	Presenter Michael Aguilar
	Date January 24, 2012

Fault Management Using Model Based System Engineering (MBSE) Tools and Techniques

NASA Engineering and Safety Center (NESC)

Michael Aguilar (GSFC)
NASA Technical Fellow in Software
NESC Software Discipline
January 24, 2012

JWST ISIM C&DH Core Flight Software (FSW)

Integration of the JWST ISIM C&DH Core Flight Software (FSW) development at GSFC and the Science Instrument FSW applications developed by the following teams.

- European Space Agency (ESA) - NIRSpec Instrument Control Electronics Flight Software
- Canadian Space Agency (CSA) and EMS Technologies - Fine Guider Subsystem (FGS) Flight Software
- University of Arizona and Lockheed Martin ATC - NIRCam Instrument ICE and FPE Flight Software
- GSFC Subsystem Software Support - NIRSpec FPE Flight Software and the Micro-Shutter Array (MSA) Flight Software
- JPL/European Consortium (EC) - MIRI FPE and ICE FSW applications and Cooler Control Electronics (CCE) Flight Software

Instrument mechanisms functional ONLY when operated at 39Kelvin!

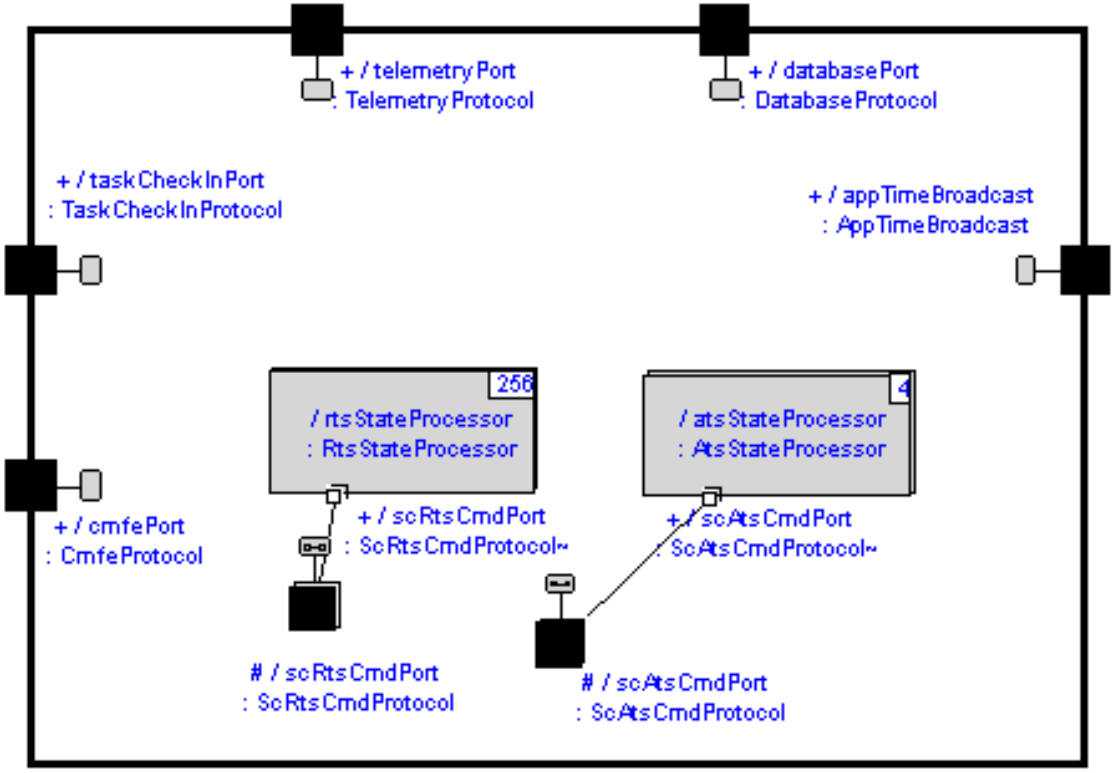
James Webb Space Telescope

This briefing is for status only and does not represent complete engineering data analysis

How do you integrate and test FM on a system that cannot be assembled and tested as one system?

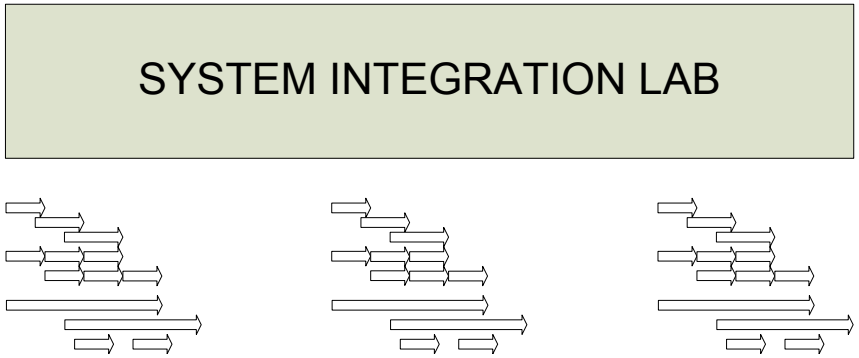
JWST FSW Rational Rose RT State Diagram Structure Diagram

- Public ports communicate with the other subsystem components
- Private ports communicate with the local capsules
- 256 Relative Timed Sequences (RTS) and 4 Absolute Timed Sequences (ATS) execute in parallel
- Code Generation
  - Task distribution
  - Message distribution



Support of Distributed Development Analysis, Evaluation, Integration, and Test

- Software Simulation can be copied and distributed as software-only integration lab
- Flight Software with COTS Hardware Interfaces promotes initial software interface verification and support through external subsystem testing, including environmental tests
- Ground System / Ops Applications with COTS interfaces supporting uplink and downlink environment



- JWST has 12 integration labs (strings) running to support SW development, SW Test development, and SW Test runs
  - 6 for GSFC, 2 for JPL, 2 for CSA, 2 for ESA
  - Exported Integration Labs provided software development environment and Ground System identical to NASA
  - Hardware interfaces were flight-like
- JWST integration at instrument subsystem level not possible due to environments required by instruments
- JWST mitigated risks at software interface, hardware interface, ground system interface
  - Exported integration labs provided consistent ground system development from developer, through integration, and into operations

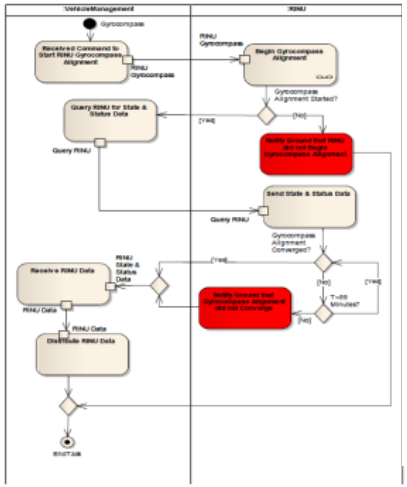
This briefing is for status only and does not represent complete engineering data analysis

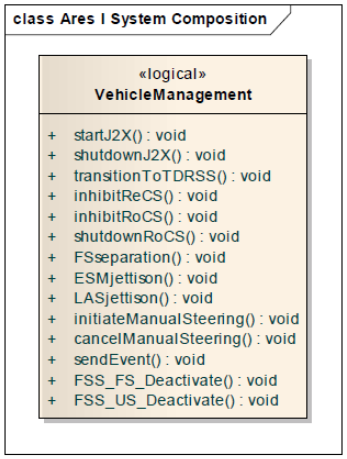
Can paper documentation be “executed” to verify FM behavior on a system that cannot be assembled and tested as one system?

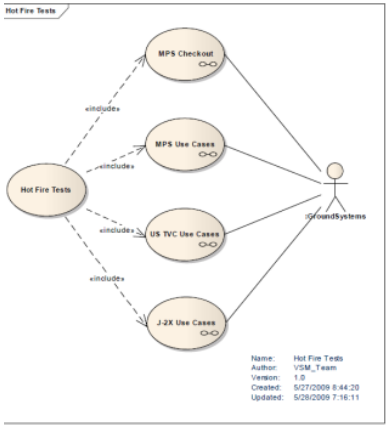
Constellation Program L2 and L3 imported to executable models to verify ICD correctness & completeness

- CxP focused on ICD ***documents***, not ICD ***models***
- What is an ICD in CxP? – different views of the same model
  - ICD documents focus on networking & information encoding, packets & protocols: CxP 70091, 70180, 70187, 70189, 70190
  - CxP 70078 (CSADD) focuses on high-level description & analysis of the systems-of-systems interactions
- CxP 70078 (CSADD) & L2 SAVIO model
  - A good but labor-intensive example of using standards (SysML, DODAF)

Integration of Vehicle System Manager (VSM) and Mission Timeline







Name: RINU Flight Operations
Description: This is a continuous task during flight for the RINU to supply State and Status Data to Navigation
Pre-condition: Autonomous Operations
Entry Event: RINU transition to Inertial Navigation
Off Nominal Event: Invalid data is received from RINU
Exit Event: End of Mission or powerdown
Related Use Cases: N/a
Actors: Navigation; RINU; Data Quality

Initiator Action	Subsystem Response
(Navigation)	
1. Navigation begins querying RINU. It takes multiple queries to get all of the State & Status data. The Delta Thetas will be queried at 50 Hz. The position, velocity, and attitude will be queried at 10 Hz with a goal of 50 Hz. Health & Status will also be queried at 10 Hz.	
	(RINU)
	2. The RINU sends the state & status data in multiple messages.
	(Data Quality)
	3a. Data Quality checks the data messages from RINU for full functionality. If the data indicates full functionality, the State and Status Data is sent to Navigation.
	3b. If the data is not indicating full functionality, notify Caution & Warning of RINU redundancy remaining and then send the data to Navigation.
(Navigation)	
4. Navigation processes and distributes RINU data to complete one cycle of the continuous task per GN&C SDD Section 3.5.2.	

