# Fault Management Technology Maturation for NASA's Constellation Program

*Keynote* for the

## *EPRI Condition Based Maintenance Conference, July 12, 2010 @ Disney Yacht Club*

*Robert D. Waterman*
*NASA John F. Kennedy Space Center*

CONSTELLATION

# Robert D. (Bob) Waterman

- 1985 – 2000 Space Shuttle Main Engine Avionics

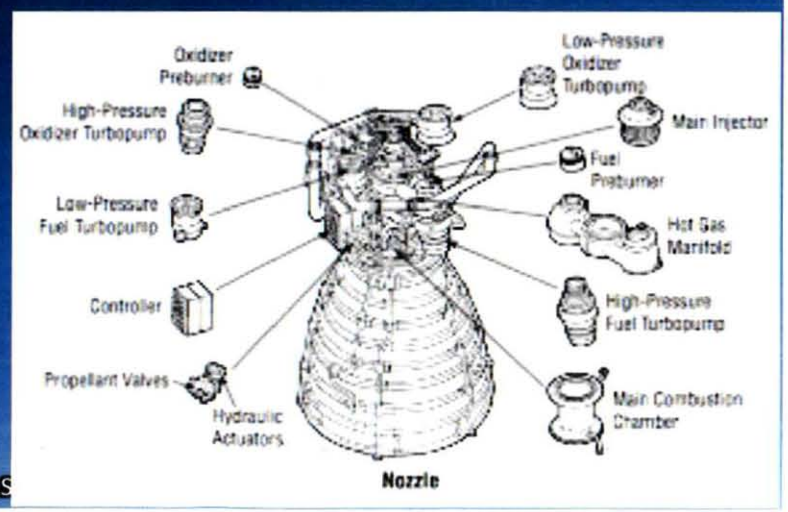
# Robert D. (Bob) Waterman

- **2001-Present**
  - Technical Integration Manager – Real-time Control Software
  - Orbital Space Plane – Avionics and Flight Software Lead
  - Strategic Technology Development Manager – Command and Control
  - Technical Lead for the Supportability of Non-Terrestrial Systems
  - Constellation Program – Ground Operations Command and Control Architect
- **Currently**
  - 21st Century Space Launch Complex – Range Interface & Control Services product group lead

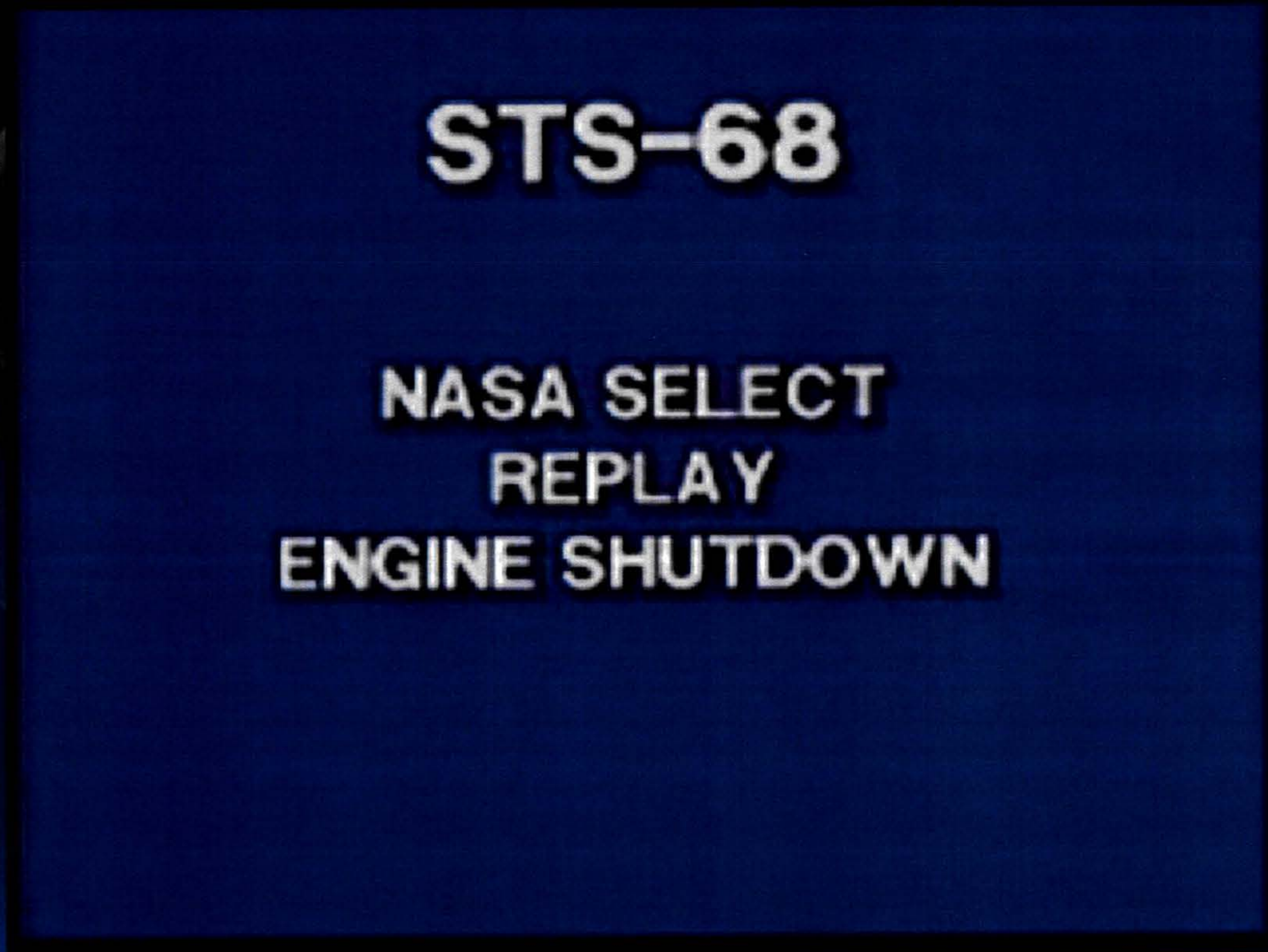
CONSTELLATION

# Space Shuttle Main Engine Avionics

- **The Space Shuttle Main Engines**
  - The three Space Shuttle Main Engines are clustered at the aft end of the Orbiter and have a combined thrust of more than 1.2 million pounds. They are high performance, liquid propellant rocket engines whose thrust can be varied over a range of 65 to 109 percent of their rated power level. They are the world's first reusable rocket engines and are 14 feet long and 7.5 feet in diameter at the nozzle exit. The Main Engine weighs approximately 7,000. Propelled by liquid hydrogen (fuel) and liquid oxygen (oxidizer), the engines operate during the entire eight-and-one-half-minute ride to orbit.



# Engine Start



# What Happened

- Engine Maintenance between flights detected corrosion in main combustion chamber requiring additional liquid oxygen injector posts to be plugged.
  - Inspections a manual and occur after every flight
- Software constants were not updated (process failure)
- Resulting combustion flow caused higher temperatures in the turbine outlet
- Temperatures were as predicted
- Temperatures were above the software redline
- Engine initiated showdown.
- Space Shuttle orbiter shutdown the other two engines

- Take away – there are many areas in both the flight and ground systems where periodic maintenance is required. NASA has been working towards predictive maintenance for some time, but as a technology program largely not in operations

