



# INCOSE (MBSE) Model Based System Engineering (SoS) System of Systems Activity Introduction

Ron Williamson, Ph.D.  
Raytheon

[ron.williamson@incose.org](mailto:ron.williamson@incose.org)

Jan 30-31, 2011

INCOSE IW11 MBSE Workshop

MBSE Wiki page: <http://www.omgwiki.org/mbse>

MBSE SoS/Enterprise Modeling Wiki page:

<http://www.omgwiki.org/MBSE/doku.php?id=mbse:enterprise>



# Outline

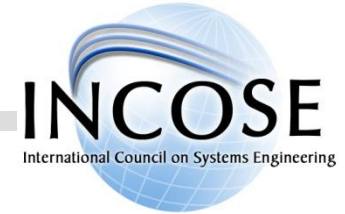


- Introduction
- Conceptual Model Summary for SoS
- Concept Representations
  - Languages
  - Frameworks
  - Patterns
- MBSE SoS Challenges
- Systems Language Models for SoS
  - SysML (System Modeling Language)
- Architecture Framework (AF) Models for SoS
  - UPDM (UML(Unified Modeling Language) Profile for DoDAF/MODAF)
- MBSE SoS Case Studies
  - Architecture Eco-System Efforts
  - UPDM and DoDAF 2.0 DM2
  - UPDM and SysML, SoaML, BPMN, BMM, etc.
  
- Questions...hold for the end of the session



# Introduction

## MBSE System of Systems

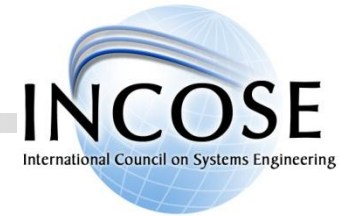


- System of Systems (SoS)...one of many definitions/characterizations
  - A class of problems that have unique characteristics, distinguishing them for “classic” systems.
    - For example, unbounded context and usage, potentially emergent behaviors, large number of complex interactions, costly to fully verify and validate a priori,...
  - These unique characteristics have lead the SE and Architecting community to investigate new languages and frameworks to help better define these key SoS characteristics
- SoS Engineering
  - Best Practices in Analysis, Architecture, Design, Development, Integration, Testing, Deployment and Maintenance
  - Modeling is increasingly critical to understanding, managing and validating
  - SoS modeling (e.g. SysML, MARTE, Modelica, eXtend, SimuLink, ...)
- SoS Architecting
  - Architecture Frameworks (DoDAF, MODAF, FEAf, Zachman, TOGAF,.....)
  - Model Based Frameworks (e.g. UPDM - Unified Profile for DoDAF/MODAF)



# Introduction SoS Engineering

## Key Concepts



	Traditional Systems Engineering	System-of-Systems Engineering
Purpose	Development of single system to meet stakeholder requirements and defined performance	Evolving new system-of-systems capability by leveraging synergies of legacy systems
System Architecture	System architecture established early in lifecycle and remains relatively stable	Dynamic reconfiguration of architecture as needs change; use of service oriented architecture approach as enabler
System Interoperability	Defines and implements specific interface requirements to integrate components in system	Component systems can operate independently of SoS in a useful manner Protocols and standards essential to enable interoperable systems
System "ilities"	Reliability, Maintainability, Availability are typical ilities	Added "ilities" such as Flexibility, Adaptability, Composeability
Acquisition and Management	Centralized acquisition and management of the system	Component systems separately acquired and continue to be managed as independent systems
Anticipation of Needs	Concept phase activity to determine system needs	Intense concept phase analysis followed by continuous anticipation, aided by ongoing experimentation

- **SoS Engineering Key Concepts**
  - Legacy Systems
  - Dynamic Reconfiguration of Architecture
  - Service Oriented Architecture Enabler
  - Protocols and Standards to Enable Interoperable Systems
  - Added "ilities" or Quality Attributes
  - Federated Acquisition
  - Independent Systems
  - Concept of Operations Critical
  - Ongoing Experimentation
  - Converging Spirals

## SoS Modeling Implications →

Saunders, T. *et al*, "United States Air Force Scientific Advisory Board Report on System-of-Systems Engineering for Air Force Capability Development," SAB-TR-05-04, July 2005



