Identifying Architectural Challenges in System of Systems

NDIA Systems Engineering Conference
October 2013

Mike Gagliardi, Bill Wood

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHEDON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM-0000673

Problem

SoS Integration and operational problems arise due to inconsistencies, ambiguities, and omissions in addressing quality attributes, capabilities and engineering considerations between constituent system architectures.

Example quality attributes: predictability in performance, security, availability/reliability, usability, testability, safety, interoperability, maintainability, force modularity, spectrum management.

Functionality and capability are critically important, but the architecture must be driven by the quality attributes. Specifying and addressing quality attributes early and evaluating the architecture to identify risks is key to success.

Concerns

Need to identify architecture issues that will turn into integration and operational problems if not addressed, early in the life cycle

End-end threads are underutilized in SoS development

Systems developed as stovepipes are difficult to integrate into SoS

Environments change at a rapid rate (e.g. threat, technology, funding, manning)

SoS Quality Attribute related problems are discovered late and expensive to fix

The Need for Augmented End-to-End Mission Threads in DoD SoS Architecture Development

DoDAF provides a good set of architectural views for an SoS architecture. However, it inadequately addresses cross-cutting quality attribute considerations.

System use cases focus on a functional slice of the system.

More than DoDAF and system use cases are needed to ensure that the SoS architecture satisfies its cross-cutting quality attribute needs.

SoS end-to-end mission threads augmented with quality attribute considerations are needed to help define the SoS Architecture and then later evaluate the SoS architecture and constituent system/software architectures.

Definitions

Vignette: A description of the geography, own force structure and mission, strategies and tactics, the enemy forces and their attack strategies and tactics, including timing. There may be associated Measures of Performance (MOP) and Measures of Effectiveness (MOE). A vignette provides context for one or more *mission threads*.

Mission Thread: A sequence of end-to-end activities and events beginning with an opportunity to detect a threat or element that ought to be attacked and ending with a commander's assessment of damage after an attack. C4ISR for Future Naval Strike (Operational)

Sustainment: A sequence of activities and events which focus on installation, deployment, logistics and maintenance.

Development: A sequence of activities and events that focus on re-using or re-engineering legacy systems and new adding capabilities

Acquisition: A sequence of activities and events that focus on the acquisition of elements of an SoS, and the associated contracts and governance

Vignettes Are the Starting Point – Example Wording

Two ships (Alpha and Beta) are assigned to integrated air and missile defense (IAMD) to protect a fleet containing two high-value assets (HVA). A surveillance aircraft SA and 4 UAVs are assigned to the fleet and controlled by the ships. Two UAVs flying as a constellation can provide fire-control quality tracks directly to the two ships. A three-pronged attack on the fleet occurs:

- 20 land-based ballistic missiles from the east
- 5 minutes later from 5 aircraft-launched missiles from the south
- 3 minutes later from 7 submarine-launched missiles from the west.

The fleet is protected with no battle damage.

Mission Threads Flow from Vignettes – Example (Non-Augmented)

