orbital products. STK is the software responsible for reading the ephemeris and generating raw reports and data for all SMEX orbit products generated in the MMOC. XFDS is the flight dynamics product in the MMOC that is then responsible for orchestrating all other steps for generating the orbital products. Figure 3 shows a screenshot of the XFDS Graphical User Interface (GUI), and Table 1 lists all the components involved in the process of generating these products.

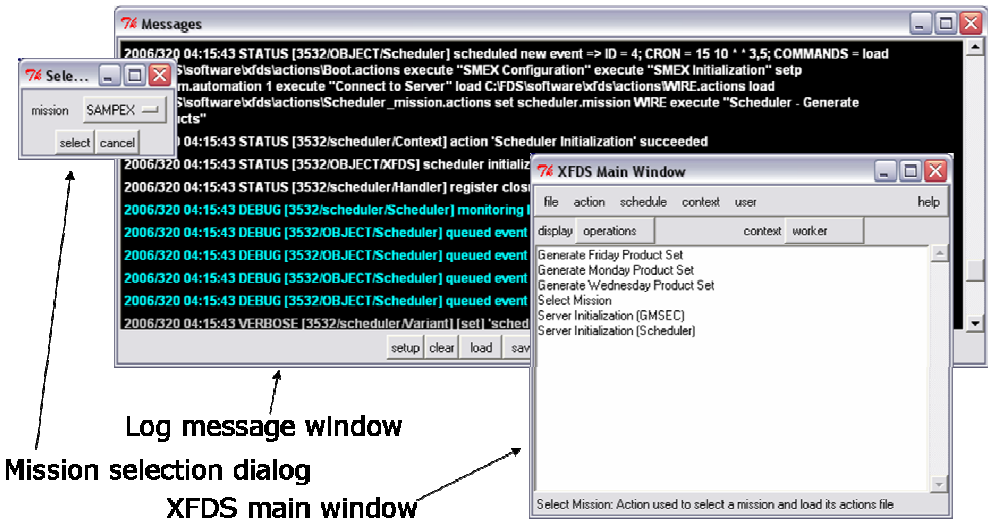

Figure 3. XFDS Screenshot

XFDS was used in the pre-GMSEC system to generate products manually. Operators were required to open the XFDS GUI for all missions multiple times a week (primarily Monday, Wednesday, and Friday). XFDS starts the STK application and sends STK commands over its Transmission Control Protocol/Internetwork Protocol (TCP/IP) socket interface to load the STK scenarios, configure the system, and generate the raw orbital products. Even before generating the products, the operators would manually check to see if the ephemeris file for that mission was delivered yet from the FDF.

In the GMSEC orbit re-engineering effort, multiple enhancements to XFDS were made to support GMSEC integration and automation. The major features added to the system, in addition to the GMSEC interface, were a scheduler, a Quality Assurance (QA) action type, the ability to run XFDS as a windows service, and failover handling.

Table 1. MMOC Orbit Components Selected

Component	Purpose
GMSEC Architecture	Consistent with the MMOC infrastructure. Provides a common messaging framework and application interface.
XFDS	GOTS Flight dynamics system used for orchestrating tools for generating raw products, reformatting, performing quality assurance checks, archiving, and delivery.
STK	Software COTS product by AGI used for modeling and generating the raw orbital products.
FDF	Flight Dynamics Facility is responsible for maintaining environment files like the UT1-UTC time offsets, star catalogs, and solar-lunar-planetary (SLP) tables, as well as generating and delivering the ephemeris files for all SMEX missions.
ANSR	Notification of system status, product completion, no delivery of ephemerides, failed product generations
Nagios	Component and system monitoring; performs recoveries.

The GMSEC interface was added to allow native integration with XFDS. Many GMSEC components that already existed only allowed access to a limited subset of features for which adapters were specifically built. XFDS implements an approach that allows access to all of its capabilities from the GMSEC interface. It allows incoming GMSEC directives to execute any of the actions loaded in the running XFDS instance. An action is the term used in XFDS to perform any of its functions. Every mission has its own set of actions stored in an actions file, which is

fully configurable. The XFDS architecture previously had features to route log messages internally since it was a multi-processed framework. The GMSEC interface leveraged this by also routing all internal log messages as log messages on the GMSEC middleware. This allows any other component to also monitor the XFDS activities. Having the GMSEC interface also allows convenient operator utilities to be easily added. Figure 4 shows a screenshot of the SMEX Request Tool, which allows an operator to easily select a few options, connect to the GMSEC middleware, and send a directive or product request to the operational XFDS system, which will process and respond to the request.

