Systems of Systems Engineering Technical Approaches as Applied to Mission Engineering

Dr. Judith Dahmann
Dr. Aleksandra Markina-Khusid
Janna Kamenetsky
Laura Antul
Ryan Jacobs

MITRE

Topics

- Mission engineering (ME)
- The relationship between system of systems engineering (SoSE) and ME
- Particular challenges of SoSE applied to missions
- Some SoSE technical approaches which address these challenges

Approved for public release. Distribution unlimited 17-3712-15

MITRE

Mission Engineering Challenge

- Systems are acquired to meet user needs in a mission context
- Mission operations are supported by sets of systems (or systems of systems) which work together to achieve mission objectives
- Systems supporting each role in a mission (i.e. kill chain) will vary over the course of the operation and be used for multiple missions

System Acquisition

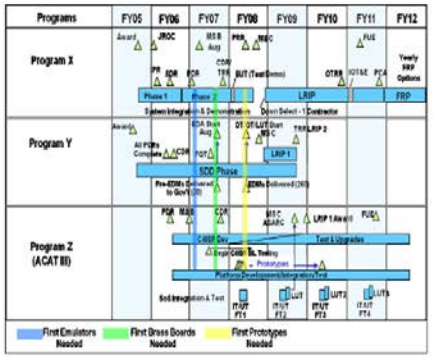
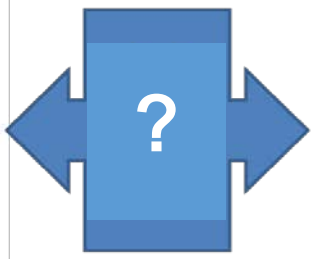
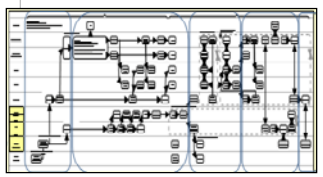


Figure 3.5-1 System-of-Systems Schedule (optional) (sample)
Note: Include an as-of date – time sensitive figure



Operations



Mission/SoS Architecture/Engineering

Mission Engineering is the deliberate planning, analyzing, organizing, and integrating of current and emerging operational and system capabilities to achieve desired warfighting mission effects

Defense Acquisition Guide Ch 3

MITRE

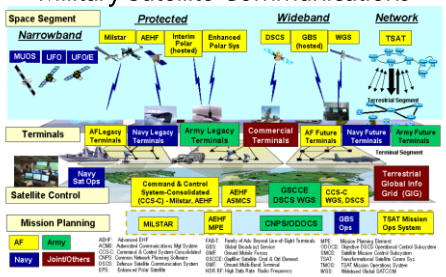
Systems of Systems in Defense

Tactical Vehicle

Platforms

A military platform (e.g. ship, aircraft, satellite, ground vehicle) equipped with independent systems (e.g. sensor, weapons, communications) needed to meet platform objectives

Military Satellite Communications



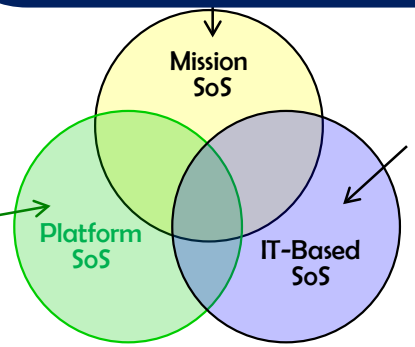
Missions

Sets of systems working together to provide a broader capability or mission

Operations Center

Information Technology

Networked information systems to support operations within or across platforms or systems to meet mission or capability objectives



Considerations in mission SoS

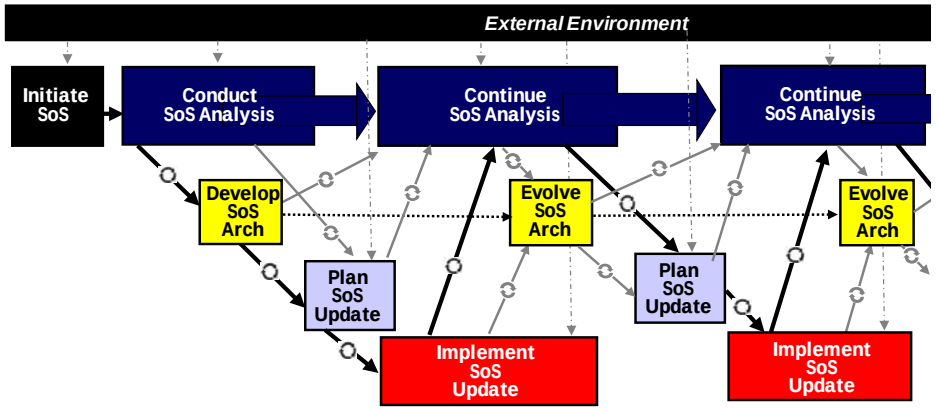
- **Mission environment**
  - Mission context - variable physical environments, threats and non-material elements - critical in driving SoS for missions
- **Composition**
  - Execution of missions is based on the employment of the set of systems available and appropriate for the mission environment
  - Performance needs of a system in the Mission SoS may vary depending on the performance of other systems in the SoS (‘AKA ‘Float and Flow’)
- **Mission ‘webs’ versus ‘threads’**
  - While there may be a logical sequence of actions for a mission, in practice there are sets of systems which support missions under different situations