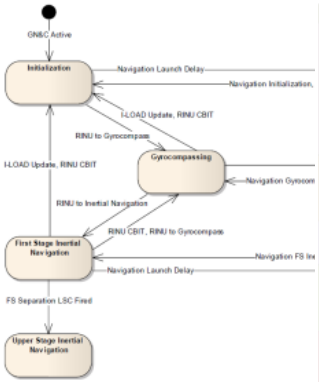
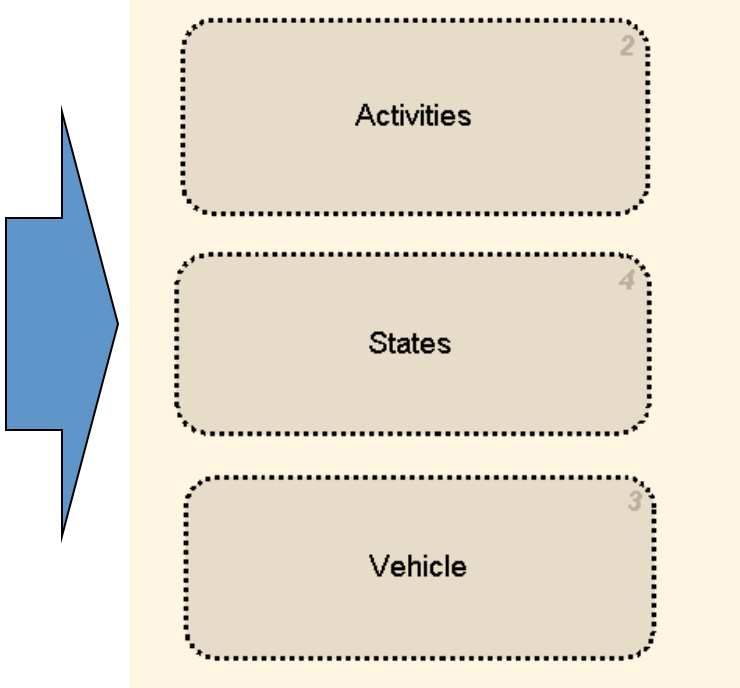


Table 4.2.2.1.2-1 Navigation States versus Vehicle States and Sub-states



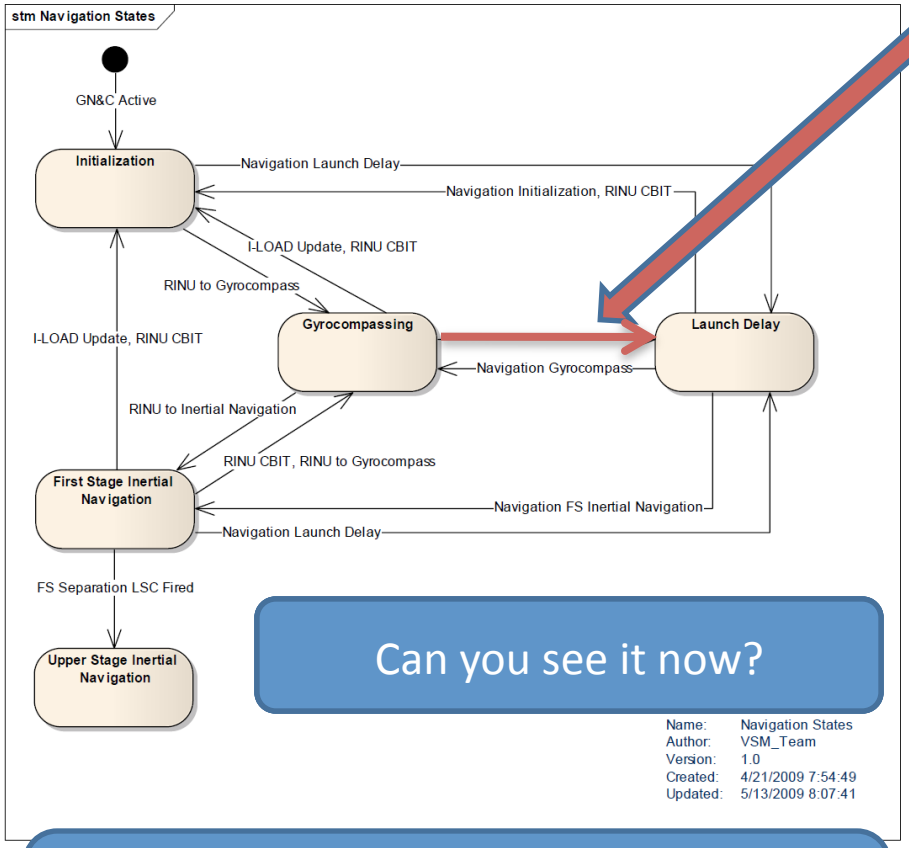
Enterprise Architect UML models of the VSM (e.g., activity diagrams, state charts, use cases) (Non-Executable)

Stateflow Representation of the VSM models (Executable)

This briefing is for status only and does not represent complete engineering data analysis

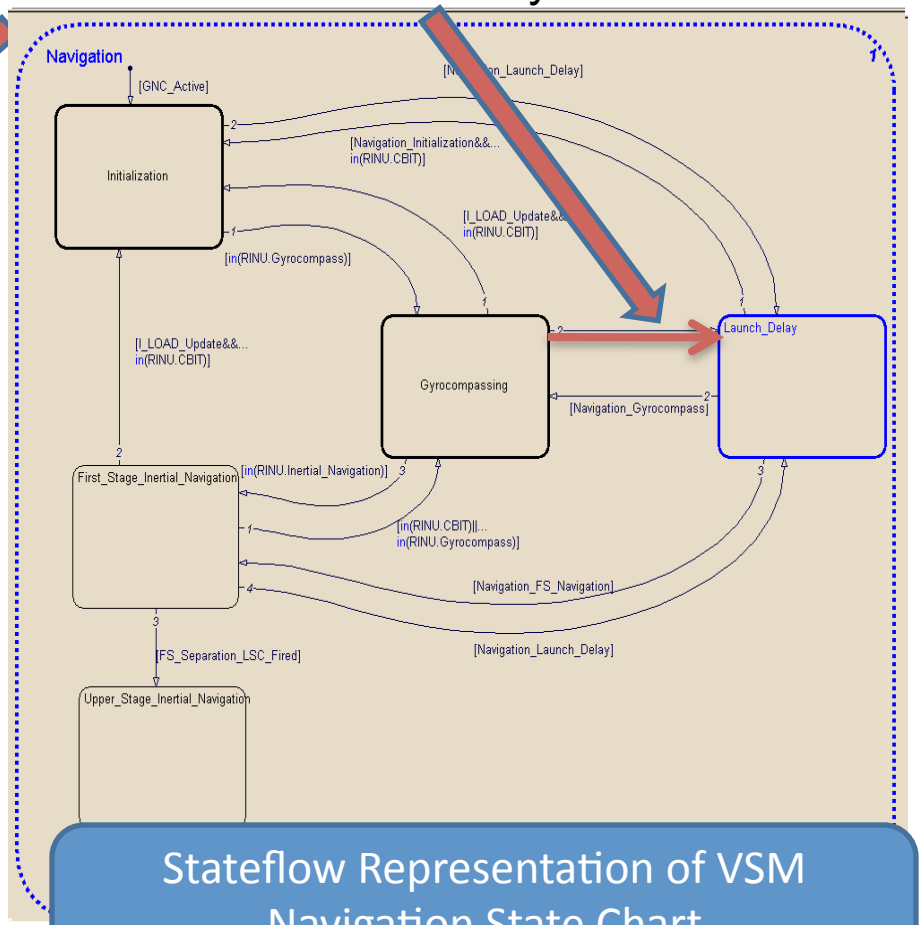
Catching Errors in VSM Design

Missing guard, causing unwanted entry into launch delay state



Can you see it now?