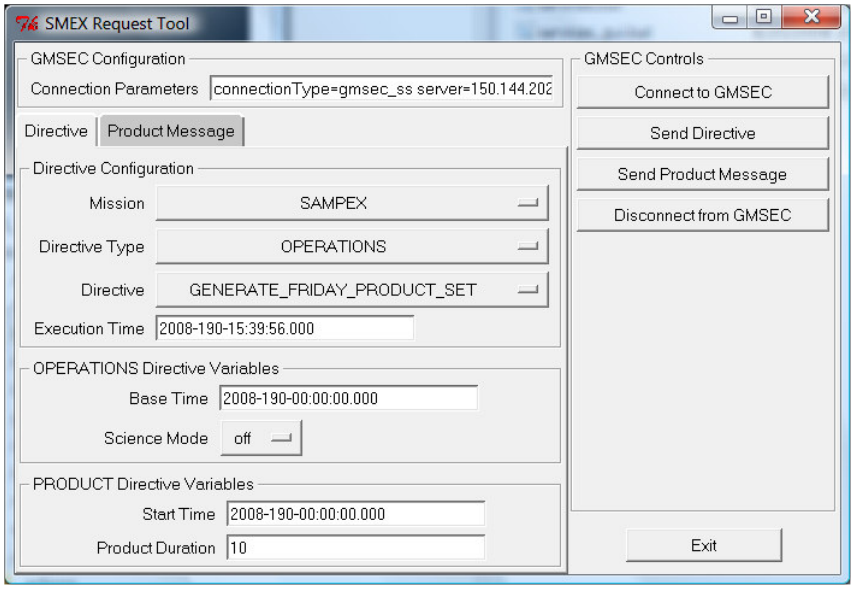

Figure 4. SMEX XFDS Request Tool

Through the GMSEC interface, XFDS also interfaces with ANSR and Nagios. For all critical errors where operators must be notified, XFDS will send a page directive to ANSR, which in turn will send an email or page through a firewall and manage a call-tree of operators. Nagios is the system that monitors XFDS and makes sure that all expected processes are running. Nagios is an open source network monitoring tool for which adapters and custom check scripts were built to enable GMSEC integration. If Nagios detects that any of the expected applications are not running, it will make several recovery attempts. If those recovery attempts fail, it will then send a notification via ANSR to get an operator involved.

The scheduler was a major enhancement to enable automation. As mentioned previously, operators had to open and drive a GUI to generate all products manually. With the development of the scheduler that modeled an automated time cron-based system, XFDS could be run as a server process and generate the products automatically. Of course, with automation extra care needs to be taken to perform better error handling. Checking for the delivery of ephemeris files was a major part of this. The system would expect the ephemeris to be delivered around 10:00 a.m. local time, but would reschedule checks every 30 minutes until it arrived. At 1 p.m., the system would start sending notifications both internal on the bus and via an external page using ANSR that the ephemeris file was not delivered yet. If the ephemeris was not delivered by 3:00 p.m., the system would cease checking and send an ANSR notification.

Adding the QA action type was another important milestone in validating the automation system. When operators generated the products manually, they could inspect the products and have a good idea of whether they were correct and valid. With automation, having the system deliver bad products could be worse than not delivering products at all. Bad products loaded onboard the spacecraft could cause a safe-hold mode in a worse-case scenario. The QA action type was added to XFDS to support this function. Before delivering products, QA action instances were configured to perform product quality assurance checks such as expected number of lines in a file, line endings, headers, tolerances for times and numbers, and value differences between orbits.

Since XFDS was running as a server process in this new GMSEC environment, it needed to be set up so it would start up automatically and not require a user to log in and start up the system manually. This would be an issue for

system reboots, including power outages when operators or system administrators are not around. To address this issue, XFDS was setup as a Windows service so it would run in the background, start up automatically at system boot, and not interfere with users logged into the system.

For an extra level of safety, better failover handling was enabled by leveraging the GMSEC architecture. Failover handling was set up to handle as many backup nodes as the mission would want. Many typical components will only allow one backup node because coordination of activities can become complex. XFDS added a few fields to the standard GMSEC heartbeat message to track the primary component performing activities and assign priorities for failover handling. Additionally, when products are generated, the node generating the products knows to also deliver the products to the other backup nodes. Using this scheme insures the highest level of reliability in an automated system.

