Selected Systems Engineering Process Deficiencies & Their Consequences

by

L. Dale Thomas, Ph.D., P.E.
National Aeronautics and Space Administration
George C. Marshall Space Flight Center

ABSTRACT

The systems engineering process is well established and well understood. While this statement could be argued in the light of the many systems engineering guidelines and that have been developed, comparative review of these respective descriptions reveal that they differ primarily in the number of discrete steps or other nuances, and are at their core essentially common. Likewise, the systems engineering textbooks differ primarily in the context for application of systems engineering or in the utilization of evolved tools and techniques, not in the basic method. Thus, failures in systems engineering cannot credibly be attributed to implementation of the wrong systems engineering process among alternatives. However, numerous systems failures can be attributed to deficient implementation of the systems engineering process. What may clearly be perceived as a system engineering deficiency in retrospect can appear to be a well considered system engineering efficiency in real time – an efficiency taken to reduce cost or meet a schedule, or more often both. Typically these efficiencies are grounded on apparently solid rationale, such as reuse of heritage hardware or software. Over time, unintended consequences of a systems engineering process deficiency may begin to be realized, and unfortunately often the consequence is system failure. This paper describes several actual cases of system failures that resulted from deficiencies in their systems engineering process implementation, including the Ariane 5 and the Hubble Space Telescope.

1. Introduction

The development of space systems, including launch vehicles and spacecraft, is a risky undertaking. The systems tend to be complex, and design margins tend to be very thin, such that the systems tend not to fail gracefully – that is, when failures occur, the consequences tend to be catastrophic and spectacular. The discussion which follows will examine seven catastrophic, spectacular system failures and how systems engineering process deficiencies contributed to those failures. Section 2 will describe a systems engineering process model against which the relevant contributing factors to the system failure will be categorized. Section 3 will briefly examine the failures of seven space systems – one launch vehicle and six spacecraft – and identify contributing systems engineering process deficiencies. An analysis of these systems engineering process deficiencies is presented in Section 4, and closing remarks are given in Section 5. A list of references, including the failure reports for each of the seven systems, is included in Section 6.

2. Systems Engineering Process Model

A system engineering process model is needed to provide a framework for characterization of systems engineering process deficiencies as causal contributors to systems failures. Rather than offer an original system engineering process model, one will be chosen from an existing system engineering standard. Numerous systems engineering standards have been developed that describe many models of the systems engineering process, and many definitions of systems engineering are provided. Comparative assessment of these standards reveals much more commonality than distinction. For instance, a comparison of two of the more noteworthy standards, IEEE-1220 *IEEE Standard for Application and Management of the Systems Engineering Process* and ISO/IEC 15288 *Systems engineering — System life cycle processes* is given in Annex C of IEEE-1220. [1] This comparative assessment provides a cross-reference between the two standards' differing terminologies and process hierarchies that illustrates their fundamental equivalence. One distinction between the standards is the more comprehensive life cycle model of ISO/IEC 15288; whereas IEEE-1220 focuses on system development and operations, ISO/IEC 15288 includes the concept and retirement portions in the life cycle model. Although any of the popular systems engineering standards would be satisfactory for purposes of this paper, ISO/IEC 15288 will be used since it is the lone international standard.

ISO/IED 15288 *Systems engineering — System life cycle processes* establishes a common framework for describing the life cycle of systems including concept, development, production, utilization, support, and retirement as depicted in Figure 1. [2] ISO/IEC 15288 consists of 25 distinct processes grouped into four general categories: Agreement Processes, Enterprise Processes, Project

Processes, and Technical Processes. The Agreement Processes are business centric processes, and not directly relevant to the analyses performed for this paper. Likewise, Enterprise Processes are strategic in nature and involve multiple projects, and hence also not relevant to a discussion of specific project failures. Still, it should be noted that the conclusions drawn from this analysis are indeed relevant to the Enterprise Processes of a national space agency or corporate supplier of space systems.

