Doing Development Right: Lessons Learned from the Development of the Cassini Saturn Mission Operations System

Johanna Gunn[1], Theresa Anderson[1], Tadas Sesplaukis[1]

[1]Jet Propulsion Laboratory, Pasadena, CA, 91001, USA
California Institute of Technology

Introduction

The Cassini Saturn Mission's operational system enables one of NASA's most complex and challenging deep space scientific missions ever. Cassini is a combined NASA Saturn orbiter and European Space Agency provided Titan atmospheric Probe with 12 orbiter and 6 probe instruments supporting 27 diverse science investigations. Taking advantage of the nearly seven year cruise to Saturn, the Mission Operations System (MOS) post-launch development team at the Jet Propulsion Laboratory has implemented a system to support the science phase of the mission that has successfully overcome many development challenges that will face future missions both big and small. Many development challenges revolved around the difficulty of system engineering a large, diverse, highly collaborative operational system that spans 10 US and 4 European operations sites, required the development or adaptation of 84 ground software programs, and required both development and validation of the system in a limited time while simultaneously supporting cruise operations.

Cassini Mission Overview

Launched on October 15, 1997, Cassini will reach Saturn in July 2004. Once at Saturn the mission will execute a highly challenging tour of the Saturnian system over four years that includes 74 Saturn orbits and 44 targeted Titan flybys. Cassini was built in a cooperative effort involving three space agencies. The Cassini orbiter was built and managed by NASA's Jet Propulsion Laboratory with the contributions of seventeen nations. The Huygens probe, to be delivered to Saturn's moon Titan early in the mission, was built by the European Space Agency. The Italian Space agency provided Cassini's high-gain communication antenna. More than 200 scientists worldwide will analyze the scientific data returned.

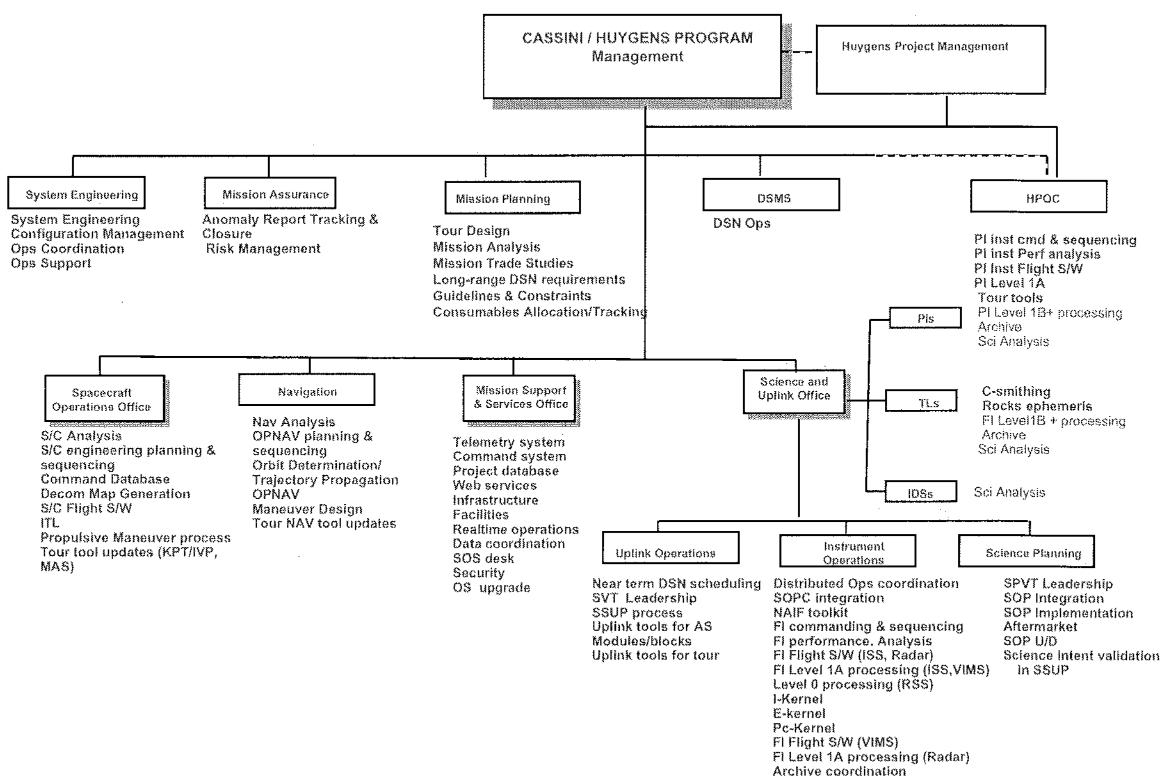
In addition to the challenge posed by the complex Tour scenario, early spacecraft design decisions to reduce pre-launch costs resulted in increased complexity to the orbiter including the loss of the scan platform resulting in significant instrument conflicts needing resolution during the operational phase. Among the constraints are a large complement of body-fixed instruments with competing science collection requirements and significant power limitations resulting in the need for operational modes that manage the total power needs of the entire flight system including the instruments. The large complement of sophisticated science instruments, each with their own science team, the multitude of scientific investigations, and the ESA Huygens Probe Mission operated in Darmstadt, Germany resulted in a large, geographically distributed science team with varying degrees of operations knowledge and experience at science operations sites.

