Model Transformation for a System of Systems Dependability Safety Case

by

J. Murphy & S. Driskell

Presented by
Judy Murphy
jmurphy@mpl.com
• Overview
  – Dependability & Safety Overview
  – Who, What, Why
  – Engineering Safety Process
  – Application to Non-Space Environment

• Phase I Dependability & Safety
  – Where We Started

• Phase II Dependability & Safety
  – Where we are today
  – Model Transformation

• Conclusion
Acknowledgement and Disclaimer

- This research was sponsored by the NASA IV&V Facility
- The views and conclusions in this talk are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government

Dependability and Safety Effort

- Address the need to identify safety-critical software requirements along with corresponding faults so that potential hazards may be mitigated early in the development of a System of Systems (SoS)
- Provides a proactive approach to the independent validation of safety requirements for systems of systems
- Provides a reusable set of artifacts for any family of spacecraft
- Provides a philosophy that can be applied to any industry
- Approach
  - Move away from mission specific device fault conditions
  - Identify, compare and contrast subsystems
  - Create fault models based on functionality vs device functionality
- Multi-phase project
  - Phase I – Initial mission specific dependability and safety case
  - Phase II – Creation of generic fault conditions for cruise/orbit
  - Phase III – Creation of fault conditions for experiments and
  - Phase IV – Creation of fault conditions for surface operations for planetary robotic missions

Are all hazards identified and mitigated
Who, What, Why

• The mission of NASA’s IV&V program, under the auspices of the NASA Office of Safety and Mission Assurance (OSMA), is to provide the highest achievable levels of assurance for mission- and safety-critical software. The NASA IV&V Program provides assurance to our stakeholders and customers that NASA's mission-critical software will operate dependably and safely.

• The NASA IV&V Program is building upon Phase 1 of spacecraft safety case study for a reusable set of artifacts for fault identification.

• Mission success and spacecraft safety are both improved through contingency hazard management and the resulting failure risk reduction.
Safety Engineering Process

• Starts with the system safety engineering activities to identify potential hazards and safety-critical functions, which are then traced through design into safety-critical hardware and software functions.

• Ends with validation and verification (V&V) of derived software safety requirements for controlling the hazard causal factors.

• Team of software engineers, who are not the members of the development team, are tasked to validate and verify the SoS’s software and requirements.
Application to Non-Space Environment

- **Business models and strategies for product:**
  - **Spacecraft Families**
    - Communication
    - Science
    - Remote sensing, etc
  - **Product Lines**
    - Cell phones, Computers
    - Medical devices, Cars
    - Financial products, etc

- **Economic valuation of products:**
  - **Spacecraft**
    - Successful launch
    - Successful pay load deploy
    - Successful science collection
  - **Product Lines**
    - New iPhone® launch - sales
    - Windows Vista® vs Windows 7® - sales
    - American vs foreign cars, etc

Thinking about the same thing in a different way
Application to Non-Space Environment

- Organizational and process designs for product
  
  **Spacecraft Processes**
  - NASA standards
  - MIL-STD-498
  - V-Model, CMMI, IEEE, etc
  
  **Product Line Processes**
  - CMMI, Six Sigma, Agile
  - Regulatory agency rules/regulations
  - IEEE, etc
  
- Service systems & their implications for product
  
  **Spacecraft Services**
  - Fault management
  - Telemetry downlink
  - Command handling
  - Experiment control
  
  **Product Line Services**
  - Cell phone alerts & applications
  - Interface design
  - Customer data access
  - Online selling
  - Onstar ®

We are not that different
Phase I Overview
Where We Started

- Built a dependability and safety case for safe-hold
  - Global Precipitation Measurement (GPM) mission
    - Studies global precipitation
  - Autonomous software for managing spacecraft hazards without ground intervention
    - Are all subsystem faults requiring safe-hold included in safe-hold monitor?
    - Are all safe-hold requirements identified?

- The IV&V analyses are model-based, striving to obtain goodness of product data in terms of three questions:
  - What is the system software supposed to do?
  - What the system software is not supposed to do?
  - What is the system software’s expected response under adverse conditions?
Process Creation

- Created a new IV&V analysis process
  - Started with an IV&V developed independent list of fault conditions specific to GPM
  - Based on previous mission experience and GPM knowledge
  - Used to help determine if there were gaps in the fault management & effects analysis (FMEA)

The right process identifies missing requirements
Process and artifacts are reusable
Sample SRM Artifact

- Mission specific safe-hold activity diagram for fault management

**SUBSYSTEMS - partial list**

**Command & Data Handling**
- Power supply part/connector failure
- RAD 750 part/connector failure
- Bulk memory part/connector failure
- Temperature & analog part/connector failure
- Payload a GPS part/connector failure

**Safety Mech. & Attitude Control**
- Propulsion I/F part/connector failure
- Solar array & high gain antenna I/F part/connector failure
- Attitude sensors and actuators I/F part/connector failure

**Guidance, Navigation & Control**
- Star tracker part/connector failure
- Sun sensor part/connector failure
- Inertial reference unit part/connector failure
- Magnetometer part/connector failure
- Reaction wheel part/connector failure
- Global positioning system part/connector failure

**Electrical Power Systems**
- Power monitor & control part/connector failure
- Battery part/connector failure
- Survival heater part/connector failure
- Subsystem I/F part/connector failure
- Instrument I/F part/connector failure

