Agenda

- Introduction
- Dedication To Safety
- Space Shuttle Flight Software Period Themes
- Space Shuttle Flight Software Accomplishments
- Look At Improvements Through Latent Product DRs
- Space Shuttle Flight Quality Measurements
- Space Shuttle Flight Software Reliability
- Space Shuttle Flight Software Lessons Learned
- Summary
- Acronyms
Introduction

- This presentation focuses on the Space Shuttle Primary Avionics Software System (PASS) and the people who developed and maintained this system.
  - One theme is to provide quantitative data on software quality and reliability over a 30 year period
  - Second theme is to focus on the people and organization of PASS

- Quality Measures
  - Pre-build Detection Effectiveness (Inspection Plus Development Test)
    - Errors found by Inspection and Development Test Pre-Build (prior to being placed under project configuration control) divided by total errors
  - Verification Effectiveness
    - Process DRs divided by (Process DRs plus Product DRs)
  - Product Error Rate
    - Product DRs divided by new, changed, deleted source lines of code. Includes only non-comment source lines of code.
Introduction

- Quality Measures
  - Consistent data relates to “code break” discrepancies
    - Requirements were supplied from external sources
  - Errors counted in three periods
    - Errors found by Inspection and Development Test Pre-Build (prior to being placed under project configuration control)
    - Process DRs found Post Build until a milestone called Software Readiness Review (SRR) for the first flight off that increment; typically occurs approximately 4 weeks prior to flight
    - Product DRs found from SRR of first flight until end of program
      - Subset of Product DRs are those which occur in either terminal countdown or in flight, called in-flight DRs
    - Additional special category of DRs are called Released Severity 1 DRs. These may be process or product DRs. These are DRs that could cause loss of crew or vehicle that are released to any field site such as the Shuttle Mission Simulator (SMS), the vehicle at KSC, or the Shuttle Avionics Integration Lab (SAIL).
Introduction

- Common themes running through lifecycle periods
  - Improvements through process enhancements
  - Improvements through automation
  - Defect removal following identification of significant process escapes
  - Impact of workforce instability
  - Early evaluator, adopter, and adapter of state-of-the-art software engineering innovations

- A significant contributor to the success of the PASS FSW organization has been the support of the NASA PASS software customers that have consistently valued quality and supported reasonable implementation schedules. NASA has also supported maintaining critical skill staffing.
Dedication To Safety
Dedication To Safety

- Developing complex human-rated flight software is a major technical challenge.
  - Perfection required to achieve the desired level of safety
    - Extremely difficult to accomplish, but can be aggressively pursued
  - Keys to the pursuit of perfection
    - Principles of Providing High Reliability Software
    - Continuous Process Improvement
    - Defect Elimination Process
Principles of Providing High Reliability Software

- Safety certification is currently based on process adherence rather than product.
- Assumption is that a known, controlled, repeatable process will result in a product of known quality.
- Process executed by personnel that are committed to safety and skilled relative to processes, system architecture, and specialized software requirements.
- Team skills and workload closely monitored by management to prevent over commitment that could result in quality breakdowns.
- Use “trusted” tools to develop, build, release and maintain the software.
- Use measurements to continuously assess the health of both the process and the product.
- Relationship between quality and reliability must be established for each software version and statistically demonstrated for the required operational profiles.
- Quality must be built into the software, at a known level, rather than adding the quality after development.
  - You cannot test quality into software
Examples Of Continuous Process Improvement

- Error Counting, Metrics
  - Formal Process
    - Structured flows
    - Formal software inspections
    - Formal inspection moderators
    - Formalized configuration control
  - Inspection improvements
    - Configuration Management Data Base (CMDB)

- Error Cause Analysis
  - Process Analysis
    - Process evaluation
    - Technical exchange seminars
    - Process maturity measurements
    - Formalized training
    - Reliability modeling
  - Reliability/complexity research
  - Development test philosophy
    - Inspection process tools
    - Formalized lessons learned
  - Structured education planning
  - Formalized lessons learned
  - Inspection process tools