Approved for public release. Distribution unlimited 17-3712-15

MITRE

SoSE Wave Model Applied to ME

Define the mission including mission threads and mission context (*Includes mission objectives, CONOPs, scenarios, key functionality, threat*)	Conduct SoS Analysis
Identify current systems supporting the mission and how they are employed (*How are we implementing the mission today?*)	
Assess mission performance to assess how well current systems work together meet mission objectives	
Identify gaps from a mission effectiveness perspective and fault isolate the source of gaps	
Identify and assess options for improving the mission effectiveness (*Including changes in how the systems are employed as well as new or different systems, systems updates and non-material considerations*)	Develop SoS Architecture
Guide systems acquisitions, from requirements through implementation to test and maintenance to assure effective mission execution	Plan SoS Update
Conduct mission level integration and test	Implement SoS Updates
Monitor mission effectiveness with changes in mission context, scenarios and threat capabilities	



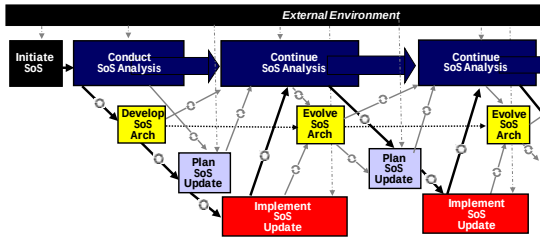
Like other SoS, SoS for missions

- Are ***not 'designed'*** top down, green field systems
- ***Evolve over time*** based on changing capability needs and systems
- Engineering follows the an **evolutionary** '***wave***' process versus traditional system 'V'

Approved for public release. Distribution unlimited 17-3712-15

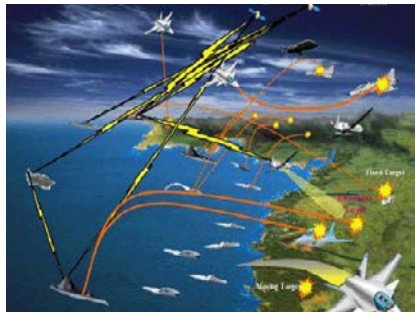
MITRE

Mission Engineering
SoSE Engineering to Meet Mission Objectives



Baseline current SoS Against **Mission Objectives**
- Assess end-to-end performance of SoS to implement mission **effects/kill chain**
- Identify **gaps**

Evaluate options and **trades across the SoS** to improve or sustain mission performance
- New TTP for the SoS
- Reconfiguration of SoS
- New/upgraded systems
- New system interfaces



Implement changes in systems, integrate and test updated SoS **mission capability**

Negotiate with systems to make changes to support **mission performance** improvement
- Plan **coordinated capability package** for mission improvement
- Coordinate technical, program and budget plans

MITRE

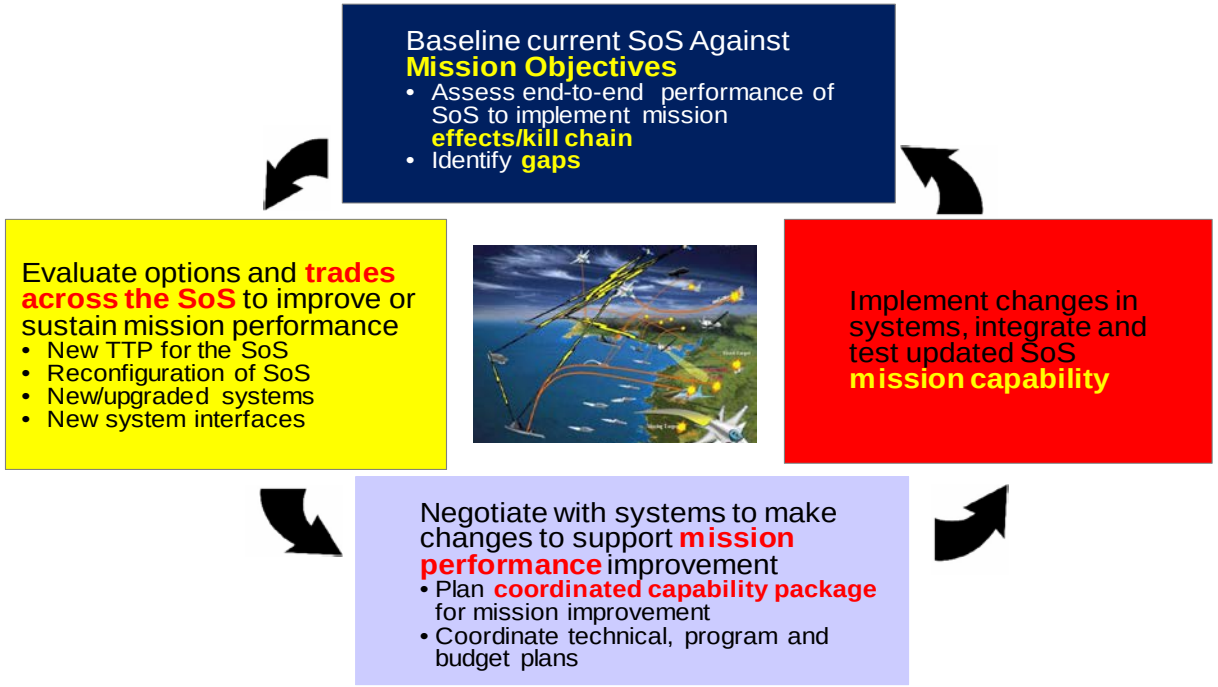
Key Activities in ME Process

A key starting point for ME is understanding current state of mission

- Operational mission objectives and CONOPS (mission threads)
- Current and planned systems
- Identifying critical, priority mission gaps

