The Cassini-Huygens Sequence Development Process

Jennifer H. Long* and William M. Heventhal, III†
Jet Propulsion Laboratory/California Institute of Technology, Pasadena, CA, 91109

Shahram Javidnia‡
Jet Propulsion Laboratory/California Institute of Technology, Pasadena, CA, 91109

The Cassini spacecraft consists of 12 instruments: 4 Optical Remote Sensing Instruments (ORS), 6 In-situ observation instruments to study Magnetospheric and Plasma Science (MAPS), one Radar instrument, and one Radio Science (RSS) instrument. When this complex mission was initially architected, much of the early emphasis was placed on the spacecraft function and design, rather than operations. The spacecraft and mission design posed significant challenges to the science and sequence development process for the four-year tour of the Saturnian system.

The science planning and sequence development process produces a comprehensive set of commands for all science and engineering activities for an approximate 40 day time period. The end-to-end sequence design process consists of five phases:

1) Integration of the Science Operations Plan (SOP), a high-level plan of science and engineering activities, detailing their timing, power, thermal, data volume, and pointing profiles

2) SOP Implementation, in which resource conflicts are resolved and activities constraint checked

3) Aftermarket and SOP Update, in which the SOP is updated while in tour using the latest information on the navigation ephemeris, and the spacecraft's and instruments' performance

4) Science and Sequence Update Process, which results in an integrated, validated, uplinkable, and flyable distributed sequence

* Sequence Virtual Team Lead, Cassini Science and Uplink Office, 4800 Oak Grove Dr. MS 230-101, Pasadena, CA 91109, MB.
† Uplink Operations Manager, Cassini Science and Uplink Office, 4800 Oak Grove Dr. MS 230-101, Pasadena, CA 91109.
‡ Sequence Virtual Team Lead, Cassini Science and Uplink Office, 4800 Oak Grove Dr, MS 230-101, Pasadena, CA 91109.

5) Execution, which includes system-level and instrument-internal real-time commands, anomaly response, and sequence pointing and timing adaptation using the latest ephemeris information

Each phase of the sequence development process had to overcome many operational challenges due to the immense complexity of the spacecraft, tour design, pointing capabilities, flight rules and software development. This paper will address the specific challenges related to each of those complexities and the methods used to overcome them during operations.

I. Introduction

The science planning process was created from a series of mission planning requirements and models from past planetary missions. From the program documents, *Cassini Operations System Functional Design* and *Cassini Operations System Functional Requirements*, the science planning process derived its driving requirements. Requirements were identified as accepted, modified, or rejected, and processes or tools were assigned to each requirement. From these basic guidelines and constraints, the Science Planning process was created. This process consists of an activity generation, integration, implementation, aftermarket and science operations plan update.

After the science planning process is complete, the products are handed off to the Uplink Operations Team (ULO). ULO leads the Sequence Virtual Team (SVT) through final verification and detailed command generation during the Science and Sequence Update Process (SSUP). ULO then uplinks the commands to the spacecraft, and acts as the custodian of the sequence commands as they execute. Any real-time engineering and science activities are coordinated through ULO during execution.

II. The Science Planning Process

A. Activity Generation, Integration, and Conflict Resolution

The first process is the activity request generation process. The following diagram shows the flow of the activity requests for tour into the Cassini Information Management Systems (CIMS), the database where the tour details would be stored.