1. 20 land-based missiles launched - X minute window

2. Satellite detects missiles - cues CMDR

3. CMDR executes re-planning – reassigns Alpha and Beta

4. Satellite sends track/target data - before they cross horizon

5. Ships' radars are focused on horizon crossing points

…

N Engagement cycle is started on each ship

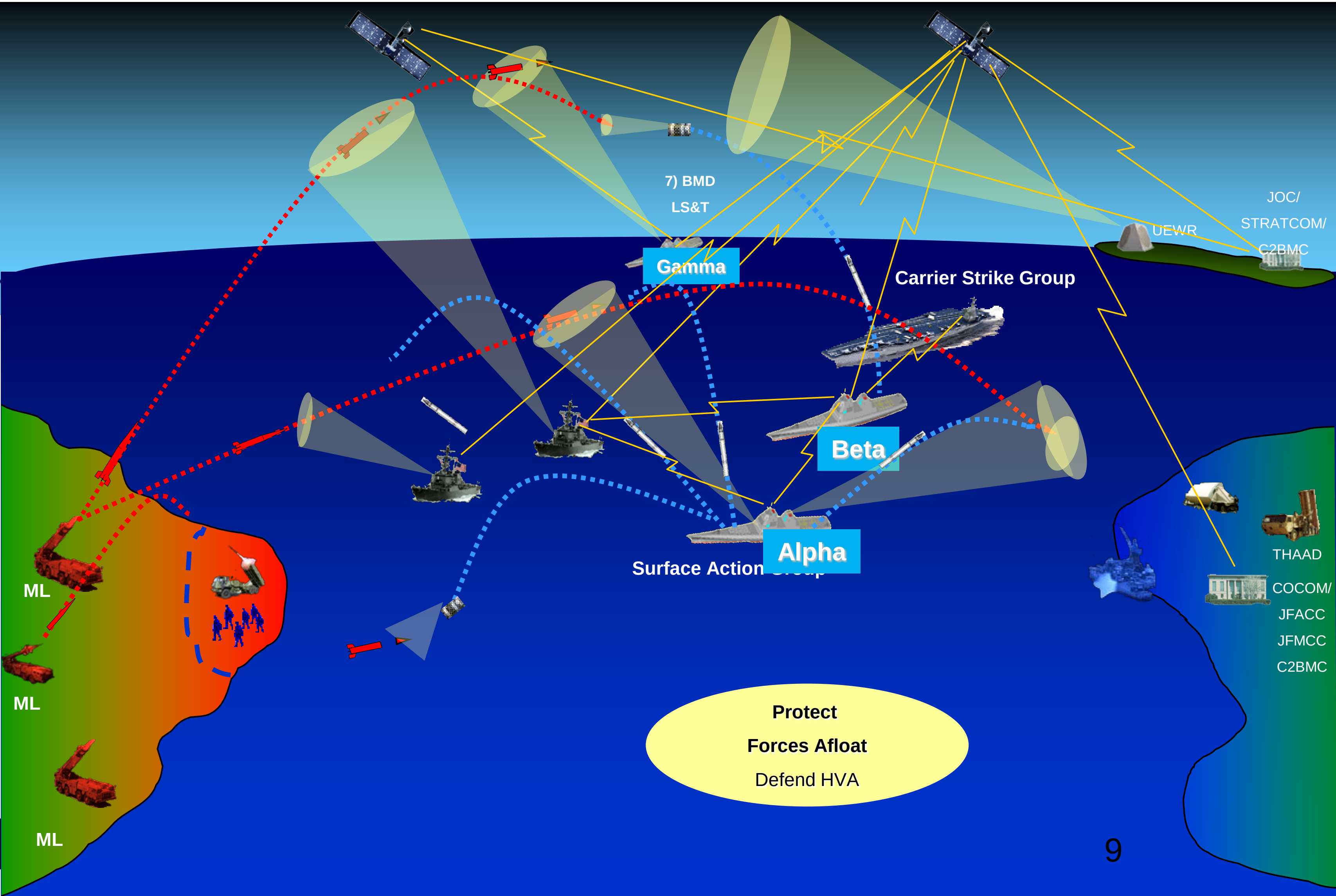
N+1. Aircraft are detected heading for fleet

N+2. SA detects missile launches – tells CMDR

N+3. CMDR does re-planning - UAVs are re-directed

N+4. FCQ tracks are developed from UAV inputs

Air and Missile Defense (AMD) OV-1 Example



Mission Thread Workshop - Goal

To augment a set of end-to-end System of Systems (SoS) mission threads with quality attribute and engineering considerations with the stakeholders.

To capture at each step of the mission thread AND each SoS quality attribute

- the engineering considerations from diverse stakeholders
- the quality attribute concerns associated with the mission thread
- the applicable use cases for the different nodes and/or systems

To develop technical challenges associated with the threads, and to aggregate the challenges over a number of MTWs

Outputs will inform and drive SoS Architecture Decisions.