- Quality Management and Process Enhancement
  - Process document assessment
  - Software development environment
    - Formalized milestone risk assessments
    - Concurrent engineering process model
    - Process improvement database
    - Project manager process reviews
  - ISO 9001 certification
  - Oracle based CMDB upgrade
    - Enhanced test tools
    - LAN based development environment
    - Formalized milestone risk assessments

- CE, SPC collaboration
  - On-line programming standards
  - Web-based troubleshooting guide
  - Re-engineering lessons learned
  - Support software inspection process automation
  - Formalized milestone risk assessments
    - Concurrent engineering process model
    - Process improvement database
    - Project manager process reviews
  - ISO 9001 certification
  - Oracle based CMDB upgrade
    - Enhanced test tools
    - LAN based development environment
  - CE, SPC collaboration

Timeline:
- 1976
- 1977
- 1978
- 1979
- 1980
- 1981
- 1982
- 1983
- 1984
- 1985
- 1986
- 1987
- 1988
- 1989
- 1990
- 1991
- 1992
- 1993
- 1994
- 1995
- 1996
- 1997
- 1998
Defect Elimination Process

Steps performed

1. Remove defect
2. Remove root cause of defect
3. Eliminate process escape deficiency
4. Search/analyze product for other, similar escapes

Process Improvement Continuously Achieved By Performing Feedback Steps 2 and 3

Original Defect

Similar Additional Undetected Defects

Defect Escaped Detection?

Defect Introduced

Root Cause
Space Shuttle Flight Software
Period Themes
## Space Shuttle Flight Software Period Themes

<table>
<thead>
<tr>
<th>Years</th>
<th>Theme</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>1978-1982</td>
<td>Initial System Development</td>
<td>Supports Incrementally / STS-1 to STS-5 Many Major Capabilities</td>
</tr>
<tr>
<td>1983-1985</td>
<td>Pre-Challenger Operations</td>
<td>Incremental Development / Reductions in Staff during 1985</td>
</tr>
<tr>
<td>1989-1993</td>
<td>Process Optimization and Stability</td>
<td>CMM Level 5 / GPC Memory/Speed Upgrade / Skilled, Stable Workforce</td>
</tr>
<tr>
<td>1994-1997</td>
<td>Transition To Loral / Lockheed Martin</td>
<td>Workforce Instability / OI-25 PTI DR Escapes / GPS Upgrade</td>
</tr>
<tr>
<td>2003-2005</td>
<td>Post-Columbia / Return-To-Flight</td>
<td>Cockpit Avionics Upgrade / Columbia Accident / Return to Flight</td>
</tr>
</tbody>
</table>
Space Shuttle Flight Software Accomplishments
**Initial Development (1978 – 1982)**