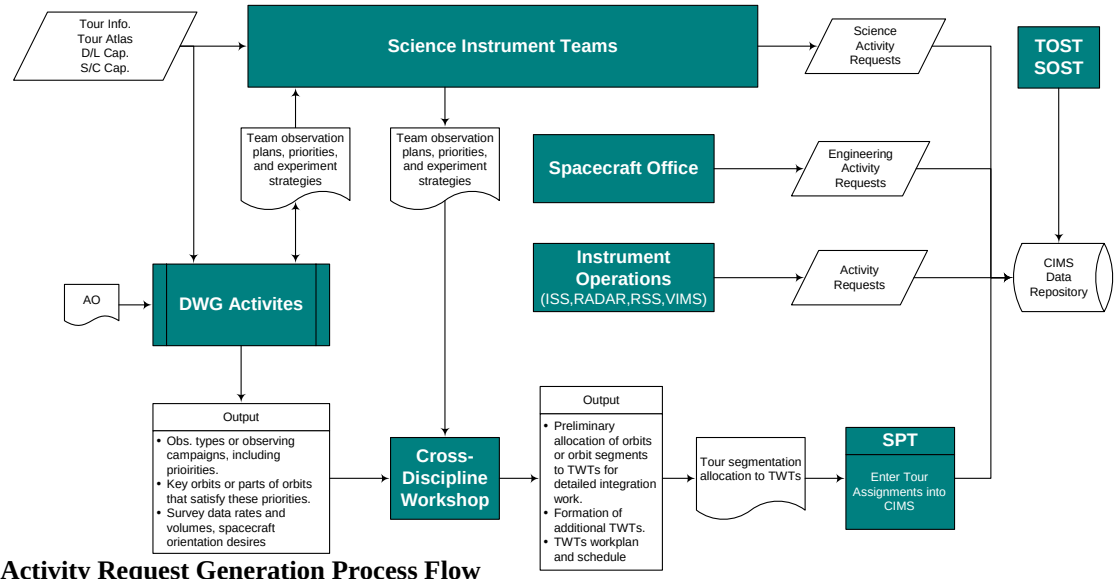

Figure 1. Activity Request Generation Process Flow

The science teams are responsible for prioritizing and developing conflict-free activity requests and plans for their experiment objectives throughout tour. The Spacecraft Office (SCO) and the Instrument Operations Team (IO) have to deliver all requests for engineering-related activities needed in the tour. These requests are then input into CIMS for use during the science planning integration and implementation processes.

The discipline working groups were then established, each taking into account the main science objectives of the mission and identifying methods throughout the tour with which these goals could be obtained. These working groups were the Rings Working Group (RWG), Atmospheres Working Group (AWG), the Magnetosphere and Plasma Science working group (MAPS), and Saturn Working Group (SWG). Once this was completed, the tour segments were assigned to different target working teams (TWT) for detailed integration.

Integration is the process in which the science and engineering activities are incorporated into a coherent timeline within each TWT. The Science Planning Engineer (SPE) aides this process by creating detailed activity reports from the information in CIMS and proposing solutions to any conflicts. The science teams and other involved parties then work the conflicts during the TWT process. Once each TWT completes a timeline, the products are archived and then later merged together for the implementation process. The following process flow reveals the details of this process.

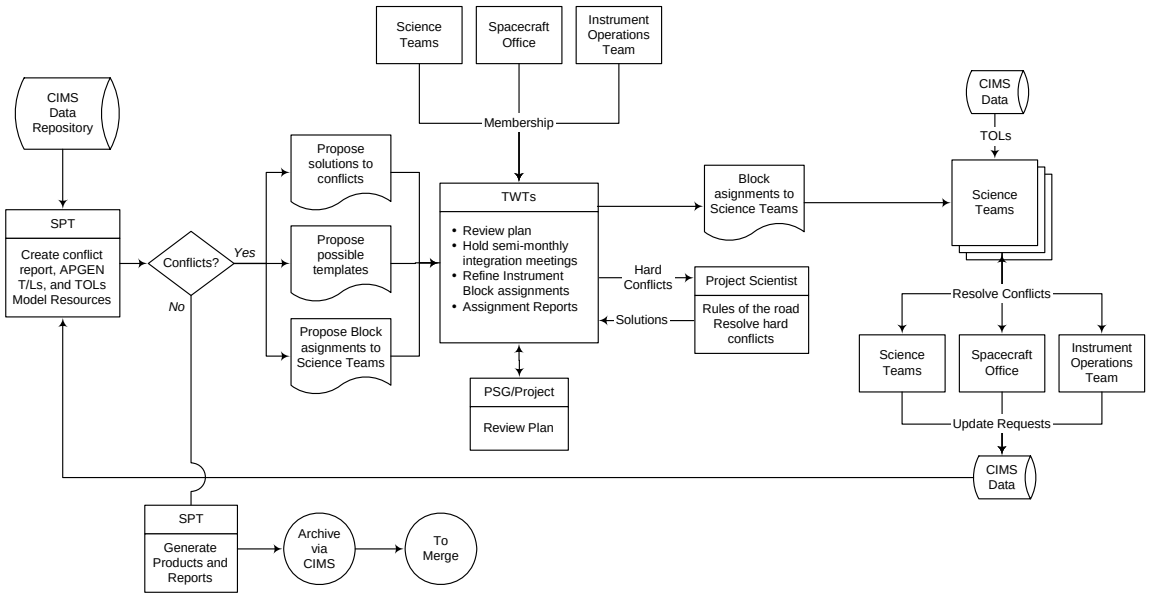

Figure 2. Integration/Conflict Resolution Process Flow

The groups responsible for integration are the Titan Orbiter Science Team (TOST), Satellite Orbiter Science Team (SOST), and the Target Working Teams (TWT): Cross-Discipline, Rings, Saturn, and Magnetospheres. A Cross-Discipline Workshop was held as part of the activity planning process, which formed the TWTs and allocated the orbits and work plan to each group by science objective.