# STS-110 Tanking

*Waiting on STS-110 MLP GH2 Leak Video from PAO*

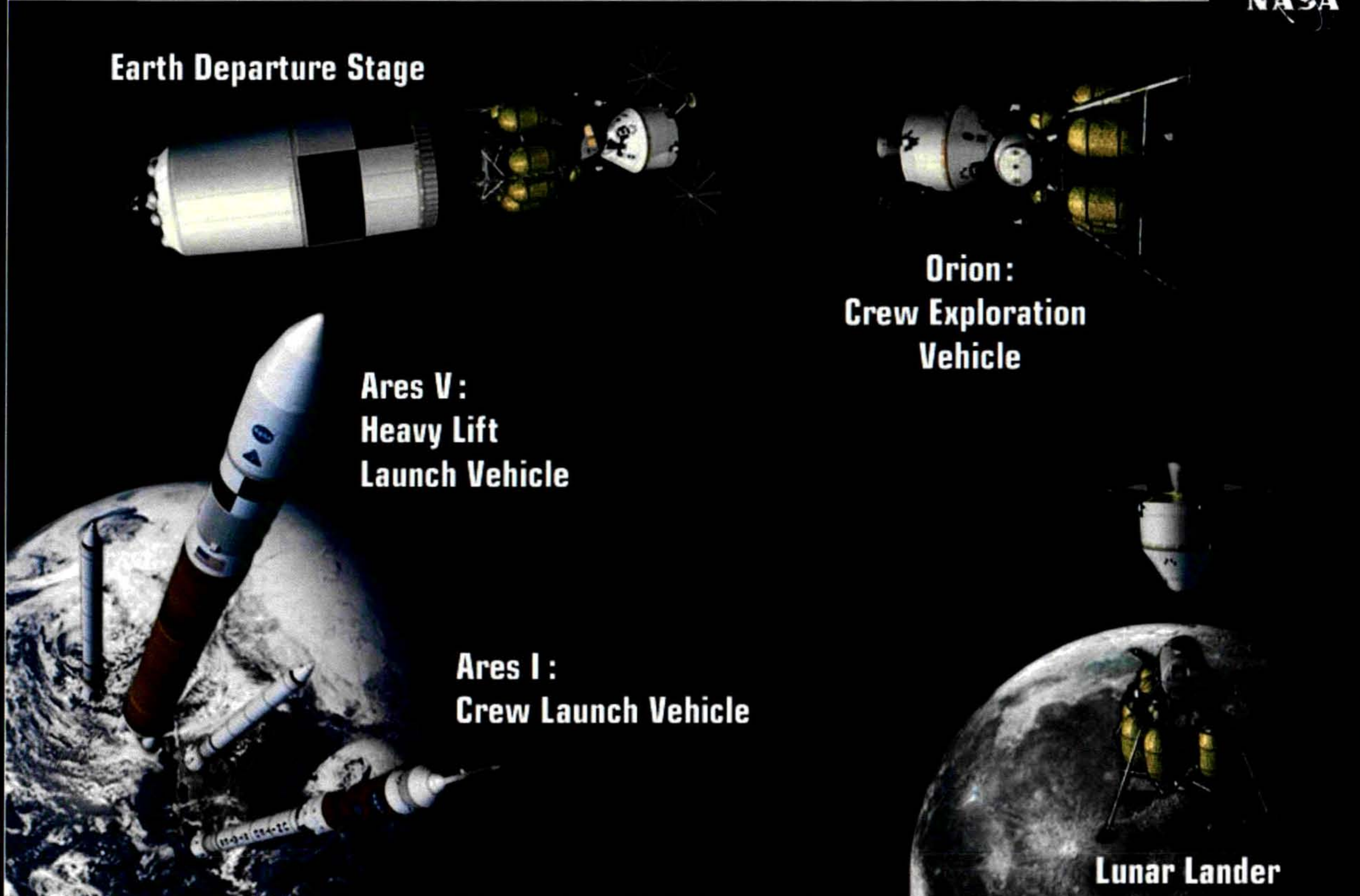
# STS-110 Tanking / What Happened

- **STS – 110 / April 4, 2002**
- **Large vapor cloud coming from Mobile Launch Platform (MLP) vent line on side 4. Probable cracked weld leaking gh2**
- **Welds used wrong filler material**
  - Operational work around - ‘clam shells’ welded over all discrepant welds.

# Transition

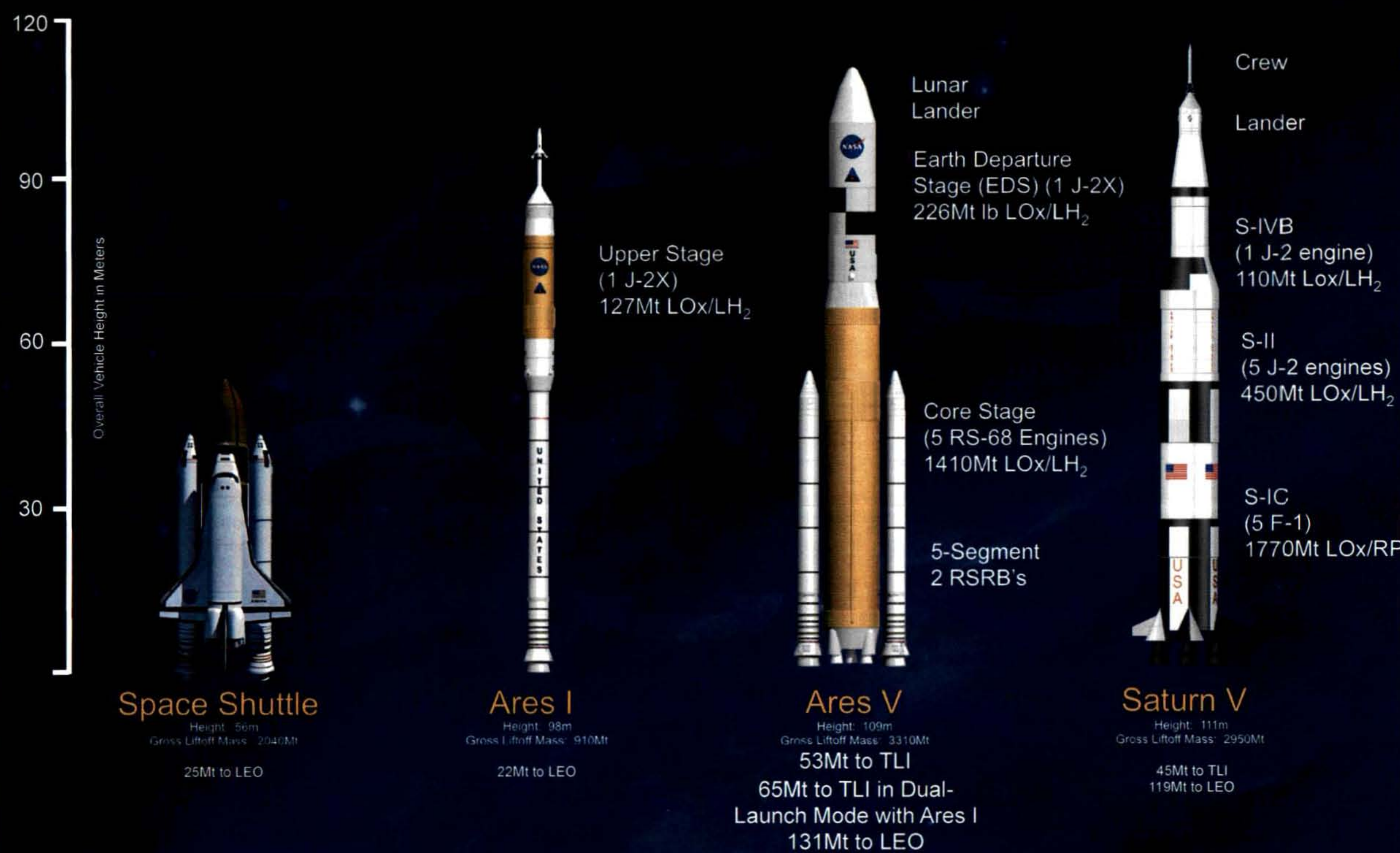
- Lessons Learned from Space Shuttle and International Space Station were applied to the Constellation Program
- Attempt to improve Operability in the design