LIFE CYCLE STAGES	PURPOSE	DECISION GATES
CONCEPT	Identify stakeholders' needs Explore concepts Propose viable solutions	Decision Options: - Execute next stage - Continue this stage - Go to a preceding stage - Hold project activity - Terminate project
DEVELOPMENT	Refine system requirements Create solution description Build system Verify and validate system	
PRODUCTION	Produce systems Inspect and test	
UTILIZATION	Operate system to satisfy users' needs	
SUPPORT	Provide sustained system capability	
RETIREMENT	Store, archive or dispose of the system	

Figure 1. System Life Cycle (ref. Figure D-1 in ISO/IEC 15288)

The Project Processes are used to establish and evolve project plans, to assess actual achievement and progress against the plans and to control execution of the project through to fulfillment. The Project Processes consist of the following individual processes:

a) Project Planning Process -- produce and communicate effective and workable project plans;

b) Project Assessment Process -- determine the status of the project;

c) Project Control Process -- direct project plan execution and ensure that the project performs according to plans and schedules, within projected budgets and it satisfies technical objectives;

d) Decision-making Process -- select the most beneficial course of project action where alternatives exist;

e) Risk Management Process -- reduce the effects of uncertain events that may result in changes to quality, cost, schedule or technical characteristics;

f) Configuration Management Process -- establish and maintain the integrity of all identified outputs of a project or process and make them available to concerned parties;

g) Information Management Process -- provide relevant, timely, complete, valid and, if required, confidential information to designated parties during and, as appropriate, after the system life cycle.

The Technical Processes are used to define the requirements for a system, to transform the requirements into an effective product, to permit consistent reproduction of the product where necessary, to use the product to provide required services, to sustain the provision of those services and to dispose of the product when it is retired from service. The Technical Processes consist of the following processes:

a) Stakeholder Requirements Definition Process -- define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment;

b) Requirements Analysis Process -- transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services;

c) Architectural Design Process -- synthesize a solution that satisfies system requirements;

d) Implementation Process -- produce a specified system element;

e) Integration Process -- assemble a system that is consistent with the architectural design;

f) Verification Process -- confirm that the specified design requirements are fulfilled by the system;

g) Transition Process -- establish a capability to provide services specified by stakeholder requirements in the operational environment;

h) Validation Process -- provide objective evidence that the services provided by a system when in use comply with stakeholders' requirements;

i) Operation Process -- use the system in order to deliver its services;

j) Maintenance Process -- sustain the capability of the system to provide a service;

k) Disposal Process -- end the existence of a system entity.

Figure 2 depicts the foregoing Project and Technical processes that provide the benchmark systems engineering process model which the system failures in the following section will be assessed.