B. Implementation

Implementation is the process by which integrated activity plans are constraint checked and turned into spacecraft command modules that will later be expanded and radiated to the spacecraft. Members of the SP team, SCO, IO, ULO, Mission Planning (MP), and the Science Instrument teams make up the Science Planning Virtual Team (SPVT) that performs this process. This was architected as a two-port process. During each port, the instrument team commands are merged into an integrated sequence that is constraint checked to ensure the spacecraft remains safe. The two-port process design was based on estimated changes made during implementation and the amount of time needed to accommodate them. For the beginning sequences, this process was developed as a three-port process and later reduced to a two-port process as the learning curve stabilized. At the end of each implementation process, the sequence history and liens are documented, and the instrument and merged sequence spacecraft activity and sequencing files (SASF) are archived in the project database (DOM). The following process flow diagram shows the specific activities during this process.

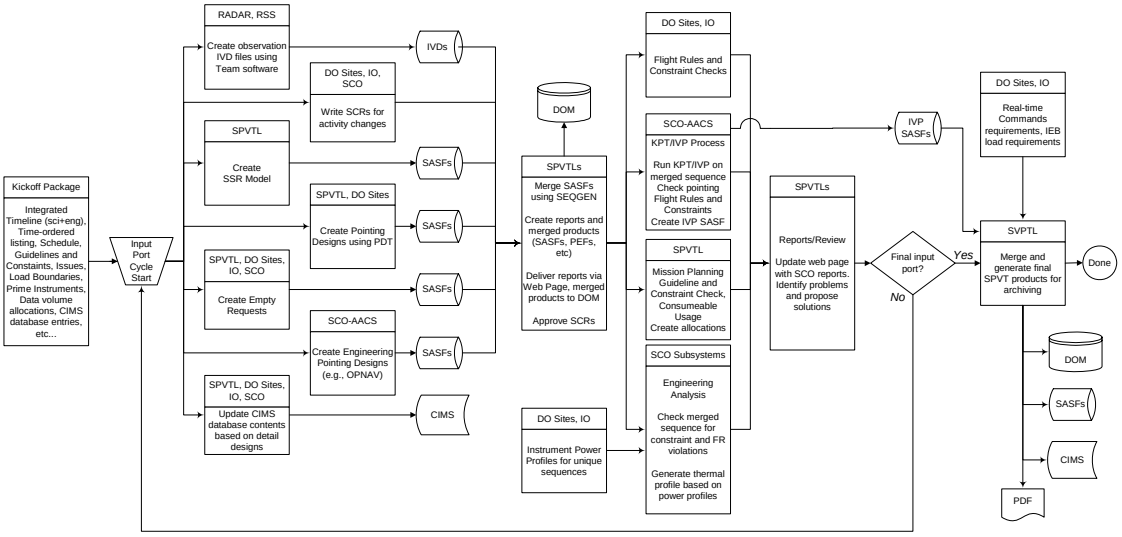

Figure 3. Implementation Process Flow

C. Aftermarket and SOP Update

The Aftermarket process occurs when science and engineering activities are updated for new discoveries or changes in the tour design. During this process, each team submits their requested changes to the SPVT Lead. These changes are costed based on the estimated work hours needed for implementation. If the work hours to implement all changes are below the limit set for that sequence, then the changes are allowed to proceed to reintegration. If there are more changes requested than work hours allocated for that sequence, negotiations are performed in order to reduce the amount of changes implemented for that sequence. There is a reintegration period during which the TWTs/OSTs manage the CIMS database to produce new products for the SSUP Process. A wrap-up meeting is schedule for the handoff of those products to the start of SOP Update.

Once this handoff takes place, the five-week SOP Update process begins. This process parallels one port of the implementation phase. During this time, updates to the pointing designs and data policing tables, which control the amount of data volume instrument teams are allowed to collect at a given time, take place. In addition, the latest ephemeredes and deep space network (DSN) station allocations are incorporated into the sequence. The sequence is constraint checked and the pointing profile is validated. At the end of SOP Update, a conflict-free, integrated SASF is handed off to the Uplink Team for their process. The following process flow diagram describes the SOP Update process.