Technical assessment of options and trades

- Fault isolating sources of gaps
- Assessing alternative approaches to addressing capability gaps



Tracking implementation, integration and test

- Given independence of systems and development schedules

Planning and funding coordinated changes in systems

- 'Capability package' which cross systems owners and development schedules

MITRE

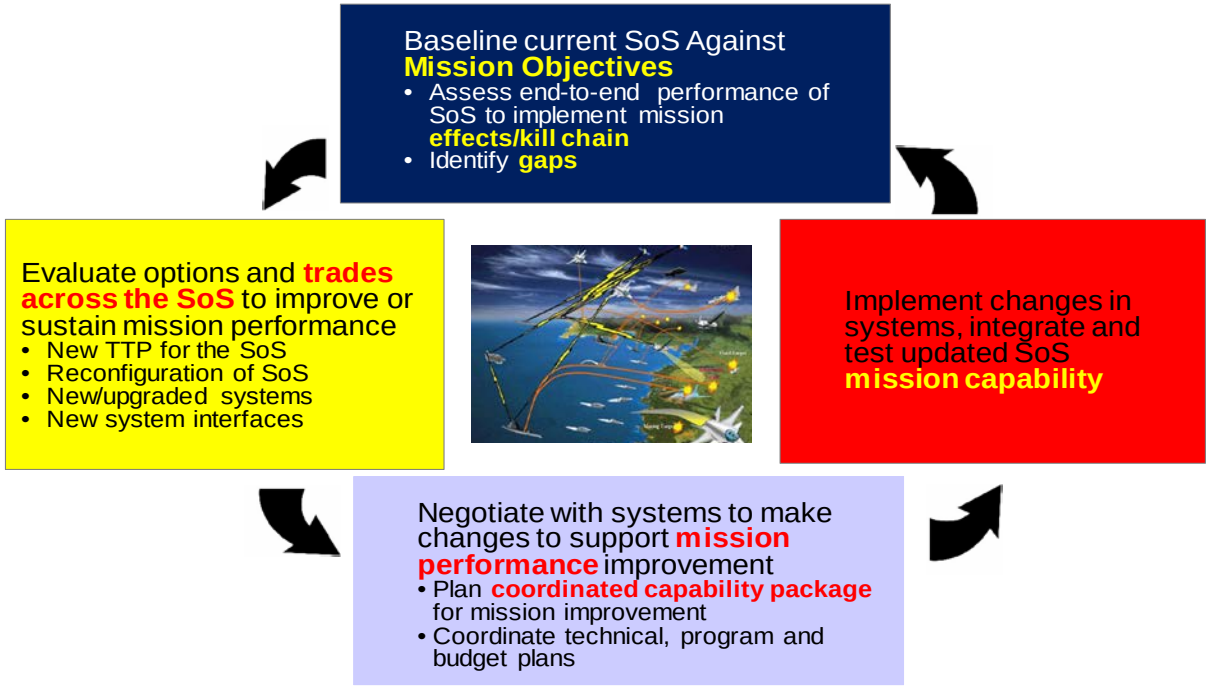
Key Activities in ME Process

A key starting point for ME is understanding current state of mission

- Operational mission objectives and CONOPS (mission threads)
- Current and planned systems
- Identifying critical, priority mission gaps

Technical assessment of options and trades

- Fault isolating sources of gaps
- Assessing alternative approaches to addressing capability gaps



Tracking implementation, integration and test

- Given independence of systems and development schedules

Planning and funding coordinated changes in systems

- 'Capability package' which cross systems owners and development schedules

MITRE

SoSE Technical Approaches to Address ME

Technical assessment of options and trades

- Fault isolating sources of gaps
- Assessing alternative approaches to addressing capability gaps

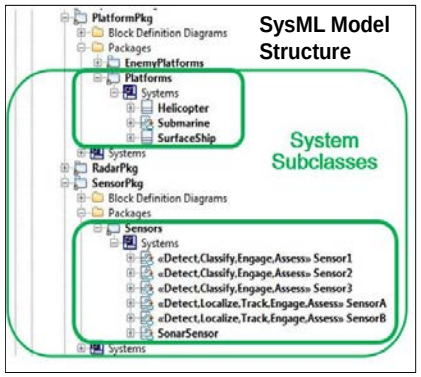
- **Mission environment**
- **Composition**
- **Mission ‘web’**