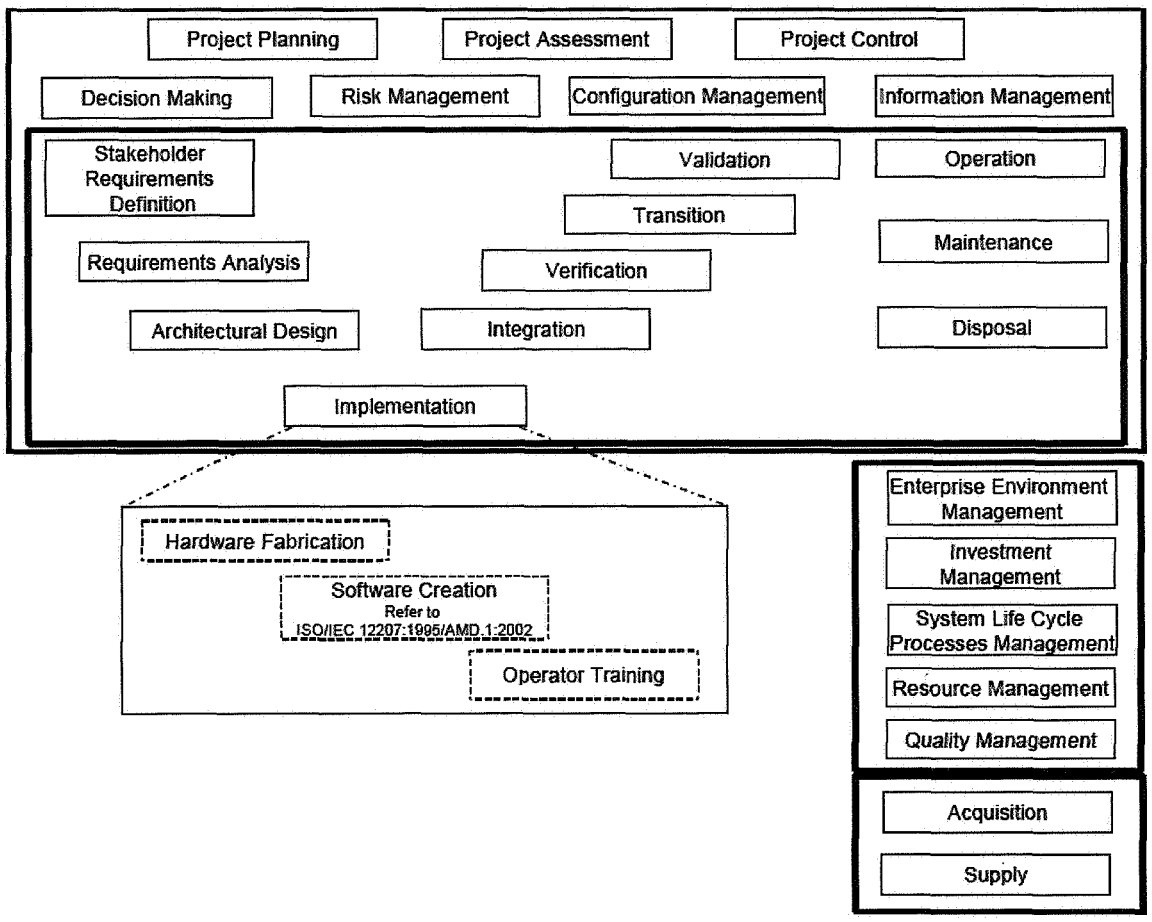

Figure 2. Systems Engineering Process Model (adapted from Figure C.1 in ISO/IEC 15288)

3. An Overview of Selected System Failures

The discussion which follows will describe the failures of various space systems from launch vehicles to scientific spacecraft. The failure scenario will be described, a summary of the cause of the failure, and then the failure cause will be put in the context of deficiencies of one or more of the system engineering processes as described in the foregoing section. In all cases, the associated

failure reports are used as sources for the failure scenario and the failure cause(s). In general, the contributing factors to the failure cause are drawn from the failure reports, although the classification of those contributing factors within a systems engineering process schema is not addressed in the failure reports. Finally, the treatment of the systems failures is not intended to be exhaustive, and only those elements of the failure reports most relevant to systems engineering are included. The reader is referred to the failure reports given in the Reference listing, which are all in the public domain, for a comprehensive treatment of the respective system failures.

3.1 Ariane 5

3.1.1 System Failure Scenario

On 4 June 1996, the maiden flight of the Ariane 5 launcher ended in a failure. Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 meters, the launcher veered off its flight path, broke up and exploded.

3.1.2 Failure Cause

The origin of the failure was flight control system, and more particularly the Inertial Reference Systems, with the primary and backup ceasing to function almost simultaneously at around H0 + 36.7 seconds. This loss of guidance and attitude information was due to specification and design errors in the software of the inertial reference system.

The inertial reference system of Ariane 5 was essentially common to a system which was flying on the version of Ariane 4 in operation at that time. The part of the software which caused the interruption in the inertial system computers was used before launch to align the inertial reference system and, in Ariane 4, also to enable a rapid realignment of the system in case of a late hold in the countdown. This realignment function served no purpose on Ariane 5, but was nevertheless retained for commonality reasons and allowed, as in Ariane 4, to operate for approx. 40 seconds after lift-off. During design of the software of the inertial reference system used for Ariane 4 and Ariane 5, a decision was made that it was not necessary to protect the inertial system computer from being made inoperative by an excessive value of the variable related to the horizontal velocity, although this protection was provided for several other variables of the alignment software. Ariane 5 had a high initial acceleration and a trajectory which led to a build-up of horizontal velocity which was five times more rapid than for Ariane 4. The higher horizontal velocity of Ariane 5 generated, within the 40-second timeframe, the excessive value which caused the inertial system computers to cease operation. [3]

3.1.3 Systems Engineering Process Deficiencies

Project Assessment -- The purpose of the review process, which involved all major partners in the Ariane 5 program, was to validate design decisions and to obtain flight qualification. In this process, the limitations of the alignment software were not fully analyzed and the possible implications of allowing it to continue to function during flight were not realized.

Architecture Design -- The specification of the inertial reference system and the tests performed at equipment level did not specifically include the Ariane 5 trajectory data. Consequently the realignment function was not tested under simulated Ariane 5 flight conditions, and the design error was not discovered.

Verification -- It would have been technically feasible to include almost the entire inertial reference system in the overall system simulations which were performed. For a number of reasons it was decided to use the simulated output of the inertial reference system, not the system itself or its detailed simulation. Had the system been included, the failure could have been detected. Post-flight simulations have been carried out on a computer with software of the inertial reference system and with a simulated environment, including the actual trajectory data from the Ariane 501 flight. These simulations have faithfully reproduced the chain of events leading to the failure of the inertial reference systems.