Cassini Mission Operations System

By plan, pre-launch development of the Cassini MOS was limited primarily to the team and data system capabilities needed for the cruise phase of the mission. The prime objective of the cruise phase was to reach Saturn with a minimum of science activities. All capabilities needed for the four year Saturn Tour, with the exception of those needed to prove the flight system capability pre-launch, were deferred until after the 1997 launch.

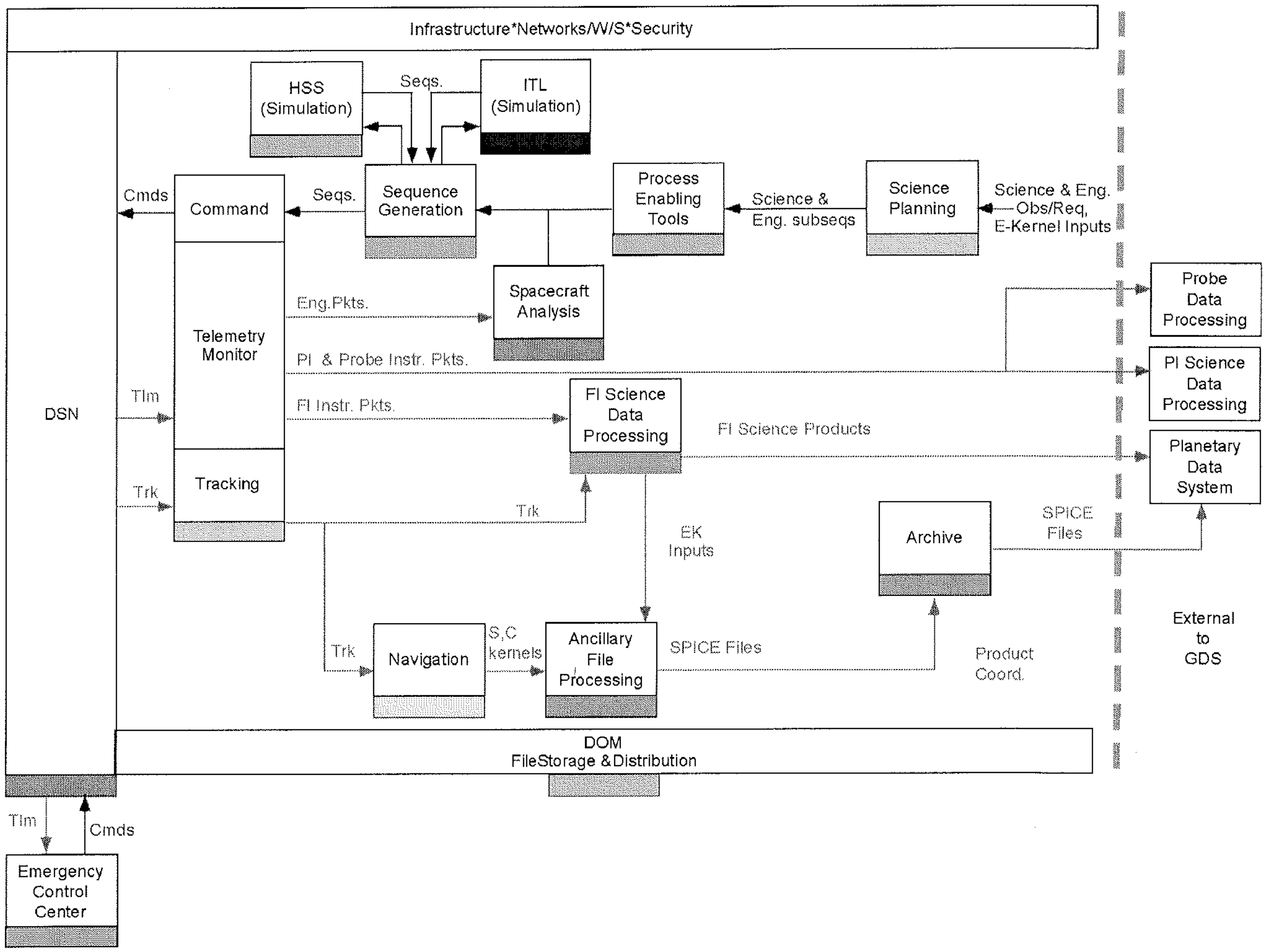
The Cassini MOS for the Saturn mission is comprised of 3 Offices (Spacecraft, Mission Support Services, Science and Uplink), which were further, divided into teams within the Offices. The Cassini MOS Offices and responsibilities are illustrated below.