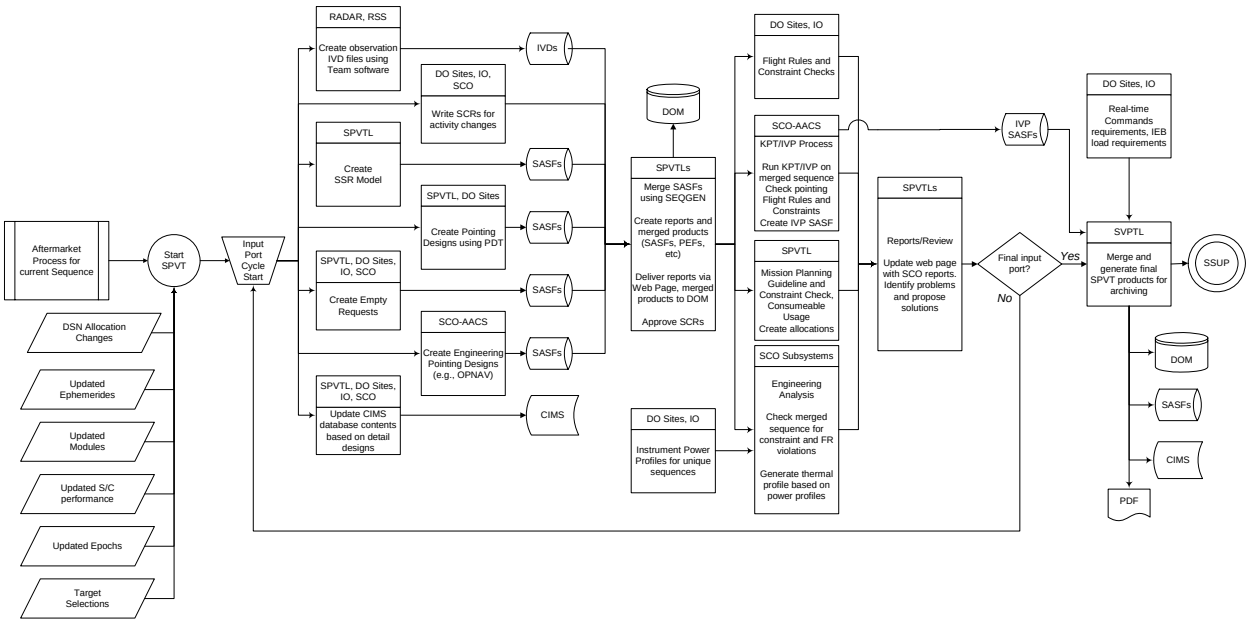

Figure 4. The SOP Update Process

III. The Science and Sequence Update Process

The Science and Sequence Update Process (SSUP) consists of four phases: Subsequence Generation (SSG), Sequence Integration, Review, and Validation (SI&V), Real-time Command Preparation, and Sequence and Command Radiation. Because the sequence commands are in their final stages of validation before uplink to the spacecraft, SSUP employs more stringent configuration management through its policies and procedures. The figure below details the steps of the SSUP process.