Augmentation Process – Per Mission Thread

Two Passes over the Mission Thread:

1) For each event in the mission thread:
   - Elicit quality attribute considerations. Capturing any engineering issues, assumptions, challenges, additional use cases and mission threads (with QA context etc.)
   - Capture any capability and/or mission issues that arise.

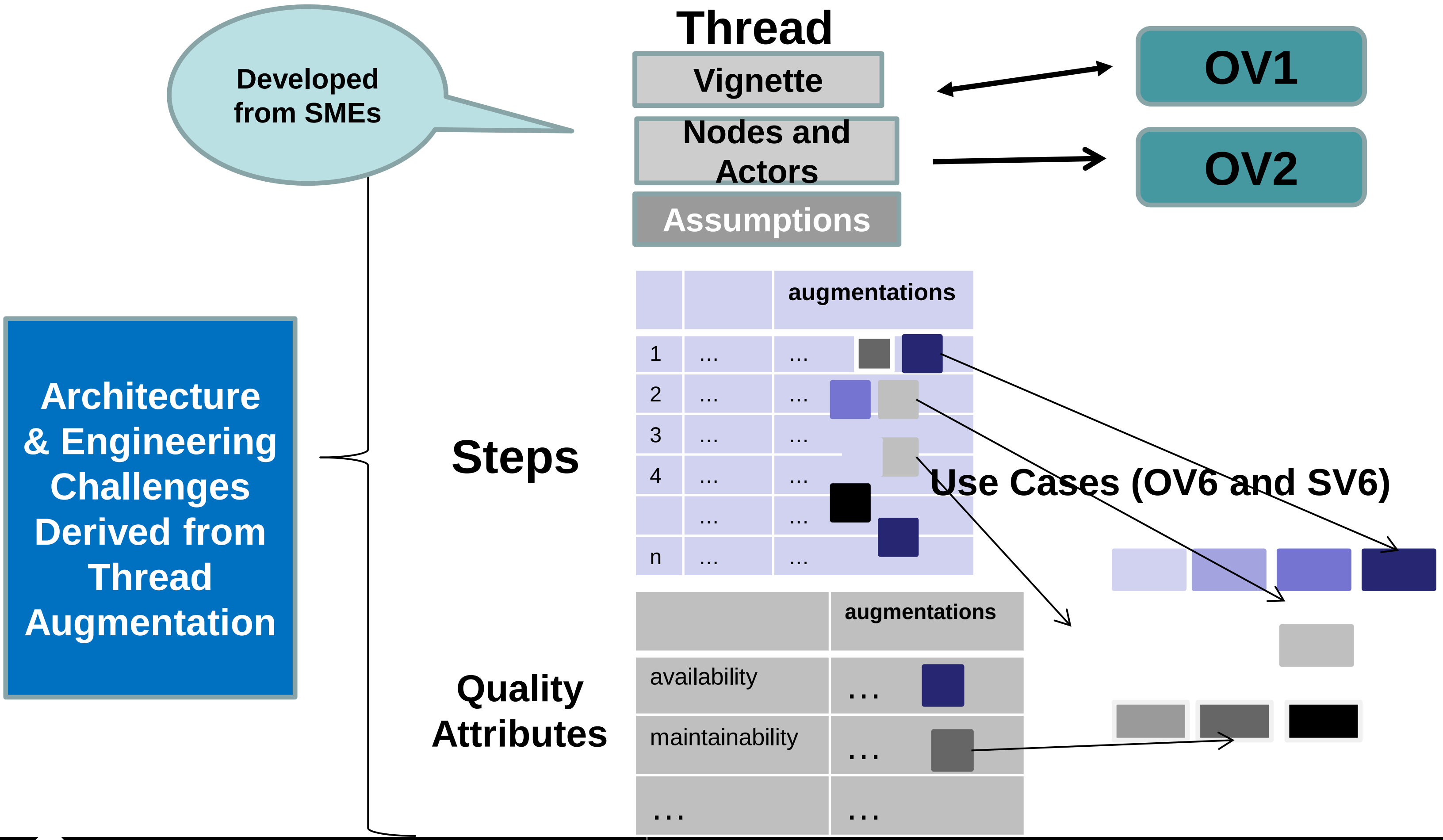
2) For each Quality Attribute - elicit any over-arching quality attribute considerations
   - Capturing any over-arching assumptions, engineering issues, challenges, additional use case and mission threads (with QA context) etc.
   - Capture any capability and/or mission issues that arise.