Enterprise Architect UML Representation of VSM Navigation State Chart (Non-Executable)



Stateflow Representation of VSM Navigation State Chart (Executable)

This briefing is for status only and does not represent complete engineering data analysis

Similar Errors Found through Modeling

Found by Tool

```
Runtime error: State Inconsistency
Block Name: VSM/Integrated (#1194 (0:0:0))
Active cluster state has no active substates
State Control (#1331 (0:0:0))
Active cluster state has no active substates
State FSS (#1429 (0:0:0))
Active cluster state has no active substates
State Guidance (#1309 (0:0:0))
Active cluster state has no active substates
State MPS (#1389 (0:0:0))
Active cluster state has no active substates
State Navigation (#1317 (0:0:0))
Active cluster state has no active substates
State Steering (#1370 (0:0:0))
```

Brief examples of errors found

Vehicle
Diagram does not have any active states prior to “Power applied to PDCU 1 and 2”

Vehicle.Prelaunch
All returns from Launch_Hold are TBD

Vehicle.Abort
No default state. Suggest adding default state as “Abort_Standby”

FSS, MPS, Guidance, Navigation, Control, and Steering
No default state, suggest adding “Standby” or “Idle”

Navigation
After Gyrocompassing, the Navigation will automatically enter the Launch_Delay state. If this is unintended, suggest placing guard to prevent this.

RINU and RGA
Both RINU and RGA use “PowerApplied” ,“OffCommand”, “PerfomCBIT”, and “gyrocompass” to transition between states.

FSS, US_TVC,
Unclear as to which are OR conditions and which are AND conditions

CTC and Imaging
Does not enter “Off” state if power is removed.

Ares-Orion Communication During Abort

Table 8-1. Proposed Orion-Ares I communications list.

Message Name	Direction	Function	Parameters
Abort Recommendation	Ares → Orion	The Ares Abort Manager sends an Abort Recommendation to the Orion when it has detected and confirmed an abort condition. Critical message, to be repeated every 40ms until message receipt confirmation is registered.	Abort Condition ID (ACID), Time to Criticality (TTC), Timestamp
Abort Recommendation Receipt Confirmation	Orion → Ares	Orion acknowledges the receipt of an Abort Recommendation message	ACID, Original Message Timestamp
Initiate Abort	Orion → Ares	Orion commands Ares to transition into ABORT MODE and perform phase specific safing actions. Critical message, to be repeated until message receipt confirmation is registered or until such time as the abort separation is made.	ACID, Timestamp
Initiate Abort Receipt Confirmation	Ares → Orion	Ares I acknowledges the receipt of an Initiate Abort message.	ACID, Original Message ID
		s initiated an uins in eated until tered.	ACID, Timestamp
		n Abort	ACID, Original Message ID
		, if the crew Abort Override Abort essage, to be mation is	ACID, Timestamp
		n Abort	ACID, Original Message ID

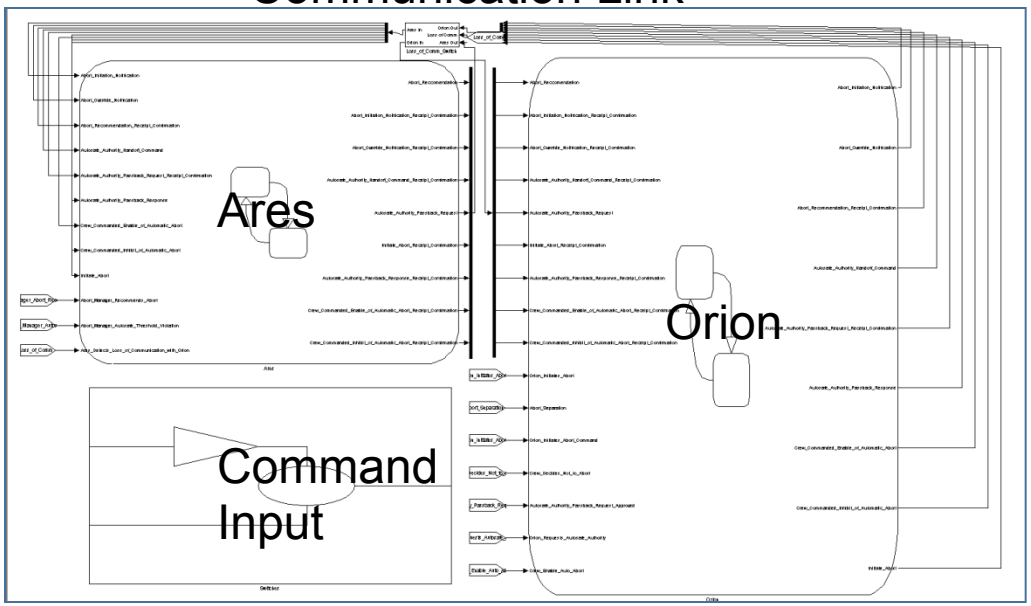
NASA
National Aeronautics and Space Administration

CxP 72244
REVISION: RFI FOR REV A
RELEASE DATE: DECEMBER 17, 2009

ARES I FAILURE DETECTION, NOTIFICATION, AND RESPONSE (FDNR) SYSTEM DEFINITION DOCUMENT (SDD)

SENSITIVE BUT UNCLASSIFIED (SBU) – ITAR
SBU designated made by Dr. Michael Watson, MSFC/EV43, Ares Projects, on May 25, 2007.
The electronic version is the official approved document.
Verify this is the correct version before use.

Communication Link



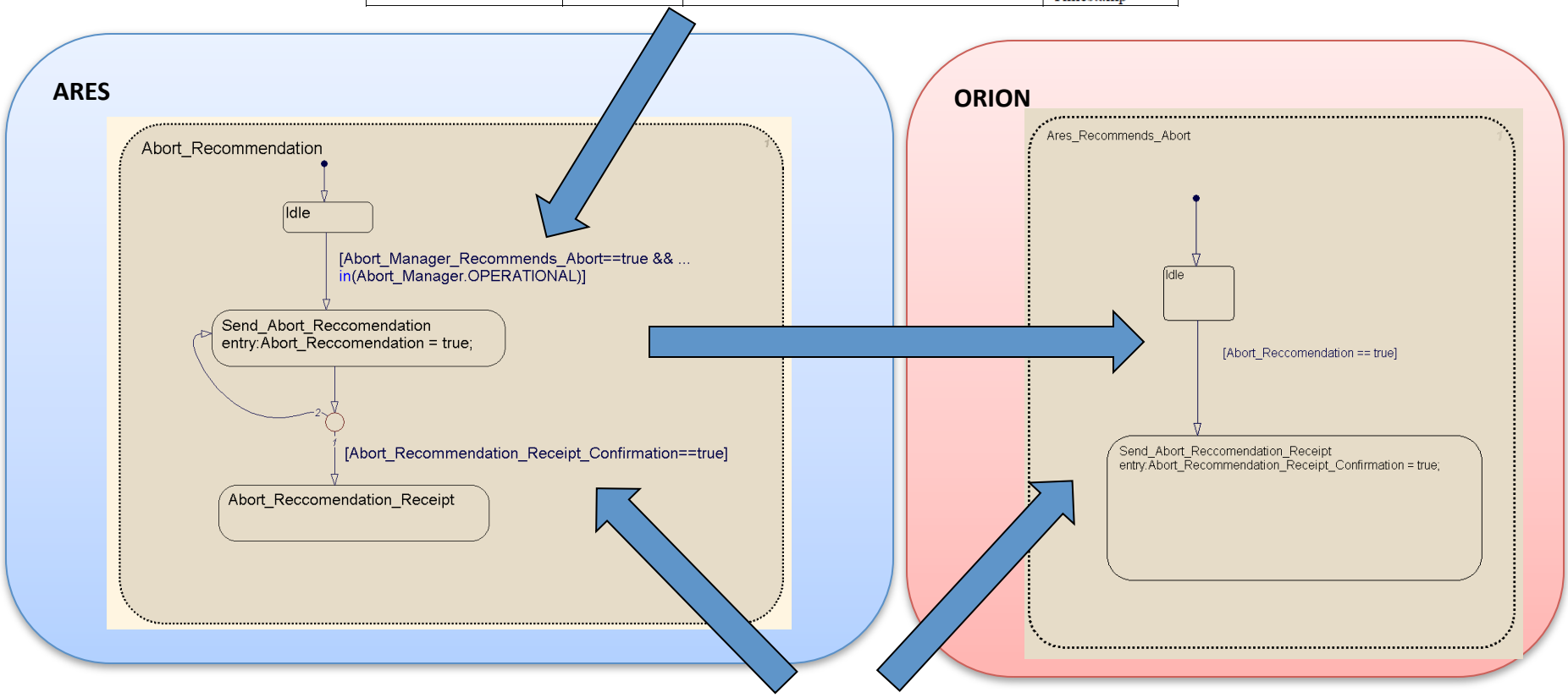
Documentation of Ares-Orion Communication During Abort (Non-Executable)

Stateflow Representation of Ares-Orion Communication During Abort (Executable)

This briefing is for status only and does not represent complete engineering data analysis

Conversion from Tabular Form to Stateflow Example

Message Name	Direction	Function	Parameters
Abort Recommendation	Ares → Orion	The Ares Abort Manager sends an Abort Recommendation to the Orion when it has detected and confirmed an abort condition. Critical message, to be repeated every 40ms until message receipt confirmation is registered.	Abort Condition ID (ACID), Time to Criticality (TTC), Timestamp