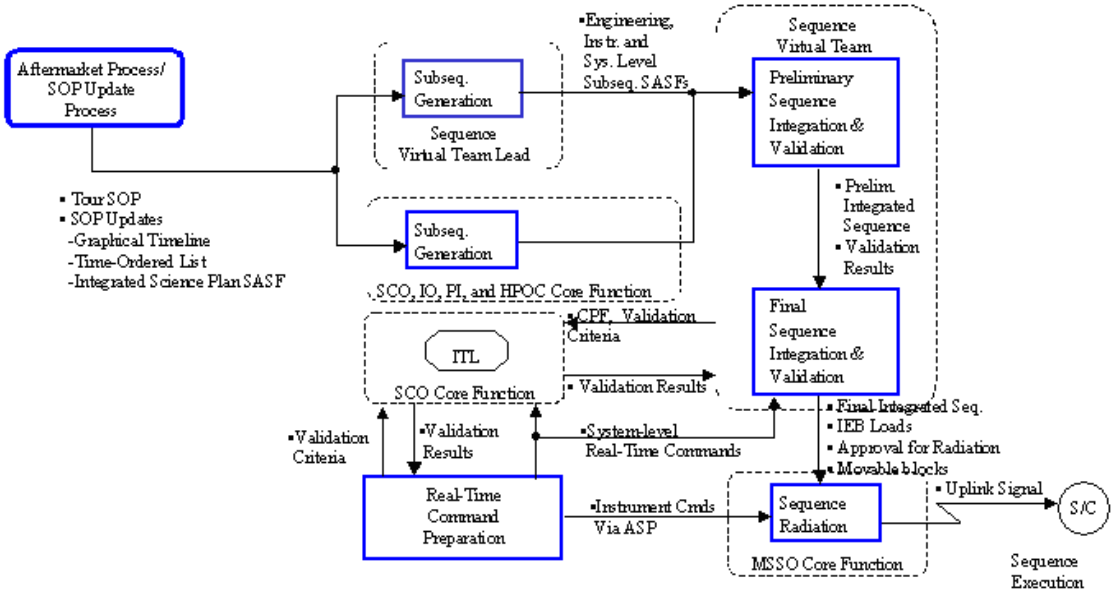

Figure 6. The SSUP Process

A. Subsequence Generation

During the SSG phase, the SOP Update products are used to create team-stripped SASFs, which provide more configuration management. Teams review these products and fill in any empty requests that were provided as placeholders during earlier development processes. Existing activities are reviewed for desired design changes or errors identified during the SOP Update process. Changes to existing designs that affect a system-level resource must be identified with a sequence change request (SCR) that is reviewed and approved by the SVT. Once approved, these changes can be submitted in the final SASF along with other accepted changes to triggers and other team internal resources.

B. Sequence Integration, Review, and Validation

The SI&V process consists of a preliminary and final phase. The preliminary phase consists of two cycles. The final phase consists of only one cycle. During the Preliminary SI&V phases, a merged product is created from the science instrument and engineering SASFs that were submitted at the end of the previous phase. The moveable block, a pre-identified portion of the sequence that may be updated in time during a later phase, is extracted and developed as its own product as well as a merged part of the background sequence to ensure compatibility. This merged product is reviewed by the instrument teams and by the spacecraft office for any conflicts or flight rule violations. The vector commands are generated from the attitude control team and merged with the background sequence to create a complete product that could be uplinked to the sequence. Changes as a result of the review of these products must be identified through SCRs for SVT review and approved by the SVT Lead (SVTL). Once approved, these changes are delivered in the SASF for the following phase. Changes for the Final SI&V phase are allowed for health and safety reasons only. First time events or special activities within the sequence are also tested using the Integration Test Lab (ITL). Instrument Expanded Blocks (IEBs), subroutines stored in instrument memory

for sequence activities, are delivered and processed during this phase. The SI&V phase results in a conflict-free, integrated, uplinkable distributed sequence.

C. Real-Time Activities

Real-time activities are prepared before and during sequence execution. Spacecraft resource users, both instrument teams and the spacecraft office, submit their real-time activities in an SASF. The SVTL adds commands to the SASF for memory management and performs constraint-checks. If the real-time activity is system-level, the appropriate teams review and approve the file before uplink. For instrument internal commands, they can be coordinated through this process, if memory management is needed, or sent via the Automated Sequence Processor (ASP). This processor will autonomously build the commands that are sent directly to the instrument and submit them for radiation over the next uplink opportunity without involvement of the SVT.

D. Sequence and Command Radiation

The Sequence and Command Radiation process prepares the DSN complexes for radiation of background sequence and real-time commands. The process requires interaction between two teams that authorize the uplink of files to the spacecraft. Once the DSN station has acquired the signal with the Cassini spacecraft, telemetry is received and initial conditions are verified, if necessary. The SVTL or Spacecraft Engineer verifies the command to be uplinked by confirming the time the file was created and the name of the file. These two fields provide unique characteristics for each command file to ensure the correct files are uplinked to the spacecraft. The mission controller verifies the file upon authorization from the SVTL or SCO Engineer and places it in the queue for radiation. This file is again verified by the SVTL or Spacecraft Engineer, and a 'go' is given to the mission controller for radiation of that file either immediately or at a specified time during that uplink opportunity. Once the file is active, the command packets are transferred to the DSN complex and radiated to the spacecraft. Verification of the file is done through registration of the program in sequence memory or execution of the commands seen by the instrument team or spacecraft subsystem.

IV. Challenges

A. Distributed Operations

1. Concept

The Cassini-Huygens Mission to Saturn and Titan is a distributed operations project. Distributed operations were chosen in an effort to reduce cost and maximize expertise. Because Cassini-Huygens was conceived as an international project, it was decided that it was more cost effective not to relocate the experts on the international components to JPL for the duration of the mission. This also allowed for better-suited instrument operating teams because the experts in each scientific field could be members of the team, and the teams could better tailor their products (science data) to the end-user community.

2. Virtual Teams

The concept of a Science Planning and Sequence Virtual Team was created to allow a core group to handle the sequence design and development processes. The use of mediums such as email, webpages and teleconferencing were essential to the success of the distributed operations concept for the Cassini-Huygens Mission. In addition, other tools and resources were developed to overcome the challenge of not having the core team of developers working in a central location.