# COMPONENTS OF THE CONSTELLATION PROGRAM



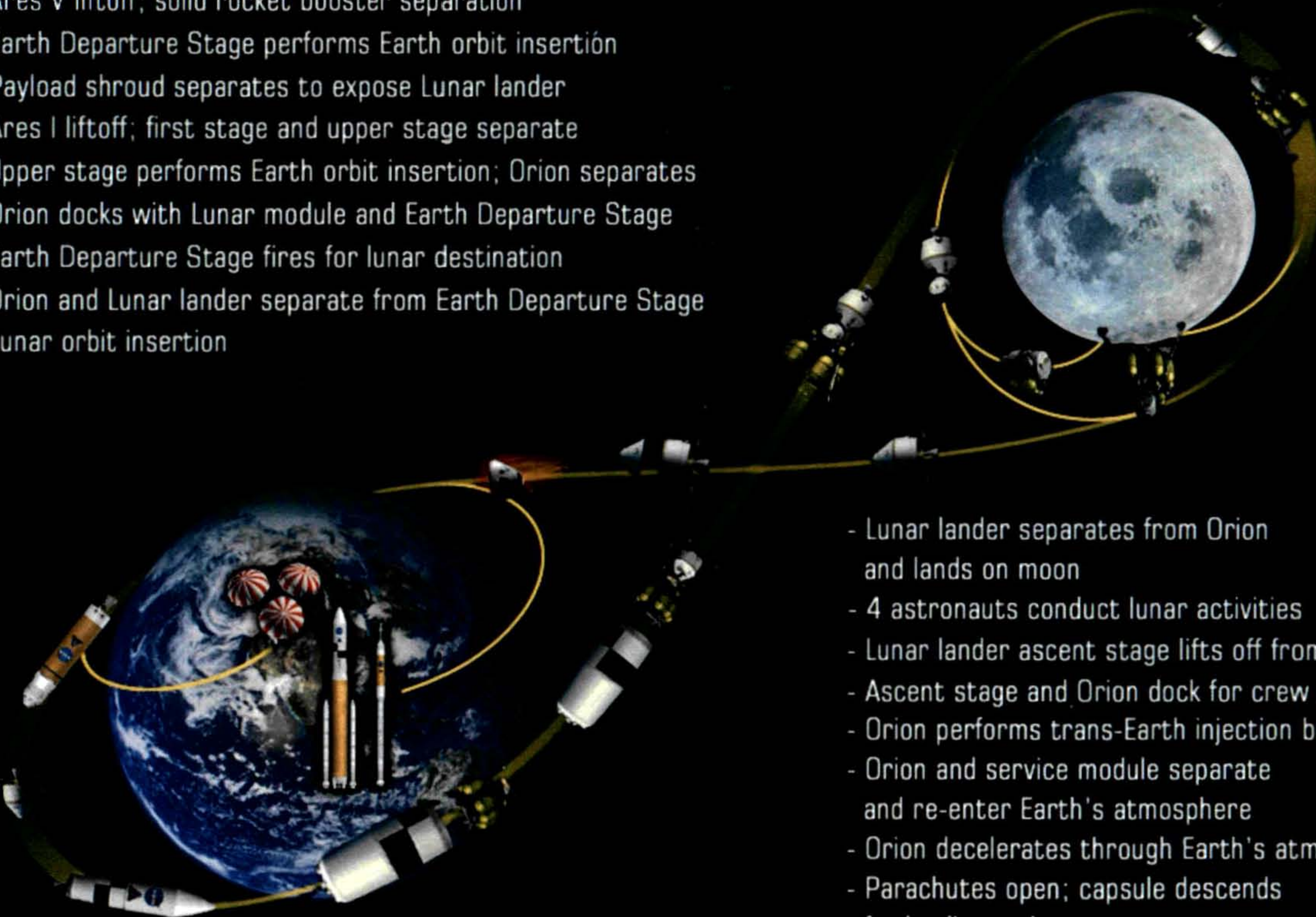
# BUILDING ON A FOUNDATION OF PROVEN TECHNOLOGIES

## Launch Vehicle Comparisons



# TYPICAL LUNAR REFERENCE MISSION

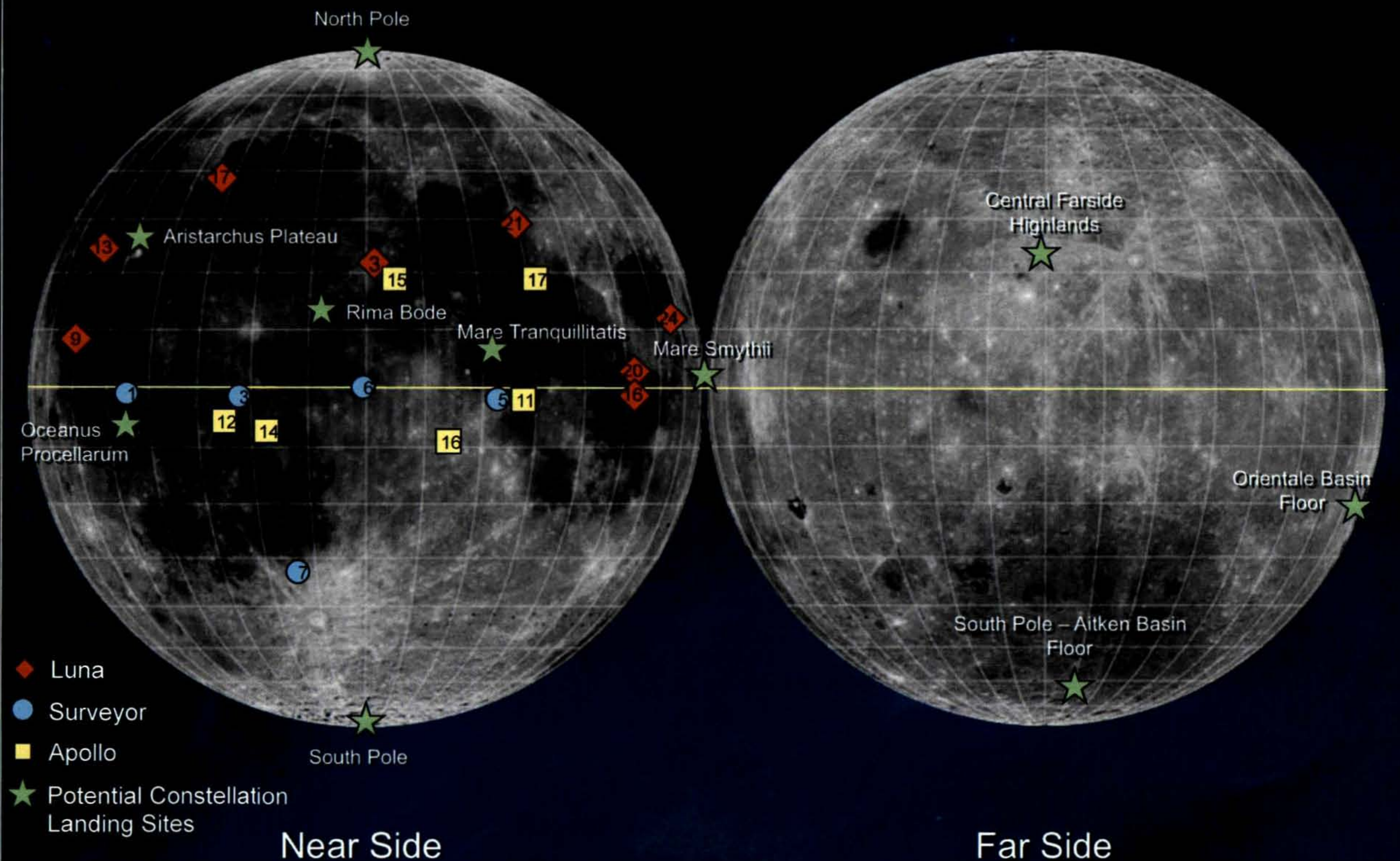
- Ares V liftoff; solid rocket booster separation
- Earth Departure Stage performs Earth orbit insertión
- Payload shroud separates to expose Lunar lander
- Ares I liftoff; first stage and upper stage separate
- Upper stage performs Earth orbit insertion; Orion separates
- Orion docks with Lunar module and Earth Departure Stage
- Earth Departure Stage fires for lunar destination
- Orion and Lunar lander separate from Earth Departure Stage
- Lunar orbit insertion



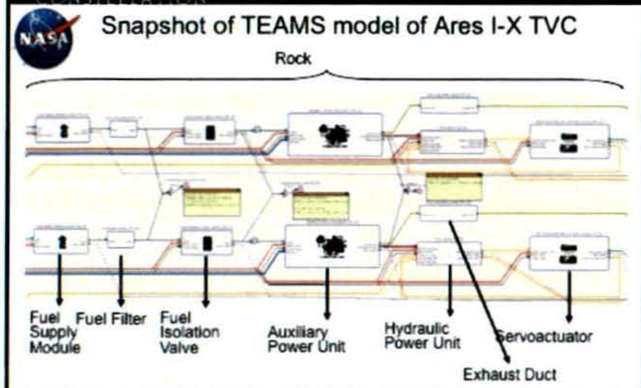
- Lunar lander separates from Orion and lands on moon
- 4 astronauts conduct lunar activities
- Lunar lander ascent stage lifts off from surface
- Ascent stage and Orion dock for crew transfer
- Orion performs trans-Earth injection burn
- Orion and service module separate and re-enter Earth's atmosphere
- Orion decelerates through Earth's atmosphere
- Parachutes open; capsule descends for landing and recovery