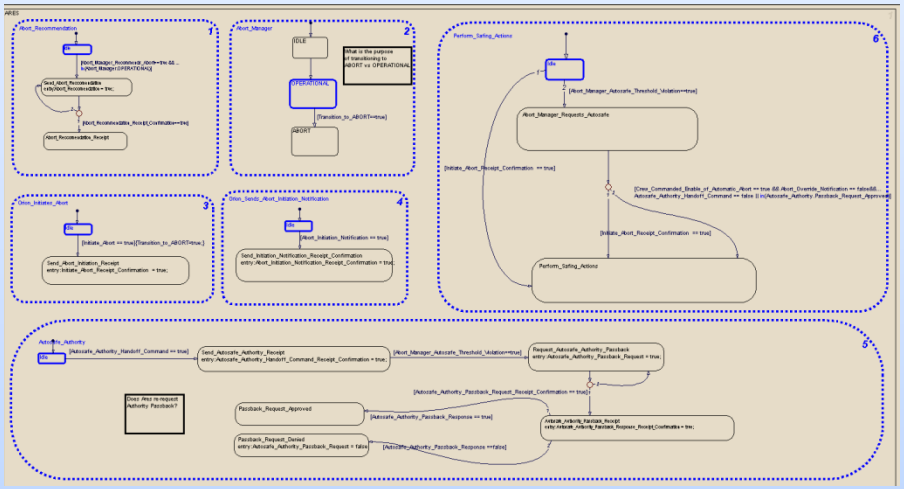


Message Name	Direction	Function	Parameters
Abort Recommendation Receipt Confirmation	Orion → Ares	Orion acknowledges the receipt of an Abort Recommendation message	ACID, Original Message Timestamp

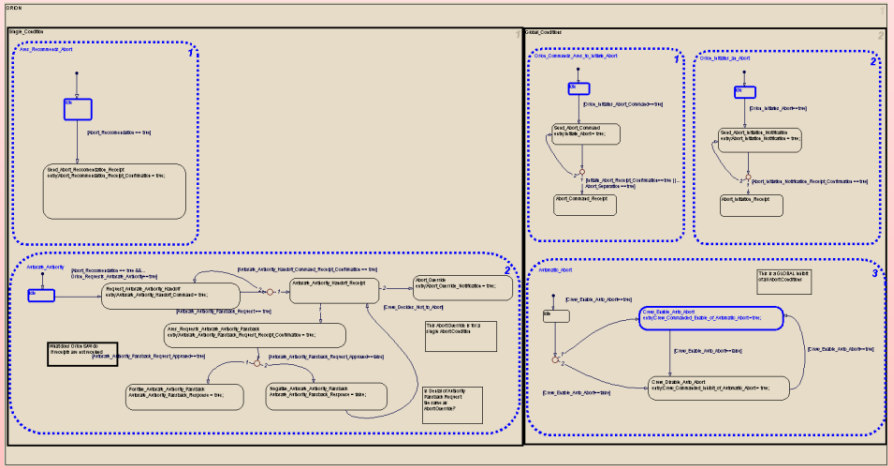
This briefing is for status only and does not represent complete engineering data analysis

Trace through “Nominal” Abort Interaction

ARES



ORION



Pre-Abort Recommendation:

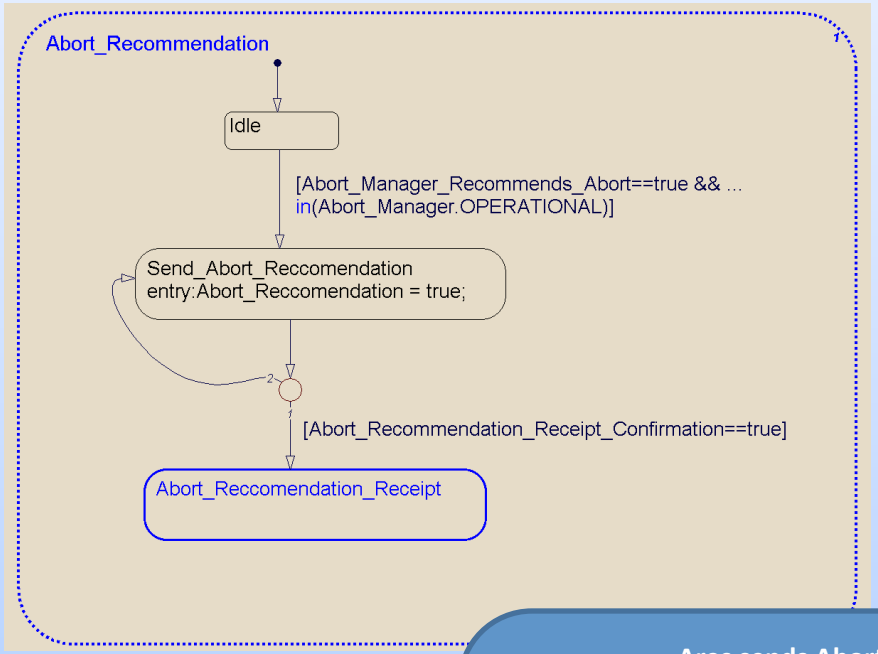
-**Ares:** All command states “idle” except for “abort manager”, which is in the “operational” state

-**Orion:** All command states “idle” except for “crew enable auto abort”, which is in the enabled state

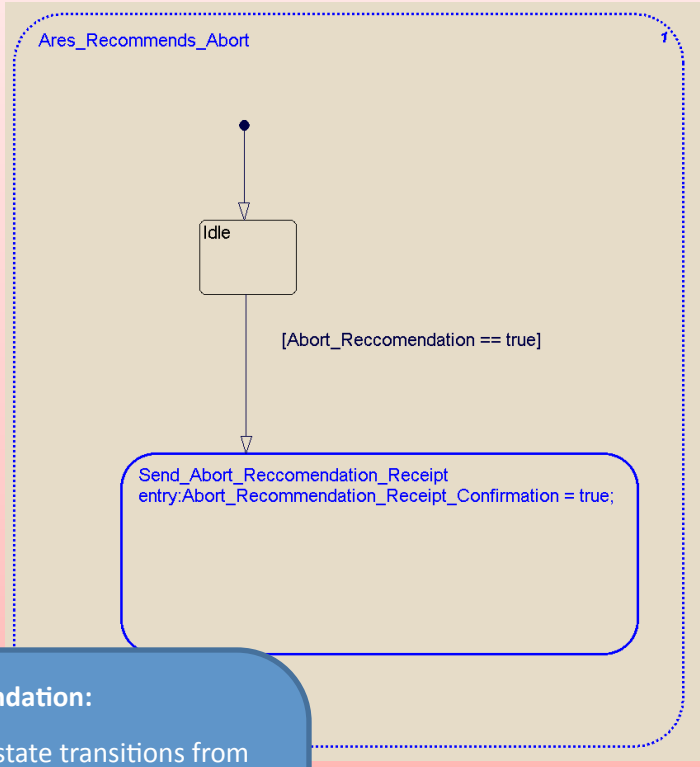
This briefing is for status only and does not represent complete engineering data analysis

Trace through “Nominal” Abort Interaction

ARES



ORION



Ares sends Abort Recommendation:

-**Ares:** Abort Recommendation command state transitions from “idle” to “send abort recommendation” and waits for Orion response. All other command states remain “idle”

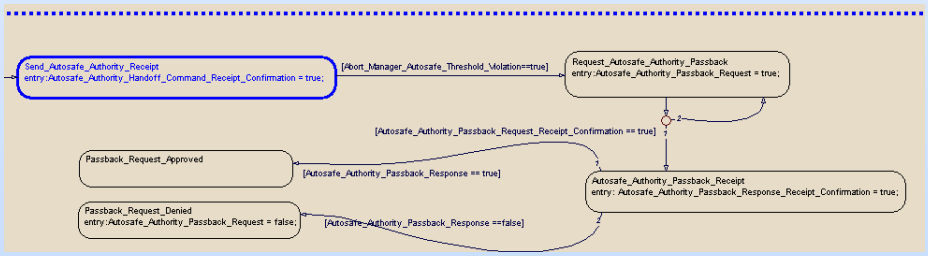
-**Orion:** Ares recommends abort command state transitions from “idle” to “send abort recommendation receipt” upon receipt of abort recommendation

-**Ares:** The abort recommendation state transitions from “send abort recommendation” to “abort recommendation receipt “

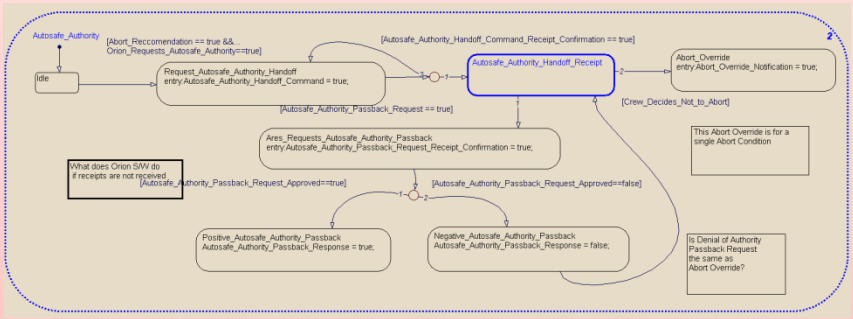
This briefing is for status only and does not represent complete engineering data analysis

Trace through “Nominal” Abort Interaction

ARES



ORION



Orion Requests Auto-safe Authority:

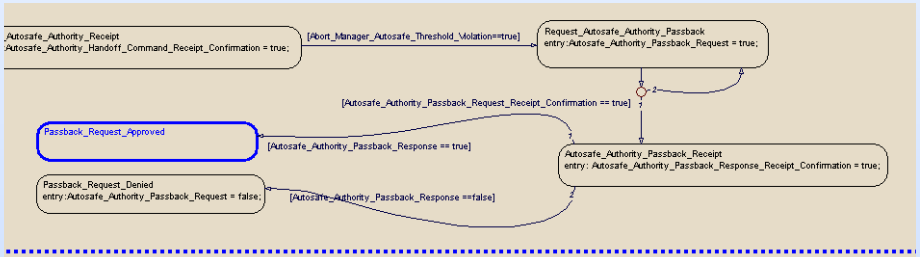
-**Orion:** Orion requests auto-safe authority after an Ares abort recommendation is received and waits for Ares to send an Auto-safe authority handoff receipt.

