International Council on Systems Engineering
A better world through a systems approach

Space-Domain Enterprise Architecture (SEA) Reference Model

Developing a Starter Kit for Space Domain Architectures

INCOSE International Symposium 2025 |
Ottawa, Canada

Space-Domain Enterprise Architecture (SEA) Reference Model

James N Martin, PhD
Distinguished Engineer
Enterprise Systems Engineering
Enterprise Effects Division

Primary Authors:
Kyle Alvarez, Mission Engineering
Edith Szarkowski, Digital SE

INCOSE Symposium
Ottawa, Canada
29 July 2025

© The Aerospace Corporation, 2025

Approved for public release: OTR 2024-004933

Primary Authors

Kyle Alvarez

Enterprise Systems Engineer with The Aerospace Corporation

Kyle develops Digital Engineering models in the Mission Engineering Department supporting various US Air Force and Space Force government programs. Mr. Alvarez earned a BS and MS in Aeronautical and Astronautical Engineering from Purdue University. He plans on leveraging systems engineering expertise for use on large projects supporting the space enterprise to guarantee mission success in a timely and affordable manner. kyle.e.alvarez@aero.org

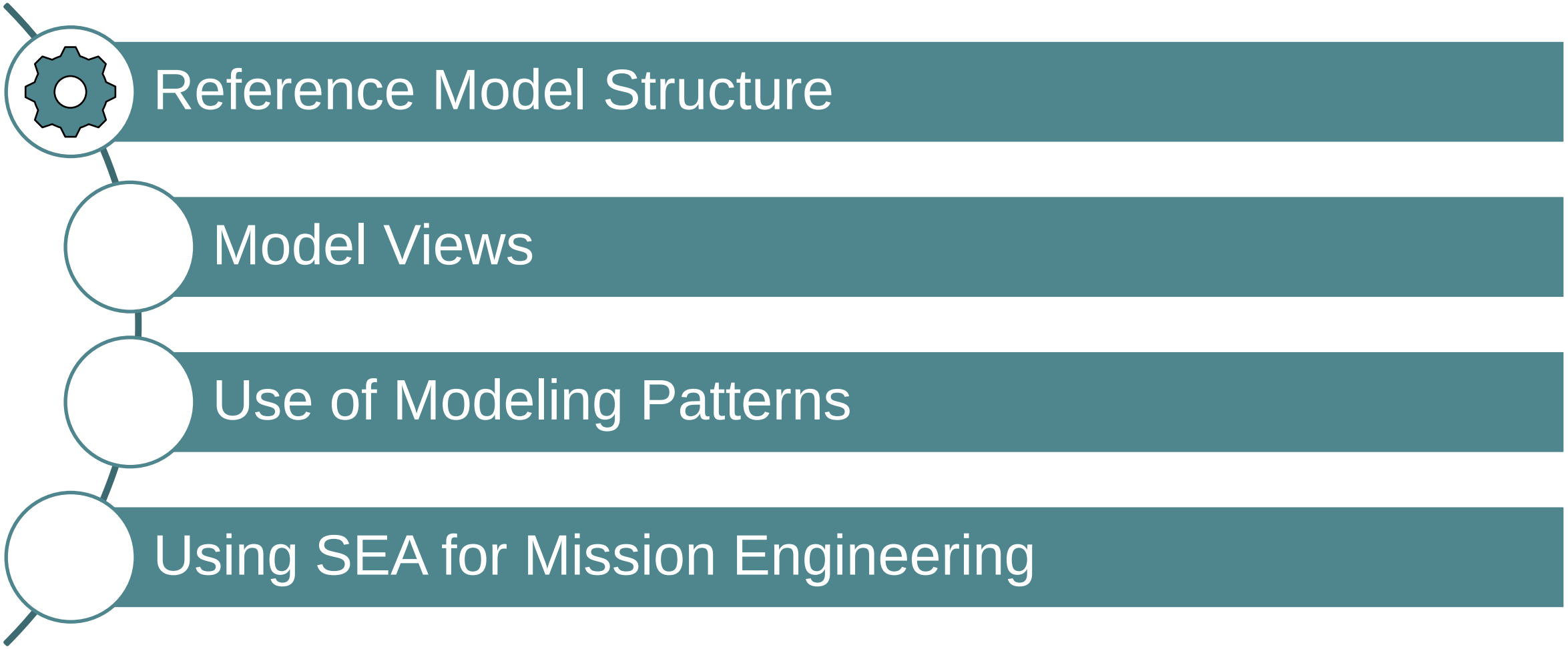
Edith Szarkowski

Systems Engineer with The Aerospace Corporation

Edith has over 15 years of experience across the system lifecycle within both the Command and Control (C2) and space domains. Ms. Szarkowski earned a BS and MS in Mechanical Engineering from University of Colorado at Colorado Springs and a MS in Space Systems Engineering from Johns Hopkins University. Her passion is to ensure her efforts in the domains are documented so operators, system administrators, and architects performing tasks have ground truth to reference in their efforts. edith.l.szarkowski@aero.org

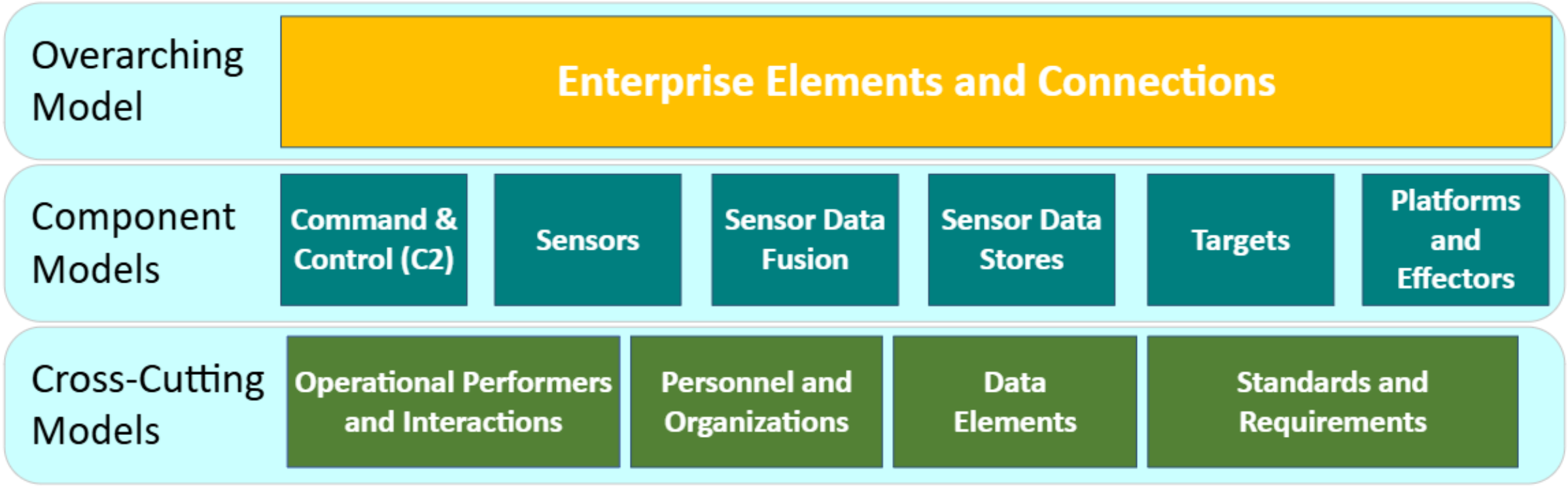
Topics

Space-domain Enterprise Architecture (SEA) Reference Model



SEA Reference Model Partitioned into Separate Model Libraries

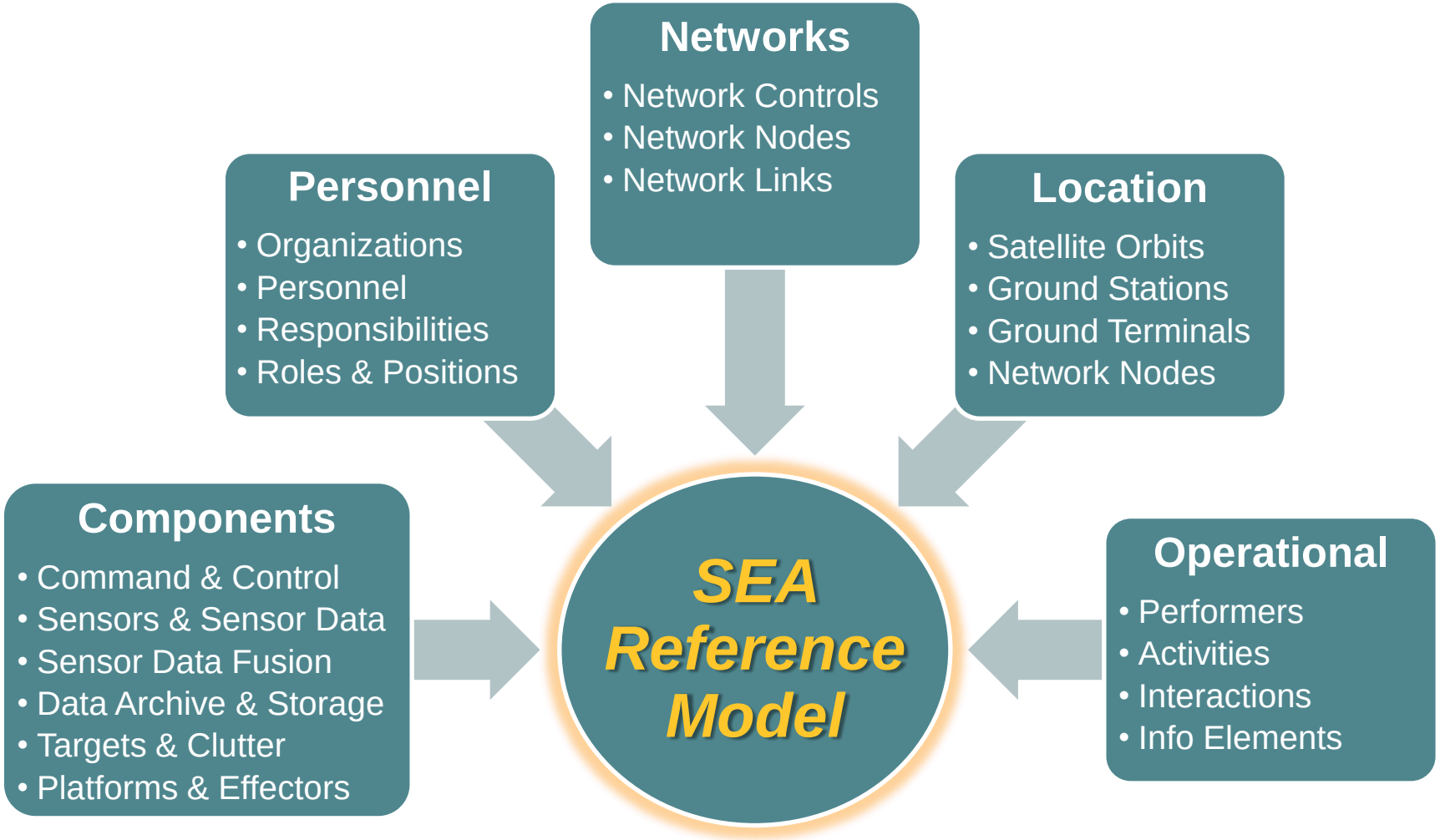
Separate models facilitate more effective model management and model development



These models span different focus areas to facilitate capturing of enterprise architecture content

What is in the Space-domain Reference Model?