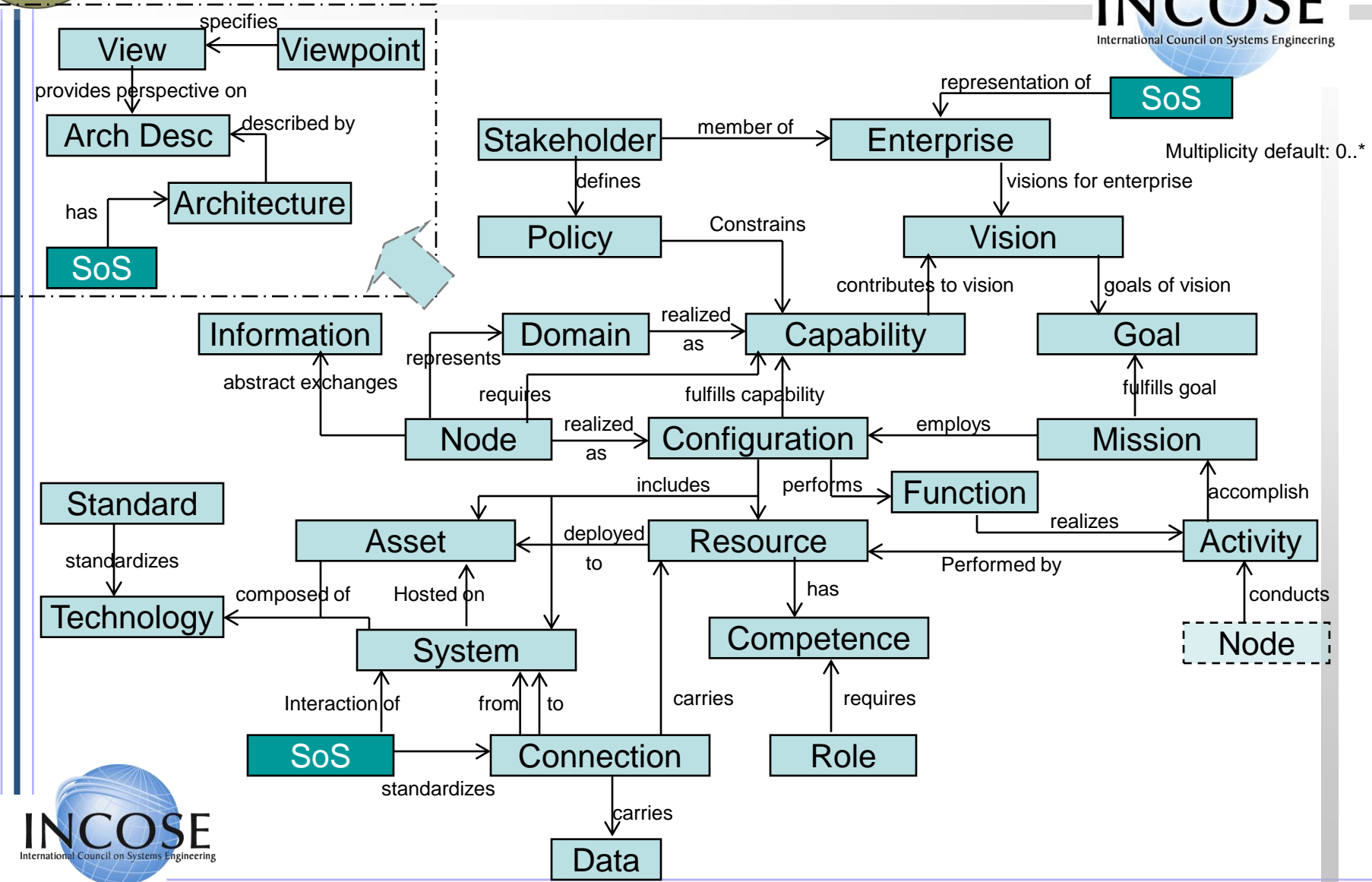


# Introduction

## ...SoS MBSE Implications

Legacy Systems	→	Models for behavior, interfaces, requirements, performance, e.g. SysML, Modelica, MARTE
Dynamic Reconfiguration of Architecture	→	Dynamic Reconfigurable models of architecture, e.g. UPDM with UML/SysML model version management
Service Oriented Architecture Enabler	→	SOA modeling language, e.g. SoaML, SOA Patterns
Protocols and Standards to Enable Interoperable Systems	→	Models for protocols, standards, interoperability, e.g. UPDM, DoDAF 2 MetaModel
Added “ilities” or Quality Attributes	→	Specialty Engineering models, e.g assurance
Federated Acquisition	→	Models for acquisition project synergy, e.g. UPDM, MODAF, DoDAF 2 MetaModel
Independent Systems	→	Models for independence in system functionality, e.g. Agent Based, federated models
Concept of Operations Critical	→	Models for CONOPs including Mission, Objectives, Courses of Action, etc. e.g. UPDM Operational Viewpoint, BPMN Business Processes
Ongoing Experimentation	→	Analysis of Alternatives models for all viewpoints and model versioning