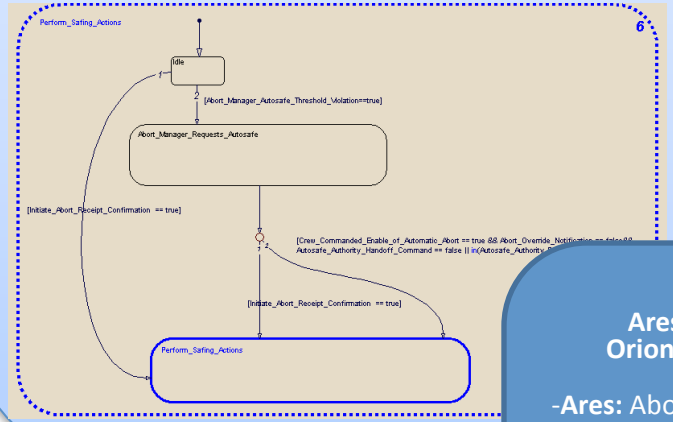
-**Ares:** Ares receives the auto-safe authority request from Orion and sends a handoff receipt and continues to monitor conditions.

This briefing is for status only and does not represent complete engineering data analysis

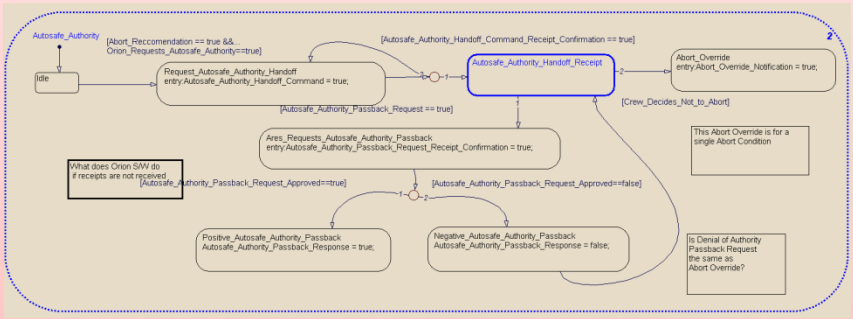
Trace through “Nominal” Abort Interaction

ARES





ORION



Ares Abort Condition Exceeds Auto-safing Threshold
Orion Approves Auto-safe Authority Pass-back Request:

-**Ares:** Abort condition exceeds the auto-safing threshold. Because Orion has the auto-safe authority, Ares sends an “auto-safe authority pass-back request” and waits for Orion to send receipt.

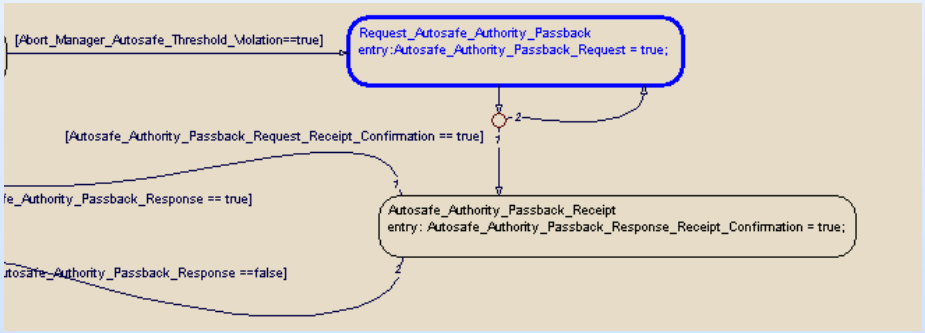
-**Orion:** Orion receives the auto-safe authority pass-back request and decides whether to approve or deny it. Orion approves the pass-back request.

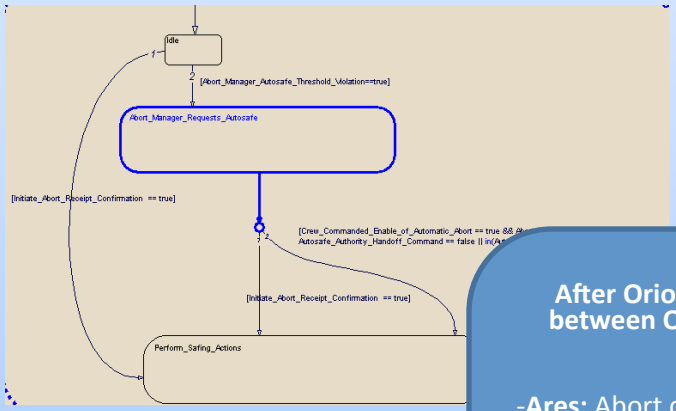
-**Ares:** Upon receipt of the positive authority pass-back response, Ares performs safing actions.

This briefing is for status only and does not represent complete engineering data analysis

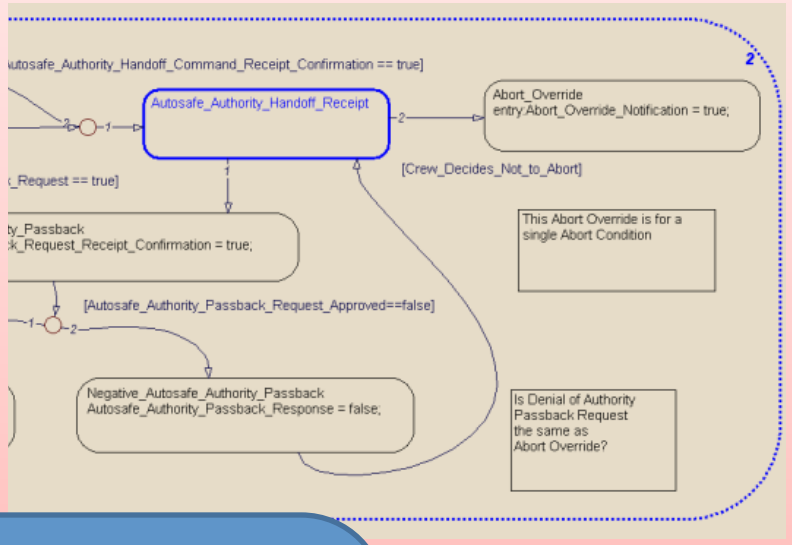
Trace through Interaction Loss of Comm. Scenario

ARES





ORION



Model Answers What if Scenario

After Orion Requests Auto-Safe Authority, Communication between Orion and Ares is Lost and Ares Exceeds Auto-Safe Threshold:

-**Ares:** Abort condition exceeds the auto-safing threshold. Because Orion has the auto-safe authority, Ares sends an "auto-safe authority pass-back request" and waits for Orion to send receipt. (Same as in "nominal" scenario)