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>- Transition from ALT work to OFT development</td>
</tr>
<tr>
<td></td>
<td>- Expansion of orbit FSW capability post STS-1</td>
</tr>
<tr>
<td></td>
<td>- First Flight Capabilities</td>
</tr>
<tr>
<td></td>
<td>- Schedule driven, heavy change request traffic</td>
</tr>
<tr>
<td></td>
<td>- Early Systems Management / Payload Management Software</td>
</tr>
</tbody>
</table>
# Pre-Challenger Accident (1983 – 1985)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scope Of Development</strong></td>
<td>▪ Rendezvous</td>
</tr>
<tr>
<td></td>
<td>▪ Full Redesigned SM/PL Capabilities</td>
</tr>
<tr>
<td></td>
<td>▪ RMS Deploy and Retrieval</td>
</tr>
<tr>
<td></td>
<td>▪ Centaur Development</td>
</tr>
<tr>
<td></td>
<td>▪ Spacelab</td>
</tr>
<tr>
<td></td>
<td>▪ Main Engine Control redesign</td>
</tr>
<tr>
<td></td>
<td>▪ Payload manifesting flexibility</td>
</tr>
<tr>
<td></td>
<td>▪ Crew enhancements</td>
</tr>
<tr>
<td></td>
<td>▪ Enhanced ground checkout</td>
</tr>
<tr>
<td></td>
<td>▪ Western Test Range (Vandenberg)</td>
</tr>
<tr>
<td></td>
<td>▪ Reconfiguration tool planning / development</td>
</tr>
<tr>
<td></td>
<td>▪ Tools and procedures planning / development for DOD flights</td>
</tr>
</tbody>
</table>
## Post Challenger Accident (1986 – 1988)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>▪ Post-51L Safety Changes</td>
</tr>
<tr>
<td></td>
<td>▪ Bailout Capability</td>
</tr>
<tr>
<td></td>
<td>▪ Abort Enhancements</td>
</tr>
</tbody>
</table>
### CMM Lvl 5 Process Under IBM (1989 to 1993)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>▪ GPC Upgrade</td>
</tr>
<tr>
<td></td>
<td>▪ Extended Landing Site Table</td>
</tr>
<tr>
<td></td>
<td>▪ OPS 3 (TAL Code) in upper memory</td>
</tr>
<tr>
<td></td>
<td>▪ Redesigned Abort sequencer</td>
</tr>
<tr>
<td></td>
<td>▪ 2 Engine Out Auto Contingency Aborts</td>
</tr>
<tr>
<td></td>
<td>▪ OV-105 Hardware changes</td>
</tr>
<tr>
<td></td>
<td>▪ On-Orbit Changes</td>
</tr>
<tr>
<td></td>
<td>▪ MIR Docking</td>
</tr>
<tr>
<td></td>
<td>▪ On-Orbit DAP Changes</td>
</tr>
</tbody>
</table>
## Loral / Lockheed Martin (1994 to 1997)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>- Mir Docking Adapter&lt;br&gt;- On-Orbit DAP Changes&lt;br&gt;- 3 Engine Out Auto Contingency Aborts&lt;br&gt;- Ascent Performance Enhancements&lt;br&gt;- Single-String GPS</td>
</tr>
</tbody>
</table>
## United Space Alliance (1998 to 2002)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scope Of Development</strong></td>
<td>3-String GPS</td>
</tr>
<tr>
<td></td>
<td>East Coast Abort Landing (ECAL) Automation</td>
</tr>
<tr>
<td></td>
<td>Automatic Reboost</td>
</tr>
<tr>
<td></td>
<td>GPC Payload Command Filter (GPCF)</td>
</tr>
<tr>
<td></td>
<td>Increased data to MEDS</td>
</tr>
<tr>
<td></td>
<td>Start of Cockpit Avionics Upgrade (CAU) builds</td>
</tr>
</tbody>
</table>
## Category Observation / Characteristics

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>▪ Last of CAU builds</td>
</tr>
<tr>
<td></td>
<td>▪ Enhanced ADI / HSI capability</td>
</tr>
</tbody>
</table>
## OI Development Continuing (2006 – 2008)

<table>
<thead>
<tr>
<th>Category</th>
<th>Observation / Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope Of Development</td>
<td>▪ Lambert Guidance Improvements</td>
</tr>
<tr>
<td></td>
<td>▪ 6x Traj display redesign</td>
</tr>
<tr>
<td></td>
<td>▪ Entry and Ascent Bearing Display additions</td>
</tr>
<tr>
<td></td>
<td>▪ RTLS ET Sep improvements</td>
</tr>
<tr>
<td></td>
<td>▪ Entry Remote Controlled Orbiter (RCO) Capability</td>
</tr>
<tr>
<td></td>
<td>▪ Elimination of old user notes and DRs</td>
</tr>
<tr>
<td></td>
<td>▪ Reduction in Horizontal Sit display code size</td>
</tr>
<tr>
<td></td>
<td>▪ Year End Roll Over (YERO)</td>
</tr>
</tbody>
</table>
Look At Improvement
Through Latent Product DRs
Number Of Latent Unknown Product DRs Flown

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Event</td>
<td>Initial System Development Supports Incrementally</td>
<td>Pre-Challenger Operations</td>
<td>Post-Challenger Revalidation</td>
<td>GPC Memory/Speed Upgrade</td>
<td>Workforce Instability</td>
<td>Cockpit Avionics Upgrade</td>
<td>Shuttle Ending</td>
<td>Shuttle Ending</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>STS-1, STS-2, and STS-3</td>
<td>With Incremental Development</td>
<td>Return-To-Flight</td>
<td>CMN Level 5</td>
<td>Survive Process Influx Of New Personnel</td>
<td>Post-Columbia Return-To-Flight</td>
<td>Shuttle Ending</td>
<td>Skill Maintenance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>With Major Capabilities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Product DRs (Unknown At The Time) Flown

Product DRs that existed on a flown system, but were unknown at the time of the flight. Discovered up to 25 years later.

