Learning from Other People's Mistakes

Most satellite mishaps stem from engineering mistakes. To prevent the same errors from being repeated, Aerospace has compiled lessons that the space community should heed.

Paul Cheng and Patrick Smith

The computer onboard the Clementine spacecraft froze immediately after a thruster was commanded to fire. A "watchdog" algorithm designed to stop the thrusters from excessive firing could not execute, and Clementine's fuel ran out. The mission was lost. Based on this incident, engineers working on the Near Earth Asteroid Rendezvous (NEAR) program learned a key lesson: the watchdog function should be hard-wired in case of a computer shutdown. As it happened, NEAR suffered a similar computer crash during which its thrusters fired thousands of times, but each firing was instantly cut off by the still-operative watchdog timer. NEAR survived.

As this example illustrates, insights from past anomalies are of considerable value to design engineers and other program stakeholders. Information from failures (and near failures) can influence important design decisions and prevent the same mistakes from being made over and over. Contrary to popular belief, satellites seldom fail because of poor workmanship or defective parts. Instead, most failures are caused by simple engineering errors, such as an overlooked requirement, a unit mix-up, or even a typo in a document. Such blunders are often too subtle to be caught during routine verification and review because flight items can only be checked against known requirements. But by understanding the types of errors that can occur, mission planners can adjust their review processes to more effectively screen them out.

> *"It's always the simple stuff that kills you.... With all the testing systems, everything looked good."*
>
> *—James Cantrell, main engineer for the joint U.S.-Russian Skipper mission, which failed because its solar panels were connected backward (Associated Press, 1996)*

The Engineering Database

As part of its overall mission assurance role, Aerospace investigates anomalies on national security space missions and uses this information to help program managers improve their systems and processes. Throughout the development, launch, and deployment of defense space systems, engineers at Aerospace analyze all anomaly reports to identify engineering errors, workmanship escapes, model deficiencies, test inadequacies, defective parts, hostile environments, and other factors that may warrant correction. These notes are all collected in a comprehensive database, which now archives more than 20,000 launch vehicle and satellite anomalies that occurred in the factory, at the launch site, during flight, and in operation.

For example, all off-nominal telemetry readings from a launch are logged into the database. A technical expert is assigned to explain each anomaly and why it occurred. Every investigation is tracked by the program office as part of Aerospace's flight-readiness verification responsibilities. Fault-management lessons, reliability impacts, and suggested corrective actions are documented.

This rigorous postflight analysis was largely prompted by the failure in 1999 of a Defense Support Program launch (DSP-19). The transfer vehicle for this mission, the Inertial Upper Stage (IUS), used thermal tape on the connector plug housing. The thermal tape was supposed to be wrapped with enough clearance to let the harness disengage. The assembly instructions stated that the tape shall be applied "within 0.5 inches of the mounting bracket flange" (instead of, for example, "no closer than 0.5 inches and no farther than 1.0 inch"). Unaware that the parts had to unfasten, and thinking that they should wrap the tapes as tightly as possible, the technicians applied the tapes so close to the flange that the separator jammed, preventing stage separation. Afterwards, engineers reviewing the telemetry from previous IUS flights realized that the connector had jammed every time. In fact, seven of the previous flights dodged failure only because the taped connectors were jerked apart when they hit the allowable stops. Unfortunately, the warning signs in the telemetry data were not heeded.

Gleaning Lessons from Failures

To disseminate information like this in a format that is readily absorbed, Aerospace began publishing a series of "Space Systems Engineering Lessons Learned" bulletins. To date, more than 120 of these papers have been produced. Each tells a failure story, spotlighting three questions: How did the mistake occur? What prevented its detection? Why did it bring down the entire system?

The reports also spell out the specific practices that could have prevented the mishap. For example, one lesson traced an instrument's power-supply malfunction in space. The problem was not caught on the ground because the test set supplied backup power, and would have been avoided if the test equipment provided metering to show the unit was unexpectedly drawing current from it. In another case, a payload was damaged during thermal vacuum testing because the test cable, which was not space-qualified, induced multipaction breakdown. Obviously, tools used in simulated space environments must be space-rated.