B. MMOC GMSEC FD Orbit Architecture

The components used in the MMOC GMSEC FD orbit system were described in the previous section. Figure 5 shows how the components are connected in the GMSEC architecture, in which FDF represents the SMEX FDF ephemeris product. The FDF currently delivers all products to the MMOC via the FTP protocol. However, a method is being explored by which the FDF would connect to the MMOC GMSEC middleware system directly or via a bridge or gateway to send a product delivery notification to the MMOC via a GMSEC message. XFDS currently polls for the delivery of products delivered from the FDF, but because the FDF would send a product delivery notification, XFDS would no longer have to poll for delivered products. XFDS would simply have an event-based trigger to the FDF delivery notification, which means that it can generate its products immediately after the FDF ephemeris is delivered to the MMOC.

In the daily product generation process, XFDS kicks off activities from its internal scheduler. All activity is published to the GMSEC middleware with log messages so operators can monitor progress of the system. In the process of generating products, XFDS runs STK in the background. Any notifications are published as GMSEC directive messages, which are picked up by the ANSR paging and notification system. Independently, Nagios is monitoring the status of all components in the MMOC.

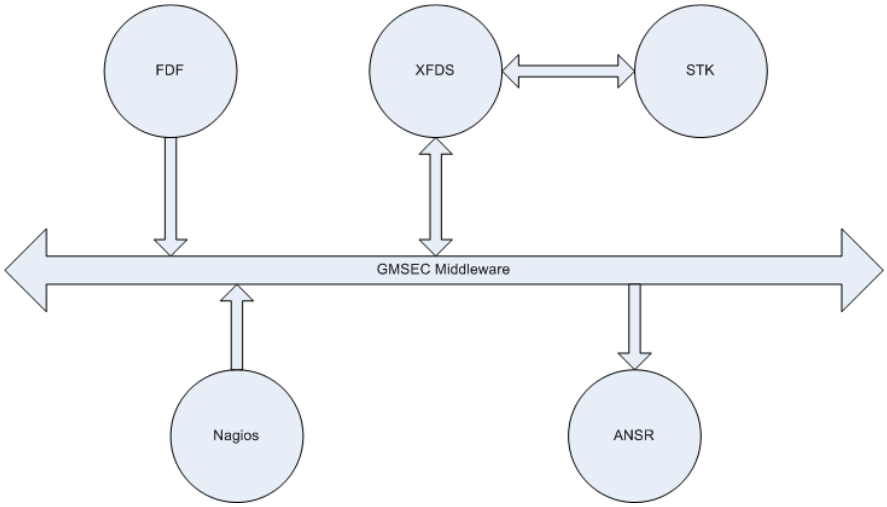

Figure 5. MMOC FD Orbit Components

Figure 6 shows the sequence of events in the daily orbital products generation process. The routine process is quite simple for the nominal cases. If product generation failed for any reason, an operator would have to interact with the system. There are multiple options for regenerating the products including: opening the XFDS GUI and generating products manually as was done in the past or using the more simplified interface of the SMEX Request Tool to send a GMSEC directive or product request message to the XFDS services. If parameters had to be tweaked, an operator may be forced to generate the products from the XFDS GUI because of limited configuration abilities from the SMEX Request Tool.