8/12/2010
## Summary Of Product DR By Period

<table>
<thead>
<tr>
<th>Years</th>
<th>New Product DRs Introduced In Period</th>
<th>Latent, Unknown Product DRs At End of Period</th>
<th>Flight Days Over Period</th>
<th>DRs Occurring In Flight</th>
<th>Latent, Unknown Severity 1 DRs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1978-1982</td>
<td>523</td>
<td>424</td>
<td>29</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1983-1985</td>
<td>109</td>
<td>322</td>
<td>147</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>1986-1988</td>
<td>16</td>
<td>240</td>
<td>None</td>
<td>No Flights</td>
<td>No Flights</td>
</tr>
<tr>
<td>1989-1993</td>
<td>22</td>
<td>140</td>
<td>291</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1994-1997</td>
<td>12</td>
<td>100</td>
<td>365</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>1998-2002</td>
<td>8</td>
<td>39</td>
<td>675</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2003-2005</td>
<td>0</td>
<td>20</td>
<td>None</td>
<td>No Flights</td>
<td>No Flights</td>
</tr>
<tr>
<td>2006-2008</td>
<td>1</td>
<td>2</td>
<td>162</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2009-2011</td>
<td>0</td>
<td>N/A</td>
<td>114</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- Supporting information available in more detail backup presentation.
Take Away From History Of Latent Product DRs

- Initial System Development period for future systems likely to be similar to that for the Space Shuttle
  - Likely first flight will have a large number of unknown software discrepancies some of which may be Severity 1
- Transition to Operations period likely to produce the maximum number of in-flight software discrepancies
  - Significant software maintenance activity as latent Product DRs are discovered, analyzed, and fixed as appropriate
  - Concurrently, program push to add capabilities which could not be implemented during the development period
  - Staffing reduction occurring from development period level to steady state operations level
  - Planned automation likely slightly behind staffing reductions based on the original automation schedule
- Quality improvements made in 1986 to 1988 for PASS were critical to the success in later years
- It takes a long time for unknown latent software defects to be identified
Space Shuttle Flight Software Quality Measurements
## Summary Of Quality Metrics By Period

<table>
<thead>
<tr>
<th>Years</th>
<th>Product Error Rate DRs / KSLOC</th>
<th>Pre-Build Error Detection Effectiveness</th>
<th>Verification Effectiveness (Percent Found By SRR)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1978-1982</td>
<td>0.8 (STS-1) to 1.1</td>
<td>Information Not Available</td>
<td>77 % to 91 % (STS-1)</td>
<td></td>
</tr>
<tr>
<td>1983-1985</td>
<td>2.8 (OI-1) to 1.1</td>
<td>40 % to 65 %</td>
<td>70 % to 80 %</td>
<td>Very Short Cycle - Release Every 4 Mo.</td>
</tr>
<tr>
<td>1986-1988</td>
<td>0.7 to 0.2 (OI-8C)</td>
<td>Near 80 %</td>
<td>60 % to 70 %</td>
<td>Return-to-flight Critical Changes</td>
</tr>
<tr>
<td>1989-1993</td>
<td>0.1 to 0.2</td>
<td>80 % to 90 %</td>
<td>80% to 90 %</td>
<td></td>
</tr>
<tr>
<td>1994-1997</td>
<td>0.1 to 0.2</td>
<td>75 % to 85 %</td>
<td>85 % to 100 %</td>
<td>Isolated Process Escape on OI-25</td>
</tr>
<tr>
<td></td>
<td>except 0.8 for OI-25</td>
<td></td>
<td>except 60 % for OI-25</td>
<td></td>
</tr>
<tr>
<td>1998-2002</td>
<td>0.1 to 0.2</td>
<td>85 % to 90 %</td>
<td>85 % to 95 %</td>
<td></td>
</tr>
<tr>
<td>2003-2005</td>
<td>CAU Canceled</td>
<td>CAU Canceled</td>
<td>CAU Canceled</td>
<td>Work on CAU required changes, Later CAU Canceled</td>
</tr>
<tr>
<td>2006-2008</td>
<td>0.0 to 0.1</td>
<td>80 % to 100%</td>
<td>95 % to 100 %</td>
<td>Reduced Flight System Changes Only, No OI Dev.</td>
</tr>
<tr>
<td>2009-2011</td>
<td>No OI Development</td>
<td>No OI Development</td>
<td>No Development</td>
<td></td>
</tr>
</tbody>
</table>
Take Away From History Of Quality Metrics

