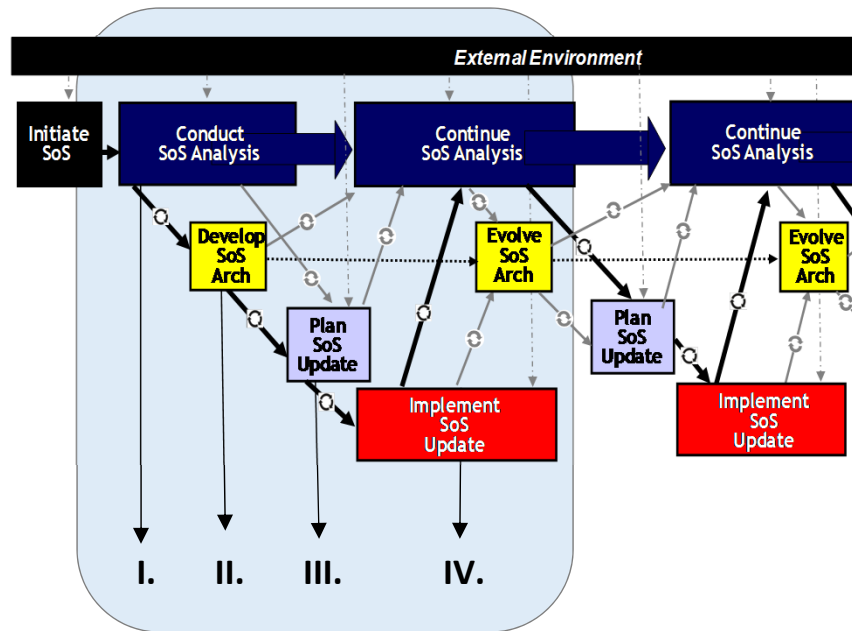


# Systems of Systems Engineering Measurement Workshop

June 13, 2016



# Measurement Through the SoS Life Cycle



**Measurement at Each Step in the Wave Model**

- At each step (I – IV)
  - What are the questions to be addressed?
  - What measures would you need?
    - Technical? Technical management?
    - At SoS level? At the system level?
  - What are the measurement challenges?
    - For technical and technical management
- Start with Acknowledged SoS
  - Assess what is different for other types

# Agenda



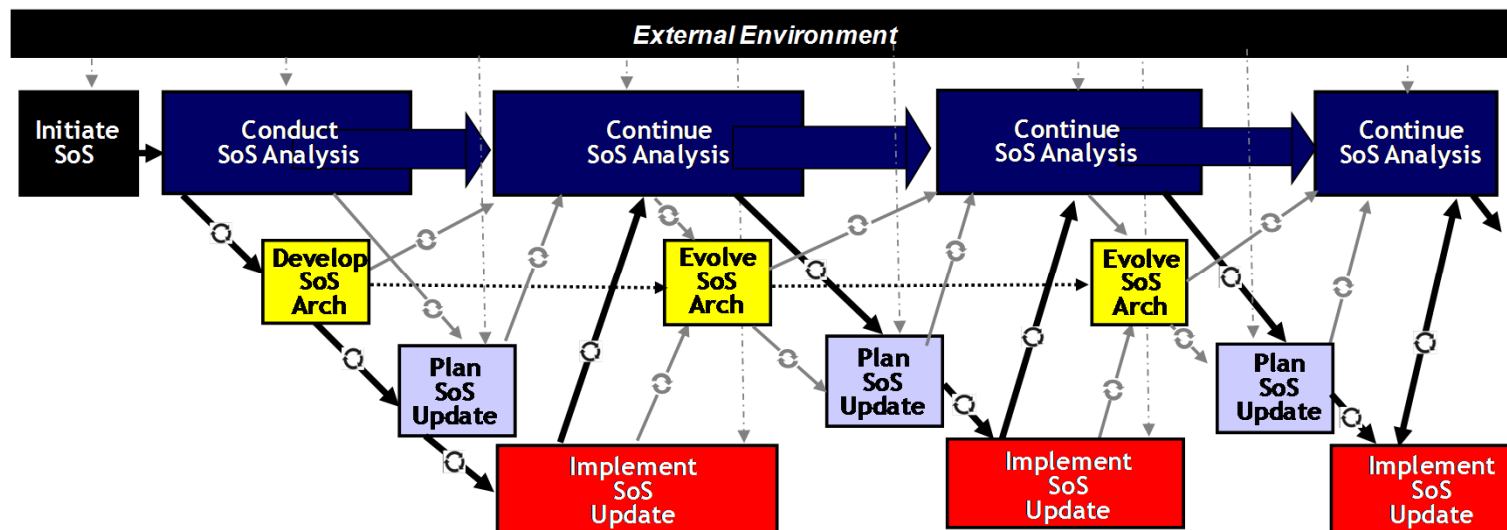
130 - 145	Intro and Setup
145 - 215	Measurement - Conduct SoS Analysis
215 - 245	Measurement - Evolve SoS Architecture
245 - 300	Break
300 - 330	Measurement - Plan SoS Update
330 - 400	Measurement - Orchestrate SoS Update
400 - 430	What would differ for other SoS Types?
415 - 500	Summary and Review