Capture any MT extensions for later augmentation

Capture Parking Lot issues – for organization, programmatic, non-technical issues that arise (will not be further pursued in the MTW).

Stakeholder Inputs are Key.

Mission Thread
(augmented via the Mission Thread Workshop)



Architecture & Engineering Challenges Derived from Thread Augmentation

		augmentations
1	…	… ■ ■
2	…	… ■ ■
3	…	… ■
4	…	… ■
	…	… ■
n	…	…

	augmentations
availability	… ■
maintainability	… ■
…	…

Nodes, Actors and Assumptions Augmentation

Name	**Protect Fleet Assets against Cruise Missile Attacks**
Vignette **(Summary Description)**	Two ships (Alpha and Beta) are assigned to air defense (AD) to protect a fleet containing two high-value assets (HVA). A surveillance aircraft (SA) and four UAVs (two pairs) are assigned to the fleet and controlled by the ships (Alpha and Beta). A pair of UAVs flying as a constellation can provide fire-control quality (FCQ) tracks directly to the two ships. A two-pronged attack on the fleet occurs: • five aircraft-launched missiles from the Southeast • three minutes later seven submarine-launched missiles from the Southwest. The fleet is protected with no battle damage.
Nodes Actors	• two ships (Alpha and Beta) • four UAVs • two HVAs • one SA • five enemy aircraft and their missiles • seven enemy submarines and their missiles
Assumptions	• Enemy aircraft are flying along a route normally used for training, and suddenly change direction and head for the fleet. They are being tracked. • The submarines are undetectable until they fire their missiles. • ***No sonabouys are deployed, but they could be in a new vignette.*** • The vignette is not concerned with counter-attacking the enemy aircraft or submarines. • It is not a wartime situation; ships are at battle condition 3. • Sea state is 3. • Ships' readiness condition is YOKE. • Alpha controls two UAVs and Beta two other UAVs. • ***Each ship has two organic UAVs.*** • During normal operations the UAVs have separate non-overlapping areas of regard (AORs). • The SA has an area of regard that will detect both the launched missiles. • The Air Defense Commander (ADC) is on-board Alpha. • ***Alpha ship's Helo is in the air.*** • ***Both ships are aware that a potentially hostile country has some fighter aircraft conducting training missions nearby.***