- STS-1 quality was extremely high
  - 0.80 Product DRs / KSLOC
  - 91% of DRs detected by flight
- Transition to operations in parallel with shortened development schedules (CI every 4 months) resulted in a sharp increase in Product Error Rate
  - OI-1, Product Error Rate 2.8 Product DRs per KSLOC
- By OI-8C (1988) Product Error Rate down to 0.2 Product DRs per KSLOC
  - Range of 0 to 0.2 Product DRs per KSLOC has been sustained since then with the exception of one capability on OI-25
- Significant effort expended on PASS changes required for Cockpit Avionics Upgrade (CAU). PASS changes not used following cancellation of CAU.
Space Shuttle Flight Software Reliability
### Summary Of Modeled Reliability By Period

<table>
<thead>
<tr>
<th>Years</th>
<th>Calendar Days Between Any Product DR</th>
<th>Flight Days Between In-Flight DRs</th>
<th>Risk To Shuttle Due To Severity 1 FSW DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1978-1982</td>
<td>6 (STS-1), 7 (STS-5)</td>
<td>7 (STS-1), 9 (STS-5)</td>
<td>1 in 327 (STS-1) to 1 in 409 (STS-5)</td>
</tr>
<tr>
<td>1983-1985</td>
<td>10 to 19</td>
<td>12 to 24</td>
<td>1 in 552 to 1 in 1072</td>
</tr>
<tr>
<td>1986-1988</td>
<td>29 at STS-26</td>
<td>90 at STS-26</td>
<td>1 in 1599 at STS-26</td>
</tr>
<tr>
<td>1989-1993</td>
<td>29 to 42</td>
<td>90 to 131</td>
<td>1 in 1599 to 1 in 2335</td>
</tr>
<tr>
<td>1994-1997</td>
<td>42 to 54</td>
<td>131 to 120</td>
<td>1 in 2335 to 1 in 3161</td>
</tr>
<tr>
<td>1998-2002</td>
<td>54 to 61</td>
<td>120 to 140</td>
<td>1 in 3161 to 1 in 3491</td>
</tr>
<tr>
<td>2003-2005</td>
<td>75 at STS-114</td>
<td>235 at STS-114</td>
<td>1 in 4212 at STS-114</td>
</tr>
<tr>
<td>2006-2008</td>
<td>75 to 88</td>
<td>235 to 276</td>
<td>1 in 4212 to 1 in 4930</td>
</tr>
<tr>
<td>2009-2011</td>
<td>88 to 94</td>
<td>276 to 294</td>
<td>1 in 4930 to 1 in 6260</td>
</tr>
</tbody>
</table>

- Risk level of approximately 1 in 1000 at January 2006 established during reliability research in the late 1980’s as a Return-To-Flight action. Variation over time based on MTBF as calendar days between any Product DR’s.
Take Away From Reliability Information