# SoSE Lifecycle

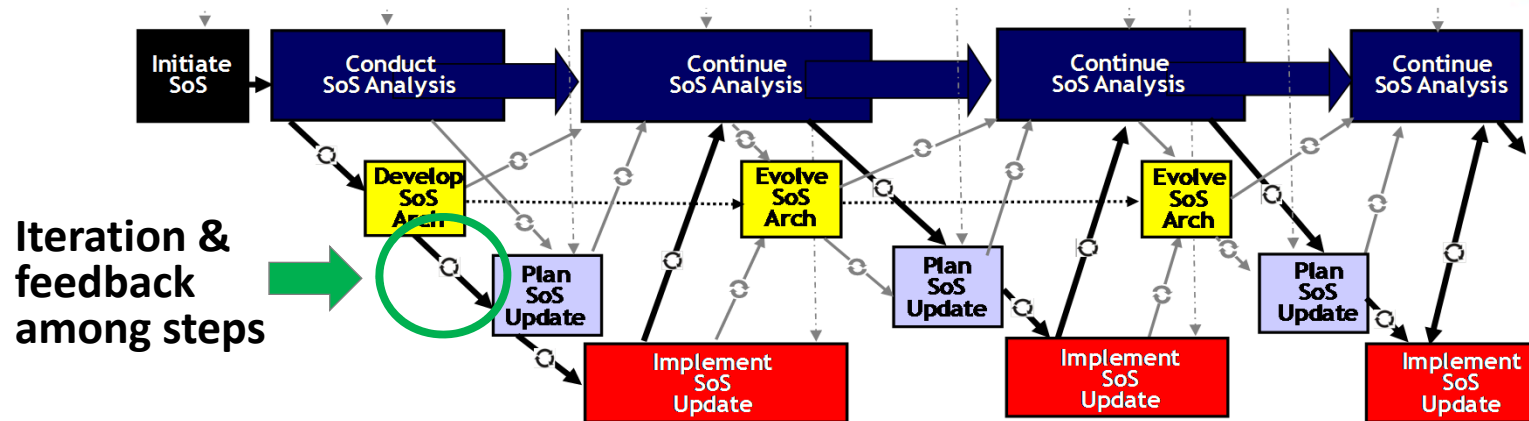


An evolutionary systems engineering approach to evolving complex systems and systems of systems

- Recognizes need for disciplined iterations to systematically address impacts of inevitable change
  - Backbone of ongoing analysis
  - Architecture evolution
  - Overlapping iterations
  - Forward movement with feedback




# SoS Wave Model Steps



- **Initiate SoS:**  
Provides foundational information to initiate the SoS
- **Conduct/Continue SoS Analysis:**  
Provides analysis of the 'as is' SoS and basis for its evolution
- **Develop/Evolve SoS Architecture:**  
Develops/evolves the persistent technical framework for SoS evolution and a migration plan identifying risks and mitigations
- **Plan SoS Update:**  
Evaluates SoS priorities, backlog of SoS changes, and options to define plans for the next SoS upgrade cycle
- **Implement SoS Update:**  
Oversees system implementations and plans/conducts SoS level testing, resulting in a new SoS product baseline
- **Continue SoS Analysis:**  
Ongoing SoS analysis revisits the state of and plans for the SoS as the basis for SoS evolution