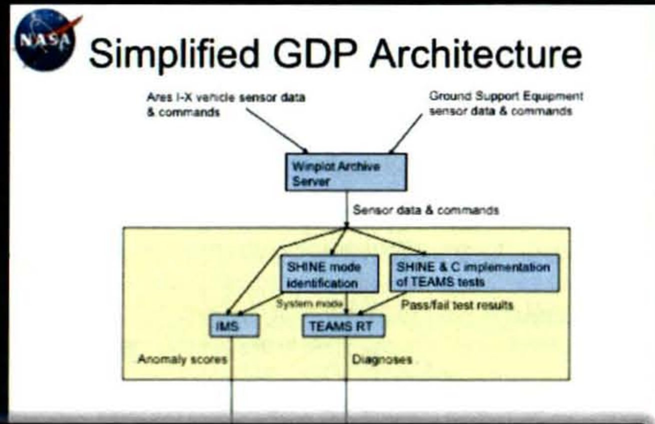
# CONSTELLATION CAN LAND ANYWHERE ON THE MOON

## Previous Missions Landed in Equatorial Band



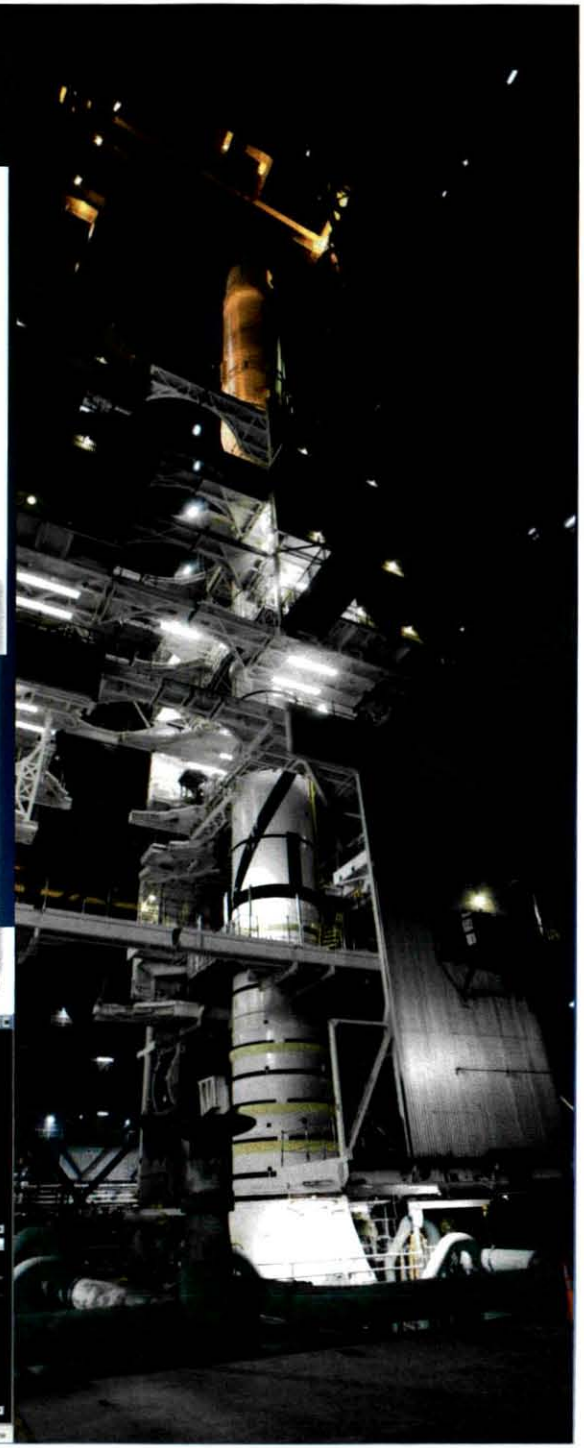
# Ares I-X Ground Diagnostic Prototype




- ✓ GDP Provided Fault Detection and Isolation for First Stage TVC System and Ground Hydraulic Support System
- ✓ GDP Provided Anomaly Detection
- ✓ GDP Was installed in Hangar AE for Ares I-X
- ✓ The prototype ran on live data from Ares I-X during all powered on testing in the VAB and at PAD-39B through End of Mission

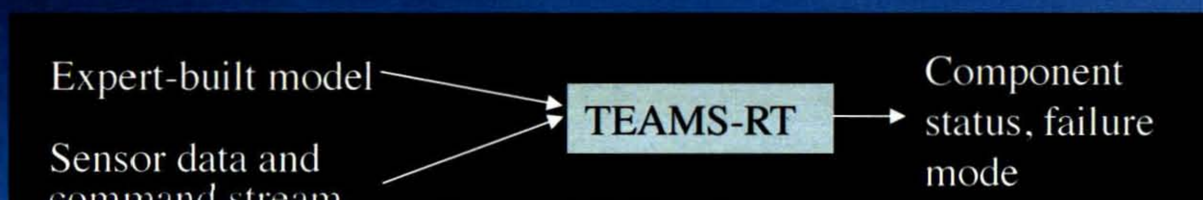




# Fault Detection and Fault Isolation Using TEAMS

## (Testability Engineering And Maintenance System)

- **TEAMS is a suite of tools for developing model-based fault isolation systems**
  - TEAMS-Designer, TEAMS-RT, and TEAMS-RDS
- **Model captures a system's structure, interconnections, tests, procedures, and failures**
  - Functional dependency model captures the relationships between various failure modes and system instrumentation
- **TEAMS-Designer used to create functional fault models from FMEA reports, fault trees, schematics, instrumentation lists, operational use cases, and other technical documentation**
  - Can be developed incrementally, adding knowledge as designs mature
  - Model-building requires system knowledge and modeling expertise
- **TEAMS-RT used for real-time isolation**
  - Input is set of health status indicators (pass/fail test results) + Dependency matrix (D-Matrix)
    - e.g.: exceedances, operator observables, manual tests
  - Output is a list of bad, suspect, good, and unknown components
- **TEAMS-RDS used for real-time operations**
  - Provides Session Management and Archival Service
  - Includes TEAMS-RT

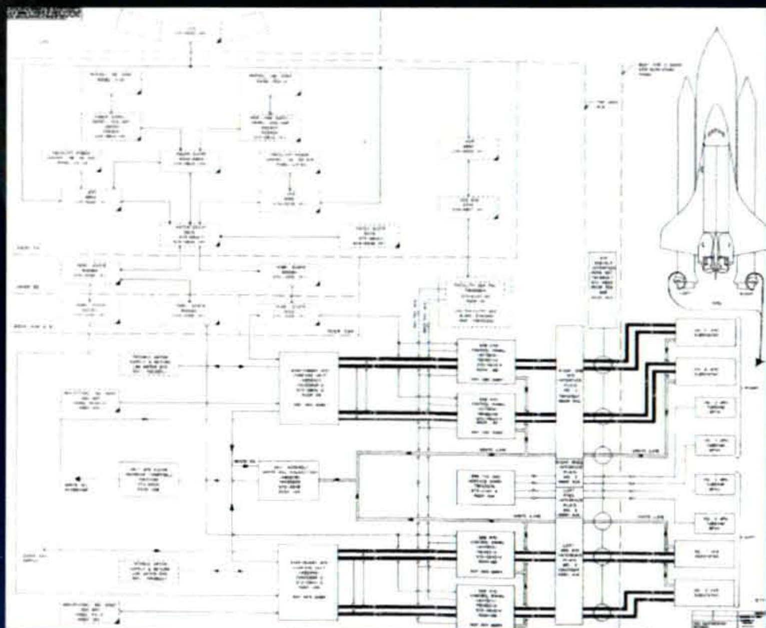


# Fault Modeling Using TEAMS:
## Modeling Process

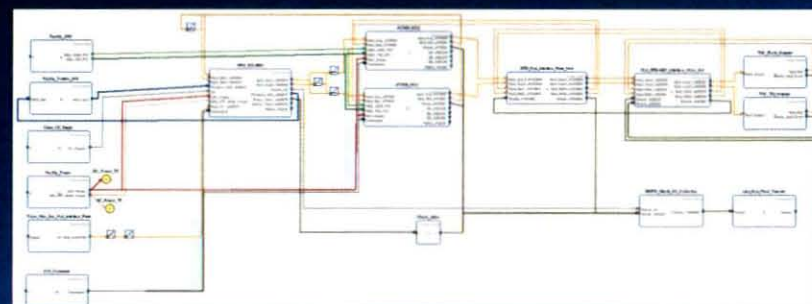
**Step 1: Build subsystem functional fault model**