- Early failure rates (calendar days between Product DRs) were on the order of 10 days.
  - Almost identical in-flight failure rate (flight days between in-flight DRs)
- Trend of risk of Severity 1 DR (loss of crew or vehicle)
  - Early flights on the order of 1 in 400
  - STS-26 on the order of 1 in 1600
  - Current flights on the order of 1 in 6000
    - Severity 1 DR small sample size insufficient to accurately assess probabilities
    - Estimates are believed to be conservative (actual risk is lower)
- Extraordinary improvement in-flight failure rate from STS-51L (Challenger Accident) to STS-26 (Return-to-Flight)
  - From 24 flight days between in-flight DRs to 90 days (factor of 3 increase)
- Major contributors
  - “Revalidation” Return-to-flight audits and actions
  - Full implementation of automatic flight reconfiguration
  - Factor of 3 increase in Software Production Facilities testing capacity driven by AP-101B/AP-101S concurrent production
Space Shuttle Flight Software
Lessons Learned

- This section provides summaries of key items for lessons learned from in-flight DRs and released Severity 1 DRs.
- Data from across all periods was developed separately. Then, lessons learned were consolidated.
Lessons Learned Over the Years

- Scenarios
  - All possible scenarios must be identified, addressed in requirements, accommodated via design/code, and tested.
    - Insure that proper initialization will occur under all scenarios supported by the software.
    - Requirements must address what is to be done for failed hardware under all scenarios supported by the software.
  - Scenario analysis must identify the maximum ranges for parameters under all scenarios. Variable precision must correctly support the maximum ranges.
  - Many scenarios-related problems have extremely small timing windows. Very unlikely to detect during testing only. May require “Multi-Pass” analysis methods to insure identification.
  - Implement robust scenario testing. Adequate test facility resources required including resources for off-nominal testing.
Lessons Learned Over the Years

- Interface Testing
  - Software Interface Control Document requirements must be explicitly verified in an end-to-end manner.
    - Two in-flight DRs due to failure to verify PASS SM to Spacelab ICD.
      - Both required in-flight patches due to mission objective impacts for specific payloads.
- Analyze Error Logs
  - Anomalies occurring which resulted in signatures being written to software error logs may go undetected unless the software error logs are analyzed after the run.
    - Procedures are documented to always require analysis of software error logs after completion of the test.
    - Test may be designed to automatically stop if a certain types of error conditions occur such as GPC errors.
      - Allows for capturing full software memory at the time of the error condition.
Lessons Learned Over the Years

- **Simulation Models**
  - Collect appropriate data during simulated hardware tests so that any anomaly occurring is identified.
  - Models in the software test environment must provide valid outputs
    - Timing related software problems will be impossible to detect in simulation unless the hardware models provide random variation similar to actual hardware characteristics.
  - Using checkpoints and restart capabilities may additionally limit the number of opportunities to observe timing related software problems even if the simulation supports random hardware timing variation.
    - Use of checkpoints and restart capabilities is highly desirable. These capabilities allow for efficient use of test facility resources. These capabilities also greatly enhance the ability to duplicate problems by providing a duplicate software/environment state just prior to the event being duplicated.
Lessons Learned Over the Years

- **Hardware / Software Integration Testing**
  - Using checkpoints and restart capabilities may limit the number of opportunities to observe timing related software problems.
    - Multiple tests from the same source checkpoint will have one fixed set of software internal timing relationships.
  - Collect appropriate data during hardware tests so that any anomaly occurring is identified.
  - Latent defects can remain in the FSW multiple years until scenario and hardware re-action timing align.

- **Manual processes**
  - Many processes such as late updates to flight reconfiguration are initially done manually. Three in-flight DRs in the 1983-1985 period introduced this way. Manual processes require continuous management oversight to ensure rigorous and correct execution.
Lessons Learned Over the Years

- “Apparently Unrelated” Changes
  - Multiple “apparently unrelated” changes can collectively produce unexpected erroneous consequences.
    - Regression testing required to insure software functions continue to work correctly.
  - Ever present risk to “stumble” into maintenance traps once the maintenance trap is introduced into the software.
    - Maintenance trap is typically the result of waiving programming standards for some small short term savings in implementation time, code space, or computer performance.
    - These short term savings are insignificant compared to the long term project cost incurred during maintenance phase.
Lessons Learned Over the Years

- Essential to formalize management and lead analysts responsibility for assessing skills proficiency and work performance history for every individual on every team and evaluate risk based on skills mix with closed loop responsibility to program manager.
  - Every change to human rated flight software must be implemented with the same professional attention to detail by knowledgeable and motivated personnel.