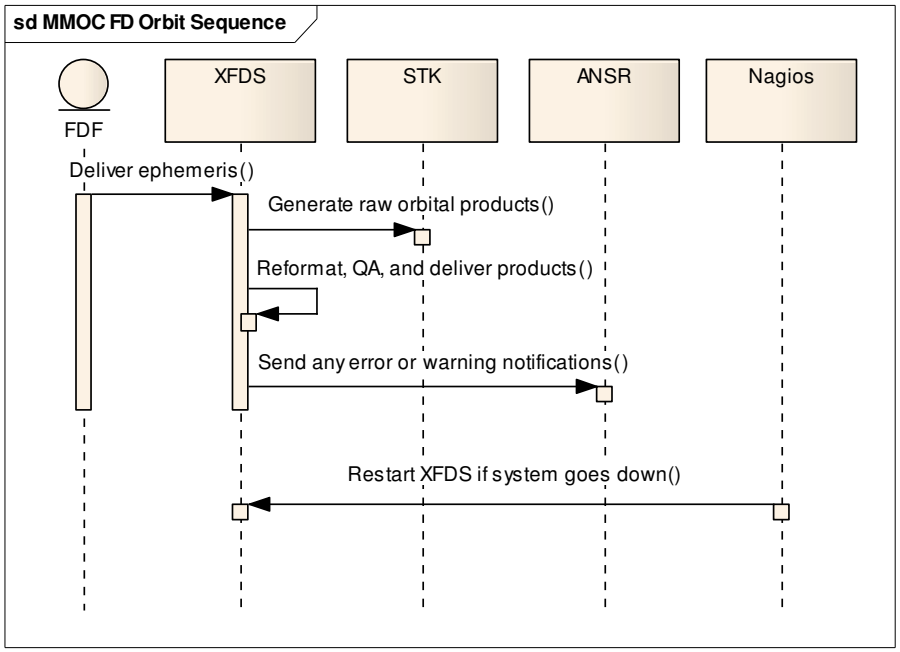

Figure 6. MMOC FD Orbit Sequence Diagram

IV. Multi-Mission GMSEC FD Attitude Re-engineering

A. Summary of MMOC GMSEC Attitude Re-engineering

In the next MMOC GMSEC FD implementation, engineers plan to integrate the MMOC GMSEC Attitude system with the MMOC GMSEC Orbit system and utilize current GMSEC interfaces with Matlab and the Integrated Test and Operations System (ITOS) telemetry and command (T&C) system to implement the attitude functions for WIND, ACE, and SOHO in the MMOC. The Matlab-based Multi-Mission Three-Axis Stabilized Spacecraft (MTASS), the Multi-Mission Spin Axis Stabilized Spacecraft (MSASS), and the Real-Time Attitude Determination (RTADS) systems will retrieve telemetry directly from the GMSEC middleware, perform attitude functions, and interact with other elements of the ground system through the GMSEC interface to support operations and automation both in real-time and playback modes. The MMOC Attitude Re-Engineering Implementation Planning Phase was completed in 2007, and the software development phase recently started in 2008. The implementation plans for the re-engineered attitude system are detailed in this section.

Many of the GMSEC components planned for deployment in the MMOC FD attitude system were described in the previous section. Table 2 lists all of the MMOC attitude components that have been selected for performing attitude operations using the GMSEC architecture, including telemetry processing, attitude determination and prediction, guide star and high gain antenna monitoring and planning, and various utilities depending on mission requirements. The Matlab MSASS, MTASS, and RTADS systems perform the various attitude related functions.

The attitude re-engineering development recently started in spring 2008 based on the MMOC Attitude Re-Engineering Implementation Plan. Since then, the development of the GMSEC-compliant version of the MTASS/MSASS Telemetry Processor (TP) has been completed, enabling a key GMSEC component in the re-engineered attitude system. This was one of the initial steps in the Attitude Re-Engineering Implementation plan to enable the MTASS/MSASS systems to receive spacecraft telemetry data over the GMSEC middleware in both playback and real-time scenarios from ITOS. For offline modes, sequential print files from ITOS can still be used to work with a non real-time interface.

There are important prerequisites for generating attitude products, such as a mission ephemeris file, various environmental files, time offsets, and solar-lunar-planetary (SLP) ephemeris data, that can be leveraged with the MMOC Orbit System via the GMSEC middleware.

Table 2. MMOC Attitude Components Selected

Component	Purpose
GMSEC Architecture	Consistent with the MMOC infrastructure. Provides a common messaging framework and application interface.
ANSR	Notification of system status, product completion, problems, possible product delivery (e.g., maneuver plans).
ADS Executive	Coordinates the product development process. Drives MTASS/MSASS and is the primary interface to the utilities. May be XFDS.
CAT	Reacts to messages for taking actions. Used to coordinate automation and notification activities.
FORMATS	Validation, reformatting, and delivery of products.
ITOS	Telemetry and command system selected by the MMOC. Provides attitude telemetry information.
MTASS/MSASS	Matlab-based attitude product generation systems. Includes TP –Telemetry Processor; MSASS – WIND; MTASS – ACE, SOHO.
RTADS	Matlab-based real-time attitude determination system.
Nagios	Component and system monitoring; performs recoveries.
Scenario Scheduler	Schedules and invokes attitude product generation.
Utilities	Same utilities for product generation and validation.
GMAN	Maneuver planning tool, ACE.
XFDS	GOTS Flight dynamics system used for orchestrating tools for generating raw products, reformatting, performing quality assurance checks, archiving, and delivery. Provides orbit ephemeris data to the attitude system.

A three-phase attitude re-engineering implementation plan has been proposed. In Phase 1 of the MMOC Attitude Re-Engineering, the focus will be on building a multi-mission attitude system to support Wind and ACE attitude operations with an emphasis on providing the existing functionality for both missions, integrating to the GMSEC architecture to simplify data accessibility, retire obsolete hardware and unsupported operating systems, and define standards to ease the incorporation of additional missions.