A reference model can facilitate consistent data capture



Reference models enable the reuse of modeling elements that gives you a running start to your project

Why Use Separate Models?

Model partitioning approach to facilitate model management

- Large enterprise model is partitioned into separate models to help handle model complexity
- This partitioning can facilitate model reuse and modeling consistency

Advantages

- Each lower-level model is “read only” to protect data integrity and enforce standardization
- Using these standard model elements in Mission Engineering (ME) ensures consistency and compatibility between the separate models
- Enforces standard structure for capture of model data and usage in different modeling efforts
- Provides an initial starting point for the model elements and associated properties to ensure consistency of the separate models across the enterprise

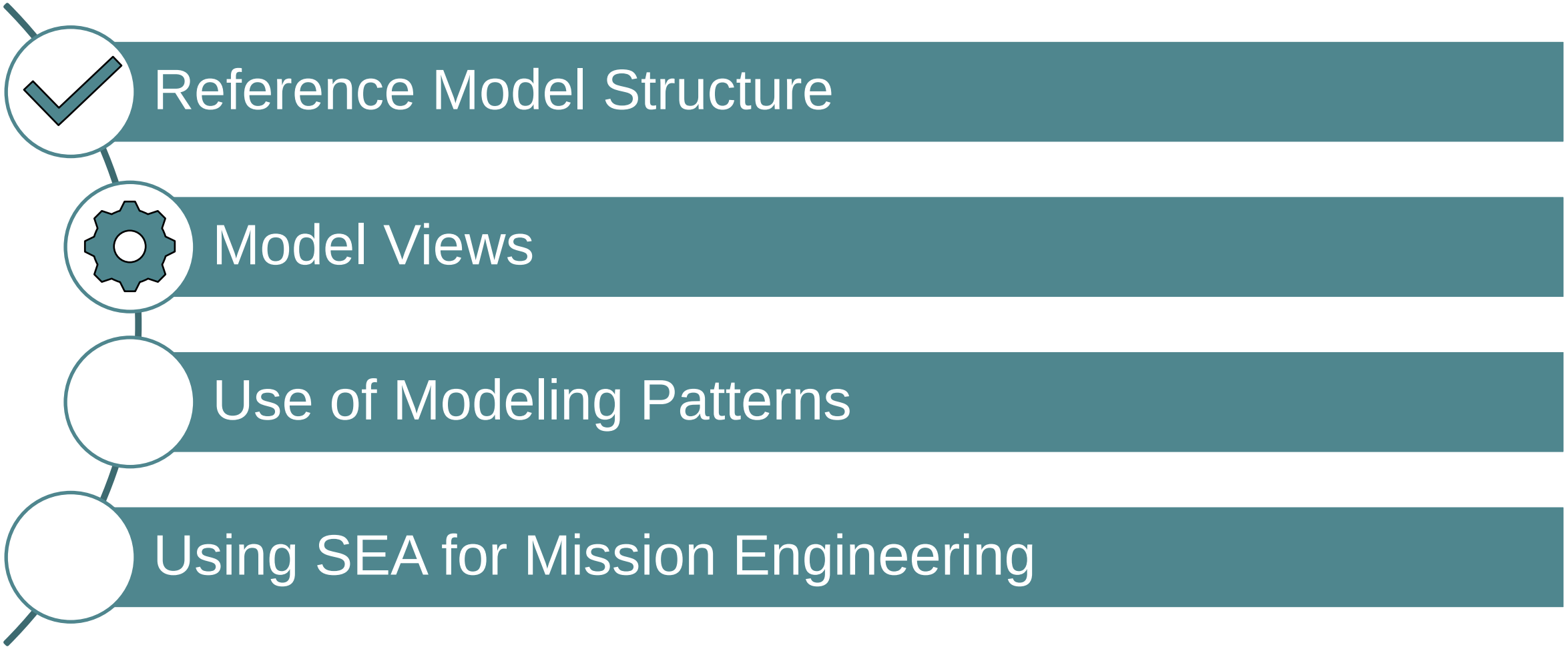
Disadvantages

- Needs careful change control and model governance to ensure orderly evolution
- Changes in model elements in one model impacts all other models that use it
- Model curation required to maintain content of models and to ensure consistency

Separate models require effort to maintain so they are consistent and enable standardized data capture to occur

Topics

Space-domain enterprise architecture (SEA) reference model



SEA Views from the Unified Architecture Framework (UAF)

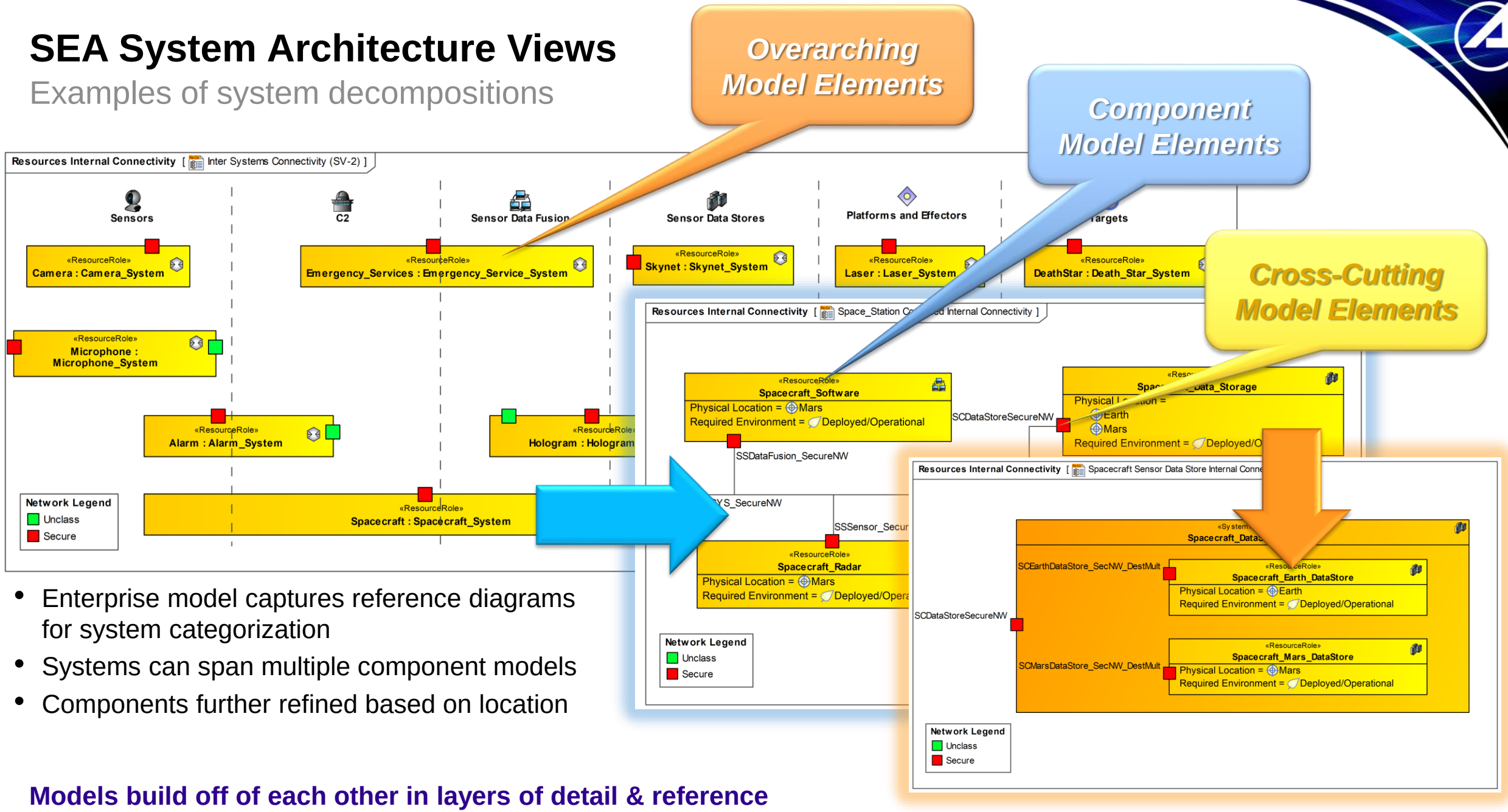
	Motivation Mv	Taxonomy Tx	Structure Sr	Connectivity Cn	Processes Pr	States St	Sequences Sq	Information If	Parameters Pm	Constraints Ct	Roadmap Rm	Traceability Tr
Architecture Management Am	Architecture Principles Am-Mv	Architecture Extensions Am-Tx	Architecture Views Am-Sr	Architectural References Am-Cn	Architecture Development Method Am-Pr	-	-	Dictionary Am-If	Architecture Parameters Am-Pm	Architecture Constraints Am-Ct	Architecture Roadmap Am-Rm	Architecture Traceability Am-Tr
Summary & Overview Sm-Ov												
Strategic St	Strategic Motivation St-Mv			Strategic Connectivity St-Cn	Strategic Processes St-Pr		-	Strategic Information St-If			Strategic Roadmaps: Deployment, Phasing St-Rm-D, -P	Strategic Traceability St-Tr
Operational Op	Require-ments Rq-Mv				Operational Processes Op-Pr	Operational States Op-St	Operational Sequences Op-Sq				-	
Services Sv		Services Taxonomy Sv-Tx	Services Structure Sv-Sr	Services Connectivity Sv-Cn	Services Processes Sv-Pr	Services States Sv-St	Services Sequences Sv-Sq	Operational Information Model Op-If	Environment En-Pm	Services Constraints Sv-Ct	Services Roadmap Sv-Rm	Services Traceability Sv-Tr
Personnel Ps				Personnel Connectivity Ps-Cn	Personnel Processes Ps-Pr	Personnel States Ps-St	Personnel Sequences Ps-Sq				Availability, Evolution, Forecast PS-Rm-A,-E,-F	
Resources Rs					Resources Processes Rs-Pr	Resources States Rs-St	Resources Sequences Rs-Sq		Rk-Pm		Resources Roadmaps: Evolution, Forecast Rs-Rm-E, -F	
Security Sc	Security Controls Sc-Mv	Security Taxonomy Sc-Tx	Security Structure Sc-Sr	Security Connectivity Sc-Cn	Security Processes Sc-Pr	-	-			Security Constraints Sc-Ct	-	Security Traceability Sc-Tr
Projects Pj	-			Projects Connectivity Pj-Cn	Projects Processes Pj-Pr	-	-	-		-	Projects Roadmap Pj-Rm	Projects Traceability Pj-Tr
Standards Sd	-			-	-	-	-	-		-	Standards Roadmap Sd-Rm	Standards Traceability Sd-Tr
Actual Resources Ar	-	-	Actual Resources Structure Ar-Sr	Actual Resources Connectivity Ar-Cn	Simulation			-	-	Parametric Execution/ Evaluation	-	-