To help reviewers increase the odds of finding mistakes, Aerospace also parsed through the lessons-learned bulletins to generate a reviewer's checklist titled, "100 Questions for Technical Review." Reviewers will more than earn their pay if just one of these items gets the response: "You know, we hadn't thought about that—we'd better check it!"

For example, one review question asks, "Has the analyst inspected the actual hardware?" This question derives from an incident where a satellite failed because the solar-array thermal model did not account for the harnesses that prevent heat dissipation; one look at the hardware would have made it obvious. Another asks, "Have designs been analytically established before testing?" This is prompted by an incident in which the spacecraft tumbled after deployment. The magnetic torque rods were set during test instead of being determined by analysis. Because the test engineer did not realize the Earth's magnetic North pole is in Antarctica, he set the phasing wrong.

US Air Force

The Genesis capsule, embedded in the sands of Utah after its parachute system failed to deploy in September 2004. The Mishap Investigation Board said the likely cause was a design error involving the orientation of gravity-switch devices.

Five Common Mistakes to Look Out For

The most common mistakes are gathered in an Aerospace report, "Five Common Mistakes Reviewers Should Look Out For." This document, available to U.S. government agencies and contractors, includes the "100 Questions" and all the lesson bulletins published through June 2007.

This report recommends that reviewers first ponder, "Could the sign be wrong?" Sign errors involving orientation and phasing (polarity) of equipment have caused numerous failures. For example, the TERRIERS spacecraft was lost because one of its torque coils had to be installed with a phase opposite of that of the other two coils—but the change was not incorporated into flight software. Likewise, a design mistake caused the solar panels on the Skipper spacecraft to be connected backward. During system integration, the magnitude of current flow between the solar array and battery was verified, but not the direction. Once in orbit, the battery drained, scuttling the mission.

More recently, the Genesis spacecraft crashed. Genesis brought back solar-wind samples. During reentry, aerodynamic deceleration was supposed to cause a parachute to unfold. Genesis's avionics design was derived from an earlier mission, Stardust, but was too complex to fit into Stardust's one-box package. Genesis designers cut and pasted Stardust's schematic into a new two-box design. The Stardust unit had been spin tested, but the two boxes in Genesis were difficult to spin together. So, believing that the design was flight proven, Genesis engineers simply verified the new assembly "by inspection." Unfortunately, nobody knew that a pair of deceleration sensors were direction-sensitive—even the Stardust drawings did not flag any polarity constraint. The sensors were by chance turned sideways in the new layout, and as a result, the parachute could not open.

The second major questions is, "How will last-minute configuration changes be verified?" Satellites are frequently modified after factory testing. Placeholder blankets need to be swapped out, flight connectors mated, and databases updated. Some items, such as brackets to secure hardware during ascent, have to be installed at the launch site. Nonflight items such as test plugs and dust covers must be removed. Late changes, especially those made in the heat of a countdown, have caused several failures, in

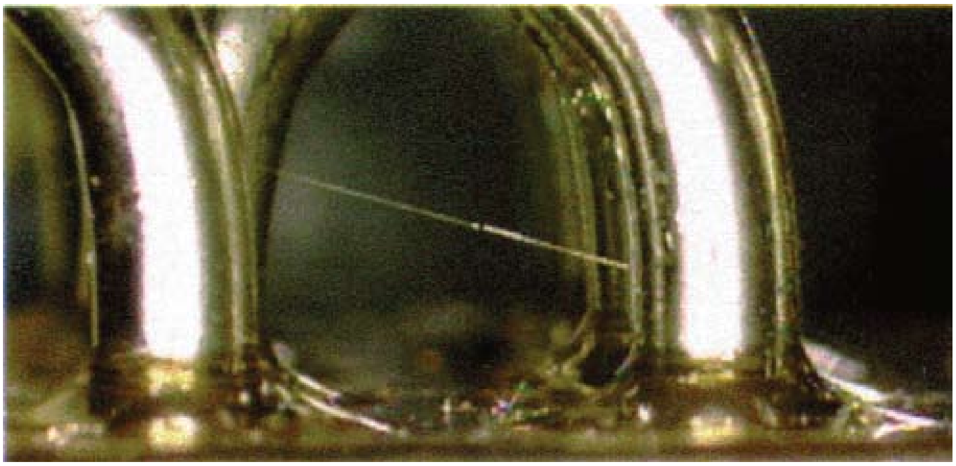
U.S. Government Satellite and Launch Vehicle Failures from 1990 to 2006