3.2 Hubble Space Telescope

3.2.1 System Failure Scenario

The Hubble Space Telescope (HST) was launched aboard the Space Shuttle Discovery on 24 April 1990. During checkout on orbit, it was discovered that the telescope could not be properly focused because of a flaw in the optics. Both of the high resolution imaging cameras (the Wide Field/Planetary Camera and the Faint Object Camera) showed the same characteristic distortion, called spherical aberration, which must have originated in the primary mirror, the secondary mirror, or both. Continued analysis of images transmitted from the telescope indicated that most, if not all, of the problem lay in the primary mirror.

The Corrective Optics Space Telescope Axial Replacement (COSTAR) was subsequently developed to counter the effects of the flawed shape of the mirror. COSTAR was a telephone booth-sized instrument which placed 5 pairs of corrective mirrors, some as small as a nickel coin, in front of the Faint Object Camera, the Faint Object Spectrograph and the Goddard High Resolution Spectrograph. COSTAR was successfully installed on the first HST Servicing Mission, and HST has since functioned nominally.

3.2.2 Failure Cause

Investigation of the manufacture of the mirror proved that the mirror was made in the wrong shape, being too much flattened away from the mirror's center. During the manufacture of all telescope mirrors there are many repetitive cycles in which the surface is tested by reflecting light from it; the surface is then selectively polished to correct any errors in its shape. The error in the HST primary mirror occurred because the optical test used in this process was not set up correctly, and thus the surface was polished into the wrong shape.

The critical optics used as a template in shaping the mirror, the reflective null corrector (RNC), consisted of two small mirrors and a lens. When, during the course of the failure investigation, the RNC was measured, it was determined that the lens was incorrectly spaced from the mirrors. Calculations of the effect of such displacement on the primary mirror show that the measured amount, 1.3 mm, accounted in detail for the amount and character of the observed image blurring. [4]

3.2.3 Systems Engineering Process Deficiencies

Project Assessment -- Clear indications of a problem in the HST optics were evident from auxiliary optical tests made at the time. A special optical unit called an inverse null corrector, designed to mimic the reflection from a perfect primary mirror, was built and used to align the RNC; when so used, it clearly showed the error in the RNC. A second null corrector, made only with lenses, was used to measure the vertex radius of the finished primary mirror. It clearly showed the error in the primary mirror. Both indicators of error were dismissed at the time.

Implementation – the HST primary mirror, as manufactured, was optically not within specification. The error was ten times larger than the specified tolerance.

Verification -- No verification of the reflective null corrector's dimensions was carried out by Perkin-Elmer after the original assembly. No integrated test of the HST optics was performed, which would have detected the spherical aberration in the primary mirror.

3.3 Mars Observer

3.3.1 System Failure Scenario

Mars Observer was launched from the Cape Canaveral Air Force Station on 25 September 1992. The eleven month cruise phase from Earth to Mars was relatively trouble-free, with only a few anomalies noted, all of which were corrected. The first of a series of maneuvers designed to insert the spacecraft into an orbit around Mars had been planned to take place on 24 August 1993. The sequence of events leading to the first maneuver began as scheduled on 21

August. The first action in this sequence involved pressurization of the propulsion system, initiated and controlled by a sequence of software commands previously stored in the spacecraft computers. In accordance with the mission's published flight rules, the transmitter on the spacecraft had been turned off during the propellant-tank Pressurization Sequence on 21 August; as a result, there was no telemetry during this event. No data from the spacecraft have been received since that time.

3.3.2 Failure Cause

Despite extensive analysis of the circumstances surrounding the mission failure of the Mars Observer spacecraft, clear and conclusive evidence pointing to a particular scenario was not discovered. The most probable cause of the loss of downlink from the Mars Observer was a massive failure of the pressurization side of the propulsion system. The most probable cause of that failure was the unintended mixing of nitrogen tetroxide (NTO) and monomethyl hydrazine (MMH) in the titanium tubing on the pressurization side of the propulsion system. This mixing was believed to have been enabled by significant NT0 migration through check valves during the eleven-month cruise phase from Earth to Mars. [5]

3.3.3 Systems Engineering Process Deficiencies

Project Planning -- The system failed to react properly to a program that changed radically from the program that was originally envisioned. The Mars Observer that was built departed significantly from the guiding principles originally established for the program, yet the acquisition and management strategy remained unchanged. The use of a firm, fixed-price contract was inappropriate to the effort as it finally evolved.

Information Management -- The discipline and documentation culture associated with, and appropriate for, commercial production-line spacecraft is basically incompatible with the discipline and documentation required for a one-of-a-kind spacecraft designed for a complex mission. Mars Observer was not a production-line spacecraft.