Model Layers

- *Overarching*
- *Component*
- *Cross-Cutting*

SEA System Architecture Views

Examples of system decompositions

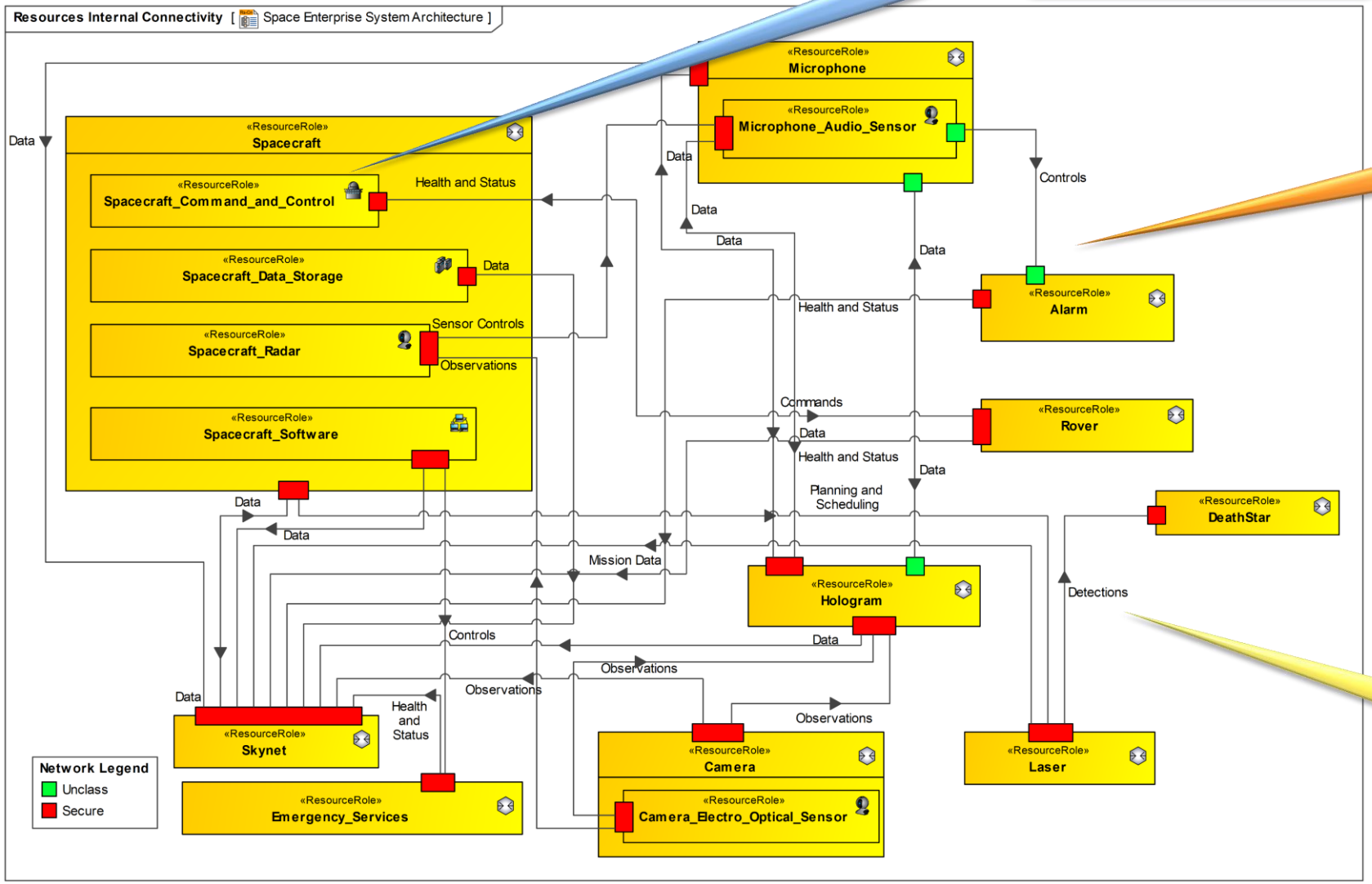


- Enterprise model captures reference diagrams for system categorization
- Systems can span multiple component models
- Components further refined based on location

Models build off of each other in layers of detail & reference standardized elements when applicable (e.g., legends)

SEA System Data Flow View

Example of systems connectivity



Component Model Elements

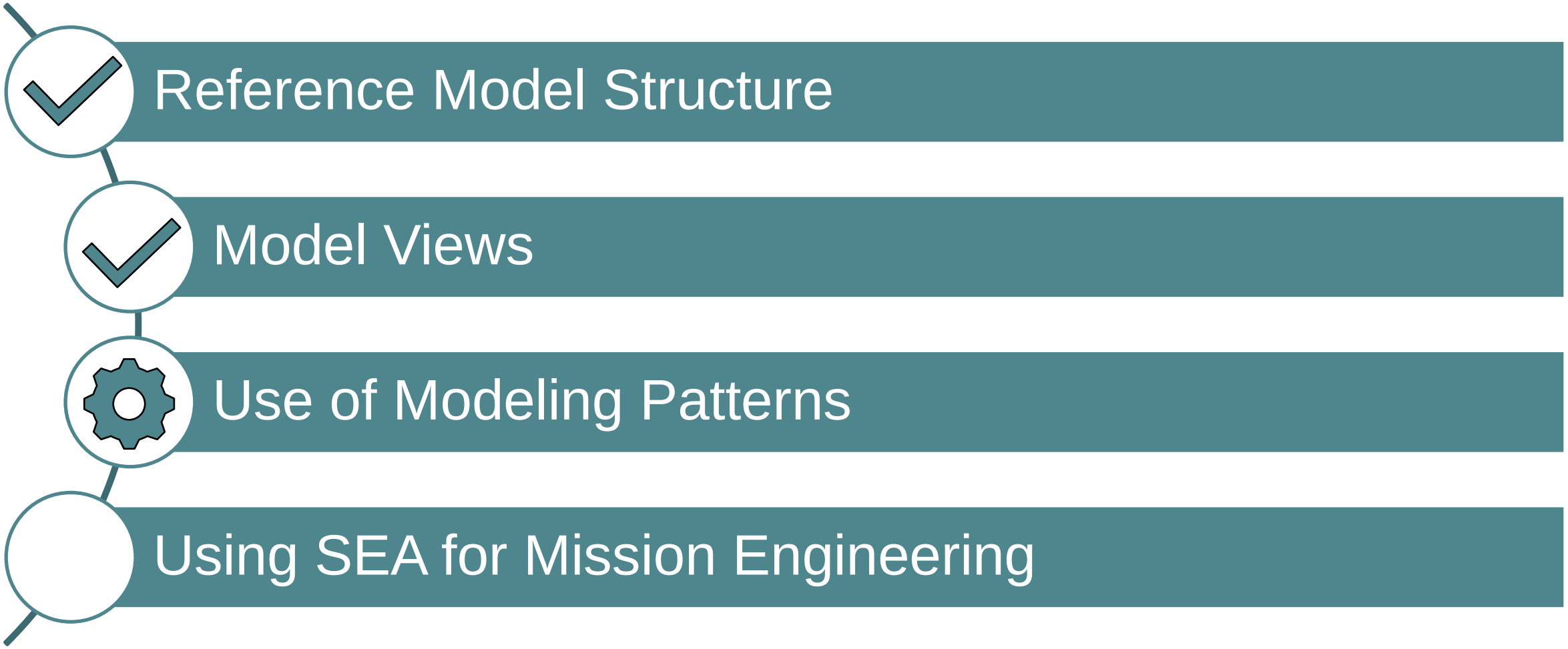
Overarching Model Elements

Cross-Cutting Model Elements

Leveraging reference information from common models allows for consistent system connectivity capture