- **Transformation of energy, material, signal within the system**
- **Basic system connectivity, interfaces, interactions**
- ➢ **Insufficient to do any analysis or to be a diagnostic engine**

*Knowledge captured from subsystem schematics/diagrams/etc. and converted into TEAMS model*



**Hydraulic Support System Block Diagram**



**Functional Model in TEAMS**

(Modeling process courtesy of Ares FFA Team)

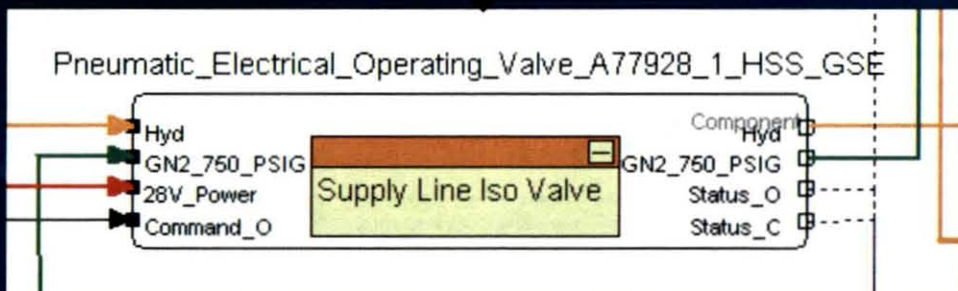
# Fault Modeling using TEAMS :

## Modeling Process

**Step 2: Populate failure modes of components**

- **Extracted from FMEA**
- **Added as "lowest level" nodes inside each component**

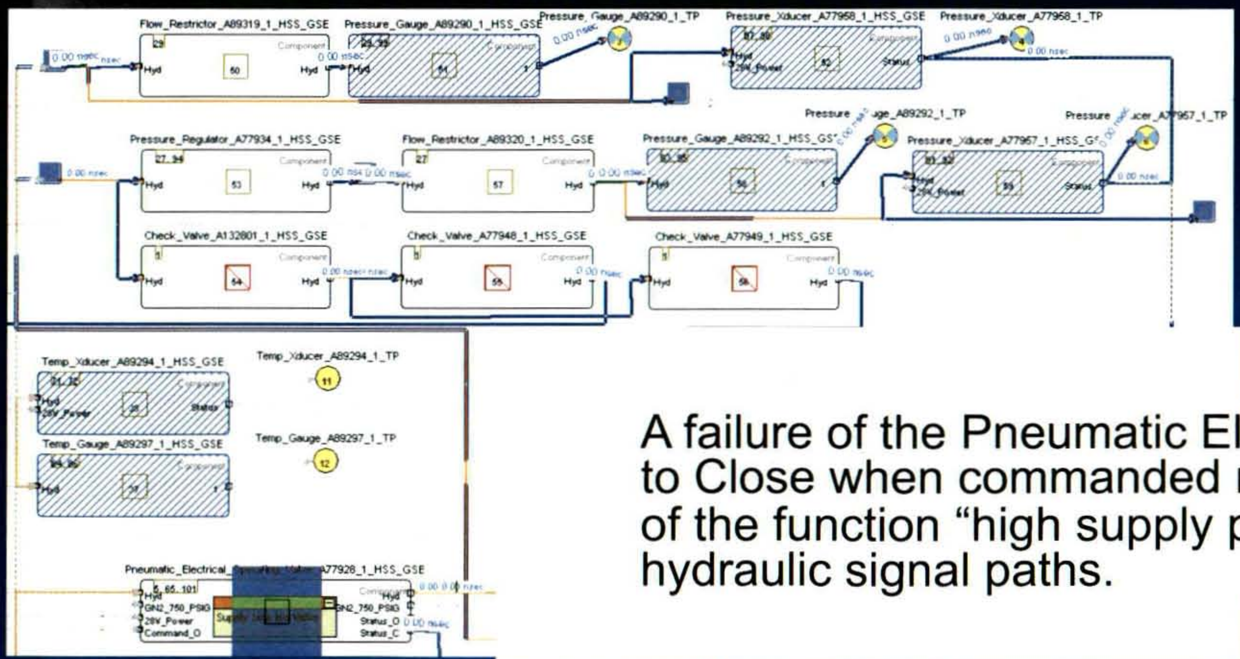
| | | | | | | |
|---|---|---|---|---|---|---|
| A77928 | Pneumatic / Electrical Operating Valve | Remotely control flow of hydraulic fluid to SRB | Fails open | Loss of hydraulic flow control to the SRB could delay operations. | No effect. | 3 |
| | | | Fails close | Inability to flow hydraulic fluid to SRB would delay operations. | No effect. | 3 |




# Fault Modeling using TEAMS : Modeling Process

## Step 3: Determine failure effect propagation paths

- **Each failure mode produces a specific effect / set of effects**
  - Propagate along physical paths (fluid, thermal, electrical)
  - Implemented using TEAMS functions
  - Formalization of FMEA

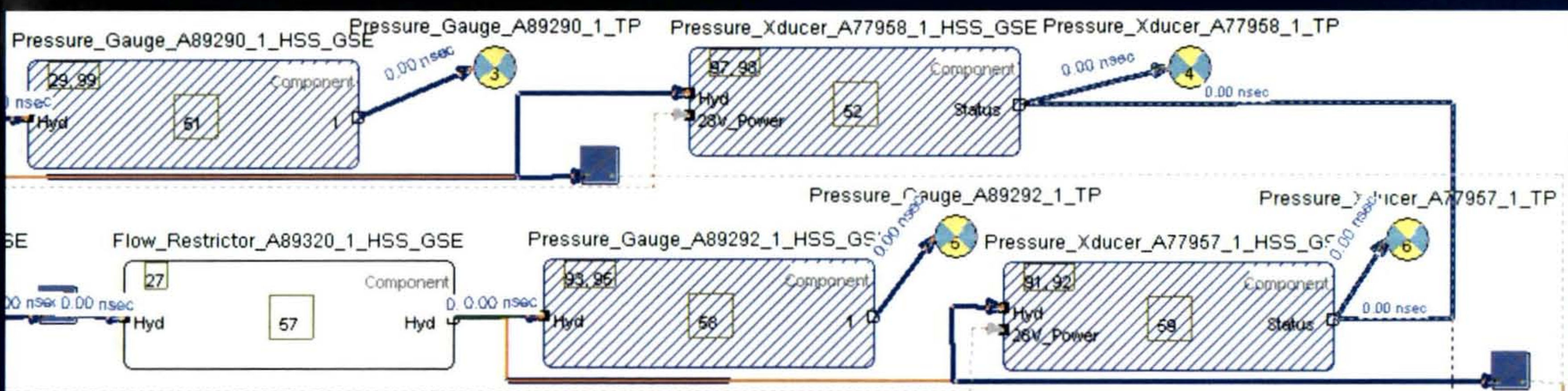


A failure of the Pneumatic Electrical Operating Valve to Close when commanded results in the propagation of the function “high supply pressure” over the hydraulic signal paths.

# Fault Modeling using TEAMS :
## Modeling Process

**Step 4: Identify sensors and test points**

- **Function model represent the location of all sensors**
- **The sensors are represented using nodes**
- **Each sensor is associated with TEAMS "test points"**




The test points that represent pressure gauges and transducers detect the function "high supply pressure," as indicated by the cyan and yellow coloring of the circular nodes.