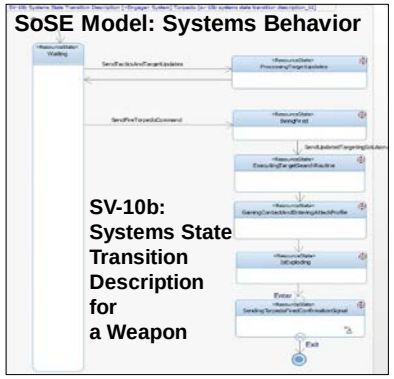
- **Scalable model-based approaches to SoS architecture representation**
- **Analytic approaches to SoS architecture assessment**
- **Assessing impacts of SoS architecture changes on operational mission outcomes**

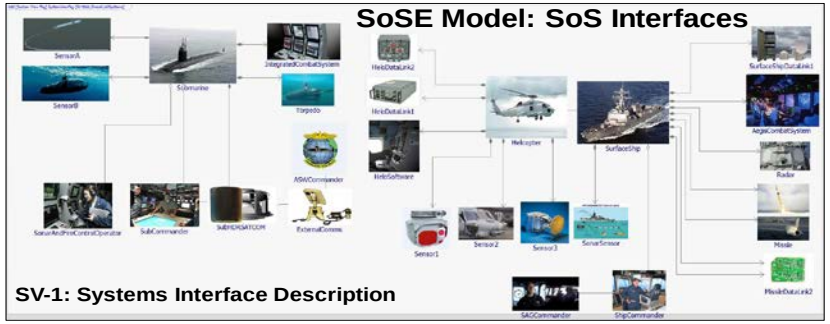
Approved for public release. Distribution unlimited 17-3712-15

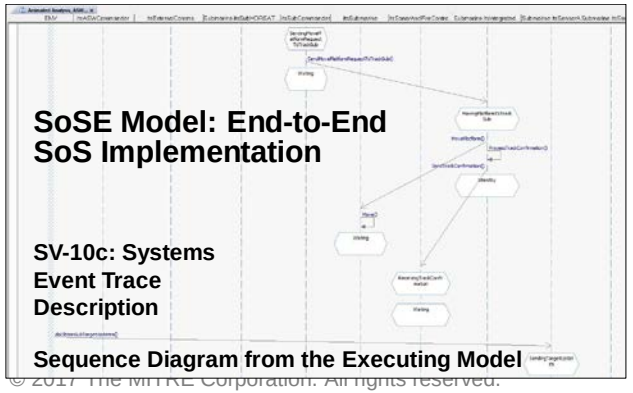
MITRE

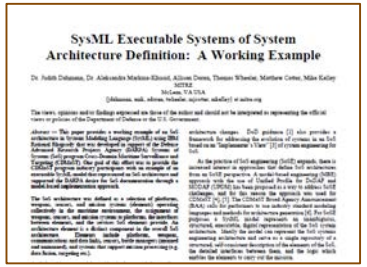
Model-Based SoSE



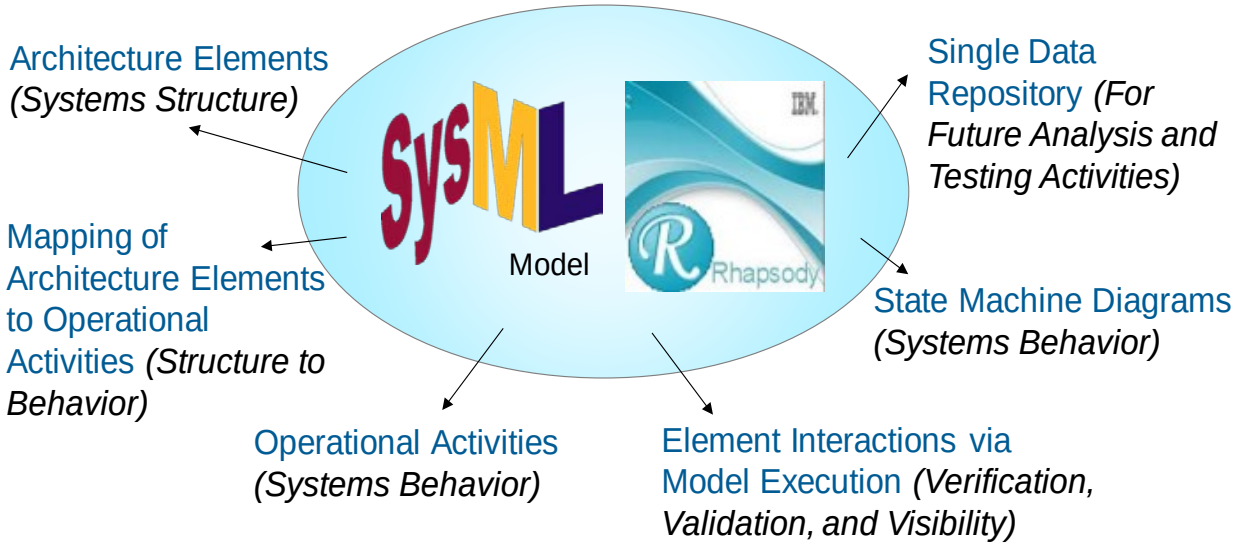







SysML Executable Systems of System Architecture Definition: A Working Example

- **For SoSE purposes, SysML model represents an unambiguous, structured, executable, digital representation of the SoS system architecture**