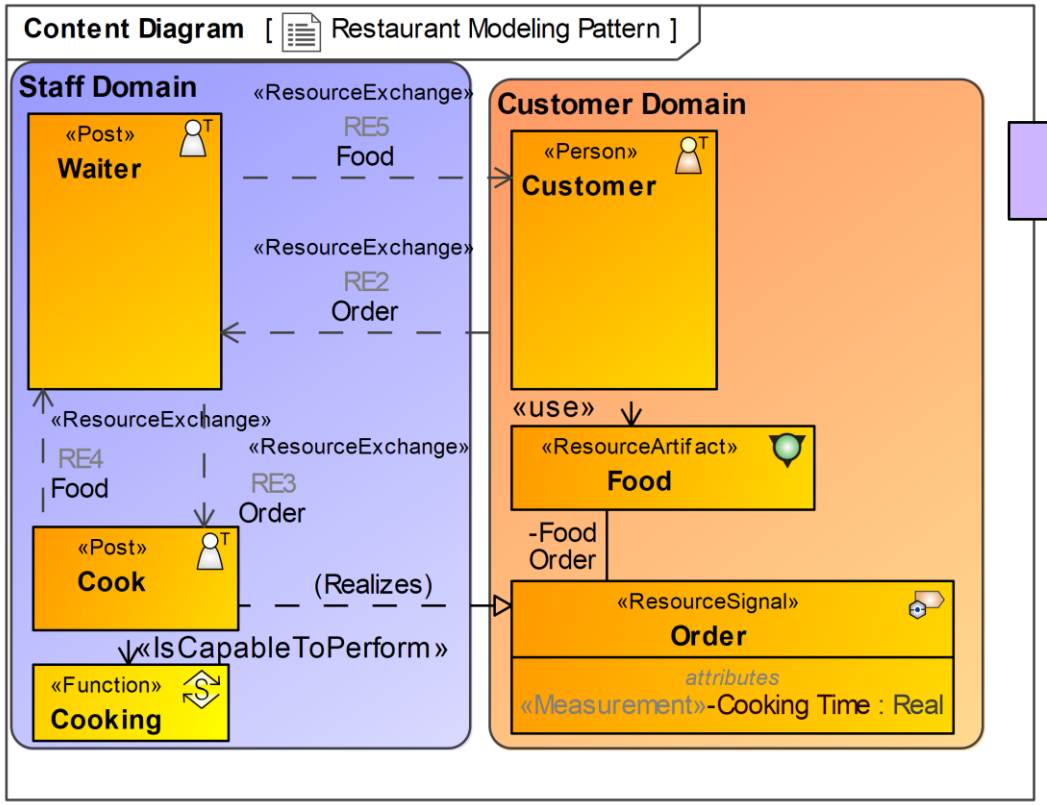
Topics

Space-domain enterprise architecture (SEA) reference model



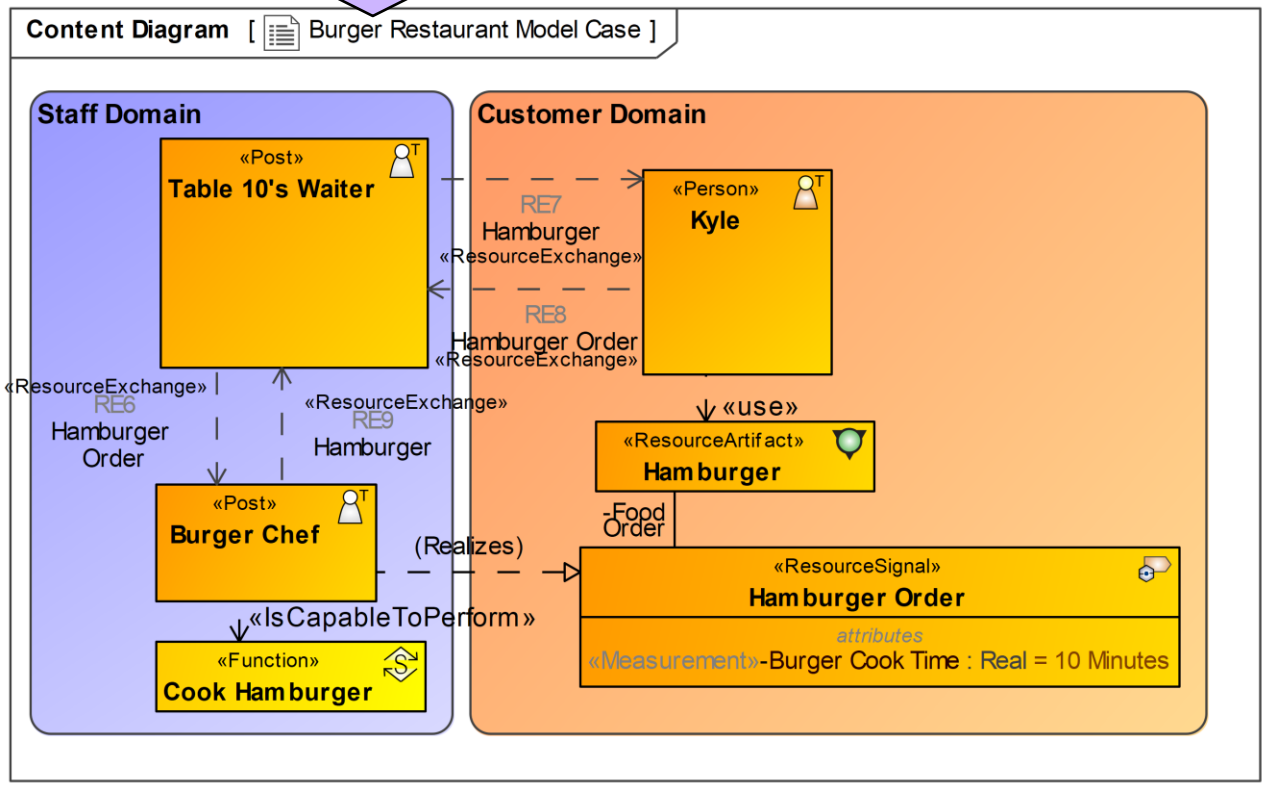
Modeling Pattern Example

Simple modeling pattern for restaurants applied to a Hamburger Restaurant Enterprise



Modeling Pattern

Modeling Pattern Applied to the Model



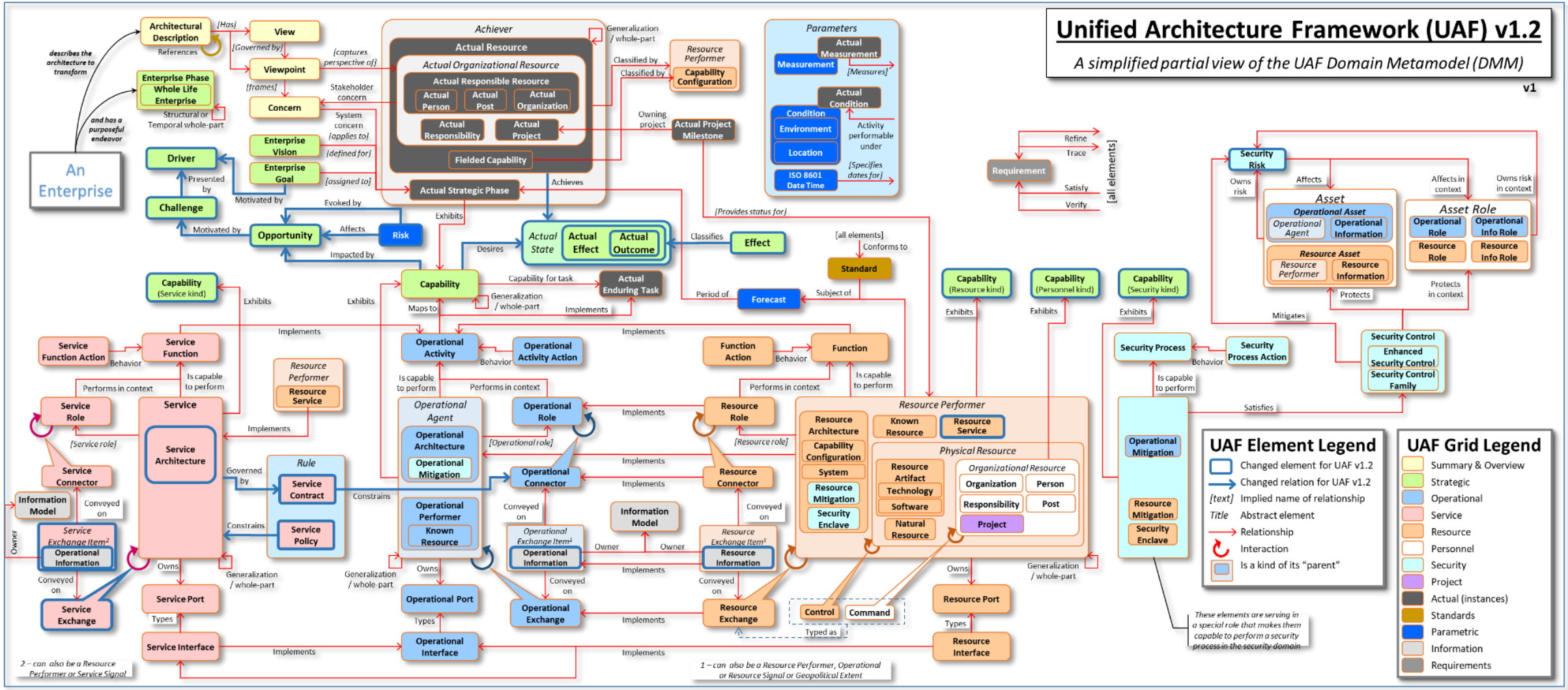
Why Use Modeling Patterns?

Commonly used in software engineering – a useful approach for model “engineering”

- Modeling efforts often struggle with using a modeling language consistently and appropriately for the domain of interest
- A modeling pattern defines how a modeling language is to be used for a variety of modeling projects in an organization
- Modeling patterns help to ensure:
  - Consistent use of a modeling language’s elements for the subject area the modeling pattern is intended to cover
  - Consistency across different models of the same subject area
  - Appropriate use of modeling language elements to represent architectural concepts in the domain of interest

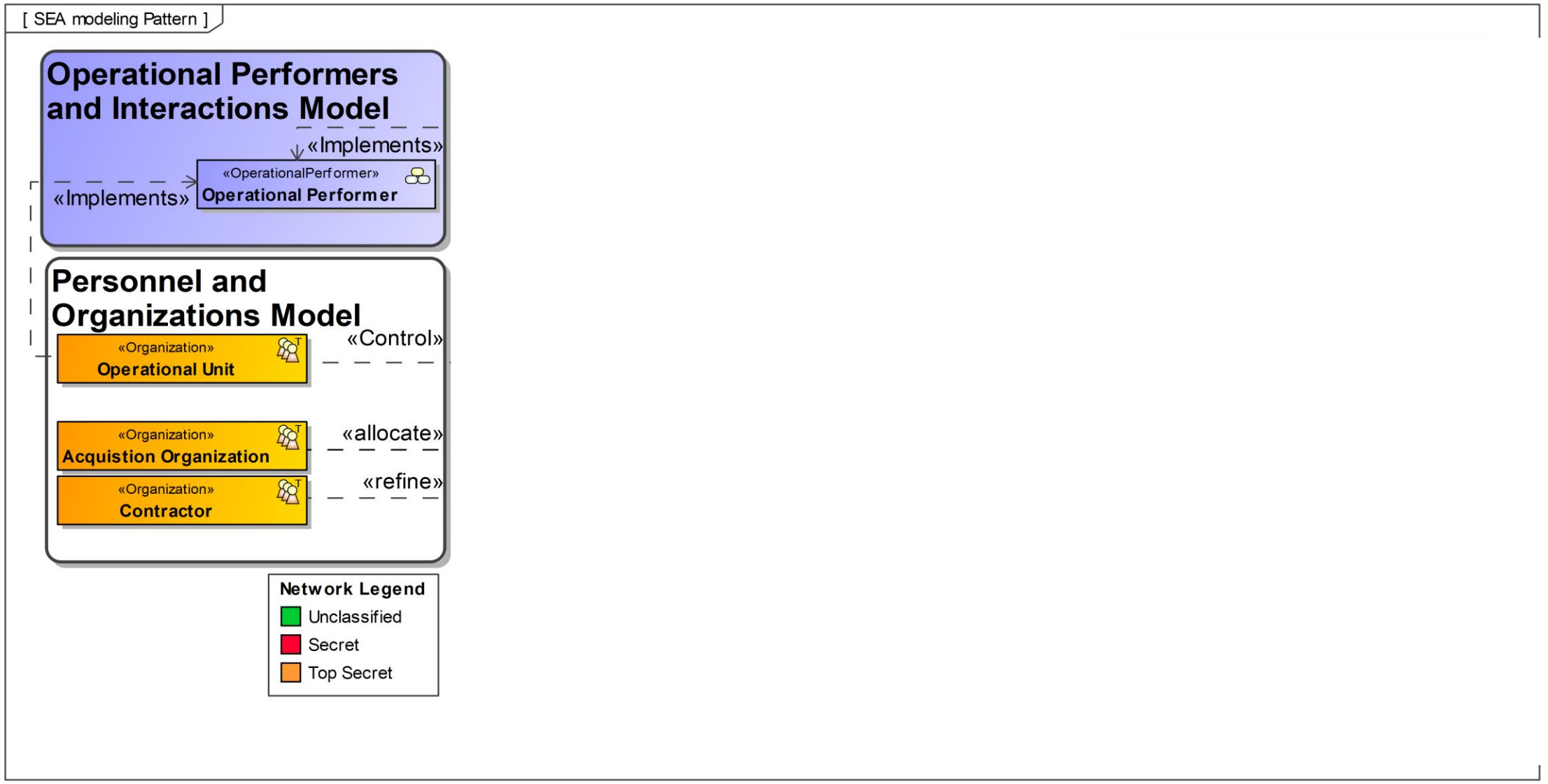
Modeling patterns help the modeler understand intended purpose and scope of the model