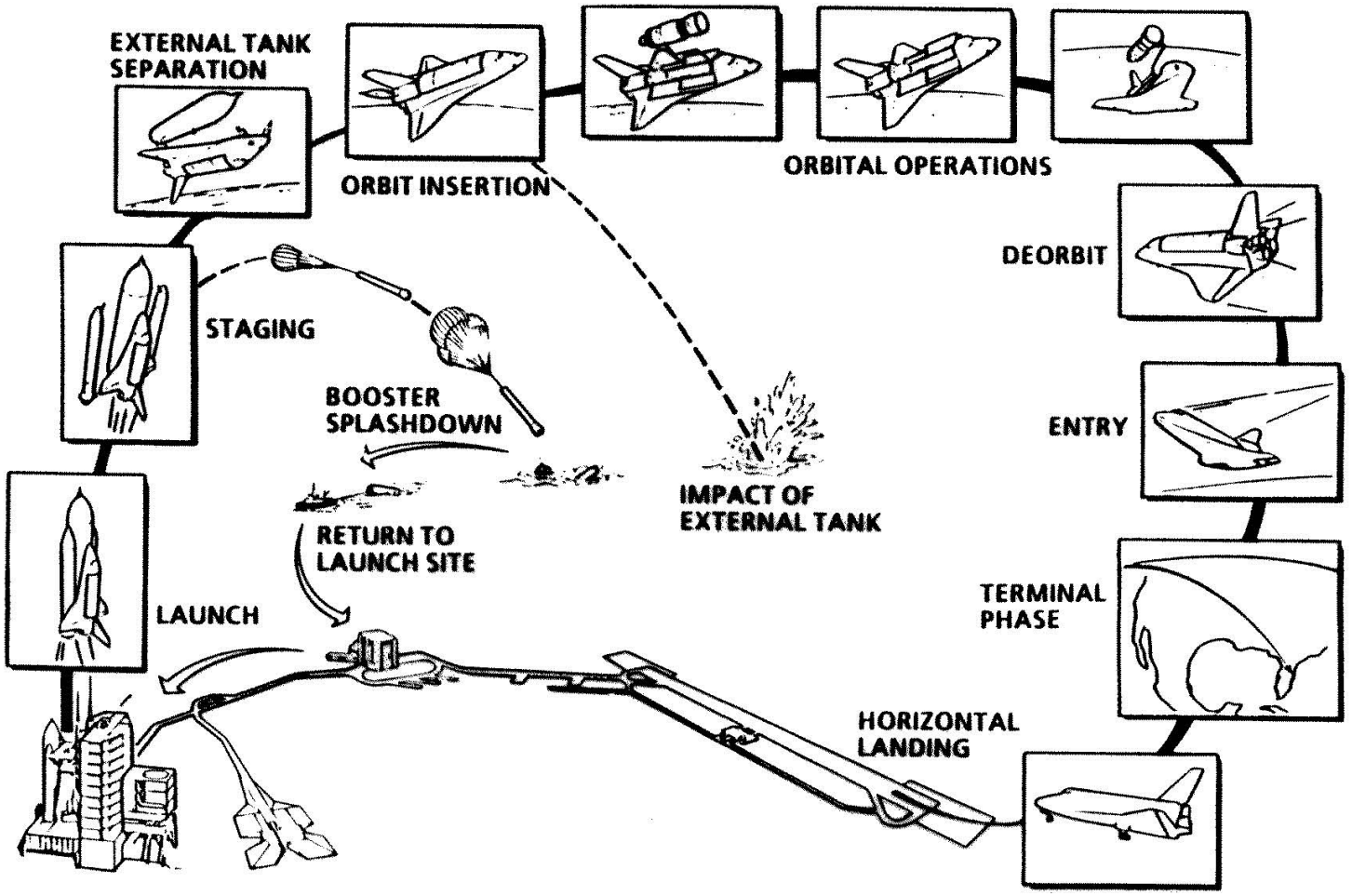
-**Orion:** Because communication is severed between Ares and Orion, Orion is not aware of the auto-safe authority request and remains in prior states.

-**Ares:** Ares can NOT perform safing actions.

This briefing is for status only and does not represent complete engineering data analysis

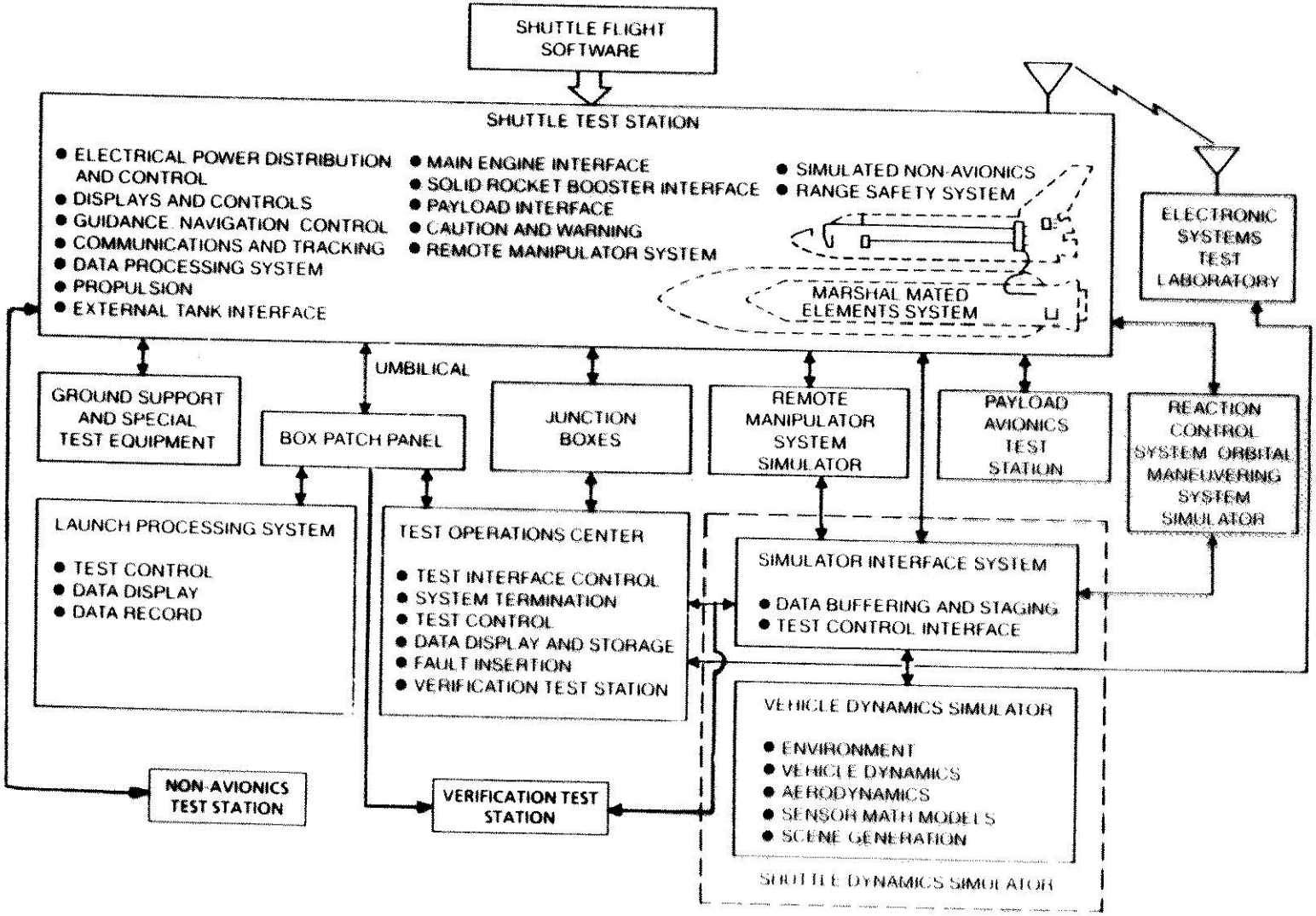
Can models of major system interfaces allow testing and verification of the systems and reduce the complexity of verification labs?

Shuttle Avionics Integration Lab (SAIL) Mission Capability



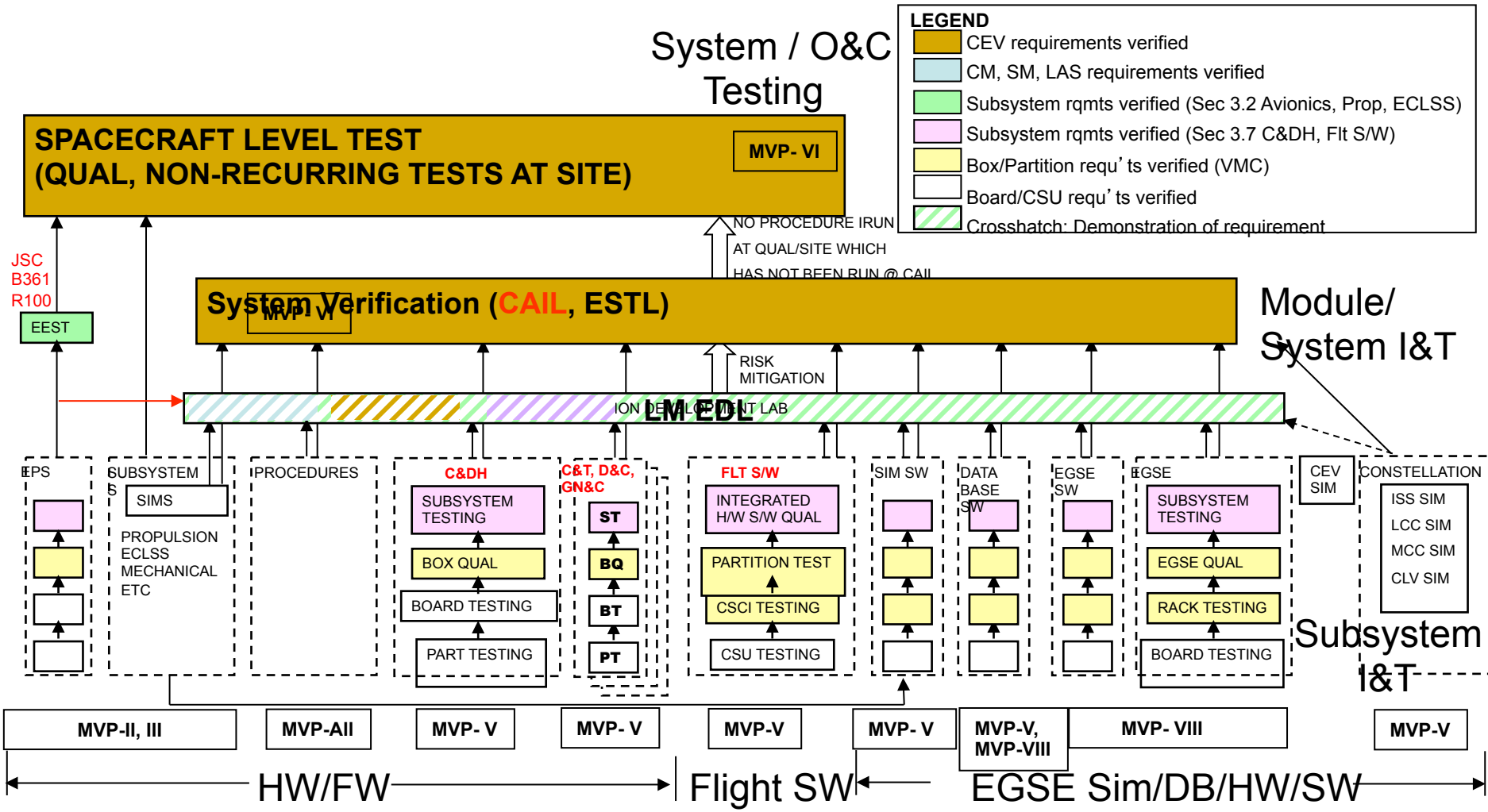
This briefing is for status only and does not represent complete engineering data analysis