3. Design tool and Information distribution

Due to the distributed operations system, one of the challenges was to effectively coordinate the software and development tools to the distributed operations sites. Export regulations added a layer of complexity in distributing software to our international partners. The use of the internet was crucial in this effort, with the creation of downloadable design software and centralized information databases. The science planning process utilized the web to ease the interface with the distributed teams with the CIMS database. This was the database in which science planning manages the overall pointing designs, telemetry modes, power modes, data volume allocation and timing resources for the SOP. By making this a web based resource, all members of the planning team could refer to the same resource in a timely manner. In addition, the real-time command generation process utilized a web-based command approval request form through which the distributed teams could initiate or provide approval to their commands for uplink.

Due to the distributed operations team, another problem encountered was how to provide configuration management and file distribution to the configuration files used by the various sequence design software tools. The Science Planning and Sequence Team Leads became the custodians for these ancillary files used during each

sequence. The Sequence Phase List of Ancillary Files (SpLAF) is the central listing of those files that is published to the distributed object manager (DOM). In order to ease the interface for the distributed operations teams, the list is also collected into a downloadable tar bundle for use with the download version of the pointing design tool (PDT). In addition, a system was developed to mirror the ancillary file directory structure created at JPL, so distributed teams could download one file and utilize the same file locations that were available to local users at JPL.

The introduction of web based solutions such as these provided significant time and money savings to the operations costs on the Cassini-Huygens Mission.

B. Science Observations

1. Pointing Design

Early in the design of the Cassini spacecraft, a descoping effort eliminated the high precision scan platform (HPSP) and the turn table (TT) for the science instruments. The instruments were then to be fixed-mounted to the basebody on two pallets. While the elimination of these elements did simplify the structural and thermal design by deleting deployments and eliminating the sunshade, which would then be done by the high gain antenna (HGA), it also eliminated the decoupled pointing of the instruments and the HGA. This meant that the ORS instruments would have to acquire data without the HGA pointed at Earth. This provided significant complication of the pointing strategy and science trades that had to be made during the planning process.

As a reaction to this, every science opportunity could no longer be optimized for all science objectives. Titan flybys, for instance, were divided up amongst prime science instruments. One flyby would be a Radio Science (RSS) prime flyby for an occultation, whereas the next flyby may be for the Ion Neutral Mass Spectrometer (INMS) to take measurements of the titan atmosphere. In addition, attitude strategy spreadsheets (SPASS) were employed during the science planning process to manage pointing for each instrument. Prime pointing instruments had to accommodate riding instruments on observations, which increased the amount of coordination needed for pointing designs and changes.

Another challenge that had to be overcome due to the loss of the scan platform and turn table was to maintain the safety of the overall spacecraft pointing profile while allowing individual instruments to take control of the pointing for their individual observations. A waypoint strategy was developed in which a safe attitude was chosen for a given period of time. Science observations had to turn to and from that waypoint attitude in order to facilitate coordination between science activities. If a design is not conflict free by the end of the sequence development process, it can be removed from the sequence, and the spacecraft is will remain at the designated waypoint during that time. While this strategy does not optimize science observation time, it does help prevent the spacecraft from pointing in an unsafe orientation.

2. Trajectory Changes

Another challenge during the sequence design process was how to create pointing profiles that were robust to trajectory updates and could be updated based on the latest ephemeris information. Those challenges were overcome through the use of turn margin, moveable blocks, and pointing updates during sequence execution. The pointing uncertainties for the reference trajectory used in sequence development were analyzed to estimate turn margin needed during pointing activities. This turn margin is held in pointing designs until late in SSUP when the last chance to accept an update to the reference trajectory has passed.

For science opportunities that are most sensitive to changes in pointing and rely on the latest trajectory information, moveable blocks are utilized. There are two types of moveable blocks: ground and live. Ground moveable blocks (GMB) are shifted in time when a new reference trajectory is incorporated during sequence development. Live moveable blocks (LMB) are shifted based on new trajectory information that is available during sequence execution. These time shifts are performed by relating all science observations to an epoch time that can be updated in the CIMS database and incorporated into the sequence products without redesign of each observation.

One final method to respond to late changes in the ephemeris information is the live update process. The live update process can include the live moveable block time shift or can be an update to only the vector definitions. During this process, the latest trajectory information available during sequence execution is analyzed and run through the pointing software. The inertial vector definitions are updated and overlaid onto the existing definitions that are onboard the spacecraft for a given time period. The figure below details the update process during sequence execution.