Enterprise Modeling Ontology



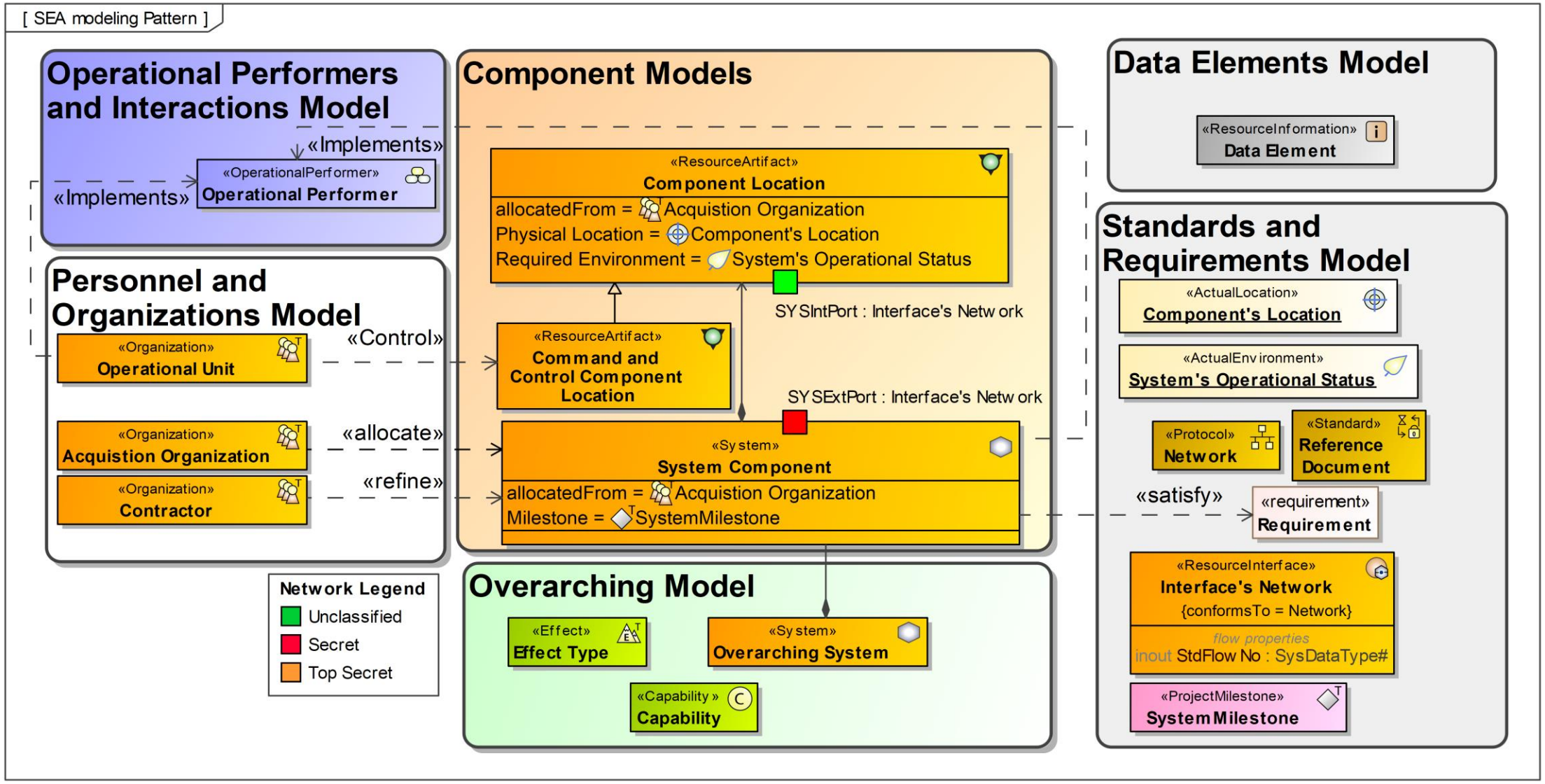
Copyright © 2022 OMG. All rights reserved.

Modeling Pattern for the Space Enterprise Architecture (SEA)



A reusable pattern for using the UAF ontology to build a space reference model

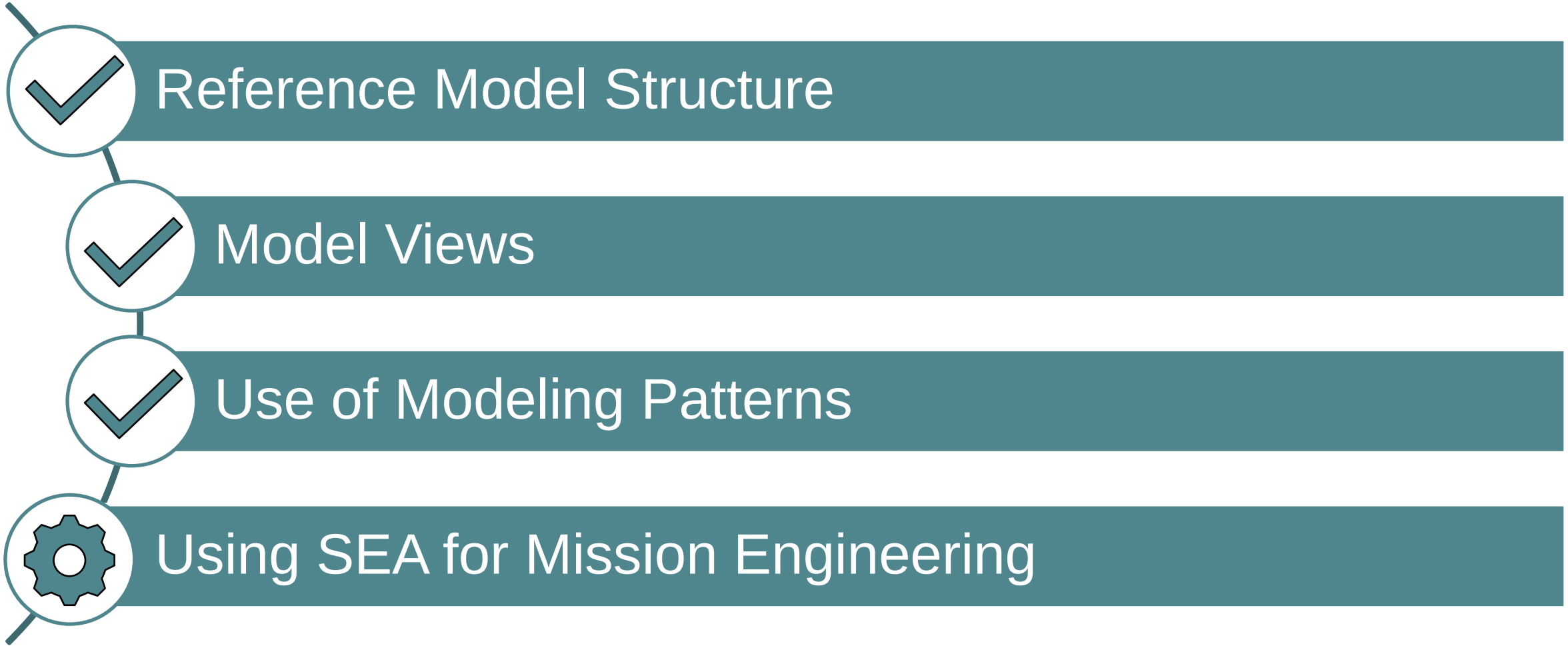
Modeling Pattern for the Space Enterprise Architecture (SEA)



A reusable pattern for using the UAF ontology to build a space reference model

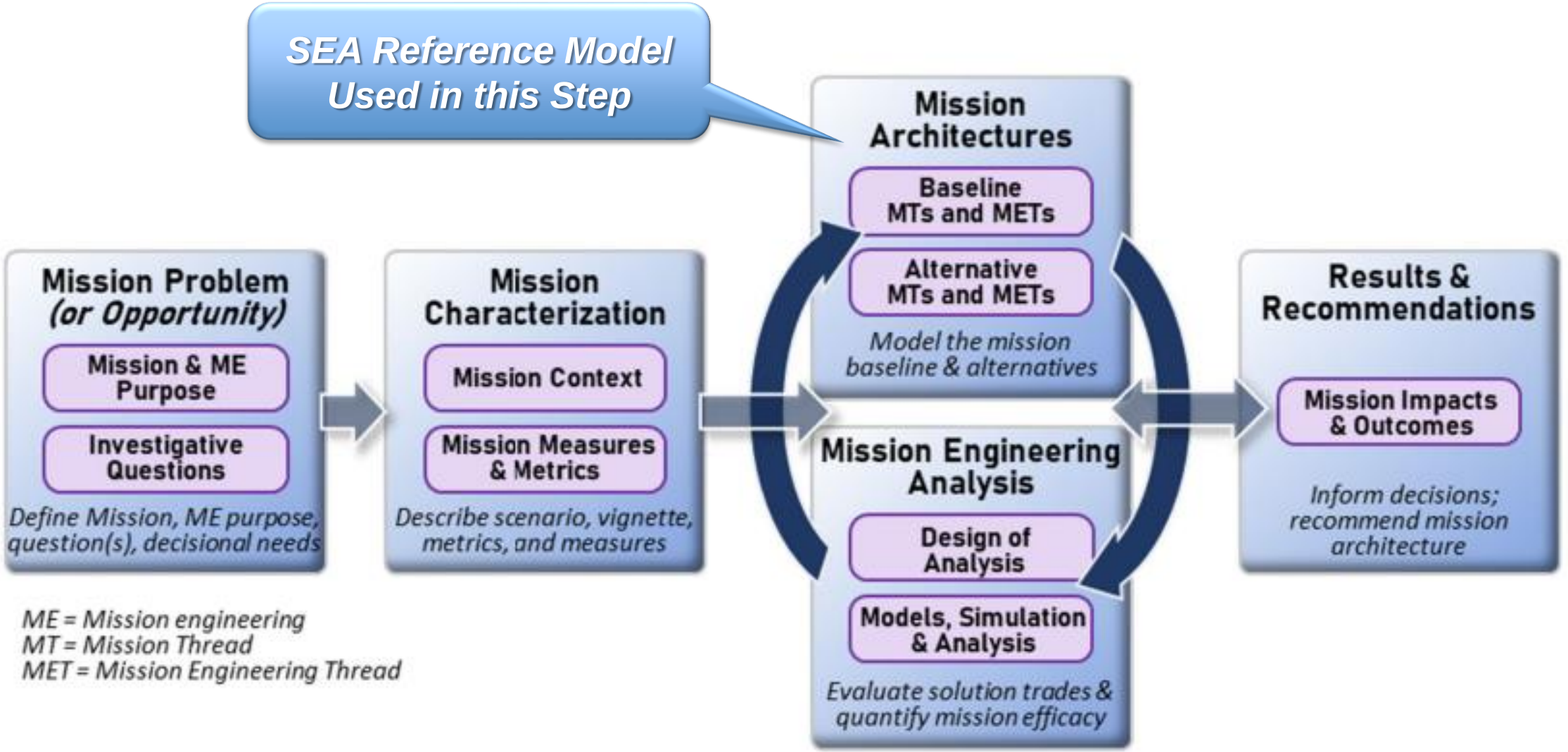
Topics

Space-domain enterprise architecture (SEA) reference model



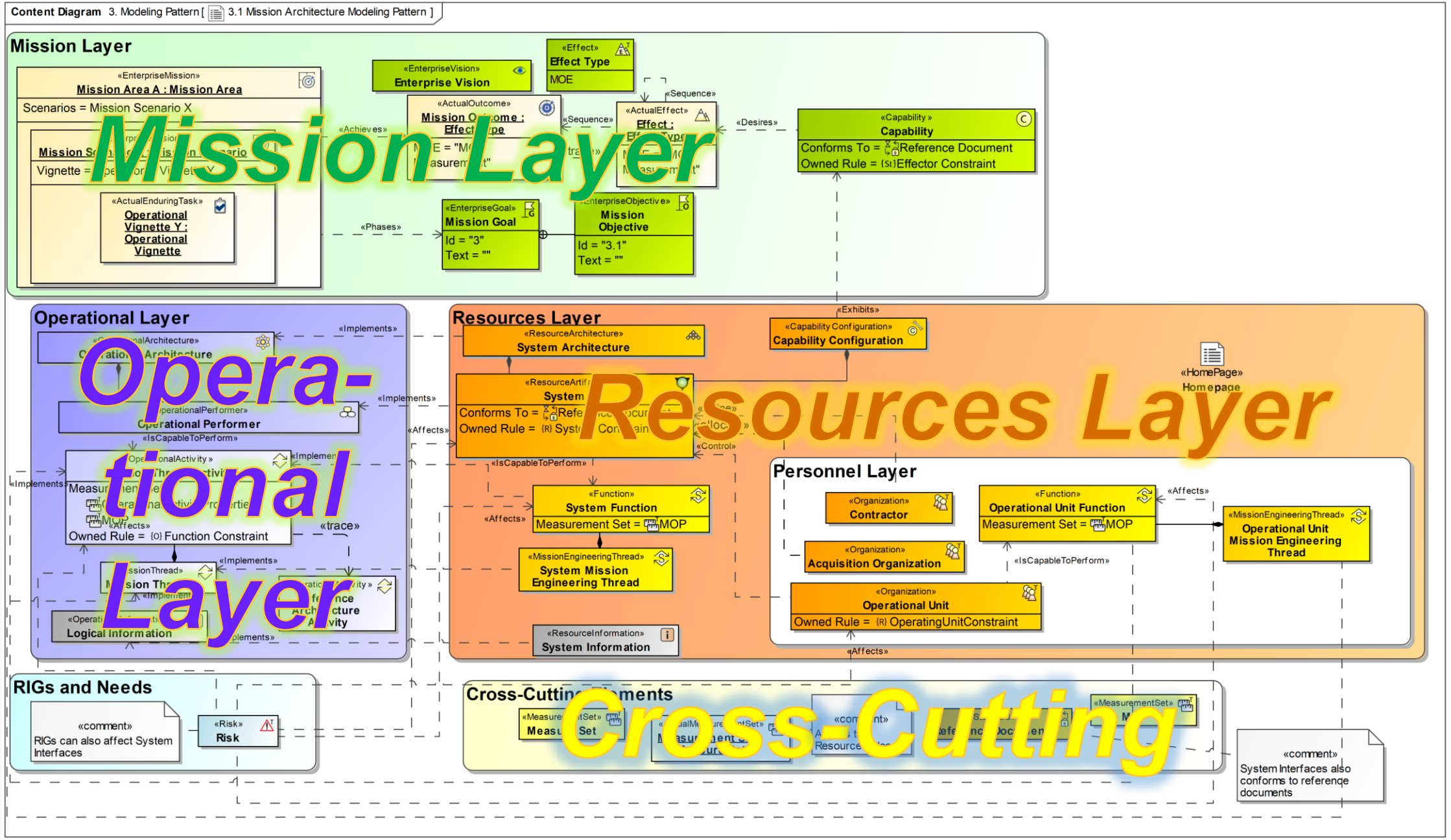
Role in the Mission Engineering (ME) Process