These charts show U.S. Government satellite (below) and launch vehicle (facing page) failures from 1990 to 2006. In contrast to common beliefs, mishaps are seldom due to poor workmanship, defective parts, or environmental factors. Notice that engineering mistakes—such as a typo (Contour), an overlooked requirement (MPL), and a unit mix-up (MCO)—caused the majority of failures. These embarrassing snafus are often so subtle that they are not caught during verification and review.

> *"Space is unforgiving; thousands of good decisions can be undone by a single engineering flaw or workmanship error, and these flaws and errors can result in catastrophe."*
>
> — Defense Science Board, 2003

Launch Date	Program	Cause	Engineering Mistake	Technology Surprise
04/90	Hubble	Flawed manufacturing equipment misshaped the mirror. Quality assurance relied on the same equipment and could not perceive the mistake, even though independent checks indicated a defect.	x	
07/92	TSS-1	Tether deployment mechanism was jammed by a bolt added at the launch site.	x	
09/92	Mars Observer	Corroded braze jammed a regulator, causing a breach of the propulsion line. The regulator was not qualified for a long mission.	x	x
08/93	NOAA 13	The battery charger was shorted by a long screw. The unit had low dimensional tolerance and experienced thermal stress outside its qualification envelope.	x	
10/93	Landsat F	Pyrovalve exploded because of an unpredictable mechanism.		x
01/94	Clementine	Computer froze, disabling the fault management software. Patches were prepared but not uploaded.	x	x
05/94	MSTI 2	Cause unknown, attributed to impact by micrometeoroid or debris, or charging.		x
12/95	Skipper	Solar array was miswired. Test did not ascertain current direction.	x	
02/96	TSS-1R	Contamination in the tether caused arcing.	x	
08/97	Lewis	Satellite lost orientation. The stability simulation was performed for the wrong orbit, the attitude control was improper, and the satellite was not monitored.	x	
10/97	STEP-4	Satellite/launcher resonance caused vibration damage. The resonance was known but not addressed.	x	
10/98	STEX	Solar array failed—analysis was run on wrong configuration.	x	
12/98	MCO	Burned up because of unit mix-up in the navigation software.	x	
01/99	MPL	Requirement flowdown omission resulted in premature shutdown of descent engine.	x	
03/99	WIRE	Unexpected start-up transient fired pyros, causing the loss of all cryogen. The inhibit circuit design was flawed, and inappropriate test configuration masked the chip's misbehavior.	x	x
08/01	Simplesat	Transmitter suffered arcing in vacuum.		x
11/01	Genesis	Parachute could not deploy because of deceleration sensor misorientation brought about by a layout change.	x	
07/02	Contour	Spacecraft broke up upon firing of an embedded solid motor. The plume analysis used to qualify the design by similarity relied on an AIAA paper that had a critical typo.	x	
04/05	DART	Spacecraft failed to rendezvous with its target satellite because of a combination of GPS software bug and flawed navigation design.	x	x