Architecture Design -- Too much reliance was placed on the heritage of spacecraft hardware, software, and procedures, especially since the Mars Observer mission was fundamentally different from the missions of the satellites from which the heritage was derived. The design of the propulsion system was not appropriate for the long duration dormancy required.

Implementation -- The use of a fault-management software package that was not fully understood.

Verification -- The original philosophy of minor modifications to a commercial production-line spacecraft was retained throughout the program. The result was

reliance on design and component heritage qualification that was inappropriate for the mission. Examples of this reliance include the failure to qualify the traveling wave tube amplifiers for pyrotechnic firing shock.

3.4 Wide-field Infrared Explorer (WIRE)

3.4.1 System Failure Scenario

The WIRE spacecraft was a planned four-month survey in the 12 and 25 μm infrared color bands. It was launched on 4 March 1999, and a problem was detected during its second pass over the ground station. The spacecraft was spinning and did not maintain a stable position. The instrument cover had been ejected at approximately the time that the WIRE Pyro Box was powered on. As a result, the instrument's solid hydrogen cryogen supply started to sublimate faster than planned, and the venting of the resulting effluent caused the spacecraft to spin up. The spacecraft reached a spin rate of sixty revolutions per minute before being brought under control. Because of the loss of solid hydrogen cryogen, the instrument could not perform its scientific observations, and the mission was declared a total loss.

3.4.2 Failure Cause

The cause of the failure was a digital logic error in the instrument pyrotechnic control electronics. The variable turn-on characteristics of the Field Programmable Gate Array (FPGA) used in the pyrotechnic control circuitry were not adequately considered in the electronics design. The FPGA application for the WIRE instrument did not account for the finite time it takes the FPGA to ramp up at turn-on and establish a stable configuration. That ramp-up time is proportional to the elapsed time since the device was last powered down, as capacitors internal to the FPGA dissipate their charge over time. During integration and testing, the system was energized every day, so the FPGA internal capacitance never had a chance to fully dissipate. Prior to launch, this part of the satellite circuitry had been powered off for about two weeks. [6]

3.4.3 Systems Engineering Process Deficiencies

Project Assessment -- A significant contributing factor which should have been able to prevent the WIRE mishap was the failure to investigate the source of the signal which caused the Electro Explosive Device (EED) Simulator to "latch" upon Pyro Box power-up during spacecraft integration. The incident was incorrectly attributed to excessive sensitivity of the EED Simulator and then an incorrect argument of similarity to other pyrotechnic device driver electronics was made; in fact it probably was an indication of the transient that caused the in-flight mishap.

Information Management -- A contributing factor to the mishap was the lack of documentation for the Actel A1020 FPGA's power-up transient characteristics in the device data sheet. This information was available in the FPGA Data Book and Design Guide in two application notes. Likewise, the lack of documentation for the Vectron 200 kHz oscillator's start time in the device data sheet was also a contributing factor.

Architecture Design -- The cause of the WIRE mishap was logic design error. The transient performance of components was not adequately accounted for in the design. The failure was caused by two distinct mechanisms that, either singly or in concert, resulted in inadvertent pyrotechnic device firing during the initial Pyro Box power-up. The control logic design utilized a synchronous reset to force the logic into a safe state. However, the start-up time of the crystal clock oscillator was not taken into consideration, leaving the circuit in a non-deterministic state for a significant period of time. Likewise, the startup characteristics of the FPGA were not considered. These devices are not guaranteed to follow their "truth table" until an internal charge pump "starts" the part and the uncontrolled outputs were not blocked from the pyrotechnic devices' driver circuitry.

Verification -- There was no system level end-to-end test with live pyrotechnic devices. The absence of this test coupled with the low fidelity simulators may be considered a contributing factor to the mishap. There has been no evidence of any component failure. Another contributing factor to the mishap was the lack of fidelity of the Electro Explosive Device (EED) Simulator. This device does not accurately simulate a pyrotechnic device used in the WIRE instrument. This possibly prevented a large current transient from being registered on the power input lines during spacecraft test. Additionally, the EED Simulator does not provide adequate information about all input signals capable of firing a pyrotechnic device.

3.5 Genesis

3.5.1 System Failure Scenario

Genesis was one of NASA's Discovery missions, and its purpose was to collect samples of solar wind and return them to Earth. Launched on 8 August 2001, Genesis was to provide fundamental data to help scientists understand the formation of our solar system. Analysis of solar materials collected and returned to Earth would give precise data on the chemical and isotopic composition of the solar wind. On September 8, 2004 the Genesis sample return capsule drogue parachute did not deploy during entry, descent, and landing operations over the Utah Test and Training Range. The drogue parachute was intended to slow the capsule and provide stability during transonic flight. After the point of expected

drogue deployment, the sample return capsule began to tumble and impacted the Test Range, at which point vehicle safing and recovery operations began.