SEA Reference Model supports the creation of space domain mission architecture models

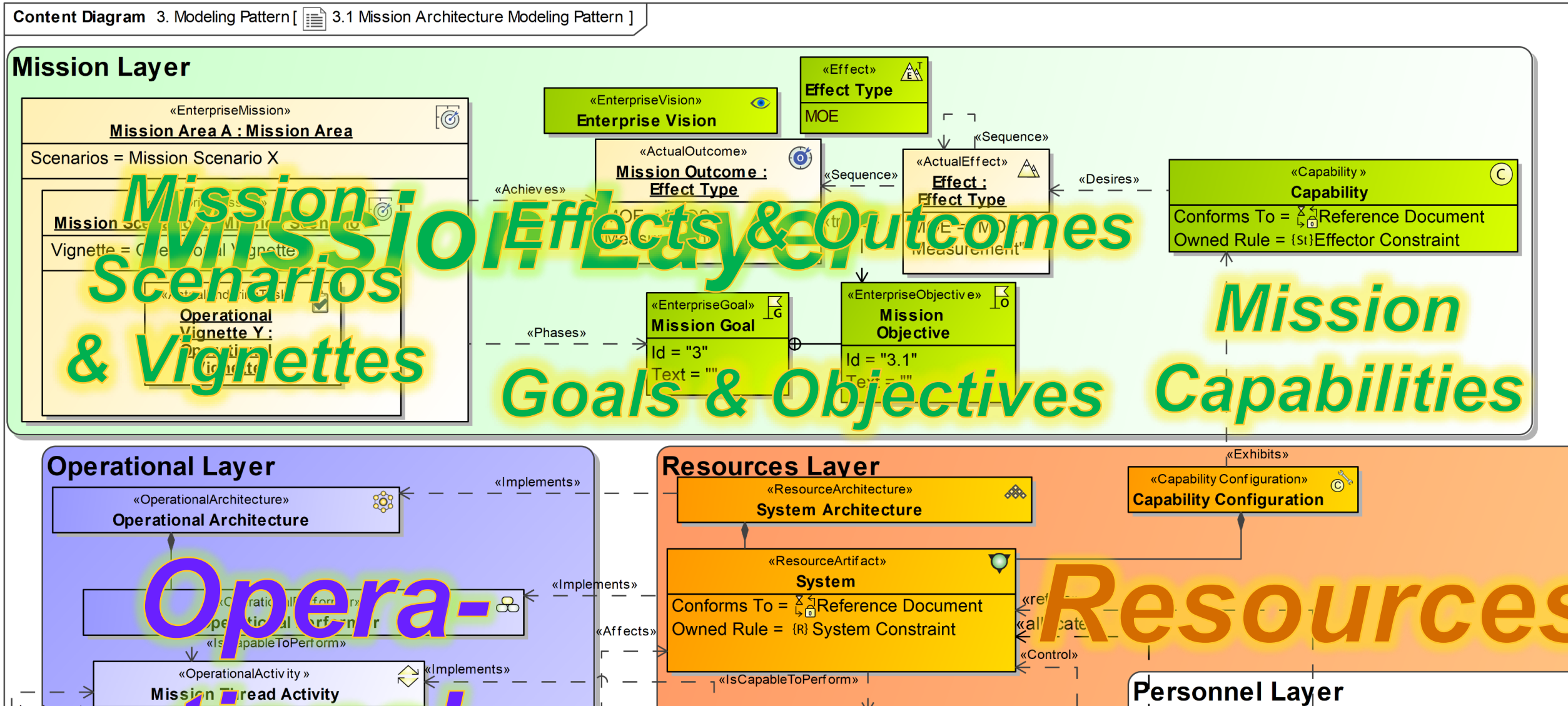


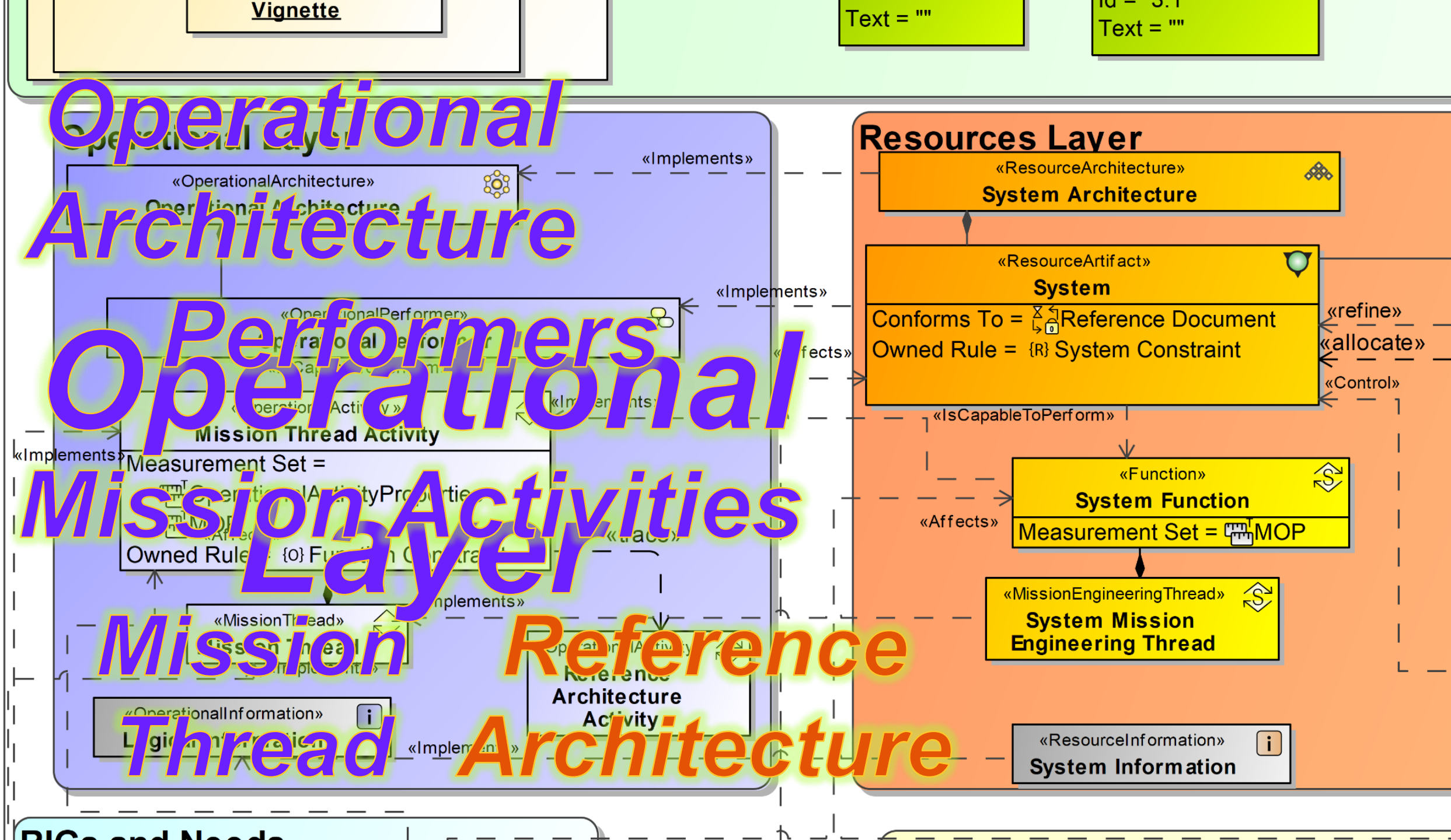
Reference model serves as a "template" for space domain mission architectures