In Phase 2, the emphasis will be on integrating SOHO into the attitude system, adding mission specific utilities and implementing automation. During this phase, the ADS Executive GMSEC interface evaluation and implementation will begin. The plan is to use XFDS to provide the ADS Executive role and manage traffic across the GMSEC bus for the attitude system, similar to the orbit system described in the previous section. The ADS Executive will initiate attitude related activities from its internal scheduler. All activity will be published to the GMSEC middleware with log messages so operators can monitor progress of the system. In the process of generating routine attitude products, the ADS Executive, XFDS, will run Matlab in the background. Any notifications will be published as GMSEC directive messages, which will be picked up by the ANSR paging and notification system. Independently, Nagios will be monitoring the status of all components in the MMOC. Attitude automation in Phase 2 will consist of routine attitude product generation, including definitive and predictive attitude solutions and plots, sensor predictions, slew related reports (maneuver command sheets, momentum management), and guide star reports, as well as attitude product quality assurance.

The final phase of the MMOC GMSEC attitude re-engineering will be a fully automated GMSEC attitude event-based system in which defined events such as the start of a ground station pass would trigger events in the attitude system. The focus will be on higher level automation and full GMSEC implementation with completely automated system scheduling and monitoring. Figure 7 represents the automation strategy for the MMOC GMSEC Attitude system. Automated functions would include:

- Event triggers in the MMOC ground system to start attitude processing.
- Nominal Attitude Determination and Maneuver Support:
  - Retrieving telemetry from ITOS via GMSEC bus
  - Running TP
  - Retrieving and verifying Guide Star data
  - Running ADS (Attitude Determination System)

    - Comparing ADS to Guide Star data
    - Running utilities/scripts
  - Validation and QA of ADS solution and utility products
  - Operator notification of success/failures
  - Delivery of products
  - Monitoring and processing of maneuvers
- Retrieval of Ephemeris and Environmental Files from MMOC Orbit System.

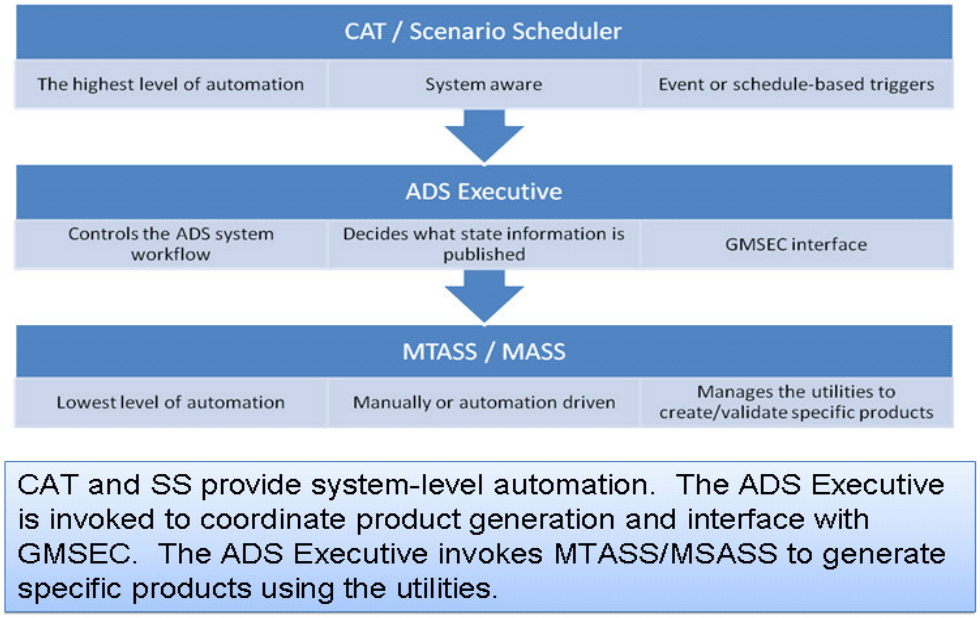

Figure 7. MMOC Attitude Automation Strategy

The ITOS T&C system will play a critical role by providing the attitude system spacecraft telemetry data via the GMSEC middleware in both playback and real-time modes. ITOS will also manage archives of all the raw spacecraft telemetry data. The re-engineered attitude system will use ITOS GMSEC publishing messages to the middleware in its event based system, as well as use the GMSEC interface to request spacecraft telemetry over the bus for both MTASS and MSASS. ITOS can provide the actual mnemonic values over the GMSEC bus or it can provide a sequential print, which is simply a file of pre-defined spacecraft telemetry mnemonics, over the bus. The initial implementation will utilize the sequential print product. Additional data transport options are discussed in the next section.

