National Aeronautics and Space Administration

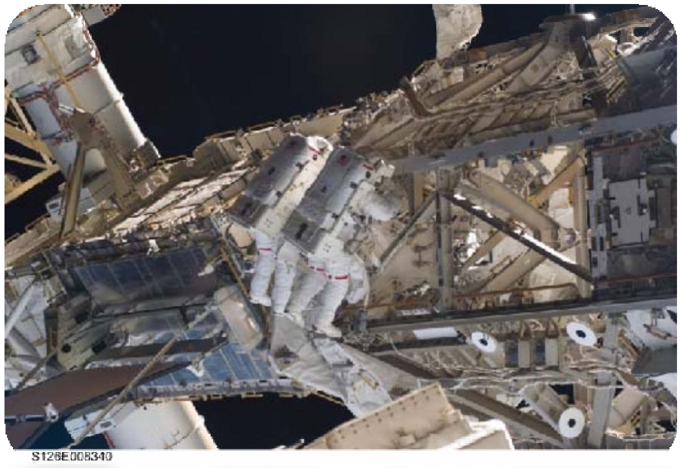
STS-126 Shuttle Software Anomaly

Leadership ViTS Meeting

May 2009

Bryan O'Connor
Chief, Safety and Mission Assurance

Jim Lloyd
Deputy Chief, Safety and Mission Assurance



This and previous presentations are archived at:
http://pbma.nasa.gov/pbma_main_cid_584

THE CLOSE CALL

STS-126 launched on November 14, 2008. When the shuttle reached orbit, two automatic functions failed:

- **S-band/Ku-band handover**

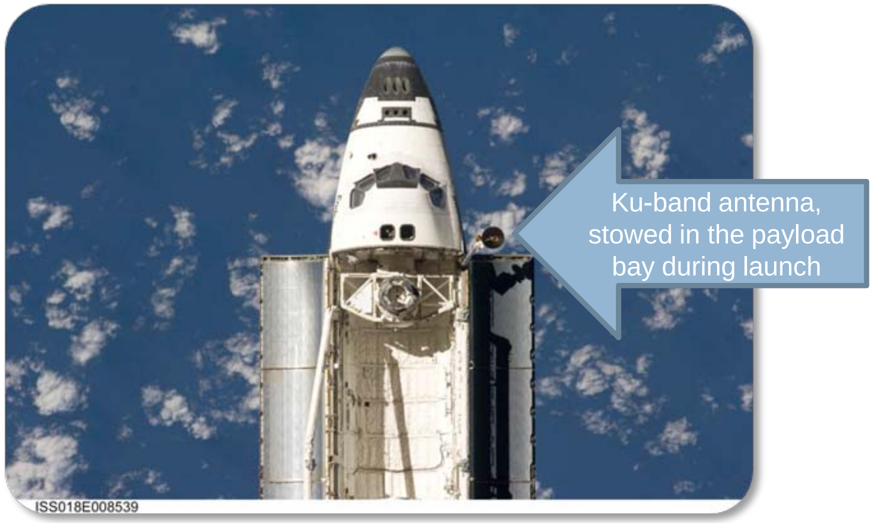
  Shuttle to ground communications that rely on radio frequencies use S-band frequencies (1,700-2,300MHz) during launch, then automatically switch to the more powerful Ku-band (15,250-17,250MHz) in orbit. This handover failed.

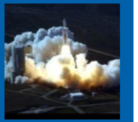
- **Payload Signal Processor (PSP) port shift**

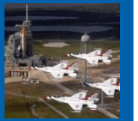
  The shuttle communicates with its payload through the PSP, which can be configured via RF link or hardwired umbilical. On STS-126, payload communications were configured for RF link during launch but failed to switch to umbilical when the shuttle reached orbit.

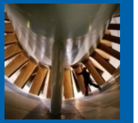
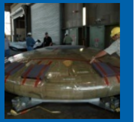
Mission control was able to manually command both the S-band/Ku-band switch and the PSP port shift.

While the software glitch did not obstruct the mission, it caught management's attention because "in-flight" software problems on the shuttle are rare.







What Happened?

The evolution of an anomaly

A software error was planted in 1989 but did not disrupt functionality until further software changes unearthed it in 2008:

OI-20* *Setting the Trap* – July 1989

- OI-20 introduced new code to the shuttle flight software. The code did not "lock down" command words.
    - The commands functioned correctly, but they would no longer function if later changes in the code shifted their alignment.
- Programmers left warning comments in the code about the need to monitor these commands over subsequent changes.

OI-29 *Camouflaging the Trap* – August 2000

- OI-29 moved the code's change history to the bottom of the code, but left the warning from OI-20 in it original location.
    - In this location, no one would see the warning without looking at the code itself. The change effectively buried the warning.

OI-33 *Falling into the Trap* – January 2007

- OI-33 introduced new code that shifted commands from their required output addresses.
- **When the shuttle reached orbit, the Ground Command Interface Logic did not receive commands to initiate PSP port moding and the S-band/Ku-band handover.**

***Operational Increment**: a flight software functional update is referred to as an Operational Increment (OI). Each OI takes about 18 months to complete and includes requirements definition, code development, system verification and mission preparation.

PROXIMATE CAUSE

A software change inadvertently shifted data in the shuttle's flight software code. Because of this shift, the software did not send configuration commands to the shuttle's Ground Command Interface Logic, and several automated functions failed.

ROOT CAUSE / UNDERLYING ISSUES

Standards and Training

- Ambiguous wording in the programming standard led programmers to believe they met the requirement.
- Programmers did not understand the potential effects of inserting or deleting data in the flight software code.