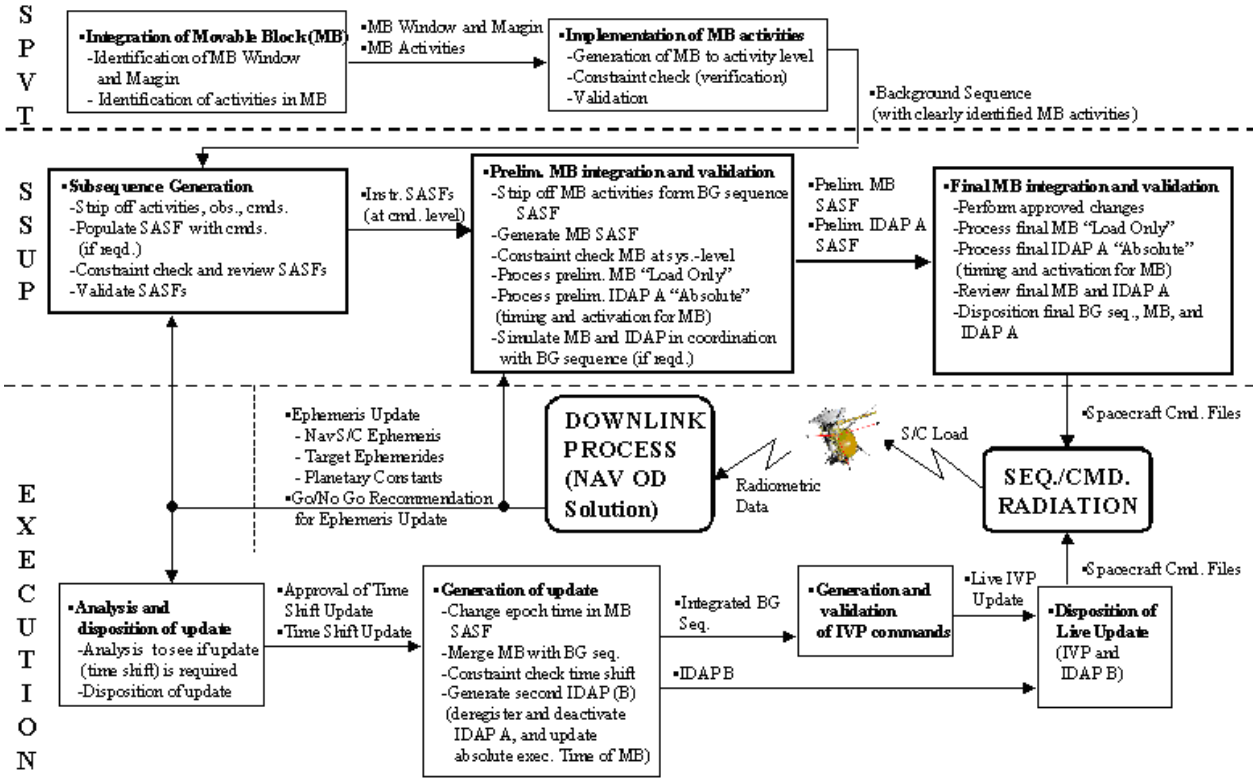

Figure 7. The Moveable Block and Live Update Process during Sequence Development and Execution

C. Software Development

1. Flight Rule Checking

The Cassini spacecraft is a complex orbiter that requires detailed flight software (FSW) and flight rules to maintain the safety and operation of the spacecraft. Cassini has defined over 330 flight rules as a method of issuing spacecraft and instrument health and safety. Violation of these rules could threaten health and safety of an instrument or spacecraft subsystem. Many of those flight rules were coded into ground software (GSW) in order to minimize the likelihood of oversight which could damage an instrument.

2. Planned FSW/GSW Delivery

Another method to lower development costs pre-launch was to defer software capabilities until closer to the need date. For example, reaction wheel control would not be needed for the first four years of flight. Consequently, Cassini launched without this capability and would later provide it as the need date arose.

3. Multi-mission software

In an effort to lower software development costs, Cassini decided to take advantage of already developed, multi-mission software. Many algorithms needed are not dependent on any specific spacecraft mission. Consequently, these algorithms can be developed once and have this development costs spread over all missions who utilize this multi-mission concept. Due to the complexity of the Cassini spacecraft and operations, many of these tools may have to be modified, which could offset the cost savings provided by multi-mission software.