3.5.2 Failure Cause

The mishap cause was the G-switch sensors, which were in an inverted orientation per an erroneous design, and were unable to sense sample return capsule deceleration during atmospheric entry and initiate parachute deployments. [7]

3.5.4 Systems Engineering Process Deficiencies

Project Planning -- A lack of involvement by NASA and Systems Engineering in vendor spacecraft activities led to insufficient critical oversight that might have identified the key process errors that occurred at the vendor during the design, review, and test of the spacecraft. However, this process was consistent with the Faster, Better, Cheaper philosophy of the time and approved of by the Discovery Program.

Project Assessment -- All levels of review, including the Genesis Red Team review, failed to detect the design or verification errors.

Architecture Design – The Genesis project team made a number of errors because of their belief that the G-switch sensor circuitry was a heritage design. Further, the prevalent view that heritage designs required less scrutiny and were inherently more reliable than new designs led to the mishap.

Verification -- Several issues led to the lack of proper testing of the G-switch sensors, including a failure to treat the G-switches as sensors, which ultimately led to the mishap. The Investigation Board recommended review and verification of heritage designs to the same level expected of new hardware/software.

3.2.6 Lewis

3.6.1 System Failure Scenario

The Lewis spacecraft was launched on 23 August 1997, with the goal of demonstrating advanced science instruments and spacecraft technologies for measuring changes in Earth's land surfaces. The spacecraft entered a flat spin in orbit that resulted in a loss of solar power and a fatal battery discharge. Contact with the spacecraft was lost on 26 August, and it then re- entered the atmosphere and was destroyed on 28 September.

3.6.2 Failure Cause

Minor rotational perturbations, possibly due to small imbalances in the forces produced by the spacecraft's attitude control thrusters, caused the Lewis spacecraft to enter a spin. Subsequent excessive thruster firings, caused by the spacecraft autonomous attempts to control in the intermediate axis mode, were sensed by the spacecraft processor which then disabled the A-side thrusters and had switched control from A-side processor to B-side processor. Excessive thruster firings on the B-side then caused the B-side thrusters to also be disabled by the processor, leaving the spacecraft uncontrolled. The single two-axis gyro was saturated, and the spacecraft was then in free drift that resulted in rotation about the principal axis, off pointing the solar array from the Sun and leaving the spacecraft without the ability to recharge its batteries. This sequence of errors, combined with the assumption that a small crew could monitor and operate Lewis with the aid of an autonomous safehold mode, even during the initial operations period, was the primary causes of the mission failure. [8]

3.6.3 Systems Engineering Process Deficiencies

Requirements Analysis -- The requirements were driven by the accommodations needed for the scientific payload that included the first spaceflight version of a hyperspectral imager. The spacecraft subsystems, for the most part, had challenging performance requirements, in such areas as pointing accuracy and thermal control, resulting in a relatively complex design.

Architecture Design -- The control system design was based on a TOMS heritage design, and was analyzed using tools developed for the TOMS program in spite of significant differences that existed between the TOMS and Lewis configurations and requirement sets. The Lewis control subsystem design was more complex than TOMS. Lewis aligned its intermediate/unstable axis of inertia toward the sun while TOMS pointed it principal/stable axis of inertia toward the sun in safe mode. TOMS measured rates in 3 axes, while rate information about the intermediate axis was deemed unnecessary for Lewis. Additionally, time out logic was used to disable the Lewis thruster electronics whenever the processor detected an "excessive" string of thruster firings, a feature was included to preclude an inadvertent vehicle spin-up and to preserve fuel.

Verification -- The simulation that was used to validate the ACS Safe Mode was flawed. The ACS design heritage was initially based on the proven Total Ozone Mapping Spacecraft (TOMS) design. The expected system performance was then analyzed using tools developed for the TOMS program. In fact, the Lewis control subsystem design was significantly more complex than TOMS because the Lewis spacecraft aligned its x-axis (intermediate/unstable), rather than its z-axis (principal/stable) of inertia toward the sun in Safe Mode. When a Lewis design modified version of the TOMS simulation was run, neither a thruster imbalance nor an initial (albeit small) spin rate about the intermediate (roll) axis was modeled. An additional factor was that the simulation was done using

mission mode parameters, not low earth transfer mode parameters that represented the condition that the spacecraft was actually in at the time of these operations. The mission mode represented a more stable attitude control condition because of lower drag forces.