# MBSE SoS Conceptual Model (partial)



Multiplicity default: 0..\*

# Some MBSE SoS Challenges

- Core Concepts have a wide range of interpretations and definitions across modeling languages
  - Duality: System of Systems and Model of Models
  - OMG Initiative: “Ecosystem” of Languages/Models
- Methodology / Discipline differences expand into SoS Engineering
  - Object Oriented vs Structured/Functional
  - Enterprise vs SoS vs System
  - Business vs Engineering Models (BPMN vs UML vs SoaML vs SysML)
  - Enterprise, Business and Technical Architecture Models (pick your favorite Architecture Frameworks)
- Example Concepts with several interpretations
  - Capability
  - Function
  - Activity
  - Requirement
  - View
  - Viewpoint
- Example Languages with overlap
  - BPMN and UML (SysML, UPDM)

UML/BPMN Integration Straw Poll (source OMG)

They remain separate standards	3
<b>BPMN is a UML profile with notation</b>	<b>6</b>
<b>Create a unified model encompassing both</b>	<b>13</b>
<b>Semantic models with UML and BPMN viewpoints</b>	<b>9</b>
BPMN replaces UML activity diagrams	4
BPMN grows to make UML not required	0
BPMN and UML are separate models, mapped with QVT	2
There are ways to make links between them	3



# Systems Language Models for SoS



## • SysML Core Concepts

- Structure, Behavior, Requirements, Parametrics
  - View, Viewpoint, Block, Part, Role, Connector, Interface, Item, ItemFlow, Activity, State, Transition, Requirement, Constraint Block,...
- SoS Core Concepts
  - View, Viewpoint, Enterprise, Mission, Projects, Milestone, Vision, Goal, Policy, Capability, Node, Configuration, Resource, System, Information, Data, Technology, Standard, Organization, Task, Activity, Measures of Effectiveness, Key Performance Parameters, “ilities”, Scenario, Workflow...

## • SysML/SoS Mapping Example (one of several approaches)

- Structure (Block,...)
  - Enterprise, Capability, Configuration, Resource, Systems, Information, Data, Technology, Organization, Milestone, Vision, Goal, Node, ...
- Behavior (Activity, State,...)
  - Function, Task, Activity, Scenario, Workflow,
- Requirement
  - Policy, Constraint, Standard,...
- Parametrics
  - MoE's, KPP's, “ilities”...

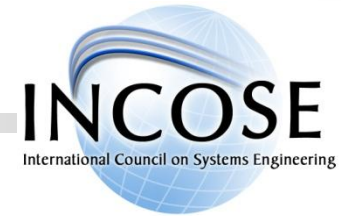
See UPDM and DoDAF Meta model  
References for mapping standards efforts







# Architecture Framework (AF) Models for SoS



- Zachman Framework
  - Perspectives, Interrogatives, Checklist
- TOGAF 9 (The Open Group AF)
  - Architecture Development Model
- FEAF (Federal Enterprise AF)
  - Reference Models (Business, Technical, Information, ...)
- DoDAF 2 / MODAF / NATO AF / ...
  - Viewpoints, Products for Capability, Operational, System, Service, Technology Standards, Information, ... Views
- ...and many additional variants of various combinations of the above frameworks



# MBSE SoS Case Studies

- Architecture Eco-System Efforts
  - Special Interest Group at OMG
  - Co-Chairs:
    - Jim Amsden (IBM)
    - Cory Casanave (Model Driven Solutions)
- UPDM and DoDAF 2.0
  - UPDM 1.0 official OMG standard
    - Co-Chairs
      - Jim Rice, NoMagic
      - Graham Bleakley, IBM
      - Matthew Hause, Atego
    - DoD
      - Walt Okon, OSD
      - Len Levine, DISA