# Focus on Acknowledged SoS



Type	Definition
Directed	Directed SoS are those in which the SoS is engineered and managed to fulfill specific purposes. It is centrally managed during long-term operation to continue to fulfill those purposes as well as any new ones the system owners might wish to address. The component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the centrally managed purpose.
Acknowledged	Acknowledged SoS have recognized objectives, a designated manager, and resources for the SoS; however, the constituent systems retain their independent ownership, objectives, funding, development, and sustainment approaches. Changes in the systems are based on cooperative agreements between the SoS and the system.
Collaborative	In collaborative SoS, the component systems interact more or less voluntarily to fulfill agreed-upon central purposes.
Virtual	Virtual SoS lacks a central management authority and a centrally agreed-upon purpose for the system of systems. Large-scale behavior emerges—and may be desirable—but this type of SoS relies upon relatively invisible, self-organizing mechanisms to maintain it.

- Assess how results would differ for other types of SoS

# 2016 Workshop Results



Area	Measurement Concept	Questions
SoS	Effectiveness	What are top-level objectives for the SoS?
		Is the SoS achieving the user capability objectives?
		How do stress conditions impact SoS effectiveness?
		How does SoS performance impact SoS effectiveness?
		What are gaps between expected and observed SoS effectiveness?
		What are root causes of gaps?
		How alternative architectures compare in terms of SoS effectiveness?
		TBD
	Performance	Is the SoS performing as expected?
		How do stress conditions impact SoS performance?
		How does the systems' performance impact SoS performance?
		TBD
System	Effectiveness	Do the systems maintain system level effectiveness when part of the SoS?
		Which systems have largest impact SoS effectiveness?
		TBD
	Performance	How are the systems performing in the SoS context?
		How do the systems contribute to the SoS?
		TBD

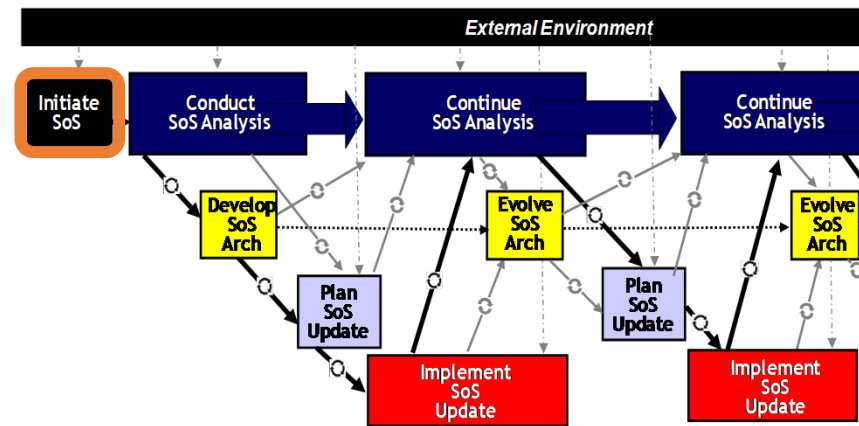
# Initiate SoS

Establish foundations for SoS engineering



Provides the basic information needed to start the SoS SE process, including an understanding of the

- SoS objectives
- The key players, their roles and expectations
- Core systems supporting capabilities



## Artifacts

- A statement of **top-level objectives for the SoS**
- Identification of key systems currently supporting **the mission or capability**
- A description of how systems in the SoS will be **employed** in an operational setting
- **Programmatic and technical information** about systems that affect SoS capability objectives
- Initial identification of risks