SAIL Functional Test



This briefing is for status only and does not represent complete engineering data analysis

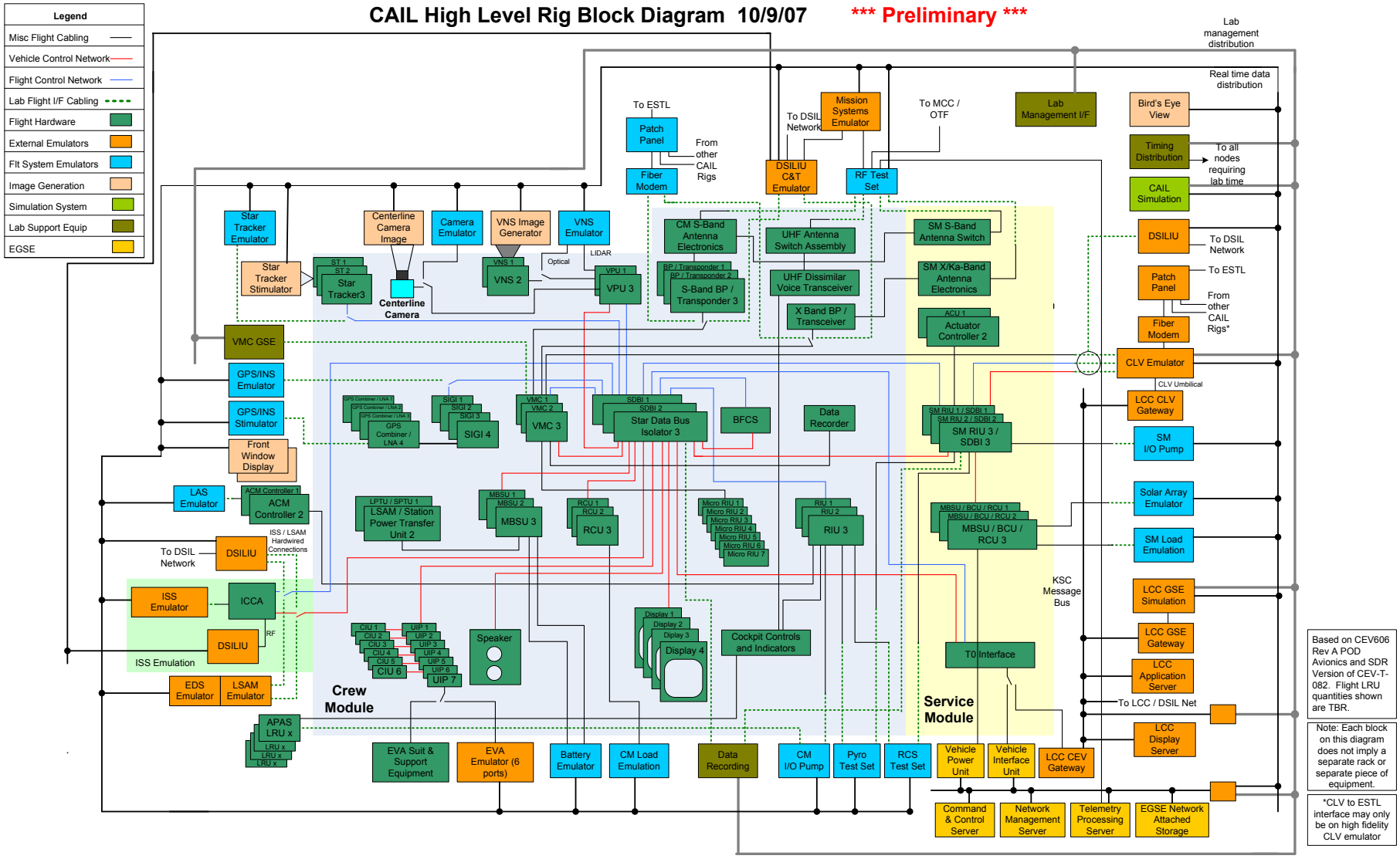
Constellation Avionics Integration Lab (CAIL) R&R in Avionics & Software V&V Plan



AV/SW Verification Aligns With CEV Verification Flow Reflected In Master Verification Plan (MVP)

This briefing is for status only and does not represent complete engineering data analysis

CAIL Test Rig Design



This briefing is for status only and does not represent complete engineering data analysis

Can models of existing software provide adequate and effective analysis of the software and hardware system?

Toyota Camry Source Code

- Source Code Language: C (originally Bell Labs, 1972)
  - Code is directly readable, comments are in Japonese
- Code Size: Over 280,000 lines of code
- Number of Modules: Over 100
- Number of files: Over 2500
- RTOS used: OSEK Compatable Operating System (NEC)
- Compiler used: Greenhills C Compiler (USA)
  - Code was compiled and executed during study
- Target Microcontrollers: NEC and DENSO ASIC designs

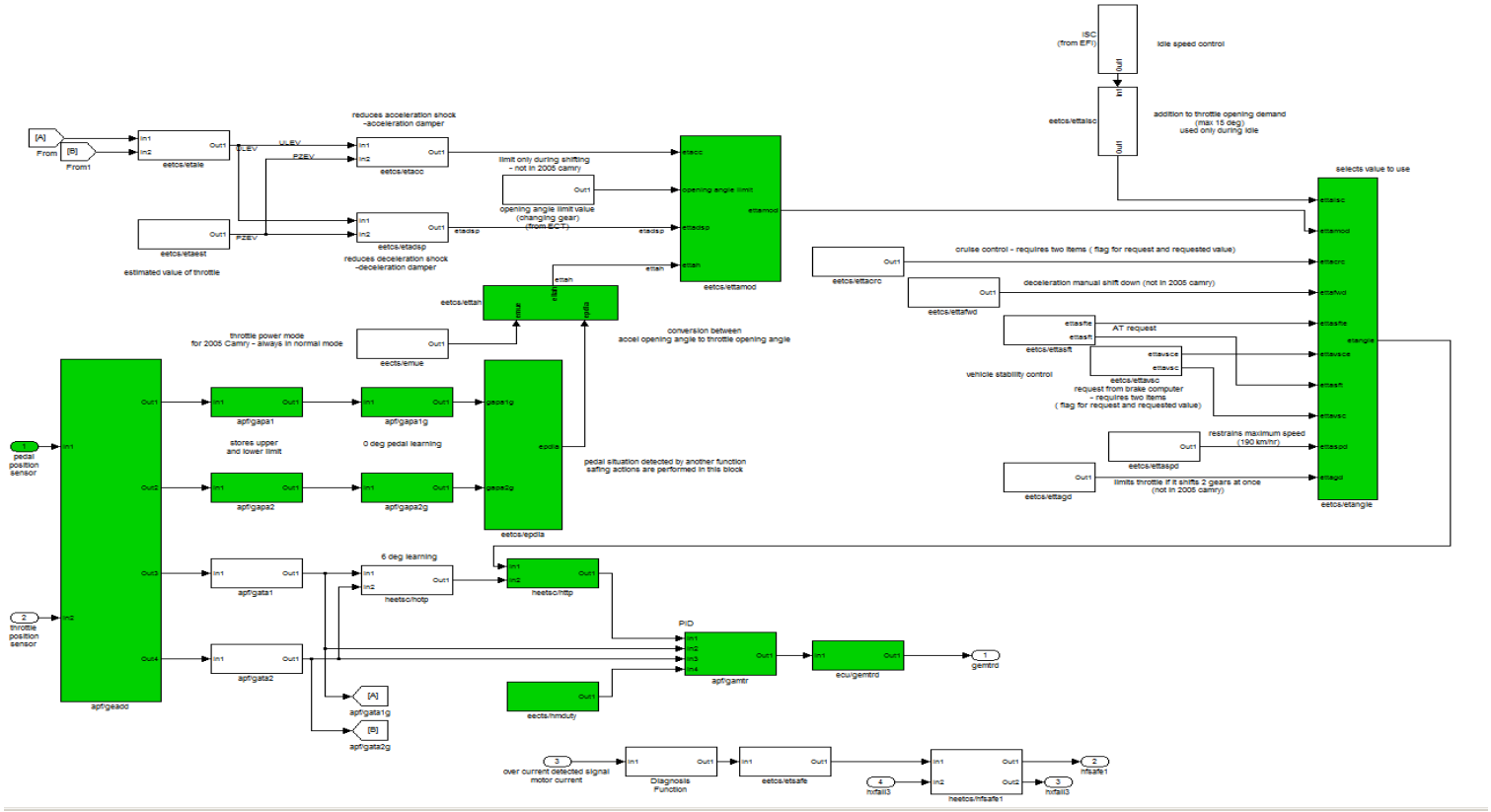
Software Algorithm Design Process

1. Identify and model the Camry algorithms from source code
2. Identify and model scheduling and timing of the Camry algorithms
3. Build executable math models that behave identically to the Camry algorithms, but can be executed within a Matlab environment on a PC
4. Stimulate models using input data of interest
5. Examine outputs of model for further analysis
6. Continuously upgrade and verify model against vehicle hardware