Cassini Functional Organization



Subsystems developed by the teams comprised the Ground Data System (GDS) as illustrated below in a high-level functional flow.

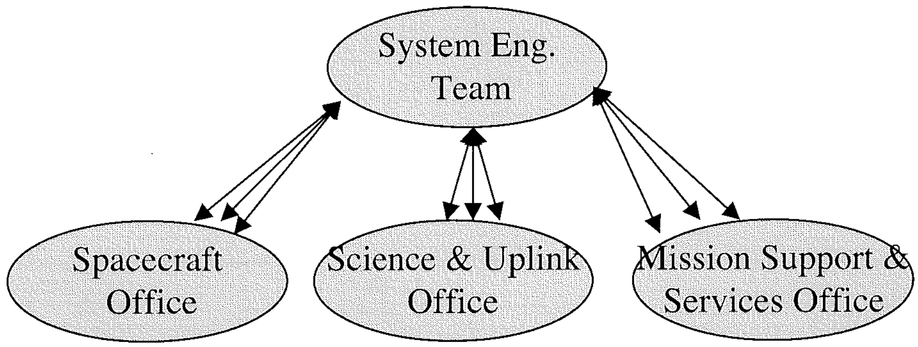
Ground Data System Functional Flow (High-Level)



Cassini System Engineering Paradigm

Pre-launch development of the operational system employed a traditional system engineering paradigm. System level engineers defined requirements based on incoming program-level requirements that were allocated down to teams who in turn allocated requirement to the subsystems they would operate during the mission. The Mission Operations Engineer (MOSE) was responsible for leadership of operations system engineering activities across all teams and their subsystems. The MOSE was supported by a team of nine system engineers who led development of the Uplink System, Downlink System, and Ground Data System, in addition to all system level training and verification and validation activities. This team was augmented by system engineers performing configuration management and GDS testing. This approach is graphically depicted in the figure following.

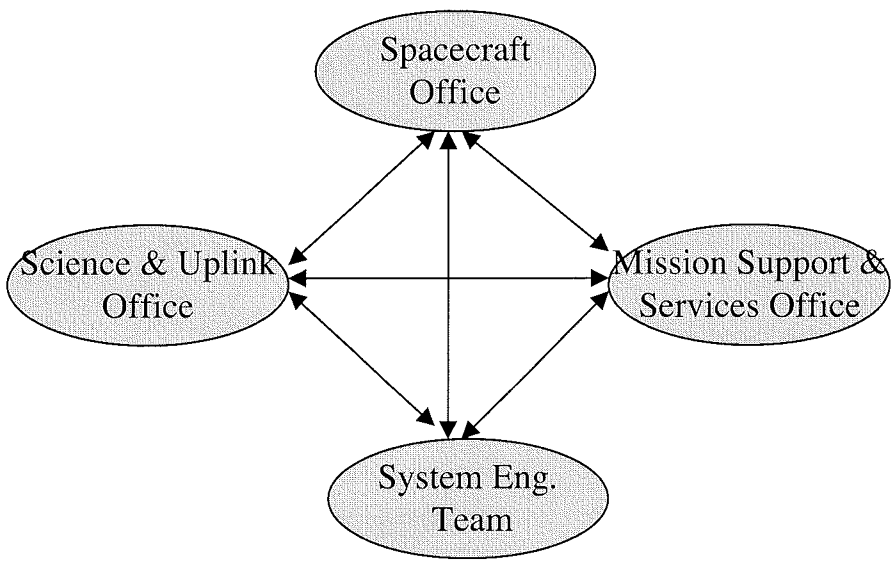
Pre-Launch System Engineering Paradigm



This paradigm was highly successful but also considered costly given available funding and resources. For development of operational capabilities for the Saturn Tour phase of the mission, the development team would be required to fly the mission to Saturn and simultaneously develop the Tour operational capability necessitating more efficient use of resources.