Transition -- The flat spin began in low earth transfer mode before mission mode was achieved. It occurred in a period when the ground stations were available but the operations crew had not yet returned to work and therefore went initially unnoticed. By the time of the discovery of the anomaly, the battery was in a deep depth of discharge (approximately 72%). At the next and final contact pass (in this low earth orbit the ground station contact times are on the order of about five minutes each), the depth of battery discharge was 82%. In preparing for this pass, the operations crew working under extreme time pressure developed a recovery plan. At the start of the contact pass the B-side thrusters were enabled by ground command, and three, one-second thruster pulses were commanded in an attempt to arrest the spacecraft rotation rate. As it turns out only the first of the three commands was executed by the dying spacecraft because the operations crew had addressed the second and third commands incorrectly. The spacecraft went out of ground station contact and was subsequently never reacquired.

3.7 Demonstration of Autonomous Rendezvous Technology (DART)

3.7.1 System Failure Scenario

On 15 April 2005, the Demonstration of Autonomous Rendezvous Technology (DART) spacecraft was successfully deployed from a Pegasus XL rocket launched from the Western Test Range at Vandenberg Air Force Base, California. The intent of DART was to demonstrate that a pre-programmed and unaided spacecraft could independently rendezvous or meet up with a non-maneuvering and cooperating satellite. The MUBLCOM satellite was DART's rendezvous target. The Defense Advanced Research Projects Agency had deployed MUBLCOM in 1999, and it remained in orbit in good operational condition following completion of its original and primary mission. During the DART mission, all went as expected throughout the launch and early orbit phases. The vehicle successfully completed its rendezvous phase as well, placing itself into a second staging orbit about 40 kilometers behind and 7.5 kilometers below MUBLCOM, even though ground operators began to notice an irregularity with the navigation system. Less than 11 hours into the mission, DART collided with MUBLCOM. MUBLCOM did not appear to experience significant damage, and the impact actually pushed it into a higher orbit. Subsequently, DART's departure and retirement phase proceeded per the original plan, and MUBLCOM regained its operational status after an automatic system reset that resulted from the collision.

3.7.2 Failure Cause

The DART navigational system used two types of sensors – Global Positioning System (GPS)-based sensors to determine position and velocity relative to MUBLCOM, and when in close proximity to the MUBLCOM, the Advanced Video Guidance Sensor (AVGS) to collect precise navigational data. As DART approached MUBLCOM, it overshot an important waypoint, or position in space, that would have triggered the transition to close proximity operations. Because it missed this critical waypoint, the AVGS was never activated to supply DART's navigation system with accurate measurements of the range to MUBLCOM. Consequently, DART continued to steer towards MUBLCOM on the basis of the GPS-based navigational data, but it was not able to accurately determine its distance to MUBLCOM. DART's collision avoidance system eventually activated 1 minute and 23 seconds before the collision, but the inaccurate perception of its distance and speed in relation to MUBLCOM prevented DART from taking effective action to avoid a collision. [9]

3.7.3 Systems Engineering Process Deficiencies

Requirements Analysis -- DART was not designed to receive commands from the ground, an approach considered philosophically consistent with the objective that DART be a demonstration of autonomous technology. However, this indirectly led to mission failure as the ground operators were unable to take corrective action when they noticed the irregularities in the navigation system – they could only sit and watch events unfold.

Architecture Design -- In DART's case, the vendor carried over many of DART's design features from the Pegasus launch vehicle approach. For example, the software architecture, which consisted primarily of a pre-programmed, timed sequence of fixed commands, worked adequately for a launch vehicle, but was not able to respond adaptively while performing autonomous in-space operations with unanticipated inputs.

4. Discussion

A summary of the systems engineering process deficiencies identified in the preceding section is shown in Figure 3. For the seven system failures reviewed, it is observed that deficiencies were observed in 8 of the 18 Project and Technical processes. Of these, three processes were dominant:

- Project Assessment (4 deficiencies)
- Architecture Design (6 deficiencies)
- Verification (6 deficiencies)

The remaining five processes had only one or two deficiencies identified. In the case of Project Assessment deficiencies, two deficiencies dealt with failure to

address discrepancies that arose during testing that would have identified the causal factor of the eventual system failure, and the other two deficiencies pertained to failure to thoroughly analyze a portion of the design owing to its presumed heritage. It is noted that two of these instances pertain to Architecture Design, and the other two pertain to Verification. In the case of the deficiencies in Architecture Design, five of the six identified deficiencies were associated with inadequate rigor applied to the design and analysis of elements of the architecture that utilized heritage hardware and software; the other deficiency was a failure to sufficiently analyze power-up transient behavior of control electronics. In the case of Verification deficiencies, five of the six identified deficiencies were associated with the utilization of heritage hardware and software; the heritage of these architectural elements led to decisions to verify configurations that differed from the flight configuration, to verify to parameters that differed from the flight parameter values. The other Verification deficiency was attributable to a failure to verify a critical item of Ground Support Equipment that was being used in the flight system verification.