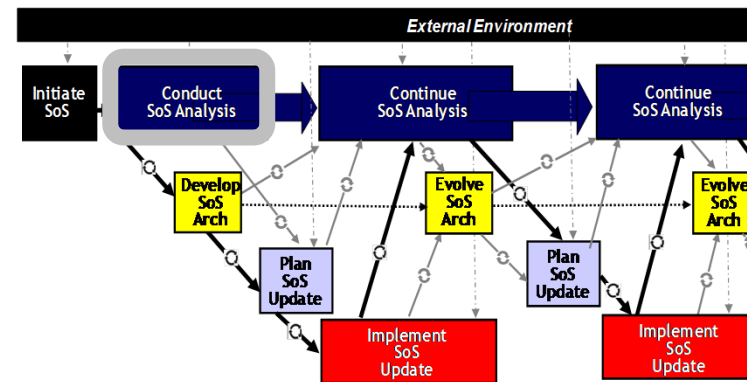
# Conduct SoS Analysis

Characterize and analyze current SoS in terms of SoS objectives



Provides an analysis of the “as is” SoS, including

- Describe current SoS
  - CONOPS
  - Systems
  - System relationships
- Assess performance against objectives
  - Expected performance
  - Actual (measured) performance
- Fault isolation
  - Source of gaps



## Artifacts

- Capability objectives
- SoS CONOPs
- Constituent system info
- SoS Technical Baselines
- SoS Performance Measures & Methods
- SoS Performance Data
- SoS Requirement Space
- SoS Risks & Mitigations
- SE Planning Elements
- SoS Master Plan
- Agreements

# Conduct SoS Analysis

Characterize and analyze current SoS in terms of SoS objectives



## From 2016 Workshop

- Questions
  - SoS effectiveness:
    - Is the SoS achieving the user capability objectives?
    - Measurement....Independent of the effectiveness of the systems to address their specific user needs
  - SoS performance
    - Is the SoS performing as expected?
  - System performance in SoS context
    - How are the systems performing in the SoS context? How do they contribute to the SoS?
- Challenges
  - Identifying SoS measures which are acceptable to multiple stakeholders
    - Particularly when not aligned to systems
  - Acquiring data on systems and interactions
    - Availability, access, consistency
  - Linking SoS performance to effectiveness

# Conduct SoS Analysis

Characterize and analyze current SoS in terms of SoS objectives



- What are the questions to be addressed?
- What measures would you need?
  - Technical?
    - At SoS level?
    - At the system level?
  - Technical management?
    - At SoS level?
    - At the system level?
- What are the measurement challenges?
  - For technical?
  - Technical management?

# Develop SoS Architecture

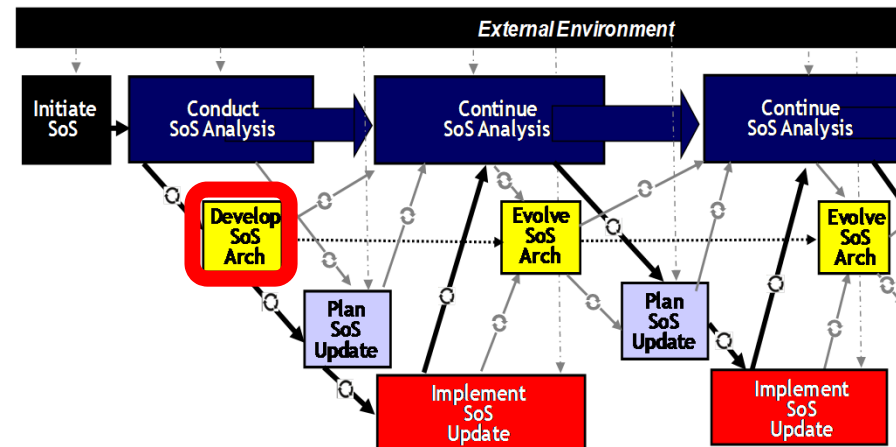
Identifying, analyzing and recommending changes to the SoS architecture



Technical analysis of changes or alternatives to the current architecture to improve SoS effectiveness or performance

- New, added, different or updated systems in current architecture
- New architecture with new ways to organize and employ systems

Produces recommendations for changes



## Artifact - SoS Architecture

- Defines the way in which the constituent systems work together
- Includes systems, SoS functions, relationships and dependencies, as well as end-to-end functionality, data flow & communications

# Develop SoS Architecture

Identifying, analyzing and recommending changes to the SoS architecture



## From 2016 Workshop

- Build on earlier measures
  - Reuse measures developed to assess the current SoS as the basis for assessing alternatives
  - Comparing change options with the current to develop recommendations based on analysis of impact on SoS effectiveness and performance
- Challenges
  - With new or added systems or systems upgrades it may be difficult to get data on both systems and interactions
    - Especially new for new systems or proposed changes
  - Linking – important when assess options
    - Changes in systems performance to SoS performance
    - Changes in SoS performance to impacts of SoS objectives

# Develop SoS Architecture

Identifying, analyzing and recommending changes to the SoS architecture



- What are the questions to be addressed?
- What measures would you need?
  - Technical?
    - At SoS level?
    - At the system level?
  - Technical management?
    - At SoS level?
    - At the system level?
- What are the measurement challenges?
  - For technical?
  - Technical management?