- Essential to collect measurements data and proactively analyze data to search for “in process” symptoms such as Pre-Build Errors found in inspections by only one inspector.
Lessons Learned Over the Years

- Specific Examples
  - Verification analyst participation in the pre-build inspection process significantly adds quality
    - Prior to mid part of Release 19 (STS-5), the Verification analysts did not participate in pre-build design/code inspections. However, they did participate in inspections of patches implemented on STS-1 due to the perceived increased risk of patch implementation over source changes.
    - Assessment of the quality of the STS-1 patches versus the STS-2 source changes for the same DR and CR implementation resulted in the observation that the STS-1 patches were of higher quality.
    - Following this conclusion, the pre-build design/code inspection process was modified to require participation of the Verification analyst.
Lessons Learned Over the Years

- Specific Examples
  - Sequential inspections (e.g., development only peer review followed by pre-build inspection) are equally effective in removing the same % of errors that exist at the start of the inspection.
    - A single inspection removes about 55 % of errors
    - Two sequential inspection each remove about 55 % of errors remaining at the start of each inspection.
    - Collectively, they remove 80 % of the errors present at the first inspection.
  - Similar results are obtained when the criteria for having a re-inspection generally results in re-inspections when undetected errors remain after an initial inspection.
    - With appropriate re-inspection criteria, it is possible to remove 80 % of the errors present at the first inspection.
Summary
## Contributors To PASS FSW High Quality

<table>
<thead>
<tr>
<th>Contributor To PASS FSW High Quality</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple releases and multiple iterations of testing prior to STS-1.</td>
<td>Delays in launch date due to TPS and SSME issues provided more testing time and more opportunities to fix identified problems.</td>
</tr>
<tr>
<td>Fully automated Flight-to-Flight Reconfiguration Process and Tools</td>
<td>Early flights had a number of System Management in-flight failures due to late manual updates.</td>
</tr>
<tr>
<td>Structured “PASS Revalidation” activities between Challenger accident and STS-26</td>
<td>Direct contributor to eliminating Severity 1 (Loss of crew/vehicle) DRs from PASS</td>
</tr>
</tbody>
</table>
| Continual enhancements of the Requirements/Design/Code/Test Inspection Processes                  | • Have appropriate participation in each type of inspection including external community participation  
                                                | • Having appropriate re-inspection criteria                                  |
| Adequate test facility functionality and capacity (equipment to execute cases on flight equivalent hardware) | Significant improvement in in-flight reliability between STS-51L and STS-26 during a period when test facility capacity increased by a factor of 3. |
| Defined criteria for selection of personnel for teams; define how to resist over commitment of critical skills. | Critical skills management has always been a priority. Re-enforced by action From OI-25 PTI DRs where team skill and over commitment were contributing factors. |
| Rigorous configuration management of all products including requirements, design, code, and tests. | Basic necessary condition                                                  |
Wrap-up

- This presentation has shown the accomplishments of the PASS project over three decades and highlighted the lessons learned.

- Over the entire time, our goal has been to
  - Continuously improve our process
  - Implement automation for both quality and increased productivity
  - Identify and remove all defects due to prior execution of a flawed process in addition to improving our processes following identification of significant process escapes

- Morale and workforce instability have been issues, most significantly during 1993 to 1998 (period of consolidation in aerospace industry).

