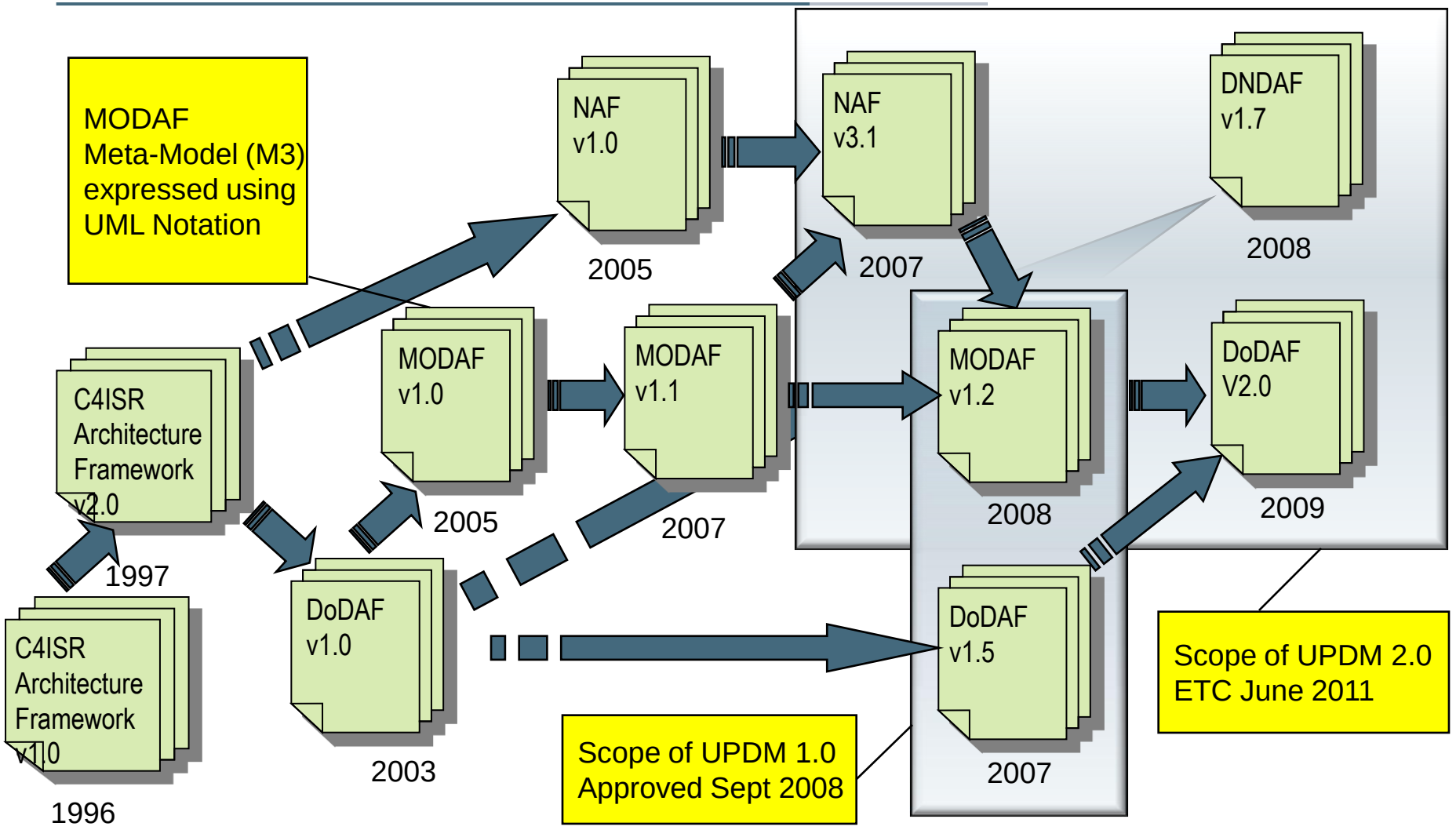
# Architecting in the Fourth Dimension
# Temporal Aspects of DoDAF 2.0