Step by Step Augmentation

Mission Steps	Description	Engineering Considerations, Issues, Challenges
1	Alpha develops the air defense plan (ADP) and rules of engagement (ROE) and sends them to Beta. The plan assigns to Alpha the area of regard (AOR) to the west, and Beta the AOR to the east. Alpha configures surveillance and weapons systems to support eastern engagements.	*1. How much is pre-defined and how much is done manually?* *2. ROE dictates a “shoot-look-shoot” defense.* 3. *How is this communicated to Beta? Using the fleets NRTC: near real-time communications*
2	The SA aircraft detects that the five enemy aircraft have changed course and are heading towards the fleet at low altitude.	*1. The enemy aircraft are within the area of regard (AOR) of the SA sensors. The SA has been tracking these aircraft and sending tracks to Alpha and Beta.* *2. Need a “fleet” SA use case*
3	SA informs both Alpha and Beta of the change.	*1. Within X seconds of detecting the change* *2. Using the Global Information Grid (GIG). Is the GIG usable for tactical near real-time data? Probably not!* *3. Need a use case on assigning the UAVs to track the aircraft at this point*
4	Alpha (and Beta) go to General Quarters	*1. ADC informs the captain who orders general quarters* *2. Using Internal Communications*
5	SA detects that missiles have separated from the enemy aircraft and informs Alpha and Beta.	*Within X seconds*
6	Alpha assigns its two UAVs to track the missiles.	*1. The legacy Defensive Engagement System (DES) cannot use external tracks to form a FCQ track.* *2. Within X seconds* *3. Does the ADC have to do this manually?* *4. Would they start tracking automatically if the missiles were within their AOR?* *5. Would they have been tracking the aircraft?*
7	The two Alpha controlled UAVs send FCQ tracks for the five missiles to both Alpha and Beta.	*1. The two UAVs can re-direct their payload to do this within YY seconds. (use case)* *2. It takes XX seconds for the FCQ tracks to stabilize.* *3. What is the comms between UAVs and Ships for maneuver and payload con-*

Over-Arching Quality Attribute Augmentation

Name of QA (filled in during Preparation phase)	**Considerations** (This column will be filled in during the Augmentation Phase)
Performance (P)	*1. The airspace de-confliction latency is heavily dependent on the number of aircraft within the strike paths.* *2. The timeline function from missile detection at specific distance from target until point of impact, including detection by both UAVs, engagement assignments, missile launching sequence, and fly out times has not been analyzed in detail!*
Availability/ Reliability (AV)	*1. What if both UAVs cannot maneuver to their respective AORs in time?* *a. They will probably have to wait until they are within the ship's radar to fire.* *b. Is this a manual decision? (tradeoff with automation)* *2. What if the ship/missile communications fails?* *a. It will probably have to fire another intercept missile!* *b. Can the other ship try to control the missile?* *3. What if Alpha/Beta Comms fails?* *a. Revert to a pre-defined separate engagement.* *4. What if Beta does not acknowledge engagement assignments? Revert to what was defined in ROE or assume that it will follow received orders or take some other option?* *a. A degraded Mode Use Case needs to be developed.* *5. Degraded modes of operation have not been detailed yet.* *6. Loss of comms. to SA.* *a. After initial detection and UAV coverage, it does not matter.* *b. Before initial detection, the UAVs will provide some coverage, but will probably have some unmonitored areas.* *c. What happens when missile goes beyond line-of-sight radar coverage?* *7. What if one of the UAVs is deemed non-functional during operations?*
Accuracy (Ac)	1. *If the tracks are relayed (see Interoperability item 2) what if they are not sufficiently accurate? Will they be?* 2. *Given multiple relay hops, how will accuracy be impacted? (Performance / accuracy tradeoff implications). How can shared resources be managed to bound latencies in this environment?*
Interoperability (In)	*1. Can a UAV that is assigned and controlled by one ship be reassigned and controlled by another ship dynamically? (Degraded mode future support?)* *2. Can FCQ information be transferred in real time from Alpha to Beta in order to target one of the missiles?*

Mission Thread Workshops - Experiences

Client	Description	MTWs	Vignettes	Mission Threads	Stakeholders
A	IRAD New platform/capability	1	1	2	8
B	New Naval Ship	13	17	37	>200
C	Battle Command	6	3	4	>100
D	Maritime Detection	2	4	4	30
E	NSF	1	3	3	15
F	Air Force Program	1	1	1	10
G	Other Govt Agency	1	4	4	12

- Identifies SoS architecture gaps, overlaps and challenges
- Identifies issues for constituent legacy system/software architectures
- Overcomes organizational stovepipes and facilitates stakeholder communication

Developing the Challenges

- Use affinity relationships to group raw data identified in the augmented mission threads
    - Quality attributes, capabilities, engineering considerations, etc.
- Description, impacts, recommendations… developed for each challenge
- Draft briefing created and vetted with sponsor, update for reality, program specific details, etc.