"SysML Executable Systems of Systems Architecture Definition: A Working Example"

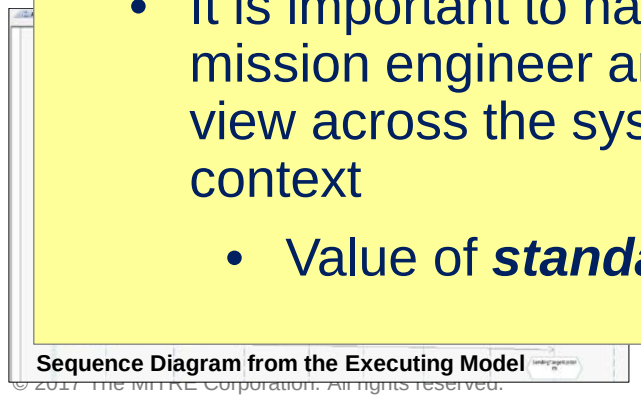
IEEE International Systems Conference
http://2017.ieeesyscon.org/

MITRE

Model-Based SoSE

Why is this important for mission engineering?

- The systems composed into an SoS architecture to support a mission are typically drawn from a **variety of specialty areas** (sensors, weapons, platforms, communications) and **diverse organizations** which bring **various perspectives** to the mission
  - ***Specificity*** provided by models can help avoid misunderstandings about system behavior, system interactions/interfaces (*Have I addressed all the needed interfaces to execute the end to end sequence of actions? Value of executable)*
- A model allows for representation of the complexity of the interrelations among systems in the mission, reflecting the variety of paths in the '***mission web***'
- It is important to have a ***commonly understood representation*** providing both the mission engineer and the constituent systems engineers a cross cutting integrated view across the systems and how they are expected to be employed in a mission context
  - Value of ***standards***-based modeling approaches



Sequence Diagram from the Executing Model

Scalable Model-Based SoSE

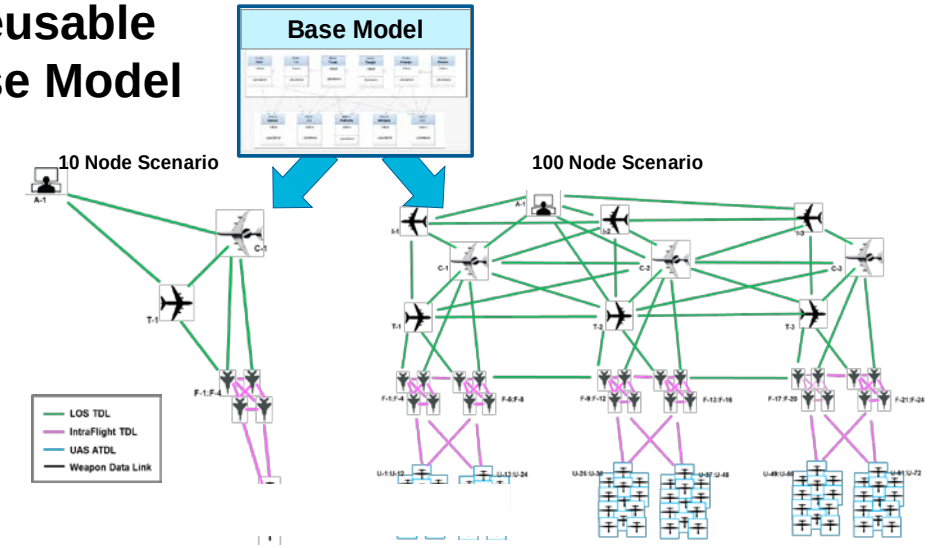
See NDIA paper 19802 for technical details

- **A key enabler of model-based SoSE is the ability to efficiently develop large complex SoS architecture model**

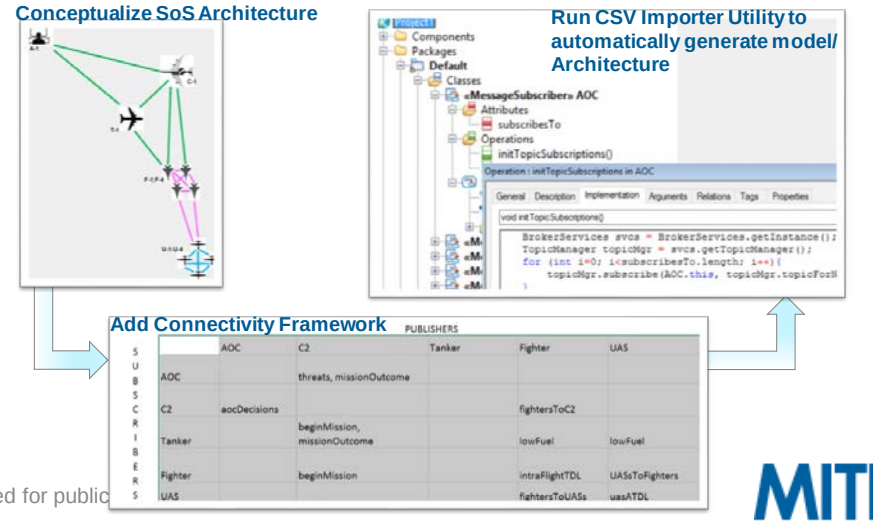
The effort required to build SoS architecture models can be reduced by starting the modeling process with a reusable **base model template**, independently of the architecture size