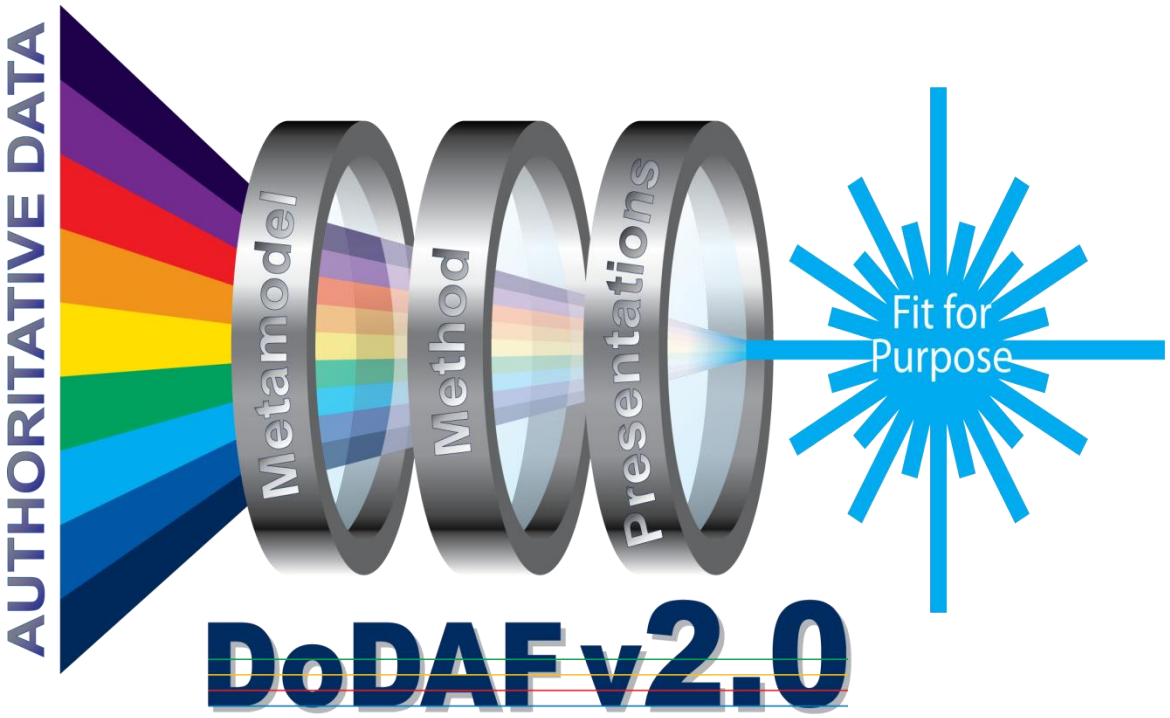
Matthew Hause – Atego, Lars-Olof Kihlstrom - Generic

# Agenda

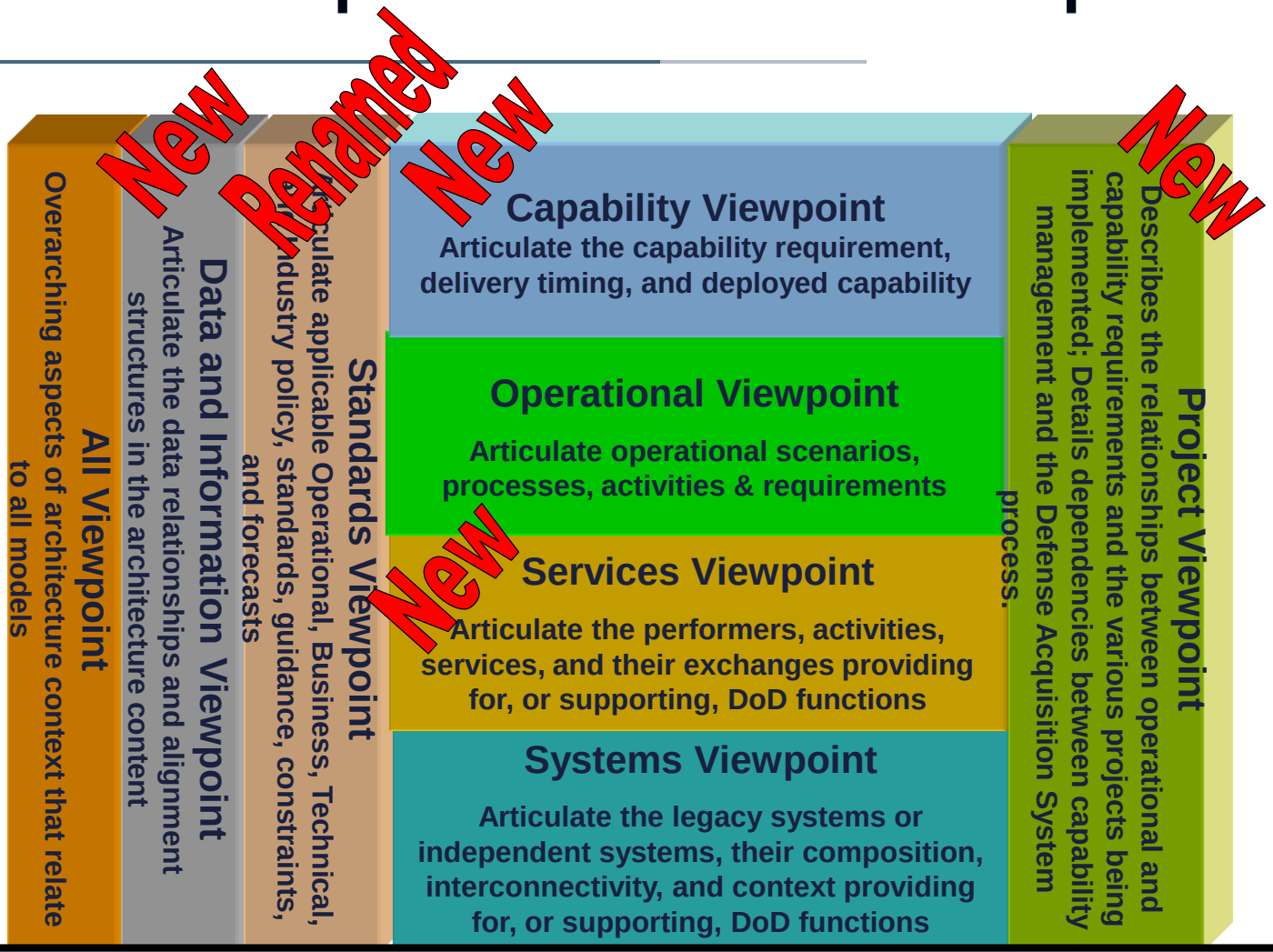
- DoDAF Background
- Ontologies
- The IDEAS Foundation
- Time and Architecture
- Examples
- Conclusion