Required Follow-up: Architecture Challenge Workshop to analyze specific challenge and develop action items to address the challenge.

Turning Challenges into Action Items

The *Challenges* need to be addressed at some point in the life-cycle or they will become problems.

The *Challenge* slides are to convince management to make decisions about which challenges to address in what time frame and take some actions to mitigate them

- BUT all the detailed items used to develop the slide are still there

Each challenge can be further reduced to a number of aspects which can be prioritized across the Challenges

Workshops can be conducted on high priority aspects to develop action items.

Challenge Rollup Across SoS Clients

#	Name
1	Usability/Automation
2	Capability Gaps
3	Resource Management / Disaster Recovery / Degraded Operations
4	Training
5	Legacy Migration
6	Collaboration

Resource Management / Disaster Recovery

Individual systems had

- Low operational reliability
- Have to re-build Situational Awareness state after recovery from failure

Disconnected operations poorly defined and managed

Degraded modes of operation inconsistently defined within SoS

- Impact of loss of high quality track data

Distributed Resource Manager could not map from large scale failure to impact on current missions to suggested recovery strategies

Degraded Operation

Many failures could occur between the initiation of strike engagements and their completion. These can involve:

- changing controlling platform of the missiles or air platforms
- loss of communications
- failure to receive/acknowledge messages from weapons
- change in operational environment

The impact of loss of communications between assets in the kill chain have not been fully considered.

An SoS strategy to support degraded modes of operation needs further development. The degraded modes of operation should all be listed and appropriate considerations made

- Provide all mission area inputs to the Architecture IPT to address SoS degraded modes of operation.
- Assess the ability to use non-organic assets to perform in the event that organic assets are not available.

Fault Management1

The SoS strategy for fault management needs further development to address the failure conditions and support the SoS degraded modes of operation.

Recommendation:

- Perform a failure mode analysis/impact study which addresses all failure conditions. Includes complete/partial loss of electrical power or cooling power and advance/maneuvering
- Develop an SoS degraded modes of operation strategy.
- Define built-in-test (BIT) and remote testing strategy; diagnostics and analysis capabilities and what test/diagnostic equipment is needed.

Fault Management$_2$

In the presence of failures, graceful degradation and continuity of operations is dependent on operator decisions which could be inconsistent.

System (or portions) is unavailable during operations. System resets are frequent. System restoration is time-consuming.

Significantly extends the timeline for contact identification and classification.

Recommendations:

- Conduct an Architecture Challenge Workshop with key stakeholders to identify actionable items, potentially include:
  - Degraded modes of operation
  - Fault model and recovery activities
  - Component unreliability
  - Disconnected operations

Summary

Can augment end-to-end threads with QA, Capability, and Engineering considerations

Identifies SoS challenges early (very good risk predictors)

Cross-discipline stakeholders can agree on thread steps

- Reduce “rice-bowls”, identify “long poles”

Good facilitation is necessary

- Enough patience to hear things through, enough control to move things along

Approach can be easily tailored and has been used for an Enterprise Service context

A core team for MTW facilitation and SoS stakeholders provided consistency

Legacy System Architecture Evaluation Using Mission Threads as a Starting Point

Mission Threads are the Starting Point

We have used mission thread augmentations to develop system specific scenarios for legacy system architecture evaluation.

Legacy system specific scenarios are developed with the SoS/EA and legacy system stakeholders.