# Fault Isolation Example

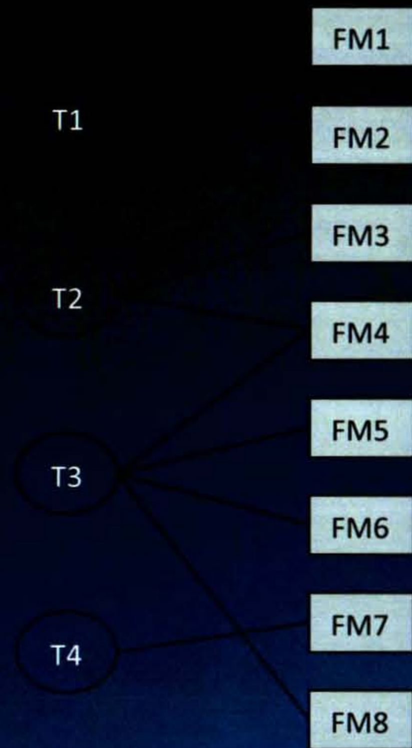
## D-matrix

Tests (observables)

Failure Modes (causes)

| | T1 | T2 | T3 | T4 |
|---|---|---|---|---|
| FM1 | 1 | | | |
| FM2 | 1 | 1 | | |
| FM3 | | 1 | | |
| FM4 | | 1 | 1 | |
| FM5 | | | 1 | |
| FM6 | | | 1 | |
| FM7 | | | | 1 |
| FM8 | | | 1 | |

1 = test can detect failure mode




Dependency matrix (D-matrix) is generated from the TEAMS Designer subsystem model

## Developing FDIR Modules - Fault Detection and Fault Isolation with TEAMS

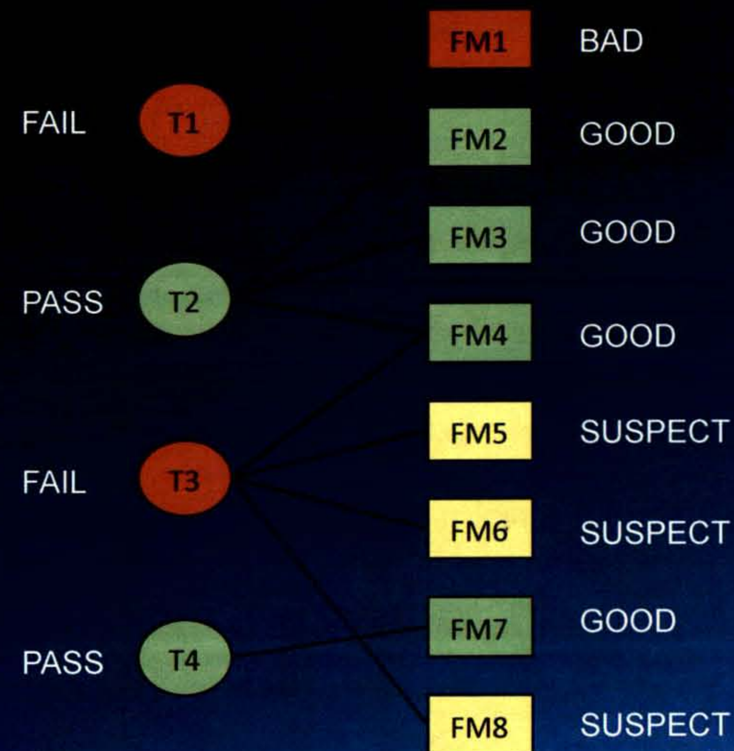
# Fault Isolation Example (cont.)

D-matrix Tests (observables)

Failure Modes (causes)

| | T1 | T2 | T3 | T4 |
|---|---|---|---|---|
| FM1 | 1 | | | |
| FM2 | 1 | 1 | | |
| FM3 | | 1 | | |
| FM4 | | 1 | 1 | |
| FM5 | | | 1 | |
| FM6 | | | 1 | |
| FM7 | | | | 1 |
| FM8 | | | 1 | |

1 = test can detect failure mode



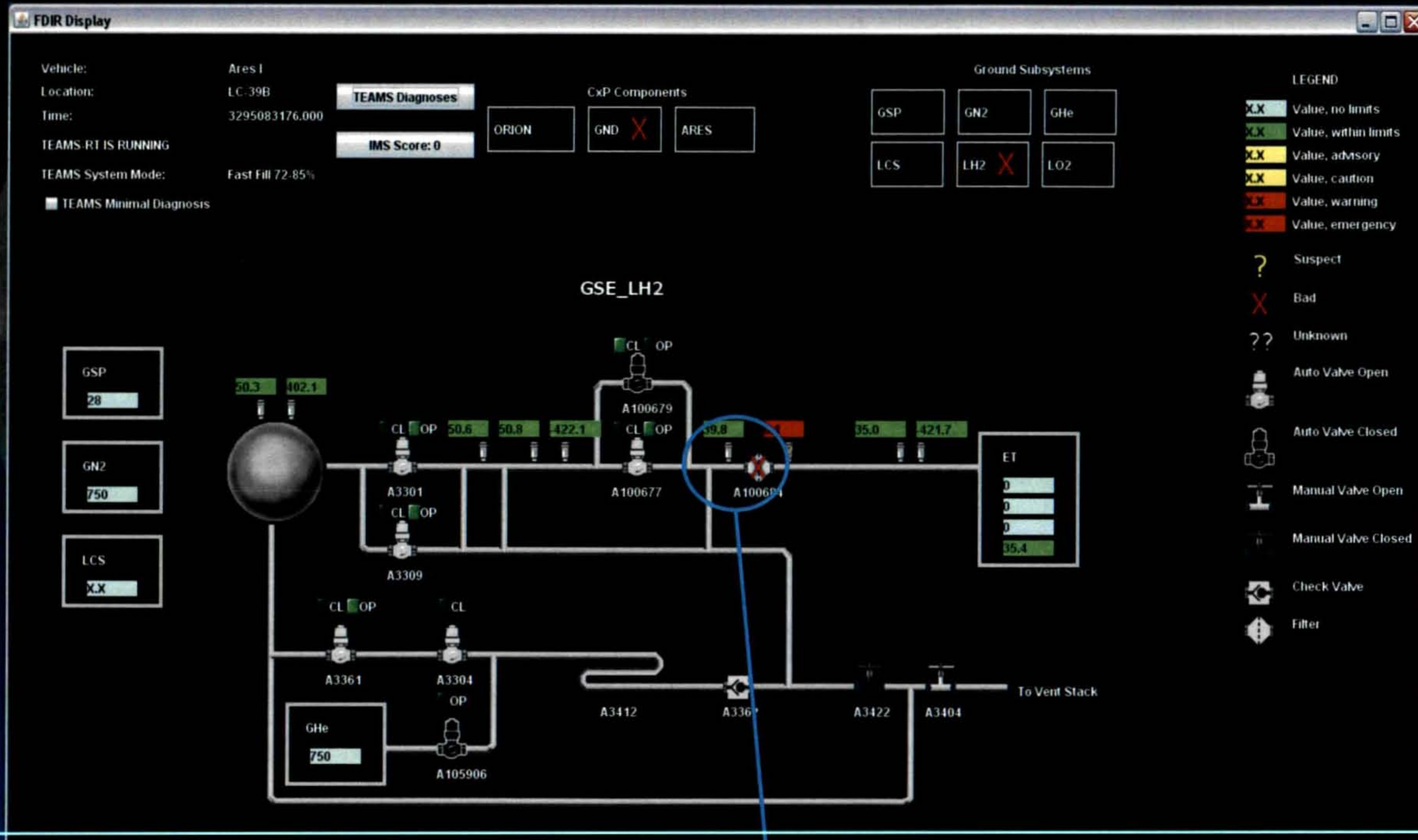

Compute *GOOD* failure modes: Every failure mode connected to a *PASS* test is *GOOD*.

Compute *BAD* failure modes: Every test that is *FAIL* has **at least one** failure mode that is *BAD*. If there is more than one failure mode that leads to a *FAIL* test, then all failure modes not labeled as *GOOD* are labeled as *SUSPECT*.

All remaining failure modes are labeled *UNKNOWN*: they are connected to tests for which we have no test information.