	Systems Engineering Processes							
Project	Project Planning	Project Assessment	Information Management	Requirements Analysis	Architecture Design	Implementation	Verification	Transition
Ariane 5		X			X		X	
Hubble Space Telescope		X				X	X	
Mars Observer	X		X		X	X	X	
WIRE		X	X		X		X	
Genesis	X	X			X		X	
Lewis				X	X		X	X
DART				X	X			

Figure 3. Summary of Observed Systems Engineering Process Deficiencies

The foregoing analysis suggests the following general observations:

1. The utilization of heritage hardware and software in a project, while perceived to reduce development risk due to its heritage, introduces risks that will manifest themselves as deficiencies in the Architecture Design and Verification processes.

2. The Project Assessment process must be acutely aware of addressing discrepancies during testing, as these may be harbingers of things to come. Likewise, the process must resist temptation to lessen design and analysis rigor when dealing with heritage hardware and software.

3. The systemic recurrence of specific systems engineering process deficiencies associated with the utilization of heritage hardware and

software suggests a weakness of Enterprise Processes, the Quality Management process in particular.

5. Remarks

For this study, a sampling of seven space system failures was assessed – one launch vehicle and six spacecraft of varying scales, developed and launched over a span of two decades, which encountered a variety of failures. One recurrent theme in the system failures was problems associated with utilization of heritage hardware. Whether this is an artifact of a small sample or a valid conclusion is not clear at this time, and suggests the need for a more comprehensive study of this phenomena.

However, even given the small sampling of projects assessed in this study, it can be concluded that a lack of rigor in execution of systems engineering processes is a contributing factor to the cited instances of space systems failures. Since all the system failures studies were mission performance failures, it should not come as a surprise that the systems engineering processes dealing with design and test were revealed to be deficient most frequently. These were projects that, after all, had survived the "infant mortality" phase of space systems, the concept and early development phases when so many space systems are cancelled due to spiraling cost estimates or the inability to form a cohesive and stable stakeholder base. Had a sampling of "programmatic" failures been included, doubtless the composition of systems engineering process deficiencies would have changed – but that is a subject for another paper.

As a footnote, it is noteworthy that although in each case discussed in this paper the original mission was a complete failure, in three cases the persistence and ingenuity of the project team allowed eventual mission success. For instance, despite failure in its maiden flight, the Ariane 5 is now flying successfully. A repair mission corrected the flawed optics on the Hubble Space Telescope, and it has since been performing magnificently. And although the Wide-field Infrared Explorer (WIRE) spacecraft planned scientific mission was a failure, the WIRE spacecraft was recovered by the mission controllers and completed a number of asteroseismology investigations for NASA's Office of Space Science as well as acting as a technology test bed for several advanced technology projects.

6. References

[1] *IEEE Standard for Application and Management of the Systems Engineering Process,* IEEE Std 1220-2005, September 2005.

[2] *Systems engineering — System life cycle processes*, ISO/IEC 15288, International Organization for Standardization, 1 November 2002.

[3] *Ariane 5 Flight 501 Failure Report by the Inquiry Board*, European Space Agency, 19 July 1996.

[4] *The Hubble Space Telescope Optical Systems Failure Report*, National Aeronautics and Space Administration, November 1990.

[5] *Mars Observer Mission Failure Investigation Board Report*, National Aeronautics and Space Administration, 31 December 1993.

[6] *Small Explorer WIRE Failure Investigation Report*, National Aeronautics and Space Administration, Goddard Space Flight Center, 27 May 1999.

[7] *Genesis Mishap Investigation Board Report*, Volume 1, National Aeronautics and Space Administration, 30 November 2005.

[8] *Lewis Spacecraft Mission Failure Investigation Board Report*, National Aeronautics and Space Administration, 12 February 1998.

[9] *NASA Report: Overview of the DART Mishap Investigation Results - For Public Release*, National Aeronautics and Space Administration, Marshall Space Flight Center, 15 May 2006.