Tools can facilitate integration of SoS connectivity information into MBE tools, tightening the coupling between subject matter experts (SMEs), software engineers, and analysts -- comma separated variable (CSV) **importer tool**

Reusable Base Model



CSV Importer



MITRE

Scalable Model-Based SoSE

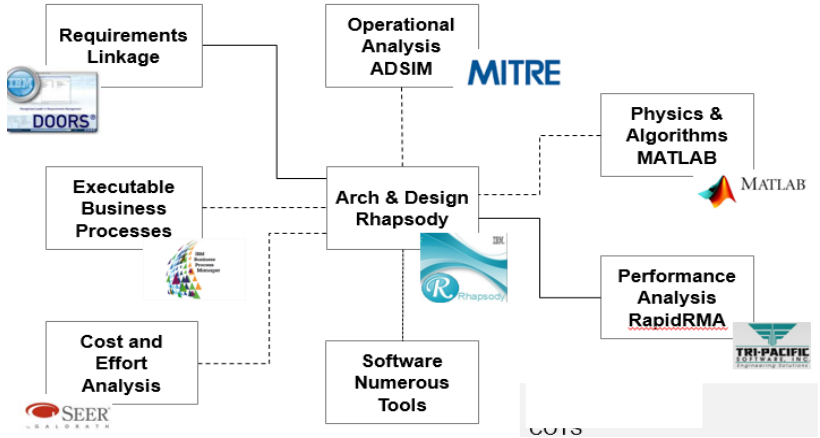
See NDIA paper 19804 for technical details

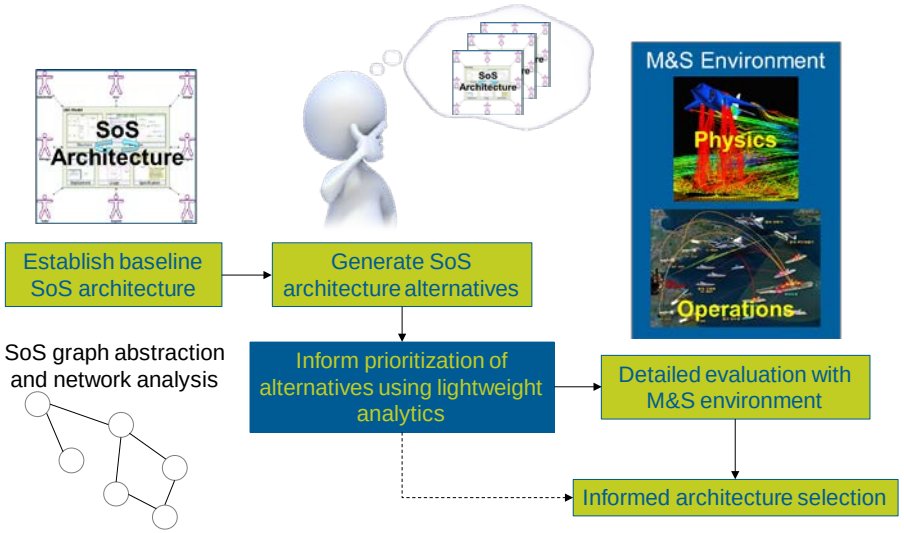
Why is this important for mission engineering?

- Missions can be large and comprise many systems, and the time required to develop a model framework for each mission architecture can raise the ***cost of entry*** for use of models to support mission engineering
- Gathering the ***needed data*** to understand the current state of a large mission can be difficult given the diversity of knowledgeable mission stakeholders.
  - Providing ***intuitive tools*** to allow stakeholders to share knowledge in a way familiar to them can build confidence and speed knowledge gathering
  - Automated transform ***directly into a model*** again lowers the cost of entry for large mission architecture, and reduces likelihood of errors or misunderstandings

MITRE

Analytic Approaches to SoS Architecture Assessment (1 of 2)



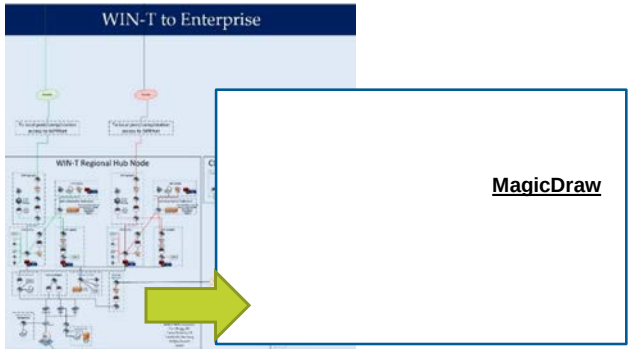


- **Representing SoS architecture in a model opens the options for analysis**
  - Interfacing a SoS model with other tools to assess performance, cost, other aspects of the SoS, provides a shared representation of the architectures for analysis from different perspectives
  - Developing approaches to assess alternative architectures is a challenge for the perspective of scalability
  - How do you identify viable options for more detailed analysis when there is such a large trade space?