Launch Date	Program	Cause	Engineering Mistake	Technology Surprise
03/90	Titan III	Satellite could not be deployed because the pyro separation harness routed firing commands to ports different from those specified by software.	x	
04/91	Atlas I	Air was sucked through a defective checkvalve during ascent and froze in a turbopump, preventing a Centaur engine from starting. Air was able to get in because of design changes that were not accompanied by reanalysis.	x	x
07/91	Pegasus	The shaped-charge detonation system caused an incomplete cut, thwarting stage separation.		x
08/92	Atlas I	Same as 04/91 Centaur failure.	x	x
03/93	Atlas I	A set screw in the throttle setting regulator came loose in flight because rework instructions neglected to require retorquing and verification.	x	
08/93	Titan IV	A propellant cut made during a repair opened in flight, instead of staying closed by internal pressure as expected, exposing the insulation to flame.		x
06/94	Pegasus XL	Vehicle lost control because of unexpected aerodynamic load. The control loop design used improper variance analysis.	x	x
06/95	Pegasus XL	Second-stage nozzle could not gimbal because of an incorrectly installed foam skid. Operator concerns regarding foam skid orientation were ignored because there was no specific orientation requirement.	x	
08/95	Delta II	Wind shear caused a large deflection, damaging the explosive transfer line and preventing one of the strap-on motors from separating.		x
08/95	Athena I	Thruster-vector-actuation hydraulic oil dumped overboard was ignited by the rocket's plume, damaging a control cable. The failure mode occurred on a similar European rocket earlier. Also, high-voltage power supply, qualified only for aviation applications, suffered arcing at high altitude.	x	x
10/95	Conestoga	Low-frequency noise of the vehicle was not filtered out, causing the vehicle to excessively actuate the vector thruster control, eventually exhausting hydraulic oil.		x
11/96	Pegasus XL	Payload did not separate, probably because of shock damage to a battery or switch.		x
01/97	Delta II	The motor case of a solid strap-on was damaged before launch and split in flight.		x
08/98	Titan IV	Wire insulation damage on the power supply caused intermittent short of the computer. Upon power-on reset, the avionics system improperly commanded the vehicle, resulting in self-destruction.		x
08/98	Delta III	A roll mode was not accounted for in the control system, resulting in loss of control.		x
04/99	Titan IV	Upper stages could not separate because vague assembly instructions misled technicians into wrapping insulating tapes too tightly. Problems repeatedly occurred on previous flights.	x	
04/99	Athena II	Nose cone failed to separate—the operation of one set of pyros pulled open the connectors for the second set. The tolerance buildup problem was manifested in an earlier flight but was not addressed.	x	x
04/99	Titan IV	A roll filter manually entered in the transfer vehicle's avionics database had an exponent error (effectively misplaced a decimal point) which caused control loss.	x	
05/99	Delta III	A defective brazing in the combustion chamber was noticed during inspection but was allowed to pass because of ambiguous drawing instructions. The chamber breached in flight.	x	
09/01	Taurus	An attitude-control valve was jammed shut by debris. The problem had occurred several times previously but was deemed acceptable because the valves had always fortuitously stuck in nonfatal positions.	x	
12/04	Delta IV Heavy	The first stage shut down too early because cavitation spoofed the sensors into reacting to phantom fuel depletion.		x
03/06	Falcon	A fuel leak, attributed to a corroded B-nut, started a fire.		x



Courtesy of NASA

Pure tin plating can grow conductive whiskers. Numerous satellites have been disabled because whiskers inside the control processors caused plasma arcing that blew the fuses.

part because late installations and removals can be difficult to verify.

An eleventh-hour analysis of the Tethered Satellite System (TSS) uncovered a structural shortfall in the deployment mechanism, making it necessary to add a bolt. The original design engineer, thousands of miles away, could not see firsthand how the modified hardware fit. The bolt protruded against a traveling ball nut, and consequently, TSS could not deploy.

The third question asks, "Can the vehicle survive a computer crash?" As the Clementine and NEAR examples illustrate, computer errors—often caused by subtle timing or memory glitches—have affected several missions. Every vehicle should be able to gracefully handle a computer fault by reverting to the "last known good state," rebooting without being stuck in endless reset cycles, remaining in a safe mode, recovering from low bus voltage, and so on.