# LH2 FDIR Fault Isolation using TEAMS:

## Diagnosis of Clogged Liquid Hydrogen Filter

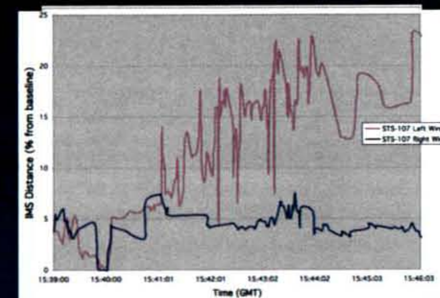



Red X and red highlighted measurement indicates component and corresponding measurement is bad.

Simulated Data

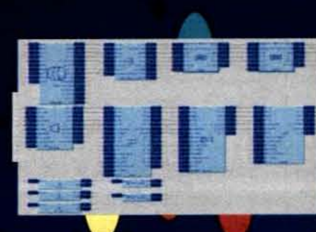
# Anomaly Detection Using Inductive Monitoring System (IMS)

IMS
Inductive Monitoring System



Deviation from nominal

Sensor Data



Nominal System Model

**Automatically learns how the system behaves and tells you if current behavior is out-of-family**

IMS developed at NASA/ARC by David Iverson

# Anomaly Detection using IMS

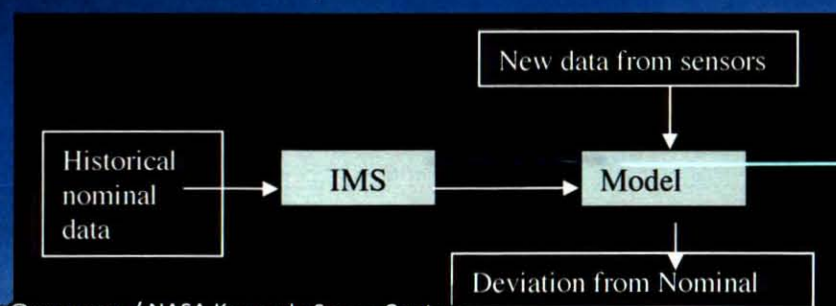
- Automatically derives models (off-line) from archived or simulated nominal operations data
  - Does not require off-nominal data
  - Does not require knowledge engineers or modelers to capture details of system operations

- Anomaly detection module can catch anomalies whose signatures are not known ahead of time

- Can detect subtle anomalies or anomalies that are not listed in the FMEA

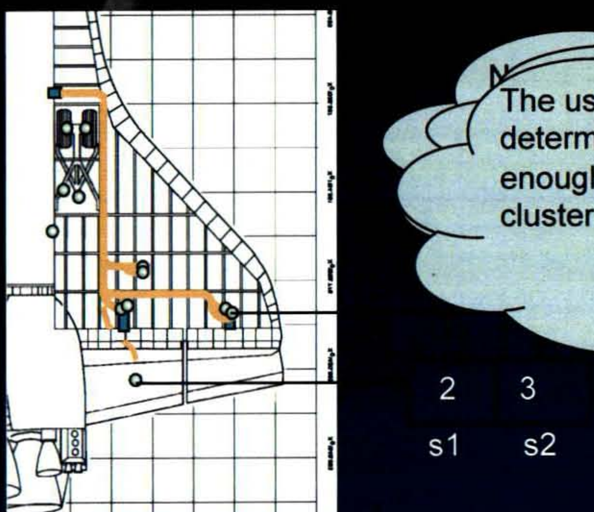
- On-line monitoring takes as input observations about the physical system (parameter values) & produces "distance from nominal" anomaly score

- Analyzes multiple parameter interactions
  - Automatically extracts system parameter relationships and interactions
  - Detects variations not readily
  - apparent with current individual
  - parameter monitoring practices



# Anomaly Detection using IMS:
## Modeling Example

Step 1: Determine sensors of interest for subsystem & form into vectors.



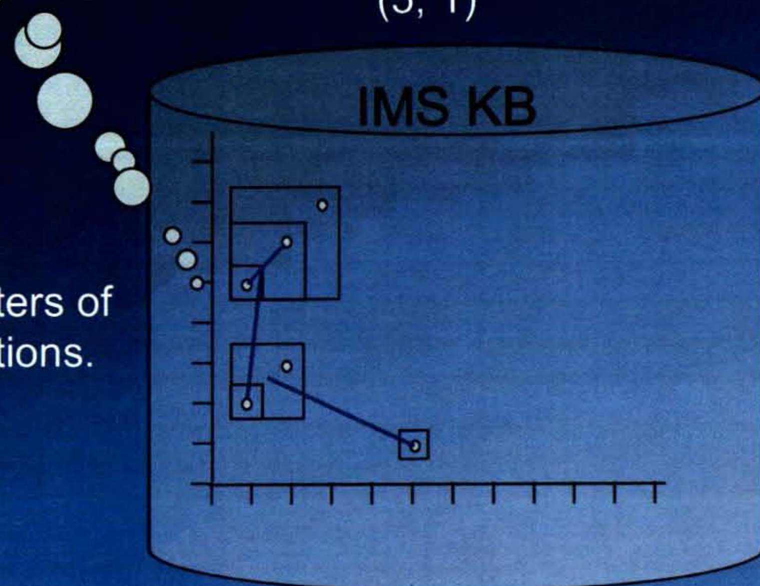

The user can customize the distance that determines whether a point is "close enough" to an existing cluster to expand the cluster vs. creating a new one.

Step 2: Train on archived data representative of expected nominal operations…

Training data set:

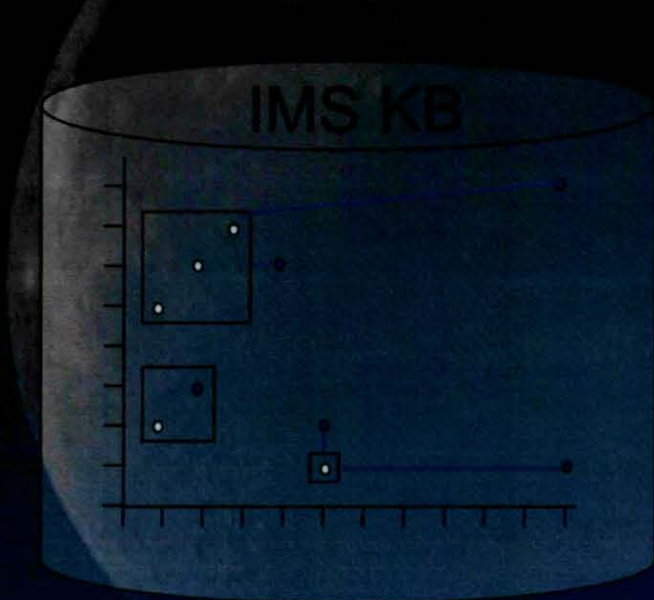
(s1, s2)
(1, 5)
(2, 6)
(1, 2)
(2, 3)
(3, 6)
(5, 1)

… Create clusters of nominal operations.



# Anomaly Detection using IMS:
## Monitoring Example

Step 3:
Using nominal operations clusters created in modeling step…

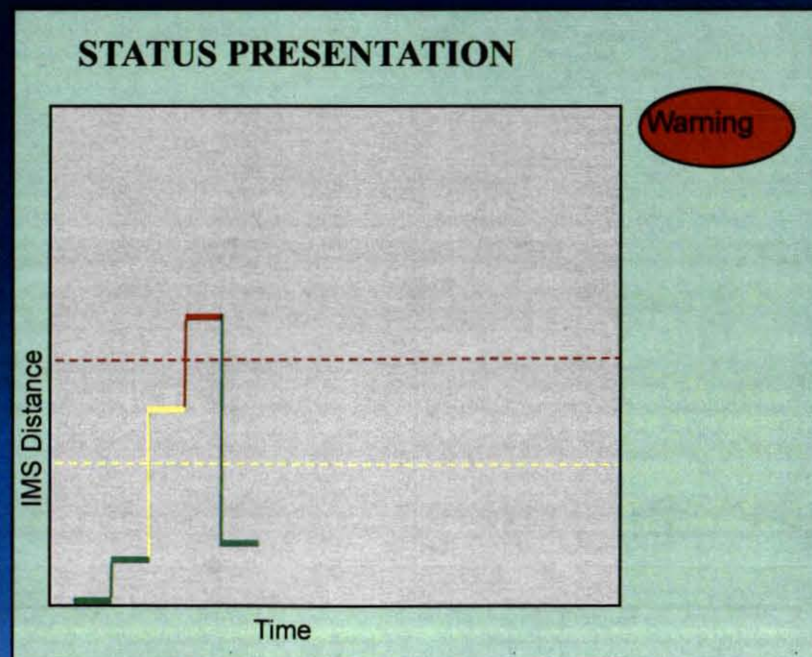



… As real time data is received, compare to nominal operations clusters…

Real-time data stream:

(2, 3)
(4, 6)
(11, 1)
(11, 8)
(5, 2)

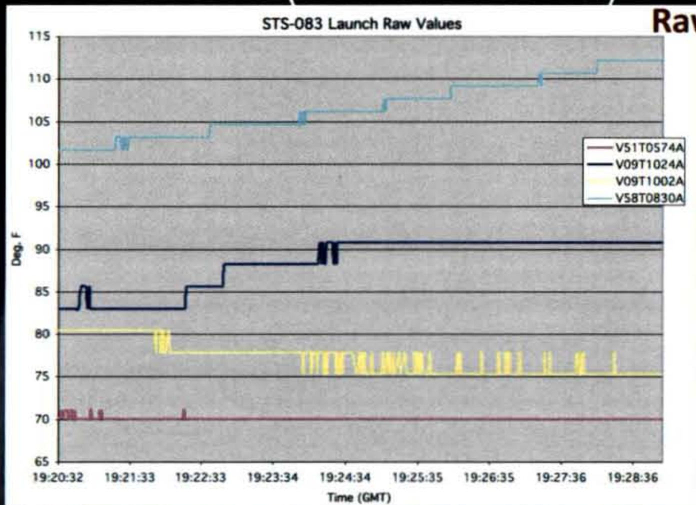
… Plot distance from closest nominal cluster to incoming data and/or issue caution/warning alert.



# Anomaly Detection Analysis of Columbia Ascent
## STS-83 Compared to STS-107

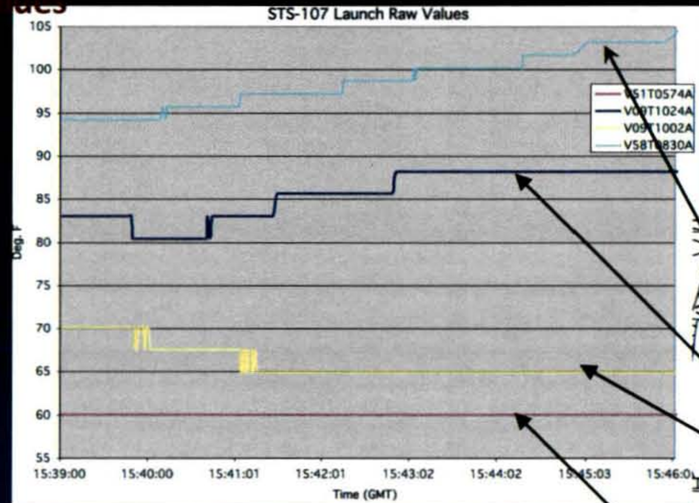
CONSTELLATION

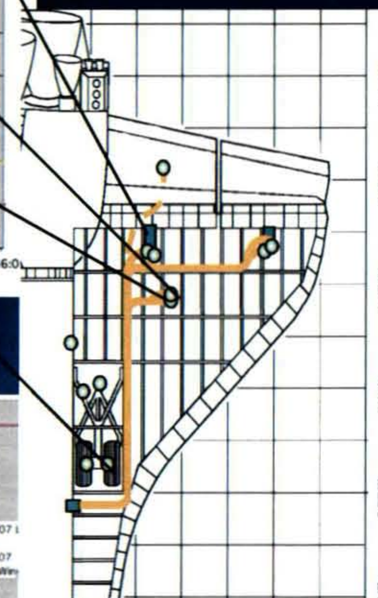
STS-83 (Nominal Ascent)




STS-107

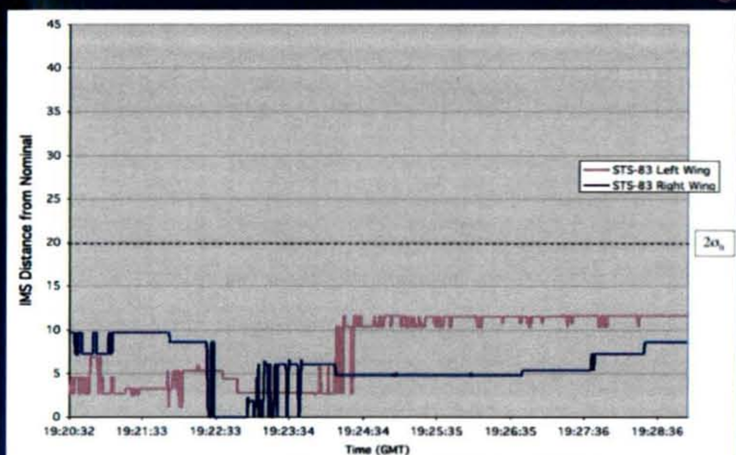
Raw Values

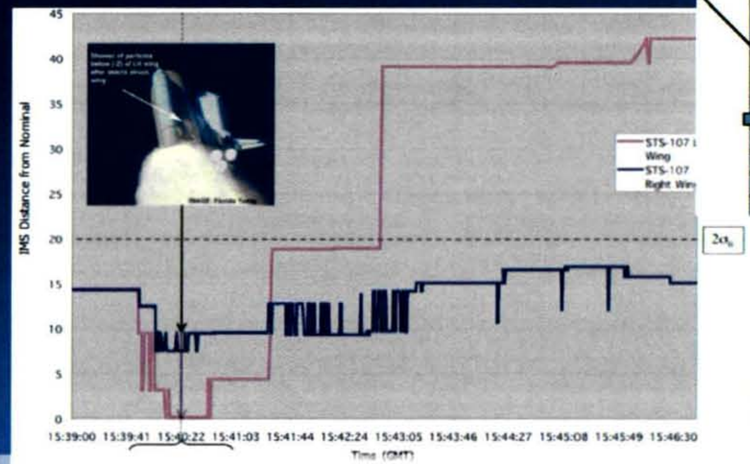


Deviation from Expected Function

Pink = Left Wing,




**There was enough (hidden) information in the STS-107 ascent telemetry data to indicate an anomaly .**
**The IMS method can help identify subtle but meaningful changes.**

Analysis courtesy
...erson/ARC

# Expected Benefits

## Many expected benefits

- Improves launch availability (reduces component of Mean Time To Repair)
  - Reduces integrated troubleshooting time (Isolation & Recovery Recommendation)
- Reduces console operator cognitive workload
  - Helps considering the reduction in console operators and non-integrated architecture of Ares / Orion subsystems
  - Supports reduction of FR personnel by 50% compared to Shuttle
- Reduces engineering support needs for Anomaly Detection and Recovery Recommendation
- Speeds assessment of flying with failed condition through trace to suspect failure modes.
- Improves time to develop flight rationale for anomalous conditions
- Fault modeling can uncover gaps in the analysis and forces analysis of Ground / Vehicle integration early
- Anomaly Detection can lead to early intervention, prevent further system damage, and reduce remediation cost and effort
- Captures subsystem design knowledge
- Provides a pathway for prognostic capabilities and Condition Based Maintenance V.S. Reactive Maintenance

## Benefits will be assessed through benchmarking, performance testing, etc.

- Initial requirement is fault isolated <= 1 second after fault detected

# Summary

- NASA will continue to pursue a robust Fault Management approach
- Fault Detection , Isolation and Anomaly Detection capabilities developed for Constellation program will be applicable to 21st Century Space Launch Complex as well as other programs
- Automated and Autonomous Response as well as Prognostics continue to be matured as technologies
- Acknowledgments – The material in this presentation was derived from work developed and publicly presented by the following: NASA ARC: Barbara Brown, David Iverson, Lilly Spirkovska, David Hall. NASA MSFC: Stephen Johnson. NASA KSC: Jose Perotti, Bob Ferrell. KSC ARSC: Becky Oostdyk

# Thank You