Extending FMECA to System of Systems (SoS) Interfaces: iFMECA

Pl’s: Leo Mayoral, Clay Smith
28 October 2009
Agenda

- Background & Problem Description
- Proposed Concept
- Possible Model for the SoS Interface
- Technical Foundation
- Extending FMECA Process to SoS Interface Analysis
- Potential Applications
- Summary
Background

- APL interest in understanding how to objectively assess failure modes for large system of systems:
  - Especially when introducing a new system into a complex and existing architecture,
  - Identifying problem interfaces during the design phase,
  - Prioritizing SE resources,
- Question: How can the Systems Engineer characterize SoS interface faults in order to prioritize resources?
System Box-Level Problem Description

- Failure mechanisms and failure modes are typically known for individual component systems
  - Usually these analyses are dictated by contract
  - Full reliability and risk analyses performed within context of the system only

- Interfaces among components systems can be uncertain
  - Defined to the level of an internal specification or requirement
  - Not completely enveloped
  - Ambiguous
  - Cause and effects not always deterministic or known a priori

- Interface issues exist even though all component systems are operating within system specifications
- Identify these interactions and prioritize their impact
System of System Problem

- A significant number of issues for System of Systems reside in the interfaces among the systems.

- This iFMECA methodology extends the current FMEA techniques to provide SoS engineers with a risk based prioritization of interfaces:
  - FMECA is one of the most widely used reliability tools (see MIL-STD-1629A)
  - Bottoms up approach
    - Functional or physical breakdown
    - For each interface failure modes are identified
    - For each failure mode identified (known or potential), determine
      - Consequence (narrative description of local, system, and SoS effects)
      - Probability of occurrence
      - Method for detection
    - Determine risk criticality
  - Rank order interfaces using risk criticality number for resource allocation (i.e., which interface to worry about first)
System of System (SoS) Problem

- A significant number of issues for System of Systems reside in the interfaces among the systems
- Interfaces are Often Complex
  - Multiplexed outputs
  - Protocol Oriented
  - Timing
  - Signal Quality
  - External Coordination
  - Network Delays
- Challenge is to find a system engineering tool that can help the PM and SE identify problem interfaces efficiently and cheaply.
FMECA Methodology: Background

- **FMECAs are used in systems to:**
  - Identify Single Point Failures,
  - Prepare diagnostic routines such as flowcharts or fault-finding tables,
  - Prepare preventive maintenance requirements,
  - Design built-in test, failure indications, and redundancy,
  - Analyze testability to ensure that hardware can be economically tested and failures diagnosed,
  - Show as formal record of safety/reliability analysis.

- **Limitations**
  - Combined effects of coexisting failures are not considered
  - Extents upward through system hierarchy, no peer-to-peer interactions
  - Process is extraordinarily tedious and time consuming for complex systems
Proposed Concept: iFMECA Methodology

- **Analyze the interface**
  - Decompose each interface to determine failure modes
    - Level of detail may vary
    - Interface dependent, several models exist to accomplish this task
  - Determine the probability of loss
    - Qualitative (ordinal scale) or quantitative (such as loss of margin)

- **Analyze the impact of interface to the function (or system)**
  - Assign a prior probability distribution based on test data, engineering judgment, or rules-of-thumb
  - Later update with Bayesian statistical methods with operational data

- **Analyze the impact of the function to the mission (or SoS)**
  - Assign a prior probability distribution based on test data, engineering judgment, or rules-of-thumb
  - Later update with Bayesian statistical methods with operational data
SoS Interface FMECA (iFMECA)

- Specific area of focus is the off-nominal performance at the interface among component systems
  - Limiting scope to these failure modes
  - Assuming that system failure is treated already

- For this case, neither System A or B has failed by its own definition, but a portion of A output is not processed by B
  - Uncertainty exists in the variability of System A output and the variability of System B threshold limit
  - Output spec of A and the input expected range of B may differ
SoS Interface iFMECA

- Probability of loss of function (LOF) for Subsystem B is a function of its inherent failure rate plus the loss of input (LOI) from Subsystem A

\[ \Pr(LOF_B) = \lambda_p t \times \Pr(LOI) \]

- For a more generalized case with multiple inputs:

\[ \Pr(LOF_B) = \lambda_p t \times \sum \Pr(LOI)_i + \text{combinatorial effects}^* \]