B. MMOC GMSEC FD Attitude Architecture

The components used in the MMOC FD attitude system were described in the previous section. In the current attitude systems, FORMATS manages and handles all attitude product validation, product reformatting and product delivery; for the final re-engineered attitude system, a GMSEC compliant system which is TBD will manage attitude product validation, formatting and product delivery. Figures 8 and 9 show how the components are connected in the MMOC GMSEC architecture and how the attitude system is integrated with the MMOC orbit products. Figure 8 represents how the components will be connected in the GMSEC architecture in an intermediate phase of implementation. Figure 9 represents how the components will be connected in the final GMSEC implementation.

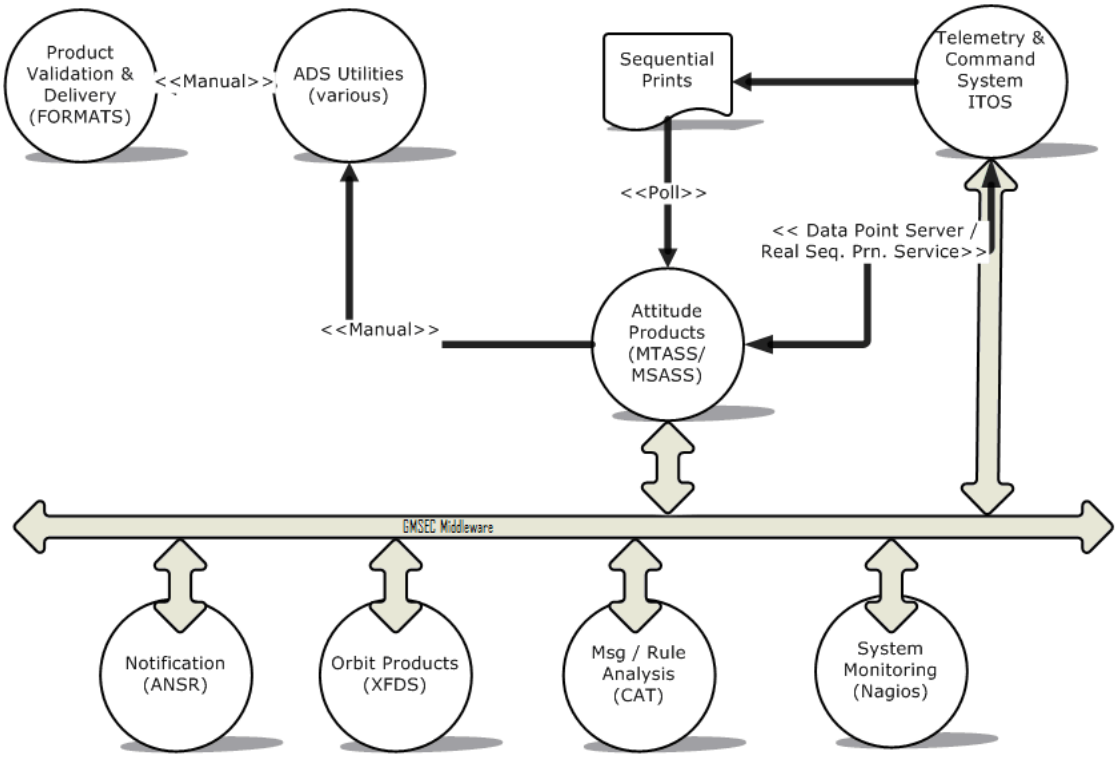

Figure 8. Intermediate MMOC Attitude System Components

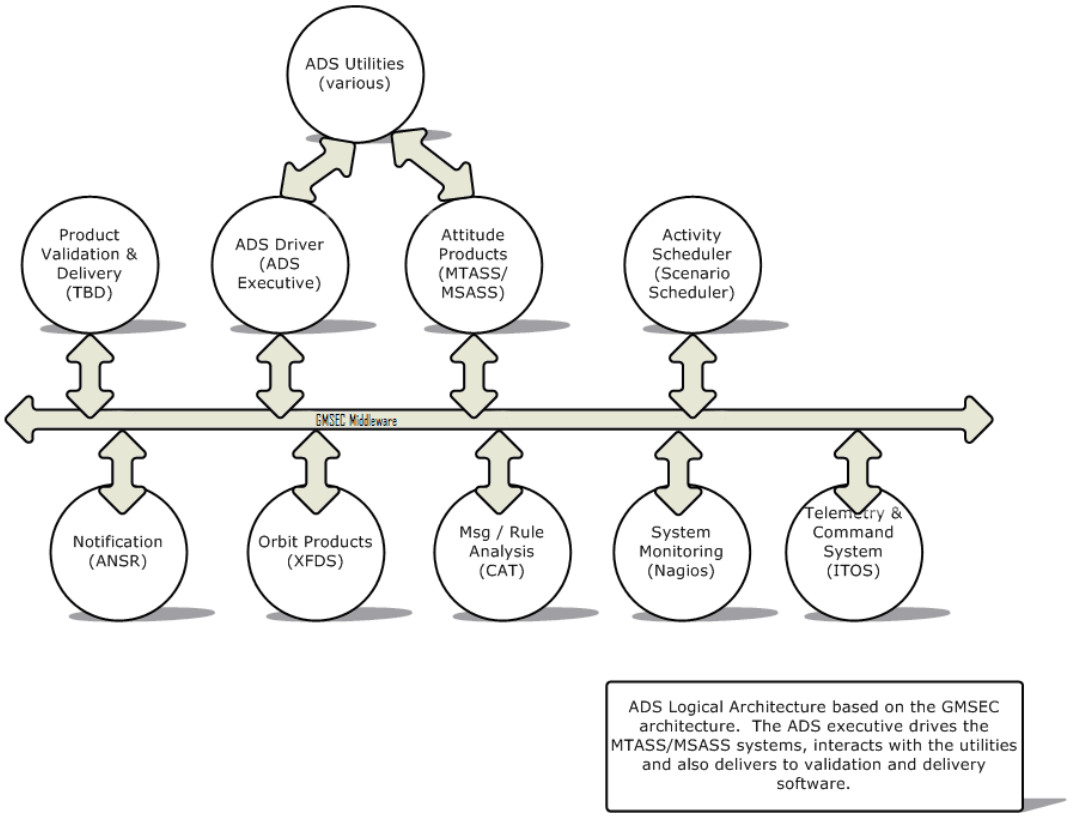

Figure 9. Final MMOC Attitude System Components

To show the sequence of events and interplay between GMSEC, COTS, and GOTS components in both playback and real-time attitude processing, Figure 10 shows the playback post-pass attitude processing sequence diagram using MTASS or MSASS, and Figure 11 shows the real-time attitude processing sequence diagram using RTADS. If product generation failed for any reason, an operator would have to interact with the system. In the post-pass sequence, either a user or an event can trigger the MTASS/MSASS attitude system to begin processing. MTASS/MSASS requests telemetry data from ITOS via a GMSEC request, and ITOS provides the data to the attitude system via the GMSEC middleware. In the real-time sequence, a user starts RTADS from the MTASS/MSASS GUI; then RTADS requests a real-time telemetry stream from ITOS via a GMSEC request; and RTADS receives telemetry in real-time over the GMSEC bus. Telemetry is buffered until a user specified span of data (from 1 to 32 seconds) is available for attitude determination. RTADS does real-time attitude data processing and updates real-time displays. RTADS continues to read telemetry from the GMSEC bus, computing real-time attitude, and updating displays until the user terminates the session from the MSASS/MTASS GUI.