4. High Speed Simulator (HSS)

HSS was originally designed and developed as a less costly, less labor intensive and faster alternative to the ITL. The initial HSS concept was a multi-mission software simulator that employs the actual attitude and articulation control (AACS) and Command and Data System (CDS) FSW to mimic the sequence execution onboard spacecraft. The concept of a high speed AACS and CDS software simulator proved to be too costly to implement, mainly due to

the complexity of the AACS FSW and its interaction with CDS FSW. An additional challenge was to find affordable hardware that can execute the sequence in a timely manner, in order to conform to the SSUP schedule. The AACS FSW simulation requirement was eventually dropped in the Cassini-Huygens mission (and its predecessor Galileo) and the HSS was strictly utilized for the CDS capabilities. The Cassini AACS team developed a FSW simulator to perform a similar function.

Advantages of HSS over ITL are numerous, including lesser requirement of expertise for operations, faster than real-time execution, availability via any workstation (versus in the ITL location), ease of setup, no need for continuous personnel support, and lack of hardware configuration requirement.

Given the advent of hardware and availability of on the shelf software, future deep space flight missions should consider development of a joint high speed AACS and CDS software simulator. This could lessen the burden of spacecraft operations on the ITL, so that resource can be more dedicated to FSW development and validation, as it was originally intended. Consequently a functional HSS can reduce the cost of the spacecraft operation by eliminating the ITL operations and maintenance cost during the later phases of cruise and orbit.

D. Spacecraft Complexities

1. Sequence Development

The complexity of the Cassini spacecraft presented multiple challenges for the sequence development process. With the distributed operations team, the distance to Saturn, and the complex operation of the spacecraft, simply planning the activities a short time in advance like other earth or mars missions was not an option. In order to perform the constraint checks and activity coordination, observations were developed in sequences that began development years in advance. Multiple sequences were developed in different phases at a given time. For instance, a sequence planned for the 2008 time frame would be in implementation phase, while the sequence that was scheduled for uplink in a few weeks would be in the final phases of the SSUP process. This placed strain on both the distributed operations team and the science planning team. Since increased funding to hire additional personnel for that time period were not available, most teams found ways to automate their processes to ease sequence development. The Cassini Mission may be faced with flight team attrition due to the length of the prime mission. However, more automated processes and standardized procedures will help offset that potential problem.

2. Memory Management

Memory management was another challenge for the Cassini-Huygens Mission. While the spacecraft had more memory capability than other missions at that time, the complexity of the commands expanded to utilize the full capacity. Because the sequences were of such great length and the instrument commands so complex, the IEB strategy was developed. Subroutines are sent to the instrument for storage in instrument memory in advance of sequence execution. During the sequence execution, instruments will include commands to load these IEB subroutines when needed. This reduces the amount of memory needed for each sequence by loading these commands into the instrument memory rather than including them as part of the sequence commands.

In addition, some instruments developed cyclic commands to minimize sequence size. These commands are a standard set of activity definitions that are defined at the beginning of the sequence. Throughout sequence execution, the instrument will refer to that cyclic definition through a single command rather than repeat the entire set of commands for that activity.

The sequence memory was divided into regions for ease of management during execution. These regions were specified for program types such as Moveable blocks, Inertial Vector updates, Orbital Trim Maneuvers, mini-sequences and small programs, and the background sequence components. Coordination between sequences was another memory management challenge. In order to accommodate both sequences in the available background sequence memory region, the sequence programs are divided into three programs: Current Master, Overlap Master, and Long Term Master. During the sequence development process, the current master is set to expire at the time that the next sequence will begin loading on the spacecraft. The overlap master and the long term master contain the commands needed for the last few days of sequence execution before the next sequence program takes over.

V. Conclusion

The sequence development process for the Cassini-Huygens Mission to Saturn and Titan is a complex process that involves years of planning and coordination between many teams. This process had to respond to challenges related to the distributed operations structure of the core development team, through its software, web-based tools, and configuration management. In addition, the pointing strategy during spacecraft operations was complicated by the descoping efforts during the spacecraft design. Methods were developed to manage safe and effective handoff

of the spacecraft orientation between teams. Because the Cassini spacecraft is so complex in its instruments and its science objectives, the science and engineering activities are planned years in advance. In order to take advantage of the latest ephemeris knowledge, margin, timing flexibility, and update processes were developed to allow for incorporation of this information at later phases of sequence development and execution. In addition, the extensive flight rules needed for this mission added difficulty to software development and simulation. To accommodate the complexity of the science and engineering sequences, the sequence memory must be managed carefully by preloading instrument specific commands before the sequence, using cyclic definitions for repeat observations, and dividing the background sequence into sections that expire when more memory is needed for upcoming activities. The lessons learned as a result of the challenges encountered during the Cassini-Huygens sequence development process will serve as a model for future complex mission operations.