- Assumptions:
  - Inherent failures are covered elsewhere
  - *Combinatorial effects from the interactions of multiple degraded inputs not yet addressed
iFMECA Methodology Criticality Number

- Mil-Std-1629 Defines a Criticality Number
- Propose an Analog for SoS Criticality Number ($C_{SoS}$):

$$C_{SoS} = \gamma \cdot \beta \cdot \Pr(LOI)$$

Where,

- $\gamma$ Conditional probability of LOM given LOF
- $\beta$ Conditional probability of LOF given LOI
- $\Pr(LOI)$ Probability of output-input mis-match

- Parameters $\gamma$ and $\beta$ based on
  - Operational data
  - System test data
  - Can be subjectively assigned and updated with Bayesian techniques as more operational experience is gathered
iFMECA Methodology ... New SE Tool

- **Analyze the interface**
  - Decompose each interface to determine the attribute ($a_i$) failure modes
    - Level of detail may vary
    - Interface dependent, several models exist to accomplish this task
  - Determine the probability of loss
    - Qualitative (ordinal scale) or quantitative (such as loss of margin)

- **Analyze the impact of the interface to the function (or system)**
  - Assign a prior probability distribution based on test data, engineering judgment, or rules-of-thumb
  - Update with Bayesian statistical methods using operational data

- **Analyze the impact of the function to the mission (or SoS)**
  - Assign a prior probability distribution based on test data, engineering judgment, or rules-of-thumb
  - Update with Bayesian statistical methods using operational data
One Possible Concept for Modeling an Interface

- A Typical Interface is Comprised of Several Interface Attributes \( (a_i) \), e.g. OSI Stack
- All \( a_i \) Must not Experience a Failure for the Interface to Work
- Viewed as a Logical “And” at the Input
- Viewed as a Logical “Or” at the Output
- All Events \( (a_i) \) are Mutually Exclusive (Assumption)
- The Occurrence of Any Event, \( (a_i) \), Causes a Degradation of the Interface
How Would the Data Be Analyzed?

1. Focus on Copper/Fiber and Wireless Connectivity
2. Ignore OSI Layers 5-7 (Session, Presentation, Application Information Layers) for Now
3. Catalogue Top Level Category Interface Faults
   - Look for Statistical Data
   - Interview for Experiential Data
4. Select a Small Subset and Analyze Failure Modes for Each
5. Correlate to Methodology
   - Validate Criticality Number
   - Validate probabilistic margin analysis
6. Document Results Formally

Examples of Potential I/F Faults:

- Copper or Optical Connection
  - Port or Interface Status is Disable or Shutdown
  - Port or Interface Status is errDisable
  - Port or Interface Status is Inactive
  - Uplink Port or Interface Status is Inactive
  - Trunking between a Switch and a Router
  - Trunking Mode Mismatch
  - Connectivity Issues due to Oversubscription
  - Common Port and Interface Problems
  - Data Signal Voltage Mismatch
  - Data Signal Voltage out of tolerance
  - Data Incompatibility
  - Noise Coupling
  - Crosstalk

- Wireless
  - Frequency Error
  - Bandwidth Error
  - Modulation Mismatch
  - Link Closure
  - Doppler Signal Errors
  - Signal Dead Spots (R² Losses)
  - Signal Integrity
  - Multipath Errors
SoS Criticality Number Extends Definition

- Mil-Std-1629 Analysis Focuses at the Box Level
- Standard Criticality Analysis considers part/board failure rate and the system impact
- Failure mode **Criticality Number** is used to convey the severity of the fault:
- Criticality Number is computed as:

\[ C_m = \beta \cdot \alpha \cdot \lambda_p t \]

- \( \lambda_p t \) Part failure rate x time (Poisson Distribution)
- \( \alpha \) failure mode ratio
- \( \beta \) conditional probability of loss of mission (LOM) \( \Pr(LOM \mid \text{Failure Mode}) \)

<table>
<thead>
<tr>
<th>Failure Effect</th>
<th>( \beta ) Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual loss</td>
<td>1.0</td>
</tr>
<tr>
<td>Probable loss</td>
<td>&gt; 0.1 to &lt; 1.0</td>
</tr>
<tr>
<td>Possible loss</td>
<td>&gt; 0 to 0.1</td>
</tr>
<tr>
<td>No effect</td>
<td>0.0</td>
</tr>
</tbody>
</table>
iFMECA Methodology