Because there were significant development tasks that had been deferred until the cruise phase of the mission, the project level system engineering function remained significant in scope. Post-launch a new "virtual" system engineering paradigm was instituted with the objectives of reducing cost and increasing productivity. The key characteristic of new system engineering paradigm was the assignment of system-wide engineering responsibilities to the implementing offices and teams at next lower level as depicted below.

Post-Launch System Engineering Paradigm



"Ownership" for elements of the system was allocated to directly to the offices. Cross-office issues were no longer worked solely by a centralized team, but instead directly between offices. Ownership for all system level elements of the system were allocated directly to the offices/teams with ownership assigned to the office/team which was the most significant stakeholder. Greater efficiency in decision making was gained by moving authority for technical decision making as low as possible in the organization. Table 1 below illustrates the assignment of system level responsibilities pre-launch under the traditional system engineering paradigm and post-launch under the new "virtual" system engineering paradigm.

Table 1, Assignment of System Engineering Responsibilities Pre- and Post-Launch

Operations Process	Post-Launch "Old" SE Paradigm Responsibility	Post-Launch "Virtual" SE Paradigm Responsibility
Long Range Mission Planning	System Engineering	Mission Planning Team
Science Planning Process	System Engineering	Science and Uplink Office
Sequencing Process	System Engineering	Science and Uplink Office
Maneuver Development & Execution	System Engineering	Spacecraft / Navigation Office
Real-time monitoring	System Engineering	Mission Support Services Office
Data Collection & Processing	System Engineering	Science and Uplink Office
Non-Real Time Analysis	System Engineering	Spacecraft / Navigation Office
Ancillary Data Processing	System Engineering	Science and Uplink Office
Science Data Processing	System Engineering	Science and Uplink Office
Science Data Analysis	System Engineering	Science and Uplink Office
Archive Generation & Validation	System Engineering	Science and Uplink Office
Status Reporting	System Engineering	System Engineering
Anomaly Response	System Engineering	System Engineering

Under the new system engineering paradigm a single engineering working team, known as the System Engineering Round Table, coordinated system engineering activities to ensure the necessary level of coordination and that no parts of the system were overlooked. Membership consisted of the now smaller, four person system-level engineering team and a single system engineer from each of the three offices. All team members shared equally the responsibility for the development of the system with the system-level engineers functioning as facilitators, system experts, and conflict resolvers.

The system-level engineers retained responsibility for developing and enforcing standards and common development processes including maintaining the software inventory and classifications, documenting system level requirements and waiver processes, overseeing requirements and design reviews, and managing the software/hardware configuration and delivery processes.

Changes to the system engineering paradigm included re-evaluation of all tasks for value-added and efficiency. Redundancy was eliminated as were a significant number of lower priority tasks. Training and the majority of verification and validation activities were delegated to the now more empowered offices, with little oversight. Overall an approximate $2M program-wide cost reduction was accomplished in FY'00 as a result of the change to the system engineering paradigm. Other advantages of the new system engineering paradigm included a more cohesive, better integrated development team and faster decision making for decisions contained in single offices.

Some drawbacks however became evident with the new system engineering paradigm including the need for a very high caliber of office system engineers with strong technical, leadership, communication, and decision-making skills. Equally challenging was finding office system engineers able to represent the large number of diverse tasks being undertaken in the offices. Although office system engineers had a responsibility to look at the "big picture" rather than focusing on their own office's needs, this was difficult where office and system needs or priorities were not the same. There were frequent questions and disagreements about ownership and responsibility for tasks. Sometimes issues would fall through the cracks because of mis-matched assumptions. Additionally office system engineers at times were not sufficiently empowered by Office Managers to make decisions or recommendations for their offices. A very high level of vigilance was needed to avoid these problems.

Summary

The long development lifetime of the Cassini Mission Operations System afforded an opportunity to compare two significantly different system engineering paradigms and to incorporate a number of different system-level development practices. Benefits of the traditional system engineering paradigm are clear and have been proven over time by many projects. The "virtual" system engineering paradigm employed post-launch allowed additional efficiencies when complemented with a strong cadre of system engineers at the office level and managers who were vigilant in managing to the new paradigm.

Acknowledgment