**High-level fault management – mission dependent**
Phase I End Result

- Ensure these hazards are managed and failure risk is reduced
- Deliver a reusable standardized spacecraft software safety case for IV&V
- Identify missing safe-hold requirements
- Provide software test scenarios
- IV&V efforts on other science missions have decided to build safety cases using this process
- This approach will be applied to other behaviors besides safe-hold
- Mapped IV&V first science list of fault conditions to Mars Science Laboratory (MSL); Fault and Failure Analysis (FFA) data
  - MSL FFA data is at a different level than the IV&V list of fault conditions, so activity was partially successful. Sufficiently and adequately mitigate the potential hazards posed to a SoS
  - This is being addressed in study Phase II
Model Transformation From Specific to Generic

- Moving from specific faults to generic faults
  - Faults are currently *device dependent* not functionality dependent
  - Faults are not always obviously or easily reusable on other missions
- Families of spacecraft may use the same underlying architecture
  - Subsystem device names are often different
- Create models and fault conditions based on the *functionality* of a subsystem at the highest level
- Created a process to go from specific to generic

Focus on functionality – not devices
• Compare space missions to each other
  – Share many of the same characteristics
  – All space missions have subsystems that deal with
    • Telemetry, command and data handling, guidance navigation and control, 1553 bus, temperatures, voltages, etc
  – Functionality of other missions uses pyrotechnics, robotic rovers and unique experiments
  – Those subsystems may have differing designs and device names, but the subsystem functionality is the common thread
Identify Subsystems

Review Subsystem/Device Functionality

Identify Third Party Rules, Regulations, Standards

Identify Common Functionality

Identify Fault Conditions Based on Functionality

Update IV&V Independent List of Generic Fault Conditions

Create/Update Generic Fault Models

Update Reuse Baseline

Provide Independent List & Models to Developers

Mission 1  Mission 2  Mission 3

Reusable process
Specific Safe-hold Fault Management Example

SUBSYSTEMS - partial list

Command & Data Handling
- Power supply part/connector failure
- RAD 750 part/connector failure
- Bulk memory part/connector failure
- Temperature & analog part/connector failure
- Payload a GPS part/connector failure

Safety Mech. & Attitude Control
- Propulsion I/F part/connector failure
- Solar array & high gain antenna I/F part/connector failure
- Attitude sensors and actuators I/F part/connector failure

Guidance, Navigation & Control
- Star tracker part/connector failure
- Sun sensor part/connector failure
- Inertial reference unit part/connector failure
- Magnetometer part/connector failure
- Reaction wheel part/connector failure
- Global positioning system part/connector failure

Electrical Power Systems
- Power monitor & control part/connector failure
- Battery part/connector failure
- Survival heater part/connector failure
- Subsystem I/F part/connector failure
- Instrument I/F part/connector failure

Mission device name dependent

The ground will create a command sequence that will put the spacecraft in safehold and command out of safehold.

Less flexible, less reusable
Generic Fault Management Example

**Subsystems** - Capability/functionality issues - partial list

**Command & Data Handling**
- Main spacecraft computer HW/SW issue
- Temperature & analog HW/SW issue
- Payload & GPS Subsystem IF HW/SW issue
- 1553 IF HW/SW issue
- Serial bus IF HW/SW issue

**Safety Mech. & Attitude Control**
- Propulsion IF HW/SW issue
- Solar array & antenna IF HW/SW issue
- Attitude sensors and actuators IF HW/SW issue
- 1553 IF HW/SW issue
- Serial bus IF HW/SW issue

**Guidance, Navigation & Control**
- Star tracking HW/SW issue
- Sun sensing HW/SW issue
- Inertial references HW/SW issue
- Magnetometer HW/SW issue
- Reaction wheel HW/SW issue
- Global positioning HW/SW issue
- 1553 IF HW/SW issue
- Serial bus IF HW/SW issue

**Electrical Power**
- Power monitoring & control HW/SW issue
- Battery HW/SW issue
- Survival heater HW/SW issue
- Subsystem IF part/connector failure
- Subsystem N IF part/connector failure
- Instrument/experiment IF HW/SW issue
- 1553 IF HW/SW issue
- Serial bus IF HW/SW issue

**Subsystems**
- Functionality IF HW/SW issue
- Functionality N HW/SW issue

**Functionality Dependent**

**Device independent**

Transformation - reusable on any mission
Generic Main Spaceflight Computer Example

- **Understand subsystem functionality**
  - Decompose into known and potential hardware and software faults
  - Peripheral Component Interconnect (PCI) status register errors
  - Excessive accumulation of uncorrectable SDRAM memory errors
  - Overcurrent/undercurrent
  - Overvoltage/undervoltage
  - CPU halt/hung
  - Etc

Compare, contrast, analyze
Applying Phase II To Your Project

• Transformation from the specific to the generic
  – Think product lines – not spacecraft
  – Apply the model transformation process
  – Replace the space mission examples with your system information
  – Decompose the system into subsystems
    • Look at projects, programs, applications, services, etc
    • Focus on functionality
  – Create models
  – Add lessons learned from previous projects
  – Establish a baseline

Reusable in any environment
Applying Phase II To Your Project

Products

Projects

Applications

Identify Subsystems

Review Subsystem/Device Functionality

Identify Third Party Rules, Regulations, Standards

Identify Common Functionality

Mission 1

Mission 2

Mission N

Identify Fault Conditions Based on Functionality

Update IV&V Independent List of Generic Fault Conditions

Create/Update Generic Fault Models

Update Reuse Baseline

Provide Independent List & Models to Developers

Reusable process
Conclusion

- Identified a proactive approach using reusable fault conditions - based on functionality
- Phase II introduces a new way to independently validate software safety requirements, via the comparison of the fault management artifacts against the IV&V team’s own list of fault conditions – based on functionality
- Helps the mission developer ensure they have identified the correct fault conditions & identifies missing requirements
  - Promotes feedback from the developer
- Builds a foundation for dependability and safety that is reusable
  - Reusable safety process identifies requirements & safety gaps
  - Applicable in any environment