- Extends the FMECA to SoS
  - Perform a systematic analysis of each SoS interaction
  - Pair-wise comparison for all output-input pairs
- Propose an Analog for Criticality Number ($C_{SoS}$):

$$C_{SoS} = \gamma \cdot \beta \cdot \Pr(LOI)$$

Where,

- $\gamma$  Conditional probability of LOM given LOF
- $\beta$  Conditional probability of LOF given LOI
- $\Pr(LOI)$  Probability of output-input mis-match

Definitions:
LOI – Loss of Input
LOF – Loss of Function
LOM – Loss of Mission

Our methodology extends the Criticality Number to a SoS by adding the conditional nature of the failures between systems.
Another method to Analyze in Interface
Output-Input Examples

- **Parameters $\gamma$ and $\beta$ based on**
  - Operational data
  - System test data
  - Can be subjectively assigned and updated with Bayesian techniques as more operational experience is gathered

- **Probability of occurrence**
  - Probabilistic measure of the interference between the input variability and the variability of the input threshold limit
  - Probability density functions obtained from system designs, testing, operations
iFMECA Methodology Advantages

- Risk-based prioritization based on calculated $C_{SoS}$
  - Input-Output pairs
  - System contribution pairs
    - Input-Output pairs sum within the receiving system
  - Significant Output-Input combinations

- Provides a **Systems Engineering Tool** for analyzing the trade space for Interfaces when introducing a new system into a SoS
  - How much should an output signal change?
- **A New Tool** to help Identify the information needed to communicate potentially mismatched information across SoS interfaces
  - Included into SoS ICD equivalents
iFMECA Methodology Execution

- Interfaces will be analyzed **not** for hardware on either side of the interface
  - Assumed to be part of the normal FMEA process already in place

- Interfaces analyzed for
  - Content communicated
  - Medium of communication
  - Protocol interoperability
  - Stress vs strength
  - Load vs endurance
Another method to Analyze in Interface Output-Input Examples

- **Case I**
  - Discrete output with discrete upper bound threshold
  - No variability is shown, therefore output will always be less than threshold
  - \( \Pr(LOI) = \Pr(I > T) = 0 \)

- **Case II**
  - Variation in output with discrete upper bound threshold
  - Some \( \Pr \) exists that the input level will exceed the threshold
  - \( \Pr(LOI) = \Pr(I > T) = \int_{T}^{\infty} f_I(x) \, dx \)
Another method to Analyze in Interface Output-Input Examples

- **Case III**
  - Variation in output with variation in upper bound threshold
  - Some Pr exists that the input level will exceed the threshold
    - \( \Pr(\text{LOI}) = \Pr(I > T) = \int_{-\infty}^{0} \int_{-y}^{\infty} f_T(y + x) \cdot f_I(x) \, dx \, dy \)

- **Many type of interactions exist**
  - Various combinations
  - Various distributions
IRaD Summary

- Shown that a New SoS Design Tool that Quantifies the Criticality of its Interfaces is Possible
  - Concept is Based on Modeling the Interface as a Combination of Boolean Variables and Employing Conditional Probability Theory to Propagate the Probability of their Failure
  - Concept is Applicable to Complex Interfaces (e.g. OSI Stack, or multi-attribute)
  - Allows for the Propagation of a Poorly Performing Attribute of an Interface to be Propagated to the Next Hierarchical Level and Address Impacts to Mission
  - Though Not Investigated, Suggests that Marginally Performing Interfaces which can Affect Overall SoS Performance May be Isolated
  - Allows the PM to Adjust Program Resources to Mitigate Poorly Designed Interfaces Early in the Design Phase by Analyzing the I/F Criticality Numbers
  - Tool is Not Radically Different – It is a Simple Extension to the Well-Understood FMECA Tool (Mil-Std-1629)
  - SoS Design Challenge: Developing and Validating the Failure Rates of the Attributes of Interface Data
Potential Follow-On Work

- Need to Typify the Types and Classes of Failures Similar to How Studies Are being Performed on the Failure of Box-Level Component Parts
- Need to Characterize the Statistical Distributions for These Interface Types and Classes of Failures
  - As a First Approximation, a Typical Normal, Poisson or Exponential Distributions could be Assumed
  - Distributions Need to be Validated on Real World Systems
- Need to Develop the Data Collection Methodology at the Design Level (Extend the Procedural Language in the Mil-Std-1629 to Address SoS Interfaces)
- Publish the Results
Questions?