MITRE

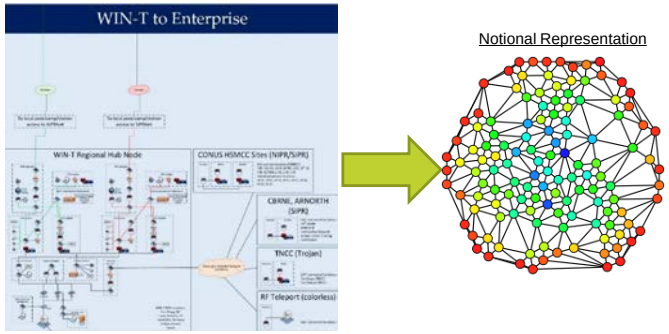
Analytic Approaches to SoS Architecture Assessment (2 of 2)

Thread Simulation

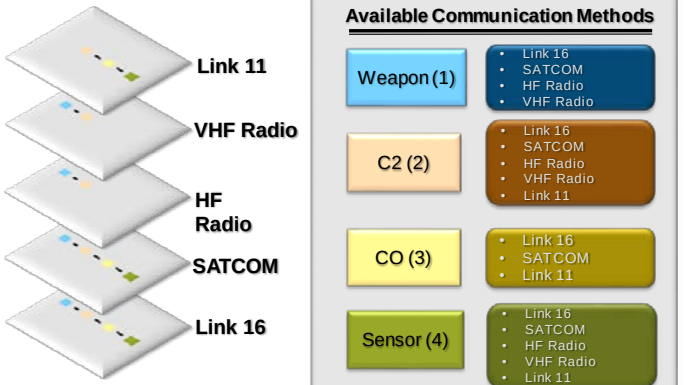


Identify Patterns and Inform Mitigation Strategies

Graph Theoretic Approach



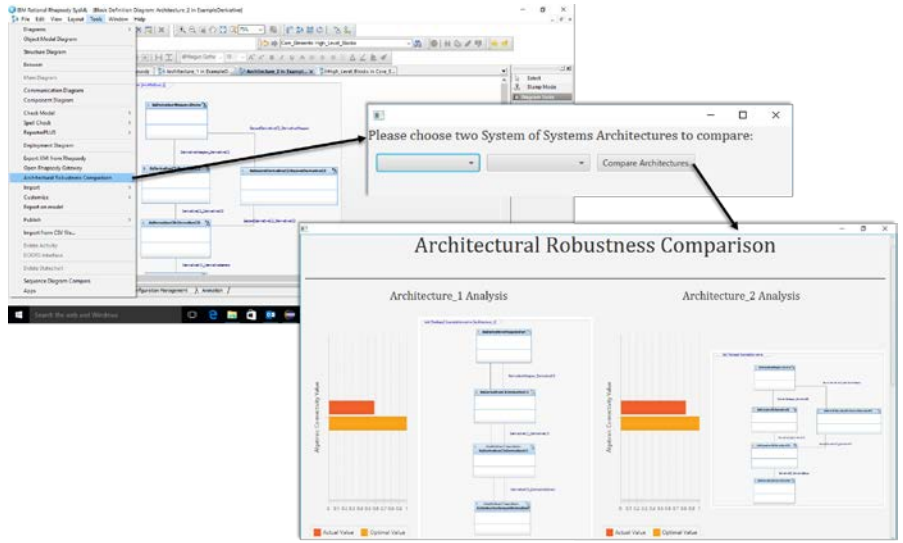
Identify vulnerable assets within the Army Network Architecture



	VHF Radio		HF Radio		Link 11			Link 16				SATCOM			
VHF Radio	0	1	1	0	0	0	0	1	0	0	0	1	0	0	0
	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0
HF Radio	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0
	0	1	1	0	0	0	0	0	1	0	0	0	1	0	0
Link 11	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0
	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0
	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1
Link 16	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0
	0	1	0	1	1	0	0	1	0	1	0	0	1	0	0
	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0
	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1
SATCOM	1	0	1	0	0	0	0	1	0	0	0	0	1	0	0
	0	1	0	1	1	0	0	0	1	0	0	1	0	1	0
	0	0	0	0	0	1	0	0	0	1	0	0	1	0	1
	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0

VHF Radio | Link 16 | Link 11 | HF Radio | SATCOM

- **Use of architecture data in a graph theoretic analysis**



See NDIA paper 19802 for technical details

MITRE

Analytic Approaches to SoS Architecture Assessment

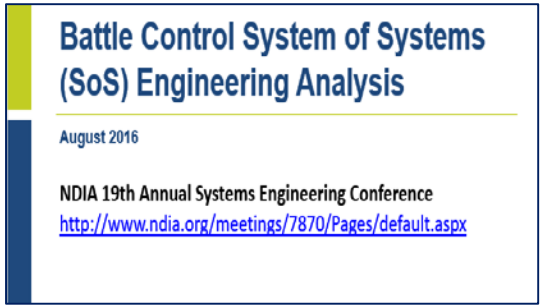
Why is this important for mission engineering?

