

Ground Software Errors Can Cause Satellites to Fail too- Lessons Learned

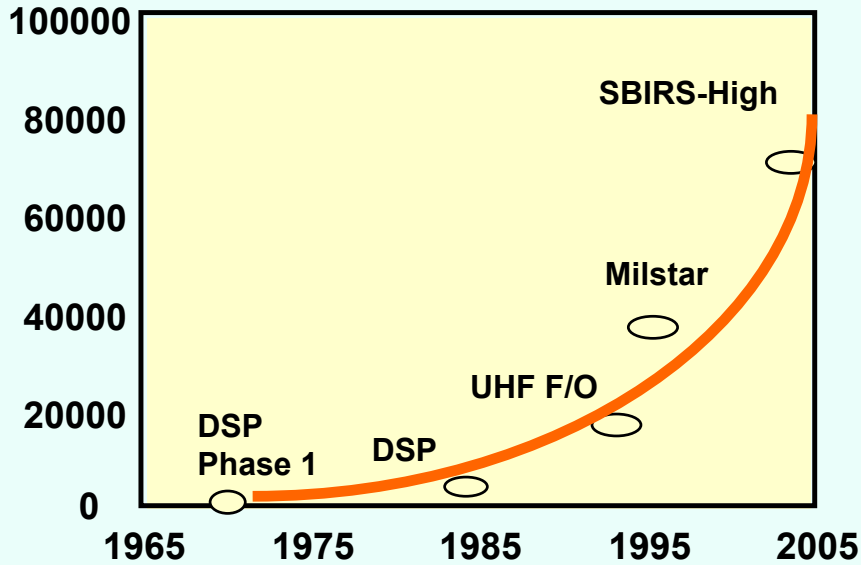
4 March 2003

Paul G. Cheng

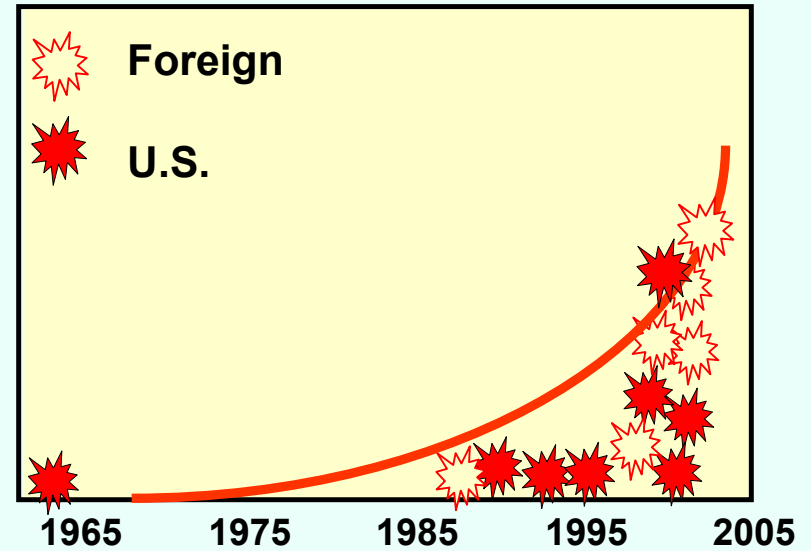
Risk Assessment & Management Subdivision
Systems Engineering Division

Software Increasingly Matters

FSW SLOC Count



SW-Related Failures



Over half of failures between 1998 and 2000 involved software

FSW SLOC = Flight Software Source Lines of Codes

1962	Mariner 1 (Atlas)	1988	Phobos 1
1990	Intelsat 6 (Titan CT2)	1996	Cluster (Ariane 501)
1991	Orbcomm X	1998	SOHO
1994	Clementine	2000	STRV x 2
1999	Milstar 2-1	2000	ICO F1 (Sea Launch)
1999	MCO	2000	QuickBird (Cosmos 3M)
1999	Terriers		
1999	MPL		

NEAR and Phobos 2 not counted

Software Mistakes Are Underappreciated

- Small error can be fatal
- Redundancy ineffective
- Risks do not necessarily decrease over time
- Involves more human factors

- Imperative to make software more robust

SE Problems Caused Most Major SW Anomalies

- Incomplete requirement implementation
 - Mars Polar Lander, Space Technology Research Vehicle (STRV)
- Improper software changes or code reuse
 - Ariane 501, Solar & Heliospheric Observatory (SOHO)
- Inadequate configuration management process
 - Terriers, Titan CT-2
- Mistakes are all too often repeated

We do not invent new mistakes, we just repeat old mistakes.

Dr. Bill Ballhaus (CEO, The Aerospace Corp.)

Aerospace's Space Systems Engineering Lessons Learned System

- Broadly scoped
 - Uses actual mishaps to concisely highlight common threads among failings
- Publish lessons each quarter
 - Software is a recurring theme
 - Widely distributed to external community

Examples of Fatal Ground Software Error

Mars Climate Orbiter Failure

- Thruster firing model, reused from a previous mission, was in metric. Thruster vendor supplied data in English units.
- In the previous mission, engineers correctly inserted a 4.45 factor to convert lb-force to Newton.
- A new thruster was used, and the vendor's new (English unit) equation was pasted into the model **without the 4.45 factor**:
 - Spec was overlooked
 - Original code had no warning remarks
 - **Ground software** viewed as non-critical
 - Truth table, manually computed, had the same mistake
 - Tests not thorough

Compounded by GN&C inadequacy, mistake turned deadly

More Examples

Sea Launch F3 (ICO) Failure*

- Need to launch at a particular time - Time variable changed name; from time_A to time_B
- Change affected the **ground software** controlling a valve
- Before Change:
“If the state is Abort (or countdown proceeds past time_A = X), close Valve A”
- Should be:
“If the state is Abort (or countdown proceeds past time_B = X), close Valve A”
- As Coded:
“If the state is Abort, close Valve A”

Valve kept open - Launch Failed

Lessons

- *Errors in ground software can be fatal too.*
- *Validate **mission-critical** element **changes** with **more** vigor than the original development.*

Fools say that they learn by experience. I prefer to profit by others' experience.

Otto Bismarck