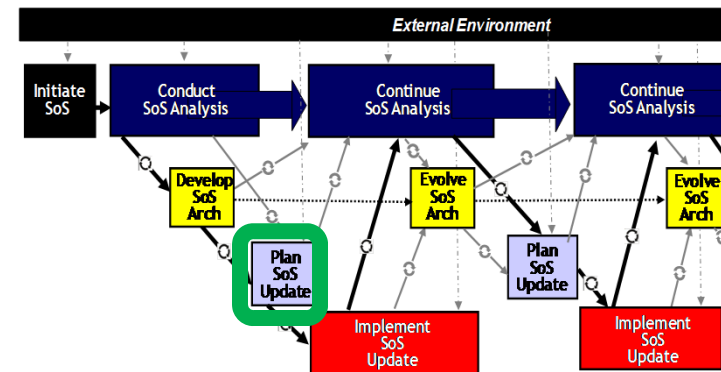
# Plan SoS Update

Plan changes to SoS



## Plan SoS Update

- Evaluate SoS priorities, options and backlogs
- Define the plan for the next SoS upgrade cycle.



## Artifacts

- An allocated baseline
- Risks and mitigations
- Agreements
- Implementation, integration & test plans
- An integrated master schedule (IMS)
- Updated Master Plan and Requirements space

# Plan SoS Update

Plan changes to SoS



- What are the questions to be addressed?
- What measures would you need?
  - Technical?
    - At SoS level?
    - At the system level?
  - Technical management?
    - At SoS level?
    - At the system level?
- What are the measurement challenges?
  - For technical?
  - Technical management?



# Orchestrate SoS Update

Plan, implement, and test system changes to SoS



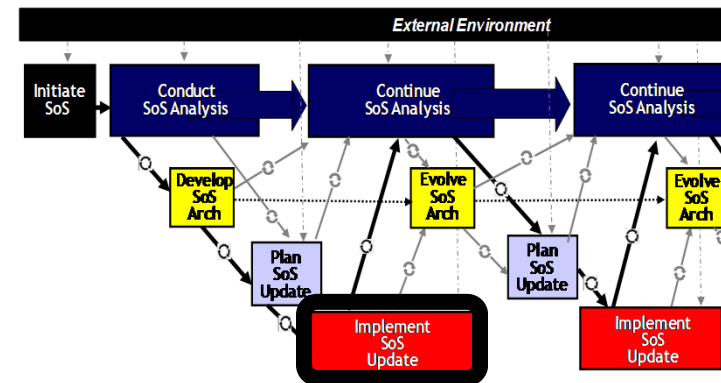
## Implement SoS Update

### Systems

- Implement and test changes systems
- SoS SE team leads SoS integration and test

### SoS SE

- Monitors system implementation
- Conducts SoS testing
- Addresses unanticipated factors encountered



## Artifacts

- SoS Test Report
- SoS Technical Plans, Requirements Space, Performance Data
- System Test Reports
- SoS IMS
- SoS Technical Baselines

# Orchestrate SoS Update

Plan, implement, and test system changes to SoS



## From 2016 Workshop

- The measures for the SoS performance and effectiveness developed in previous steps will be employed here for SoS integration and test
- Implementation is done by the systems
  - SoS SE monitors progress of the systems development of changes
- These implementation plans are in effect project level measures
  - These are currently well supported by the ICM developed by the PSM

# Orchestrate SoS Update

Plan, implement, and test system changes to SoS



- What are the questions to be addressed?
- What measures would you need?
  - Technical?
    - At SoS level?
    - At the system level?
  - Technical management?
    - At SoS level?
    - At the system level?
- What are the measurement challenges?
  - For technical?
  - Technical management?