Testing

Handover functions were not tested rigorously:

- OI-33 did not modify the code for the functions that failed on STS-126, so testing did not address these functions.

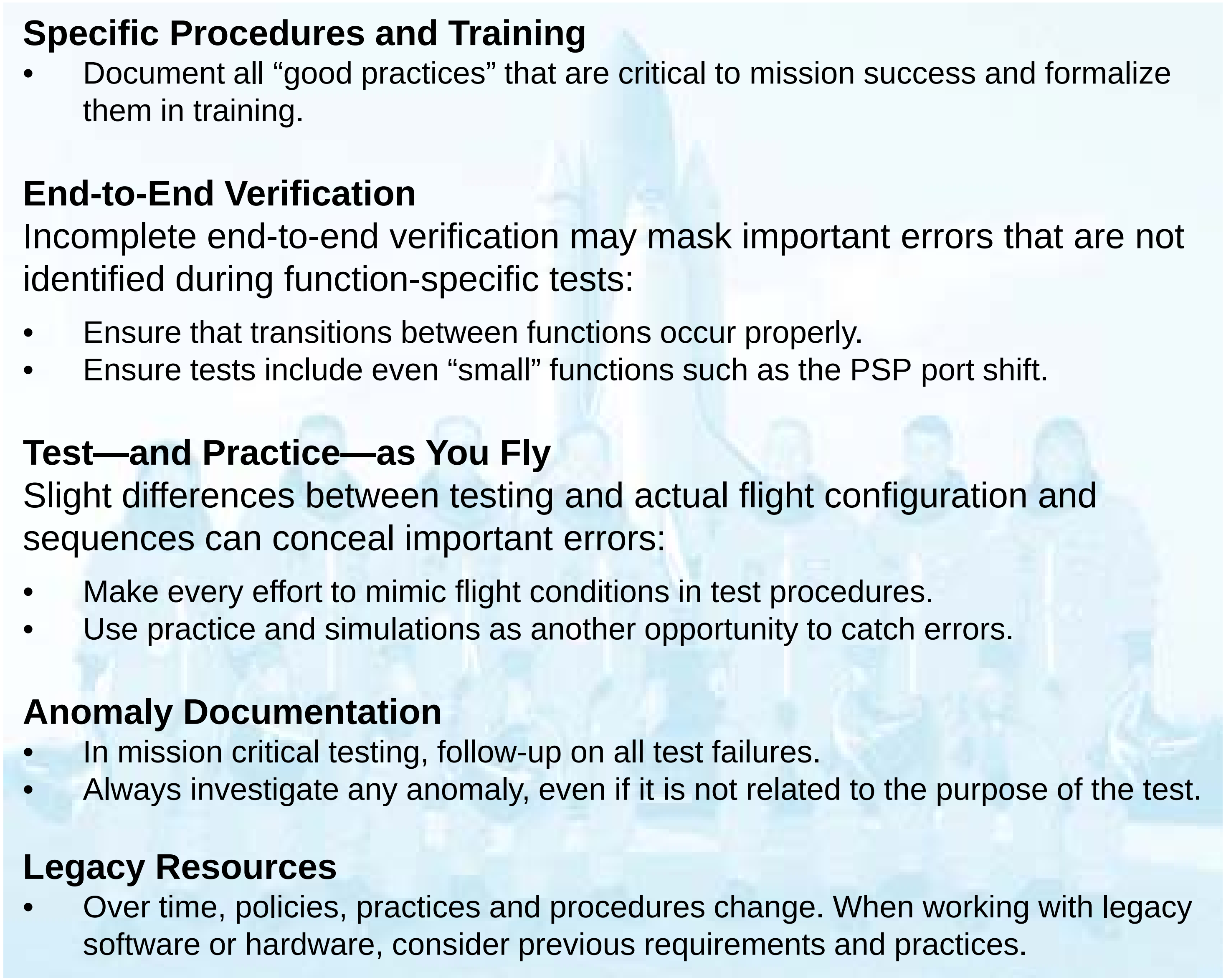
Testing the Payload Signal Processor port switch:

- The mission required only one PSP port shift, when the Shuttle transitioned from launch to orbit configuration.
- Integration tests started in the "on-orbit" configuration. Because the STS-126 payload only used the umbilical, tests did not address the PSP port shift.

Testing the S-band/Ku-band handover:

- False positive: During core testing, a handover was successful because the test isolated the command rather than testing it as part of a series of commands.
- The S-band/Ku-band handover was part of the preliminary setup for several additional tests. When the handover failed, personnel switched the configuration manually and continued tests without documenting the anomaly.
    - A history of handover problems unrelated to flight software led personnel to believe the software was functioning properly.
    - The false positive from the core testing affirmed their interpretation.
- No IV&V for non-critical software.

FOR FUTURE NASA MISSIONS



Specific Procedures and Training

- Document all “good practices” that are critical to mission success and formalize them in training.

End-to-End Verification

Incomplete end-to-end verification may mask important errors that are not identified during function-specific tests:

- Ensure that transitions between functions occur properly.
- Ensure tests include even “small” functions such as the PSP port shift.

Test—and Practice—as You Fly

Slight differences between testing and actual flight configuration and sequences can conceal important errors:

- Make every effort to mimic flight conditions in test procedures.
- Use practice and simulations as another opportunity to catch errors.

Anomaly Documentation

- In mission critical testing, follow-up on all test failures.
- Always investigate any anomaly, even if it is not related to the purpose of the test.

Legacy Resources

- Over time, policies, practices and procedures change. When working with legacy software or hardware, consider previous requirements and practices.