- The PASS project has also consulted with others, including the Software Engineering Institute, so as to be an early evaluator, adopter, and adapter of state-of-the-art software engineering innovations.
Acronyms
# Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADI</td>
<td>Attitude Direction Indicator</td>
</tr>
<tr>
<td>ALT</td>
<td>Approach and Landing Test</td>
</tr>
<tr>
<td>AP-101B</td>
<td>Initial flight computer for Space Shuttle; 104 K 32-bit full works of Memory</td>
</tr>
<tr>
<td>AP-101S</td>
<td>Upgrade flight computer for Space Shuttle; 256 K 32-bit full words of Memory (256K 32-bit FWs = 1MB 8-bit bytes).</td>
</tr>
<tr>
<td>ATV</td>
<td>Automated Transfer Vehicle</td>
</tr>
<tr>
<td>CAIL</td>
<td>CEV Avionics Integration Lab</td>
</tr>
<tr>
<td>CAU</td>
<td>Cockpit Avionics Upgrade</td>
</tr>
<tr>
<td>CEV</td>
<td>Crew Exploration Vehicle</td>
</tr>
<tr>
<td>CI</td>
<td>Configuration Inspection</td>
</tr>
<tr>
<td>CM</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>CMM</td>
<td>Capability Maturity Model</td>
</tr>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integrated</td>
</tr>
</tbody>
</table>
## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Acronym</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>DAP</td>
<td>Digital Auto Pilot</td>
</tr>
<tr>
<td>DOD</td>
<td>Department of Defense</td>
</tr>
<tr>
<td>DR, DRs</td>
<td>Discrepancy Report(s)</td>
</tr>
<tr>
<td>ECAL</td>
<td>East Coast Abort Landing</td>
</tr>
<tr>
<td>ET</td>
<td>External Tank</td>
</tr>
<tr>
<td>FSW</td>
<td>Flight Software</td>
</tr>
<tr>
<td>GPC</td>
<td>General Purpose Computer</td>
</tr>
<tr>
<td>GPCF</td>
<td>GPC Payload Command Filter</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>HIS</td>
<td>Horizontal Situation Indicator</td>
</tr>
<tr>
<td>HTV</td>
<td>H-II Transfer Vehicle</td>
</tr>
<tr>
<td>ICD</td>
<td>Interface Control Document</td>
</tr>
<tr>
<td>KSC</td>
<td>Kennedy Space Center</td>
</tr>
</tbody>
</table>
## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KLSOC</td>
<td>1000 Non-Comment Source Lines of Code (new, changed, and deleted)</td>
</tr>
<tr>
<td>MEDS</td>
<td>Multifunction Electronic Display System</td>
</tr>
<tr>
<td>MIR</td>
<td>Name of the Russian Space Station</td>
</tr>
<tr>
<td>MTBF</td>
<td>Mean Time Between Failures</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>OFT</td>
<td>Orbital Flight Test</td>
</tr>
<tr>
<td>OI</td>
<td>Operational Increment</td>
</tr>
<tr>
<td>OPS</td>
<td>Operational Sequences</td>
</tr>
<tr>
<td>OV</td>
<td>Orbiter Vehicle</td>
</tr>
<tr>
<td>PTI</td>
<td>Program Test Input</td>
</tr>
<tr>
<td>RCO</td>
<td>Remotely Controlled Orbiter</td>
</tr>
<tr>
<td>RMS</td>
<td>Remote Manipulator System</td>
</tr>
<tr>
<td>RTLS</td>
<td>Return-To-Launch-Site</td>
</tr>
</tbody>
</table>
# Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTLS</td>
<td>Return-To-Lauch-Site</td>
</tr>
<tr>
<td>SAIL</td>
<td>Shuttle Avionics Integration Laboratory</td>
</tr>
<tr>
<td>SASCB</td>
<td>Shuttle Avionics Software Control Board</td>
</tr>
<tr>
<td>SEI</td>
<td>Software Engineering Institute</td>
</tr>
<tr>
<td>SM</td>
<td>Systems Management</td>
</tr>
<tr>
<td>SM/PL</td>
<td>Systems Management/Payload</td>
</tr>
<tr>
<td>SMS</td>
<td>Shuttle Mission Simulator</td>
</tr>
<tr>
<td>SRR</td>
<td>Software Readiness Review, typically 4 weeks prior to flight</td>
</tr>
<tr>
<td>SSME</td>
<td>Space Shuttle Main Engine</td>
</tr>
<tr>
<td>STS</td>
<td>Space Transportation System</td>
</tr>
<tr>
<td>TAL</td>
<td>Transoceanic Abort Landing</td>
</tr>
<tr>
<td>TPS</td>
<td>Thermal Protection System</td>
</tr>
<tr>
<td>Traj</td>
<td>Trajectory</td>
</tr>
<tr>
<td>YERO</td>
<td>Year End Roll Over</td>
</tr>
</tbody>
</table>