# SoS Types



Type	Definition
Directed	Directed SoS are those in which the SoS is engineered and managed to fulfill specific purposes. It is centrally managed during long-term operation to continue to fulfill those purposes as well as any new ones the system owners might wish to address. The component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the centrally managed purpose.
Acknowledged	Acknowledged SoS have recognized objectives, a designated manager, and resources for the SoS; however, the constituent systems retain their independent ownership, objectives, funding, development, and sustainment approaches. Changes in the systems are based on cooperative agreements between the SoS and the system.
Collaborative	In collaborative SoS, the component systems interact more or less voluntarily to fulfill agreed-upon central purposes.
Virtual	Virtual SoS lacks a central management authority and a centrally agreed-upon purpose for the system of systems. Large-scale behavior emerges—and may be desirable—but this type of SoS relies upon relatively invisible, self-organizing mechanisms to maintain it.

- How would this differ for other SoS types?

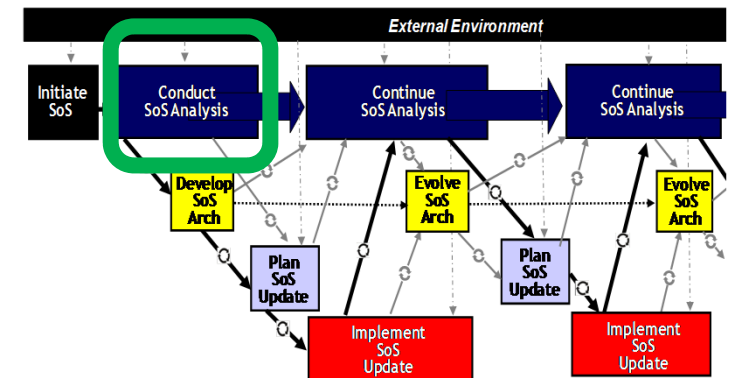
# Summary and Discussion



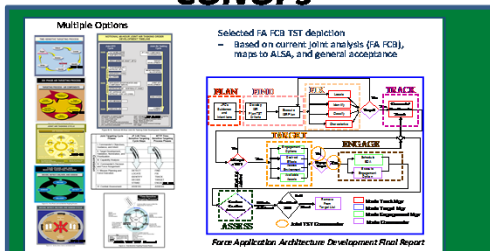
# Backup

# Conduct SoS Analysis

Provides analysis of the 'as is' and basis for SoS evolution

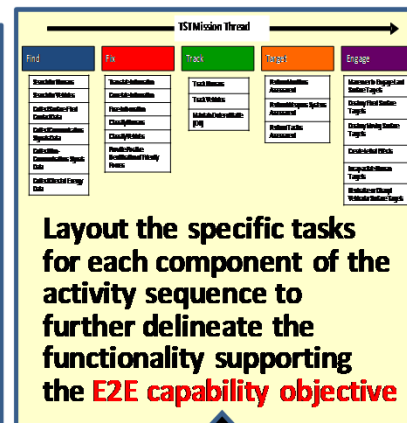


## CONOPS

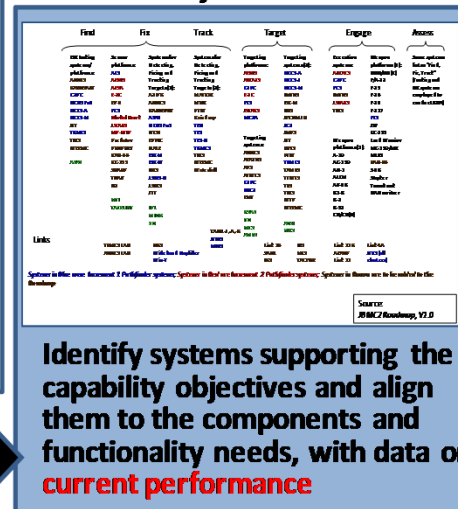


Understand operational context and developing a **CONOPS** -- Includes key steps in process and constraints on those steps; may be a set of mission threads, conditions, players and **performance objectives**