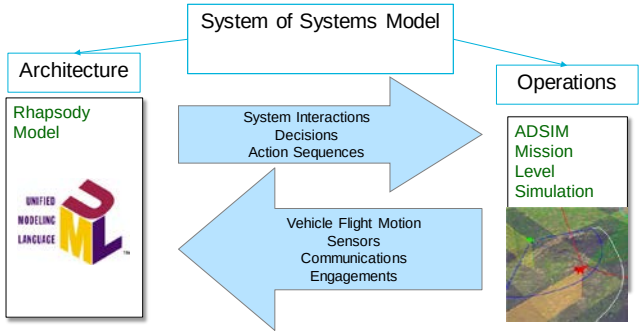
- Scale and complexity of missions require trades across multiple metrics and many solution options
- Lightweight analytic tools leverage architecture data to enable an initial quantification of mission impacts due to architecture changes
- This initial analysis can be used to filter out undesirable architecture options prior to investing resources to assess options with more detailed modeling and simulation tools

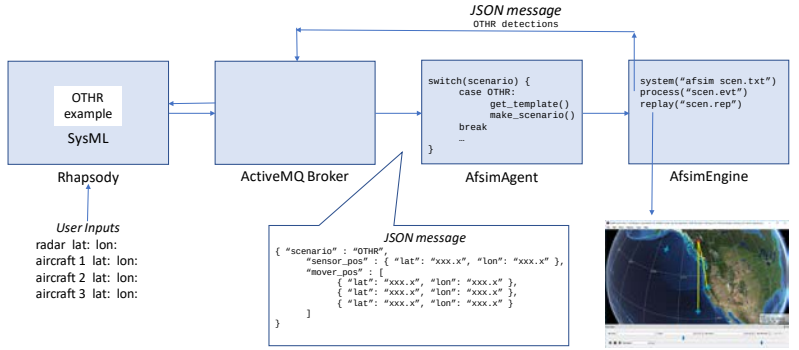
MITRE

Linking SoS Architecture to Operational Outcomes

- **Effectiveness of SoS for missions is based on mission outcomes**
  - SE analysis of SoS for missions addresses the technical feasibility of the SoS options
  - Analyzing alternative SoS architectures or specific SoS compositions also needs to consider the impact on mission outcomes, typically addressed in operational simulations or test environments
  - This includes ***developing automated interfaces between architecture models and operational simulations***, allowing for analysis of the effectiveness of the SoS in representation scenarios, following proposed concepts of employment
  - Examples include Rhapsody to ADSIM, more recently to AFSIM







MITRE

Linking SoS Architecture to Operational Outcomes

Why is this important for mission engineering?

- Mission engineering is all about achieving user operational capability
- Ensuring technical feasibility is an important prerequisite – it is key that systems work together as planned based on engineering across the systems supporting the mission
- But it is key that the mission SoS composition is fit for purpose in the mission environment – physical, threat, etc. – and when executed leads to the expected mission outcomes under anticipated conditions
- Mission SoS architectures can be complex, and it can be time consuming and error prone to have to manually instantiate these in today's operational simulations
- Automating this facilitates the conduct of the analysis of the mission effect or proposed or alternative SoS compositions, and it allows operators and commanders to see the proposed composition in their operation context

Summary

- **Mission engineering is an application of SoSE with specific driving characteristics**
- **As SoSE technical approaches and tools evolve, they provide valuable capabilities to enable technically based approaches to addressing mission engineering challenges**

Approved for public release. Distribution unlimited 17-3712-15

MITRE

Abstract

In the US Department of Defense there is increased interest in **mission engineering - the deliberate planning, analyzing, organizing, and integrating of current and emerging operational and system capabilities to achieve desired warfighting mission effects.** The Components have implemented mission engineering in areas where there is a critical interest in achieving mission capability such as ballistic missile defense or naval mission areas, and there is growing interest in addressing a broad set of mission areas through the implementation of mission integration management - the coordination all the programmatic elements - matching funding, schedules, technical improvements, resources (technical staff, development and test infrastructure, M&S etc.) across the relevant mission systems and supporting systems to develop, test, and field a phased set of mission capabilities. **One element of this is engineering of the systems of systems supporting the mission area.**

This presentation outlines the **key activities** involved in mission engineering and describes **opportunities for application of systems of systems engineering technical approaches** to these activities to provide the engineering base for mission integration and mission management. In particular, mission engineering often emphasizes the definition of the key activities need to execute the mission in the form of **mission threads or kill/effects chains and assessing gaps in mission performance.** Less attention has been paid to the various **patterns of mission activities and the engineering required to identify and assess alternatives to addressing the gaps and engineering the SoS to implement the preferred approach**. Drawing on work within the MITRE Systems Engineering Technical Center's model based engineering center, this presentation will present approaches to developing, representing and evaluating systems of systems architectures using model based methods and evaluating SoS configurations to address the functional needs of the mission which provide a set of approaches to supporting mission engineering.

Approved for public release. Distribution unlimited 17-3712-15

MITRE