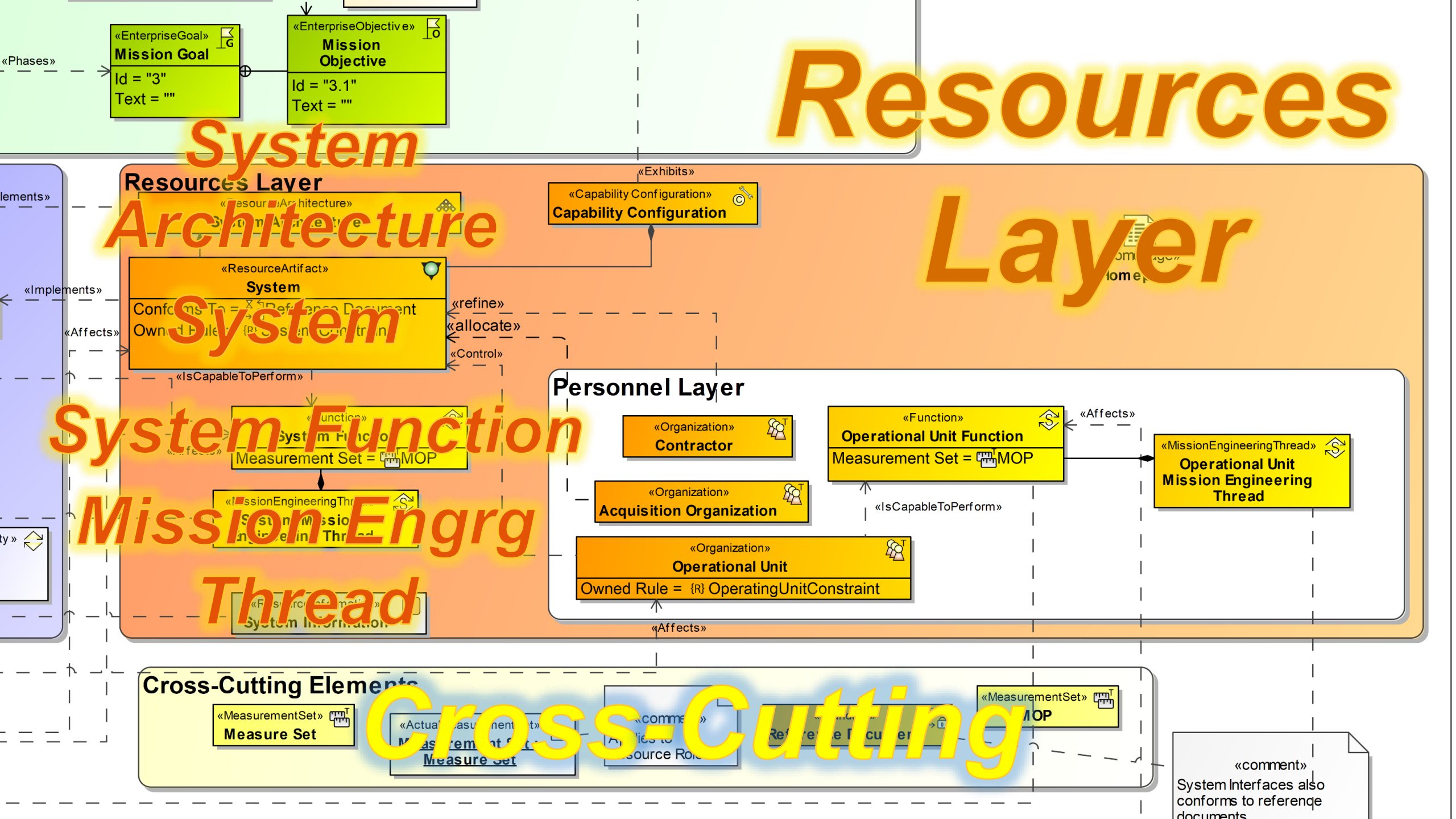
Mission Architecture Modeling Pattern

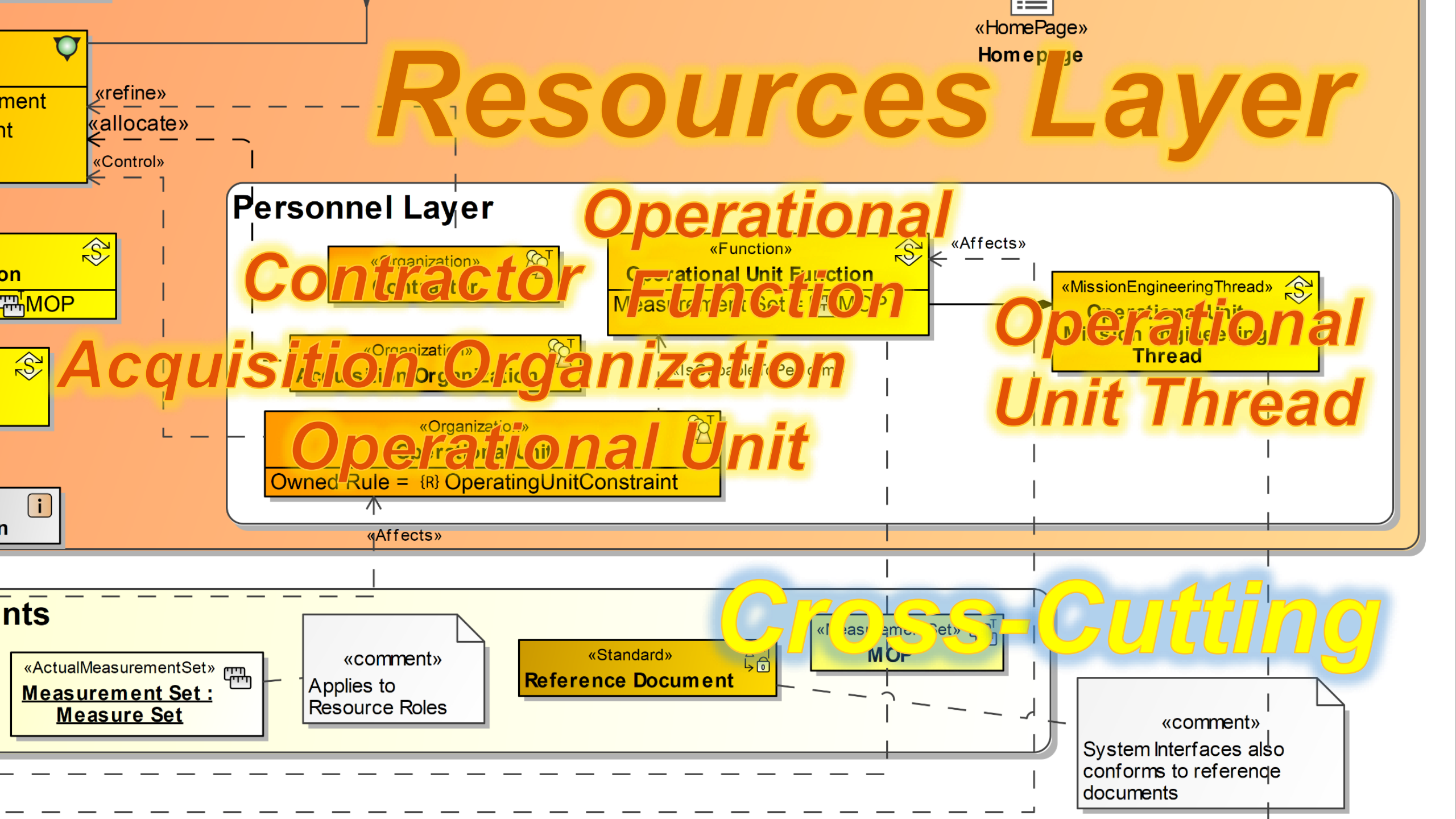


Mission Architecture Modeling Pattern



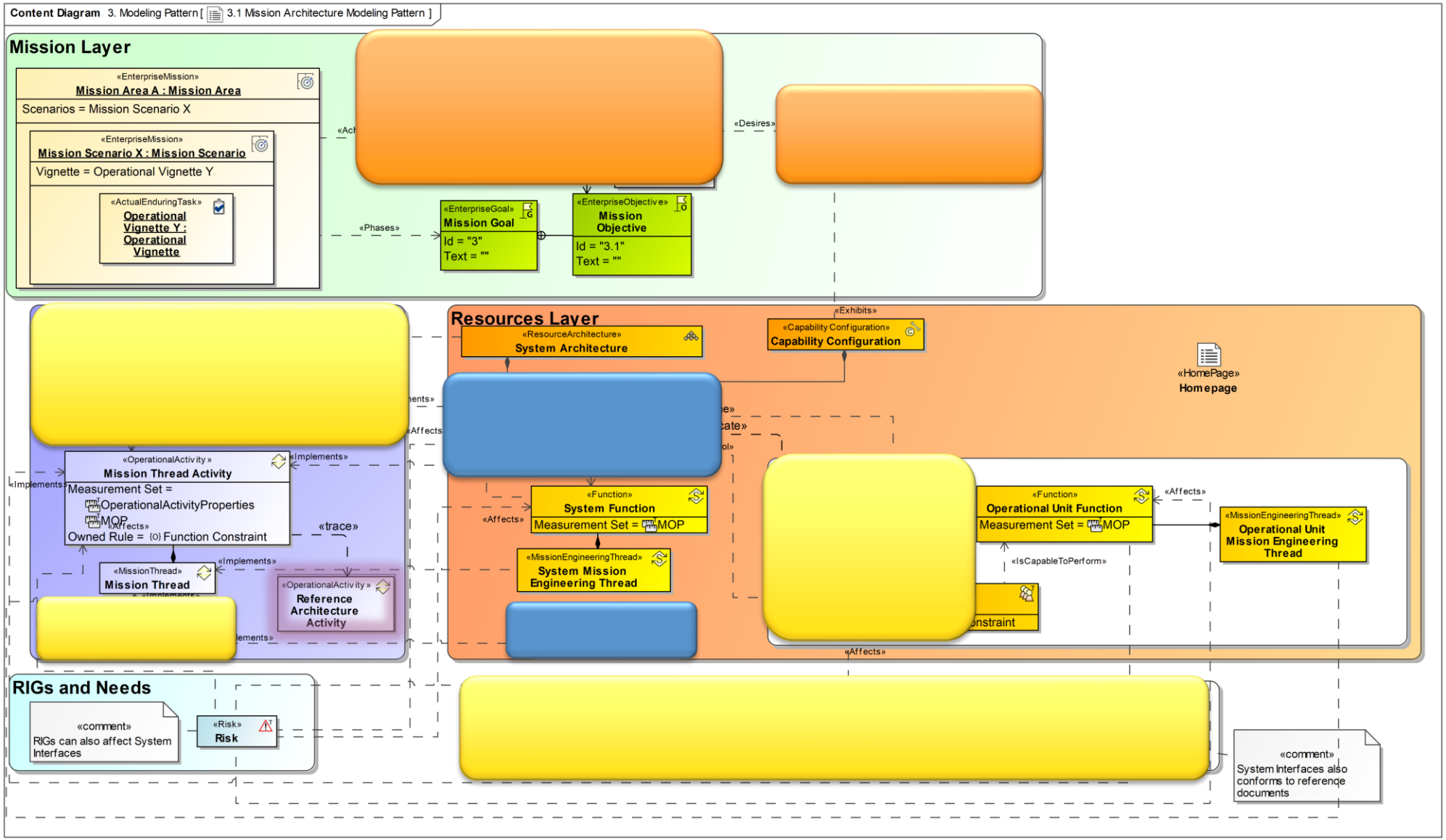

Vignette
Text = ""
Text = ""
Operational Architecture
Operational Performers
Operational Mission Activities Layer
Mission Thread
Reference Architecture
Operational Layer
«Implements»
«OperationalArchitecture»
Operational Architecture
«Implements»
«OperationalPerformer»
«Implements»
«Implements»
Mission Thread Activity
Measurement Set =
Owned Rule = {O}
«MissionThread»
«OperationalInformation»
«Implements»
Reference Architecture Activity
Resources Layer
«ResourceArchitecture»
System Architecture
«ResourceArtifact»
System
Conforms To = Reference Document
Owned Rule = {R} System Constraint
«refine»
«allocate»
«Control»
«IsCapableToPerform»
«Function»
System Function
«Affects»
Measurement Set = MOP
«MissionEngineeringThread»
System Mission Engineering Thread
«ResourceInformation»
System Information


«EnterpriseGoal»
Mission Goal
Id = "3"
Text = ""
«EnterpriseObjective»
Mission Objective
Id = "3.1"
Text = ""
«Phases»
Resources
System Architecture
Resources Layer
«Exhibits»
«Capability Configuration»
Capability Configuration
«ResourceArtifact»
System
«Implements»
«Affects»
«refine»
«allocate»
«Control»
«IsCapableToPerform»
System Function
Measurement Set = MOP
System Mission Engrg Thread
Personnel Layer
«Organization»
Contractor
«Organization»
Acquisition Organization
«Organization»
Operational Unit
Owned Rule = {R} OperatingUnitConstraint
«Function»
Operational Unit Function
Measurement Set = MOP
«Affects»
«IsCapableToPerform»
«MissionEngineeringThread»
Operational Unit Mission Engineering Thread
«Affects»
Cross-Cutting Elements
«MeasurementSet»
Measure Set
«MeasurementSet»
MOP
«comment»
System Interfaces also conforms to reference documents


«HomePage»
Homepage
Resources Layer
«refine»
«allocate»
«Control»
Personnel Layer
Operational Function
Contractor
Acquisition Organization
Operational Unit
«Function»
Operational Unit Function
«Affects»
«MissionEngineeringThread»
Operational Unit Thread
«Organization»
«Organization»
«Organization»
Owned Rule = {R} OperatingUnitConstraint
«Affects»
Cross-Cutting
«ActualMeasurementSet»
Measurement Set : Measure Set
«comment»
Applies to Resource Roles
«Standard»
Reference Document
MOP
«comment»
System Interfaces also conforms to reference documents

Mission Architecture Modeling Pattern

What parts of a space mission architecture are covered by SEA?