The safety analysis must ensure that the fault-management system cannot be spoofed into making a wrong move, as happened to Ariane 501. This rocket's flight software carried over a function, "platform alignment calculation," from Ariane 4, even though it was no longer needed. But the 501 vehicle flew a different trajectory, creating an alignment bias too large for the legacy code to compute and resulting in an "error exception." Although the exception was irrelevant, the avionics system presumed that all faults were caused by bad hardware, and should be handled by an equipment swap. It thus halted the active controller and switched to the backup, which of course immediately encountered the same error and also shut down. The rocket needlessly self-destructed.

The fourth question asks, "Is the circuit overcurrent protection adequate?" Shorting and plasma arcing have caused many failures. Fuses, circuit breakers, and other devices are intended to protect upstream components, but careful design is required to prevent the safety devices from misbehaving. Plasma arcing most frequently results from pure tin plating that grows conductive filaments (commonly known as "whiskers"). Numerous satellites, such as Galaxy VII, had been disabled because whiskers grew on the relays of their control processors, causing arcing that blew the fuses.

The IMAGE satellite's single-string transponder was protected by a circuit breaker. Radiation tripped the breaker, turning the transponder off. Unfortunately, a design flaw caused the breaker to misreport itself as "on," which prevented the onboard fault-management software from resetting the circuit. The malfunction disabled uplink, making ground rescue impossible.

The last question is, "Can pyros cause unexpected damage?" Premature firings caused the Wide-field Infrared Explorer (WIRE) spacecraft to open its telescope cover, venting all the cryogens and ending the mission in the first orbit. The mishap occurred because a logic controller chip unexpectedly asserted all outputs during power-on. Both "arm" and "fire" inhibits were commanded by the same chip (a common design oversight), and therefore failed together.

Unexpected pyro firings have also caused the death of personnel. In 2003, a Brazilian launch vehicle was being readied for launch when an electrostatic discharge introduced an arc in the unshielded pyro circuits, setting off the initiator. Because the rocket had no safe-arm devices to contain an accidental flash, it caught fire on the pad, killing 21 workers.

Taking Heed of Lessons

Ben Franklin wrote, "Experience keeps a dear school, but fools will learn in no other." In the space business, the cost of experience is very high indeed. But learning is hampered by a widespread reluctance to share failure information. Consider a 1993 incident with the Orion rocket: Someone forgot to install a critical voltage-clamp component into the test meter used to check the solid motor's firing circuit. The full test voltage ignited the rocket, killing a technician. Only after this accident was made public did two other facilities reveal that the same meter had inadvertently set off their rockets, too!

Engineers also neglect to research lessons. The WIRE mission failed because of a start-up quirk in a logic controller, which had been previously described in NASA's "Application Notes." Similarly, a Maxus rocket crashed because drained hydraulic fluid started a fire that burned through a guidance cable. This debacle prompted several programs to redesign fluid drains or add cable insulation—but the Athena program did not do so. Athena's maiden flight subsequently suffered exactly the same failure.

Aerospace is striving to improve its process for disseminating lessons learned, most importantly by incorporating them into engineering standards, mission assurance handbooks, testing guidelines, and military specifications. References to real failures should impress upon designers the importance of following (and the risks of ignoring) a given set of guidelines, specs, or standards. Still, if the space industry is to head off further mistakes, it must overcome proprietary concerns and bureaucratic inertia to share, seek out, and heed lessons learned.

Further Reading

"Five Common Mistakes Reviewers Should Look Out For," Aerospace Report TOR-2007-(8617)-1 (The Aerospace Corporation, El Segundo, CA, 2007).

J. F. Binkley, P. G. Cheng, P. L. Smith, and W. F. Tosney, "From Data Collection to Lessons Learned—Space Failure Information Exploitation at The Aerospace Corporation," *Proceedings of the First International Forum on Integrated System Health Engineering and Management in Aerospace* (Napa, CA, November 2005).

"Report of the Defense Science Board/Air Force Scientific Advisory Board Joint Task Force on Acquisition of National Security Space Programs" (Defense Science Board, 2003).