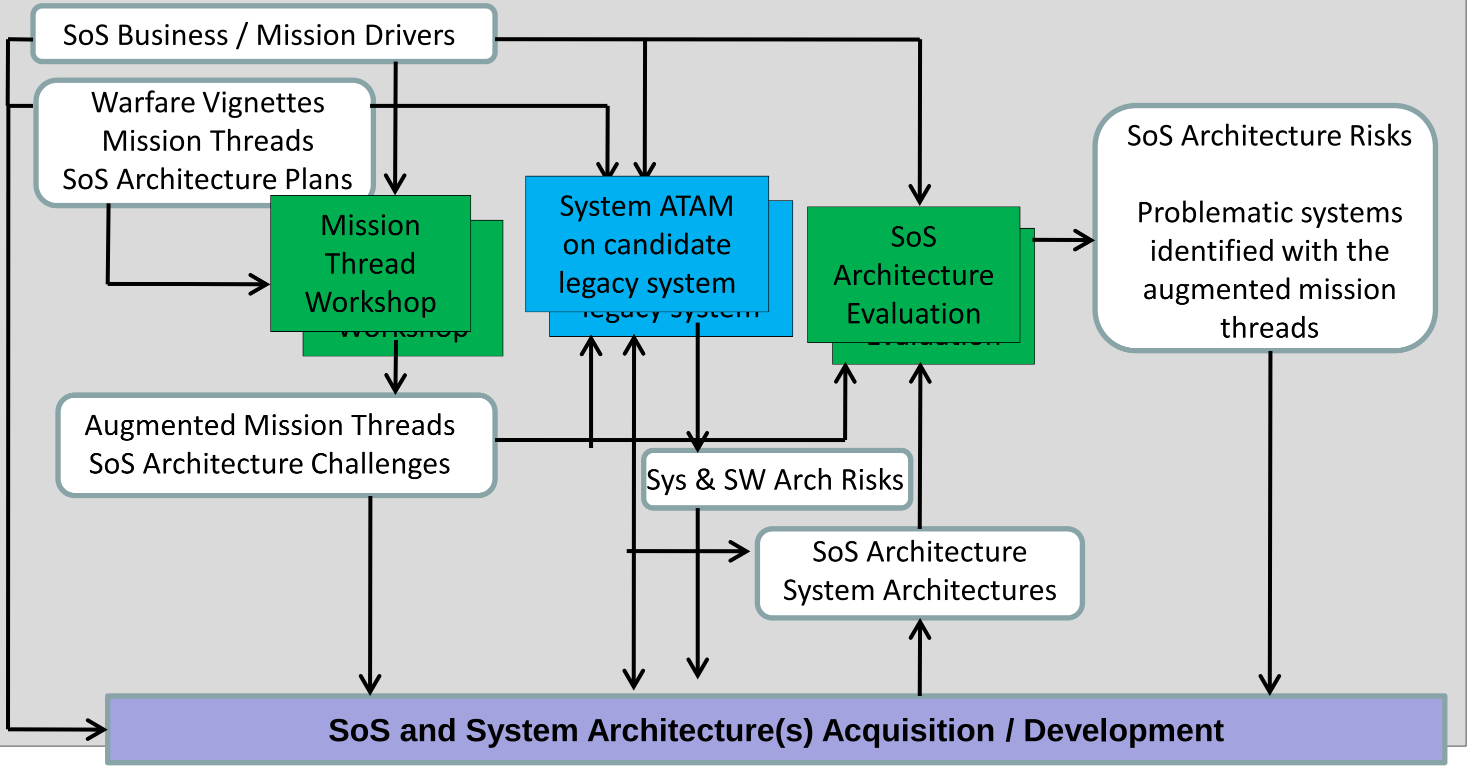
Scenarios are derived from the MT augmentations that pertain to the specific legacy system

- From each MT step augmentation
- From each Quality Attribute augmentation

Scenarios are vetted and prioritized with stakeholders and will be used to evaluate the legacy system & software architecture.