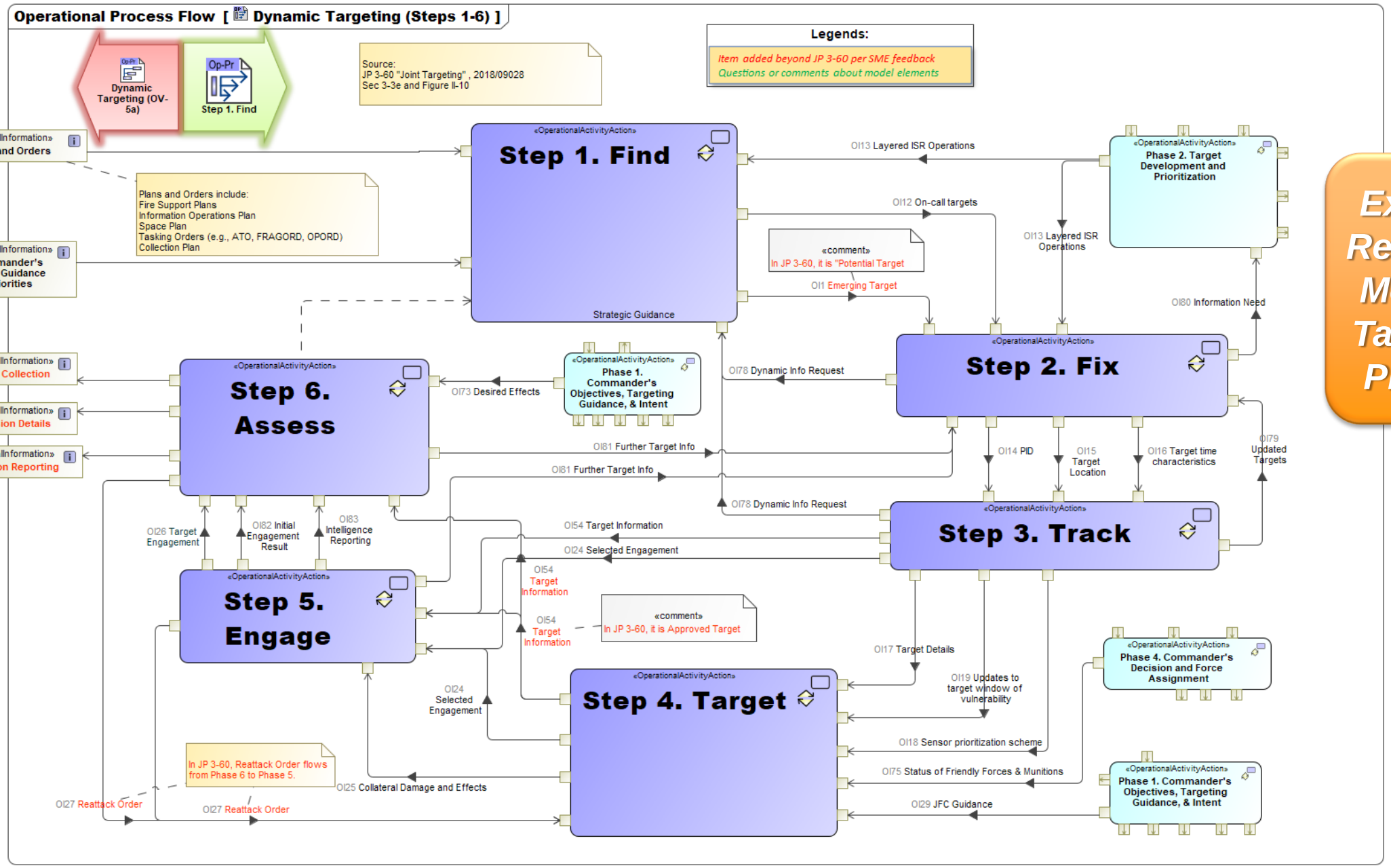


Model Layers

- Overarching
- Component
- Cross-Cutting

SEA covers significant portions of a space domain mission architecture to reduce model development time

Dynamic Targeting Process (from the JP 3-60 Joint Targeting Ref Model)



Example Reference Model of Targeting Process

Closing Thoughts

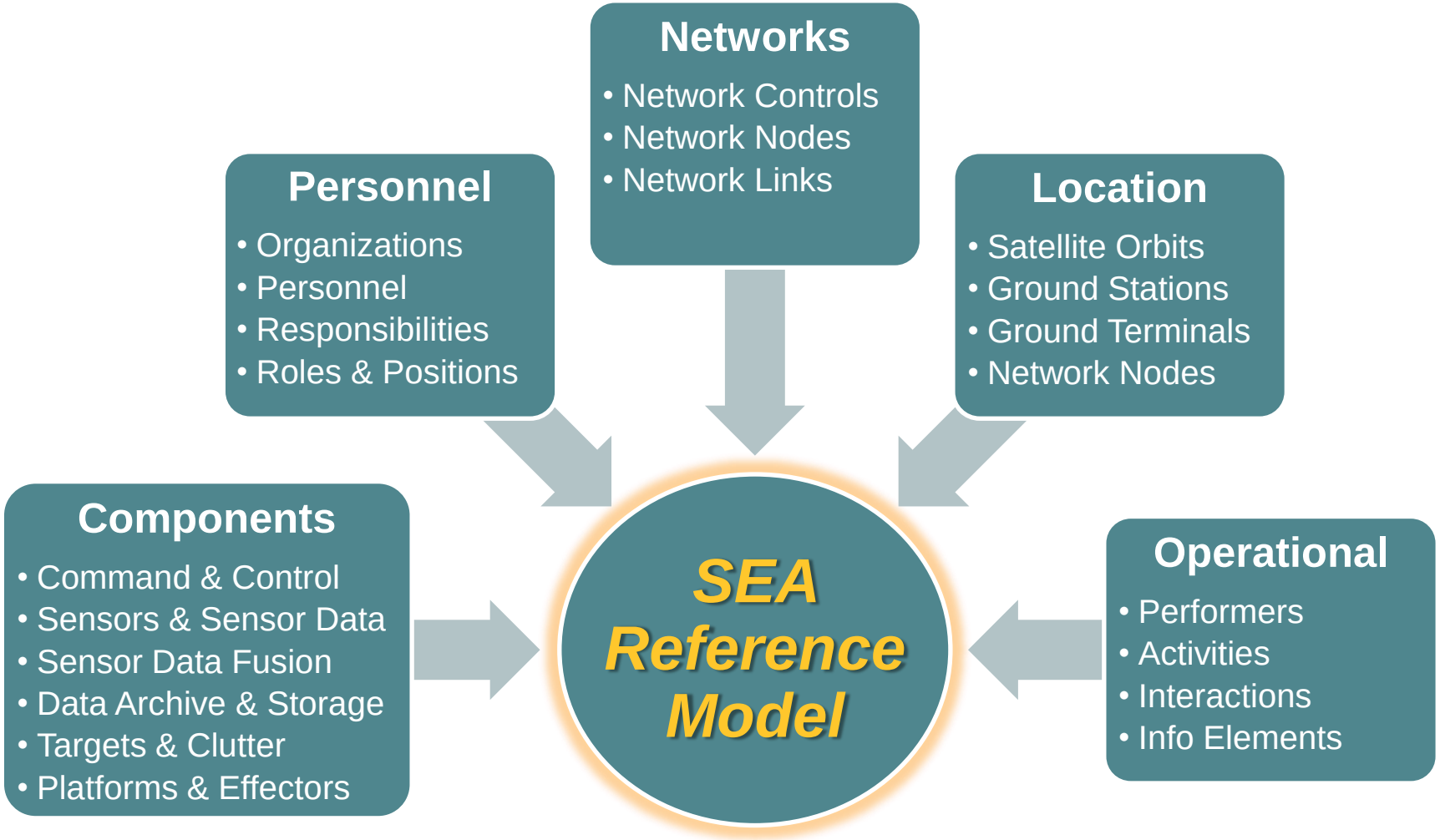
Taking advantage of MBSE and Digital Engineering methods…

- ✓ There is growing demand to produce Enterprise-level architecture models in the Space Domain to better understand the proper context for our Space Systems
- ✓ By building a Reference Architecture model, this helps to standardize representations of Space Systems and their context to reduce the time and effort of model development
- ✓ System models and Mission Engineering study efforts should align to this Reference Architecture to align our Space Systems with Military Doctrine and Concepts of Operations
- ✓ Must maintain the Reference Architecture model to ensure continued relevance and constant improvements as the industry evolves
- ✓ Will encourage harmonization and congruence between various modeling efforts

Space Domain Reference Model will save much Time & Effort in our architecting efforts

What is in the Space-domain Reference Model?

A reference model can facilitate consistent data capture



Reference models enable the reuse of modeling elements that gives you a running start to your project


Thank you

Backup

SEA Models' Definitions

- Three cross-cutting models are references
  - Personnel and Organizations: *This model contains the stakeholders, or actors, involved with the systems in the other libraries within the Space Enterprise as well as the locations where the other libraries operate. Stakeholders are Developers or Owners/Operators and directly interact with systems through the C2 systems.*
  - Standards and requirements: *This model contains the patterns/templates, classifications/networks & other standardized data pieces needed for the other libraries to reference. Common elements that are needed across multiple libraries are located in this model.*
  - Operational Performers and Interactions: *This model contains the operational performers which represent generic system types to be used to fill in mission threads. The model also defines the exchanges of information between operational performers.*
- Six component models captures a system's purpose
  - Command and Control: *This model contains the Command and Control (C2) systems within the Space Enterprise. These are systems where Actors reside and where data is both received and disseminated such that the actors can provide authority, resources, and directions to assess problems, provide solutions, and accomplish missions.*
  - Sensors: *This model contains the Sensor systems within the Space Enterprise. These are systems that provide data such as tracks related from systems categorized as Targets to Sensor Data Store systems and/or effector systems and are both ground based as well as spaced based assets.*
  - Sensor Data Fusion: *This model contains the Sensor Data Fusion systems within the Space Enterprise. These are software systems that combines the collected data from Sensor systems to provide updated information on tracks of interest for the C2 systems to take action against.*
  - Sensor Data Store: *This model contains the Sensor Data Store systems within the Space Enterprise. These are software systems that stores the collected data from Sensor systems for use by Sensors Data Fusion to consolidate.*
  - Targets: *This model contains the Target systems within the Space Enterprise. These are both space-based and ground-based assets that are the focal point for Effectors as well as what Sensor systems monitor as part of mission needs per C2 systems.*
  - Platforms and Effectors: *This model contains the platform and effector systems within the Space Enterprise. These are both space-based and ground-based assets that receive track data direction from Sensors systems against Target systems as part of mission needs per C2 systems.*
- The Enterprise Elements and Connections model stitches all other models together and captures enterprise relationships for mission engineering threads to reference

Ten models are stitched together to create a space domain enterprise reference architecture