7. Support the development of final test sequences for test upon Camry vehicles

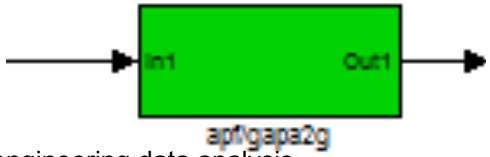
This briefing is for status only and does not represent complete engineering data analysis

Software Algorithm Models



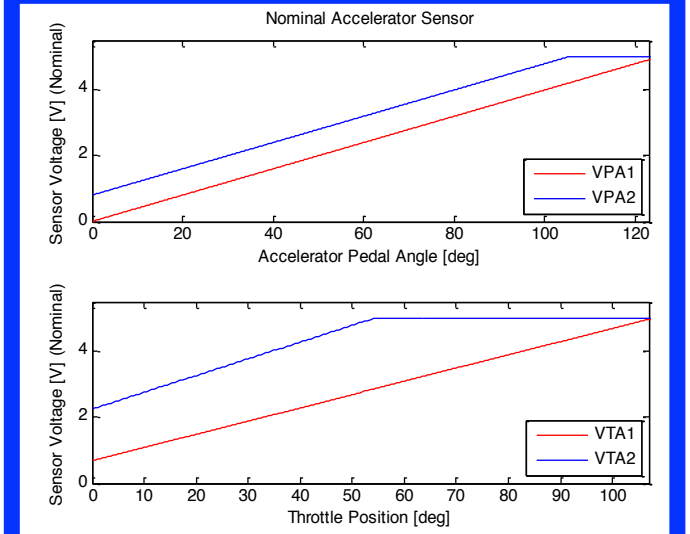
Modeling Process

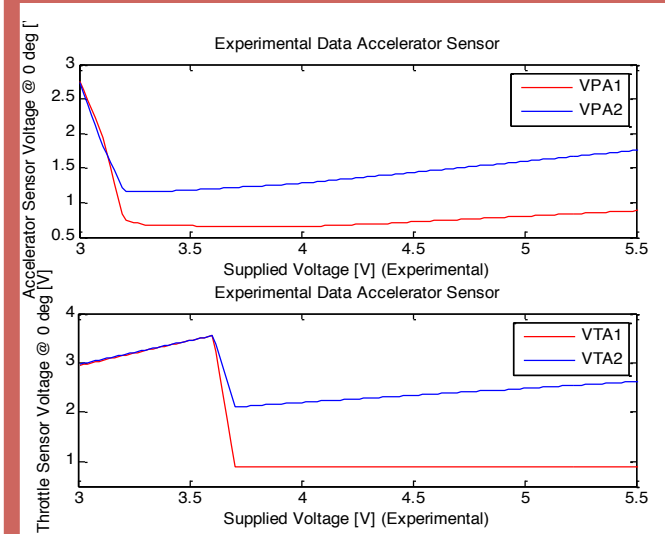
C Source Code → S-function →

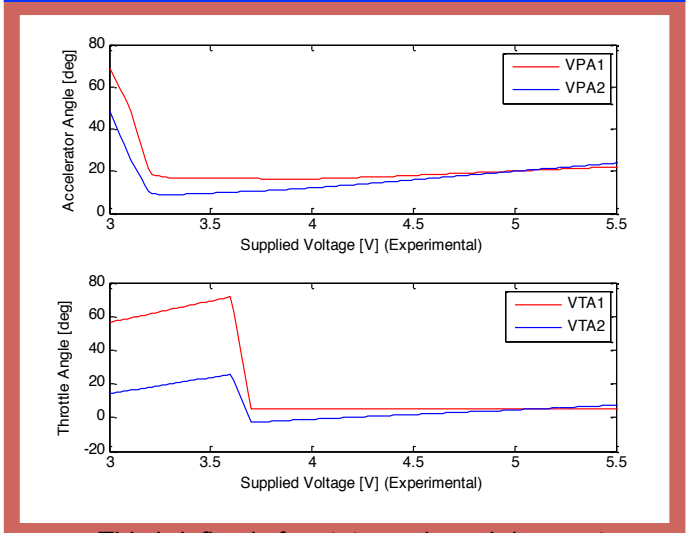


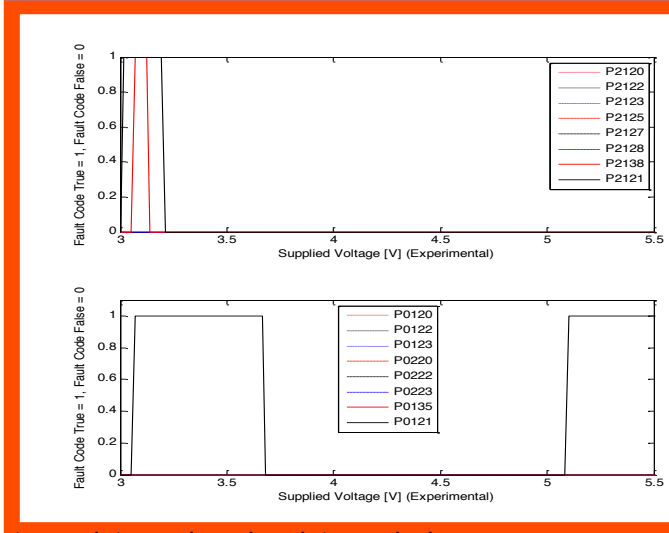
This briefing is for status only and does not represent complete engineering data analysis

Simulink Model Analysis









This briefing is for status only and does not represent complete engineering data analysis

Simulink Models Studied

- Functional Document Derived ETC Models Completed
  - Pedal Learning Algorithm
  - Throttle Learning Algorithms
  - Diagnostic Codes
  - Pedal and Throttle Sensor Characteristics
  - Pedal Angle to Throttle Commanded Angle Conversion
  - Throttle Valve Motor PID controller
  - Cruise Control
  - Idle Speed Control Module
- Enhanced Model Fidelity
  - Toyota C-code compiled within models to increase fidelity
- Functional Analysis without Modeling
  - Vehicle Stability Control Torque Limit Checking
  - Transmission Control Switch Diagnostics

Spin Logic Analysis

- Various Focused Camry Models
  - Spin Accelerator Pedal Position Learning Algorithm analysis initiated
  - Spin and C-code Accelerator Pedal Position Learning Algorithm analysis initiated
- Logical Model of Camry Software
  - Conversion of the Stateflow models into verifiable Spin models, to allow exhaustive analysis of the key properties
  - Increased fidelity of Stateflow model for further Spin analysis

This briefing is for status only and does not represent complete engineering data analysis

Spin Logic Analysis Conclusions

Model	Type	Conclusion
Interrupt Enable/ Disable Pairing	Computation	Verified
Accelerator Pedal Learning	Computation	Inconclusive
Sensor Input	I/O	Potential Issue
Motor Drive IC	Computation	Verified
Port Register Input	I/O	Verified
PWM Functionality	Computation	Potential Issue

This briefing is for status only and does not represent complete engineering data analysis

Software Timing Performance Analysis Results

- Statistical Analysis
  - Toyota provided sample sets of software task performance
    - 250 samples per second, 900,000 samples per hour
    - Test bed configured at 6000 rpm, 8000 rpm, and 9000 rpm
  - The data analysis should sufficient margin under these maximal and stressed rpm configurations.

- Worst Case Execution Time (WCET) Analysis
  - Static Analysis using aiT Tool
    - V850 Model Processor used as closest model
    - Source code "executed" within aiT Tool
    - Worst Case Execution Time determined in static test
  - Recursion issues prevented complete analysis

This briefing is for status only and does not represent complete engineering data analysis

On-Ramps?

- Identification and scope of changes/tradeoffs in system behavior.
- Interfaces protocol requirements verification.
- Failure modes / safing / fault management.
- Boot up / Abort sequencing.
- Subsystem verification / system verification.
- Test bed support, operations and maintenance support.
- Documentation consistency.