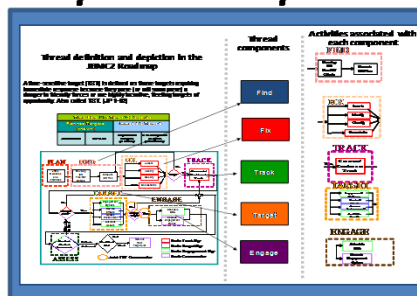
## Functional Baseline



## Current System Baseline



## Requirements Space



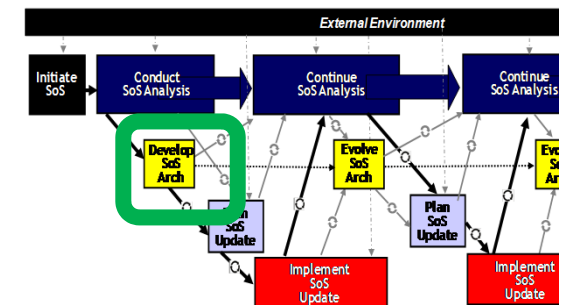
Develop an 'functional architecture' for the SoS by looking at the key functions to be supported across the 'thread' or activity sequence, including **performance objectives**

Results provide basis for architecture development and planning for SoS updates

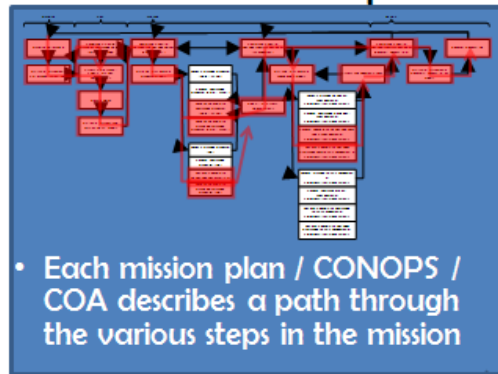
# Develop SoS Architecture



Develops and evolves the persistent technical framework for addressing SoS evolution

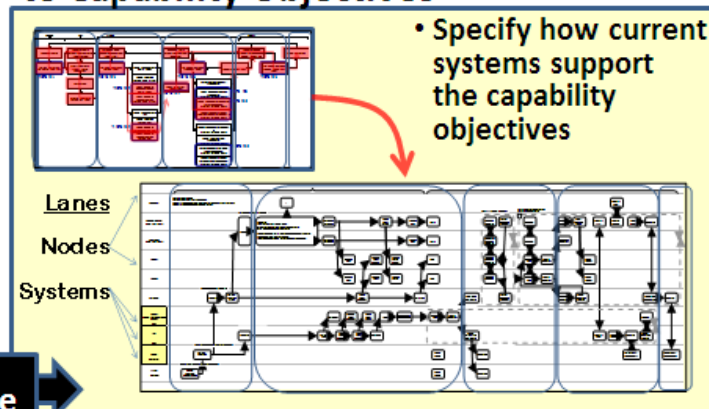


## Delineate E2E SoS Capabilities



- Each mission plan / CONOPS / COA describes a path through the various steps in the mission

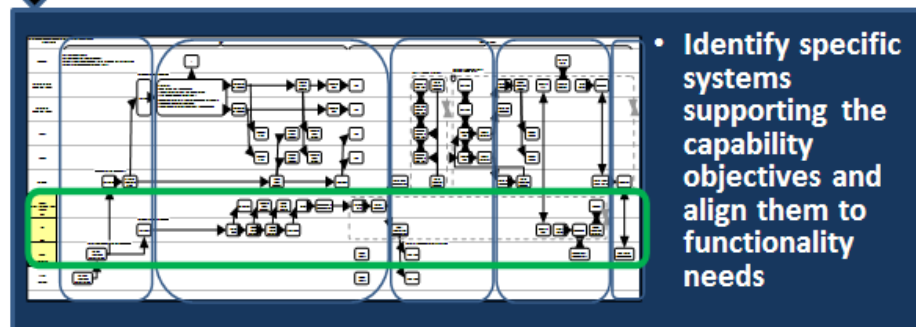
## Identify Systems Contributing to Capability Objectives



- Specify how current systems support the capability objectives

Identify and evaluate alternative approaches to organizing and augmenting systems to meet SoS needs