SoS Architecture Quality Attribute Specification and Evaluation Approach

- Early elicitation of quality attribute considerations
- Early candidate legacy system architecture evaluation
- Early identification and mitigation of architectural risks


SoS Business / Mission Drivers
Warfare Vignettes
Mission Threads
SoS Architecture Plans
Mission Thread Workshop
System ATAM on candidate legacy system
SoS Architecture Evaluation
SoS Architecture Risks
Problematic systems identified with the augmented mission threads
Augmented Mission Threads
SoS Architecture Challenges
Sys & SW Arch Risks
SoS Architecture System Architectures
SoS and System Architecture(s) Acquisition / Development

Example Legacy System Overview - 1

A legacy system was partially modernized

- Exists as a combination of Legacy (75%) and Modernized (25%)
- Exists in two data centers for recovery from catastrophic failure, and under-the-cover data mirroring (one active, one passive)

Modernized portion is:

- COTS based, relational database, service oriented, transactional, batch entries with hundreds of transactions, data warehousing, 24/7/365 operation
- Software is provisioned to processors at configuration time to form round robin load sharing
- Complex interactions between agency users and many participating government agencies and commercial users
- Specialized interfaces to the external users
- Business workflows are implemented using reliable messaging queues between business processes

Example Legacy System Overview - 2

Software architecture mostly undocumented

Data architecture documented in some areas

Original architects no longer on the job

Software maintained by developers, primarily using the software itself and some detailed design documents

Many outstanding software PTRs take a long time to resolve

Limited weekly availability of architects for evaluation

Legacy System Architecture Evaluation - 1

Approach:

- Based on MTW and ATAM concepts
- Developed three end-end business threads, based on the business drivers and elicited from business and operations stakeholders. Representing the three major end-end capabilities of the legacy system.
- Quality attributes were derived from system business drivers and interviewing the architects.
- Augmented the three end-end business threads:
    - Augmenting the steps in the threads; eliciting any legacy system specific scenarios for each step. All quality attribute concerns generated scenarios.
    - for the over-arching quality attributes, developed a legacy system specific utility tree; eliciting concerns and scenarios, using MTW templates
- All of the scenarios are taken together and prioritized (just as in Phase 1 of ATAM).

Legacy System Architecture Evaluation - 2

Approach:

- Evaluation team makeup: 4 SEI evaluators w/ facilitator,
  2 Subject Matter Experts from Program Office
- Due to limited weekly availability of architects and lack of documentation, we decided to hold a series of architecture evaluation sessions. Three times a week, two hours per session.
- Decided to focus evaluation sessions on specific topics, e.g., performance, availability, maintainability, etc.
- When all the focused sessions were completed, we executed end-to-end thread sessions.

Legacy System Architecture Evaluation - 3

Results:

- The focused sessions resulted in identifying over 100 architectural risks, 25 non-risks.
- The end-to-end thread sessions resulted in identifying 15 additional risks, mostly dealing with end-to-end issues.
- Seven risk themes were generated. Customer was very satisfied with the results.

Legacy System Architecture Evaluation - 4

Lessons Learned:

- The end-to-end business threads set the proper context for scenario generation for the specific legacy system
- Executing the end-to-end thread sessions last was very beneficial:
  - Put all the previously identified risks into context
  - Helped to focus on end-to-end type risks
  - Helped to understand (and document) the architecture end-to-end
- Lack of a documented architecture was a burden
  - Slowed down the evaluation sessions and extended the schedule. We quickly abandoned the notion of capturing architect’s hand drawings.
  - Never sure if there were places in the architecture that needed further evaluation

Legacy System Architecture Evaluation - 5

Lessons Learned:

- We were satisfied that we had covered the architecture when we finished the end-to-end thread sessions

- We needed 20 evaluation sessions (2 hours each), spanning six weeks. This is not out of line for the total amount of time needed for architects to support an architecture evaluation.

- The architects were very cooperative and open about the process and provided good information, even though they weren't the original architects.

Contact Information

SEI Technical Report: Introduction to the Mission Thread Workshop
CMU/SEI-2013-TR-003
www.sei.cmu.edu

Mike Gagliardi
Software Engineering Institute
MJG@sei.cmu,edu
412-268-7738

Bill Wood
Software Engineering Institute
WGW@sei.cmu.edu
412-268-7723