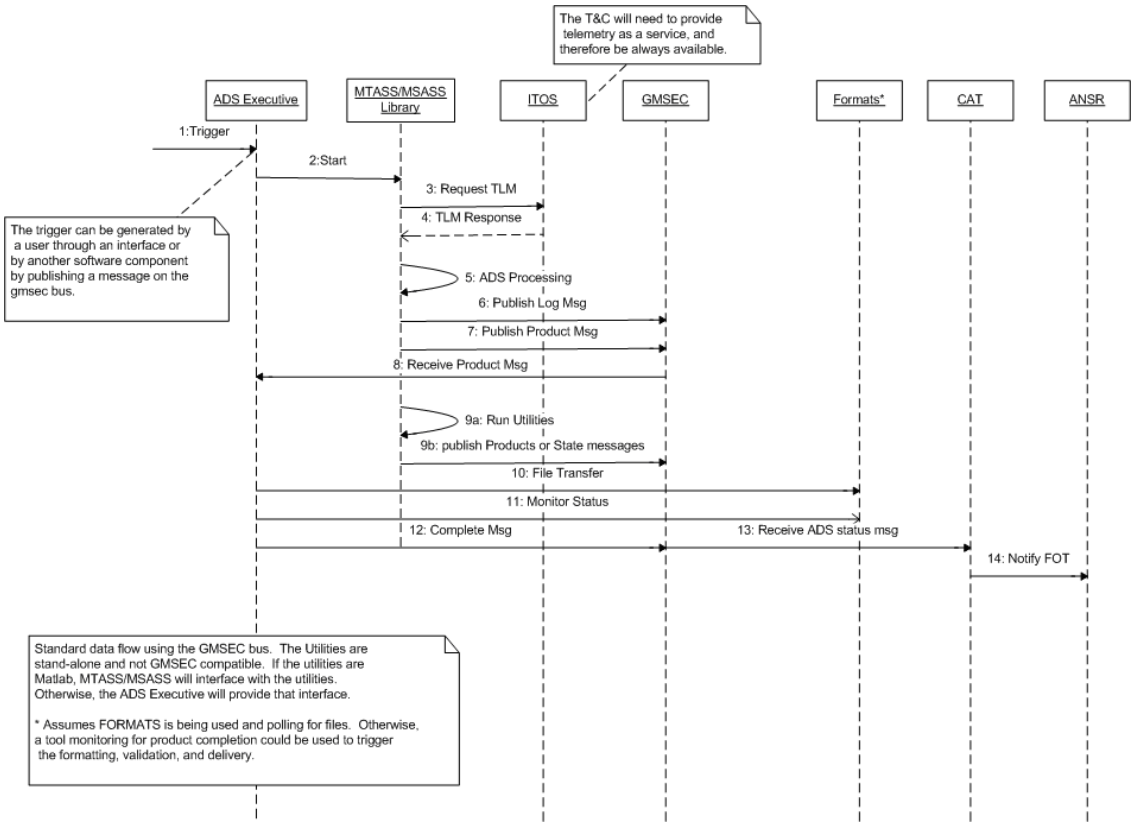

Figure 10. MMOC Post-Pass Attitude Processing Sequence Diagram

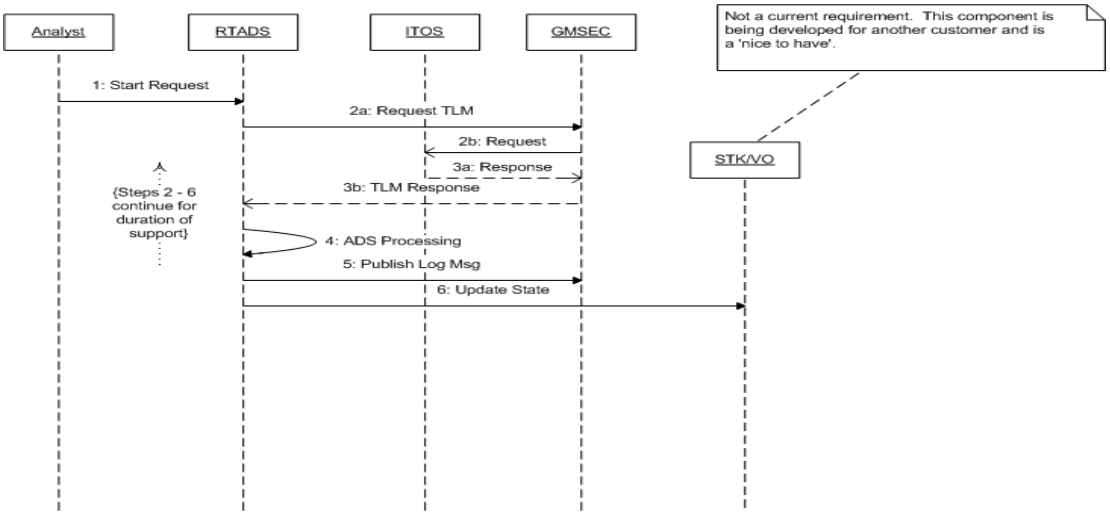

Figure 11. MMOC Real-Time Attitude Processing Sequence Diagram

While the MMOC Attitude System is planned to be a single installation of GMSEC, COTS, and GOTS components and flight dynamics tools supporting the ACE, WIND, and SOHO missions, the architecture is capable of supporting many more missions. The MMOC GMSEC Attitude Re-engineering effort is focused on building a system that is modular and standardized to enable other missions to more easily and efficiently be added to the attitude system in the future.

V. Summary

This paper provides a summary of the use of the GMSEC ground system architecture for re-engineering FD orbit and attitude operations and services for multi-satellite applications in the MMOC at GSFC. The MMOC consolidates operations for the SMEX MOC, the ACE MOC, the WIND MOC, and the SOHO MOC into one control center. The first GMSEC FD implementation ever built was completed in early 2004 for performing automated orbit operations for TRACE, SAMPEX, SWAS, and WIRE. In the next phase of the implementation, engineers plan to integrate the MMOC GMSEC Attitude system with the MMOC GMSEC Orbit system and utilize current GMSEC interfaces with Matlab and the ITOS T&C system to support operations and automation both in real-time and playback modes for WIND, ACE, and SOHO, implementing a critical mission operations function, flight dynamics, for multiple missions via a single installation of GOTS and COTS software. The consolidation of independent mission systems within one integrated MMOC multi-mission system using the GMSEC approach will reduce both direct operations costs and overall system sustaining costs.