SoS Architecture



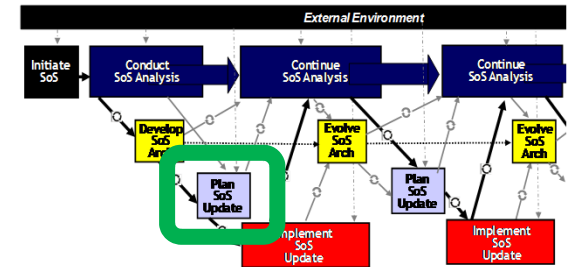
- Identify specific systems supporting the capability objectives and align them to functionality needs

Align Systems (Current Capabilities) with SoS Functional Needs

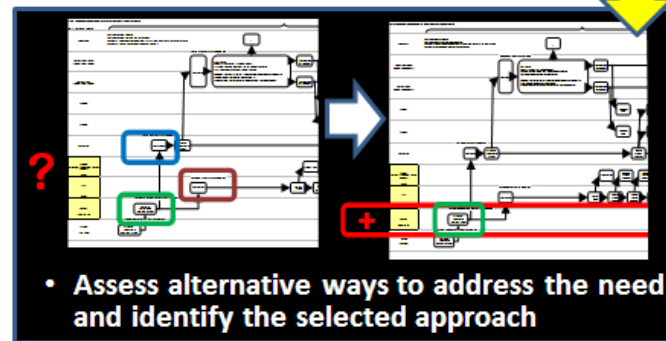
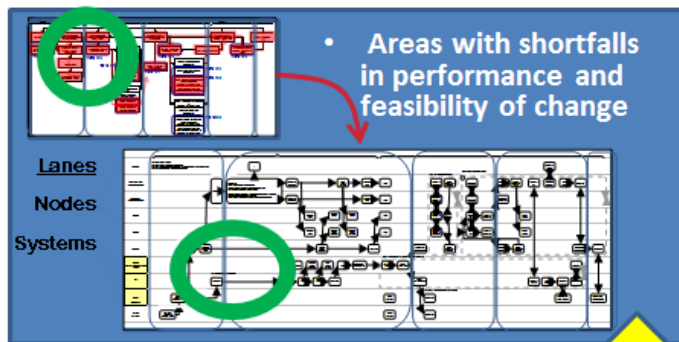


# Plan SoS Update

Evaluates the SoS priorities, options and backlogs to define the plan for the next SoS upgrade cycle



Identify Needs  
to be Addressed in this Wave



Evaluate Options for Addressing Needs

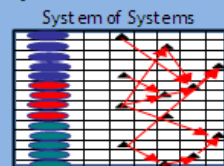
Plans for System and  
SoS Development,  
Integration and Test

SoS

- Integrated Master Schedule (Key sync points (not aggregation of plans))
- Risks and Mitigation Plans
- SoS changes and dependencies which drive testing

Systems

- Additions to system plans for development and test

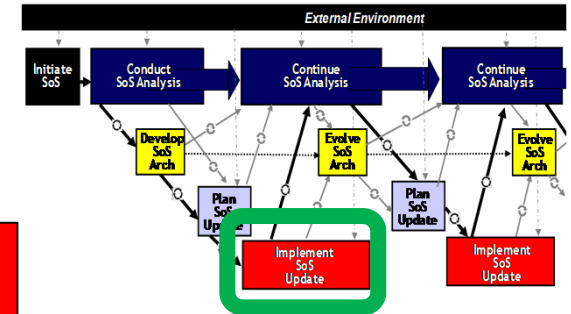


Constituent Systems



# Implement SoS Update

**Monitors implementations at the system level and plans and conducts SoS level testing, resulting in a new SoS product baseline**



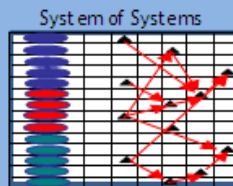
## Monitor System and SoS Development, Integration and Test

### SoS

- Integrated Master Schedule (Key sync points, not aggregation of plans)
- Risks and Mitigation Plans
- SoS changes and dependencies which drive testing

### Systems

- Additions to system plans for development and test



### Constituent Systems



19

## Review Progress And Inform Users and SE Process

- Collect and assess data from system and SoS development technical reviews and tests
- Update product baseline, architecture, performance assessments, and requirements space
- Provide input into 'Continue SoS Analysis'