# Historical Development of AF's.





# Select the Viewpoints That Fit-the-Purpose



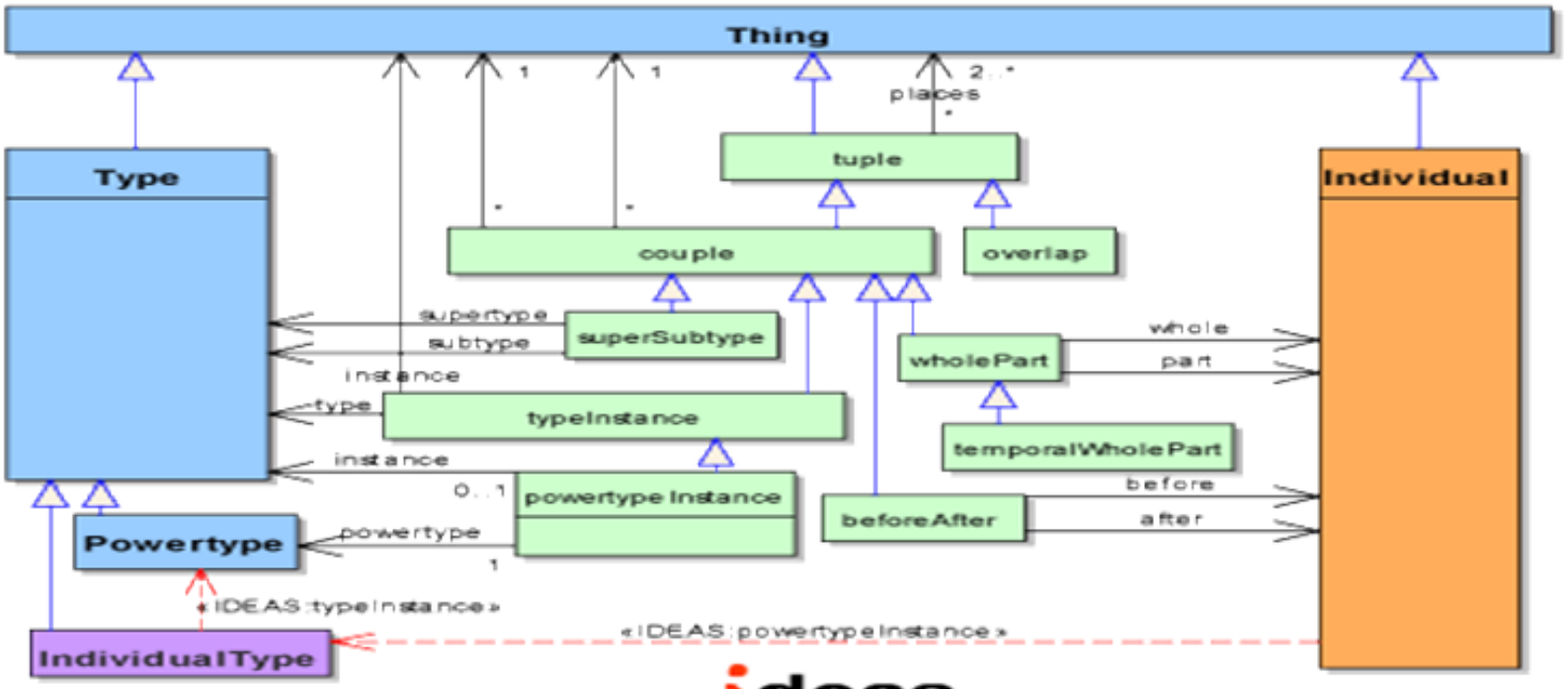

**Architecture viewpoints are composed of data that has been organized to facilitate understanding.**

# Time is of the essence

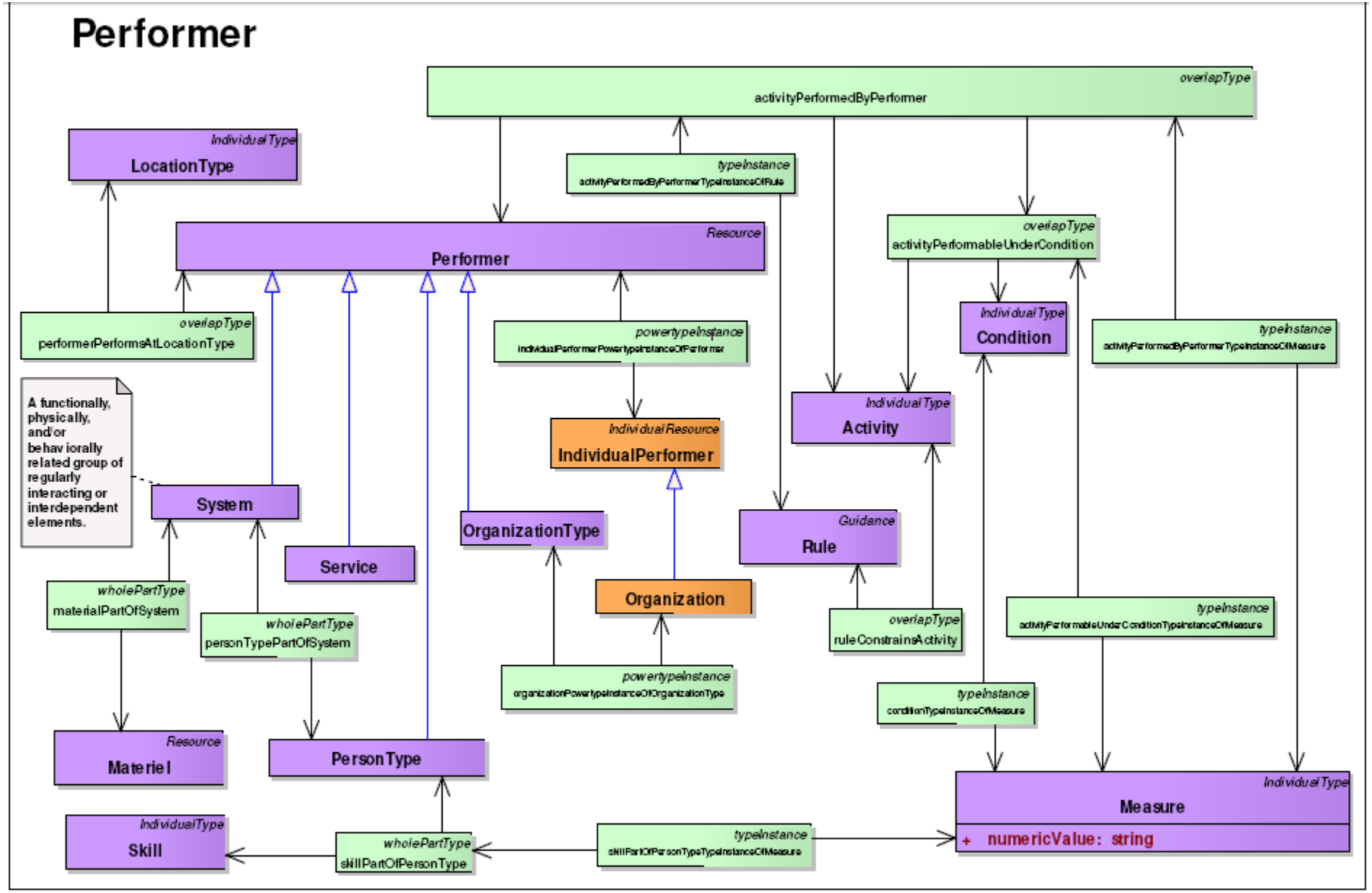
- Previously, modeling time DoDAF was less explicitly specified.
- DoDAF 2.0 provides time-based concepts in many ways
  - BeforeAfter (IDEAS foundation element)
  - BeforeAfterType (IDEAS foundation element)
  - Desired Effect (DM2 element)
  - TemporalWholePart (IDEAS foundation element)
  - TemporalWholePartType (IDEAS foundation element)
  - Work Streams
  - Project activity sequence
  - State modeling
  - Etc.

# IDEAS - Top-Level Foundation

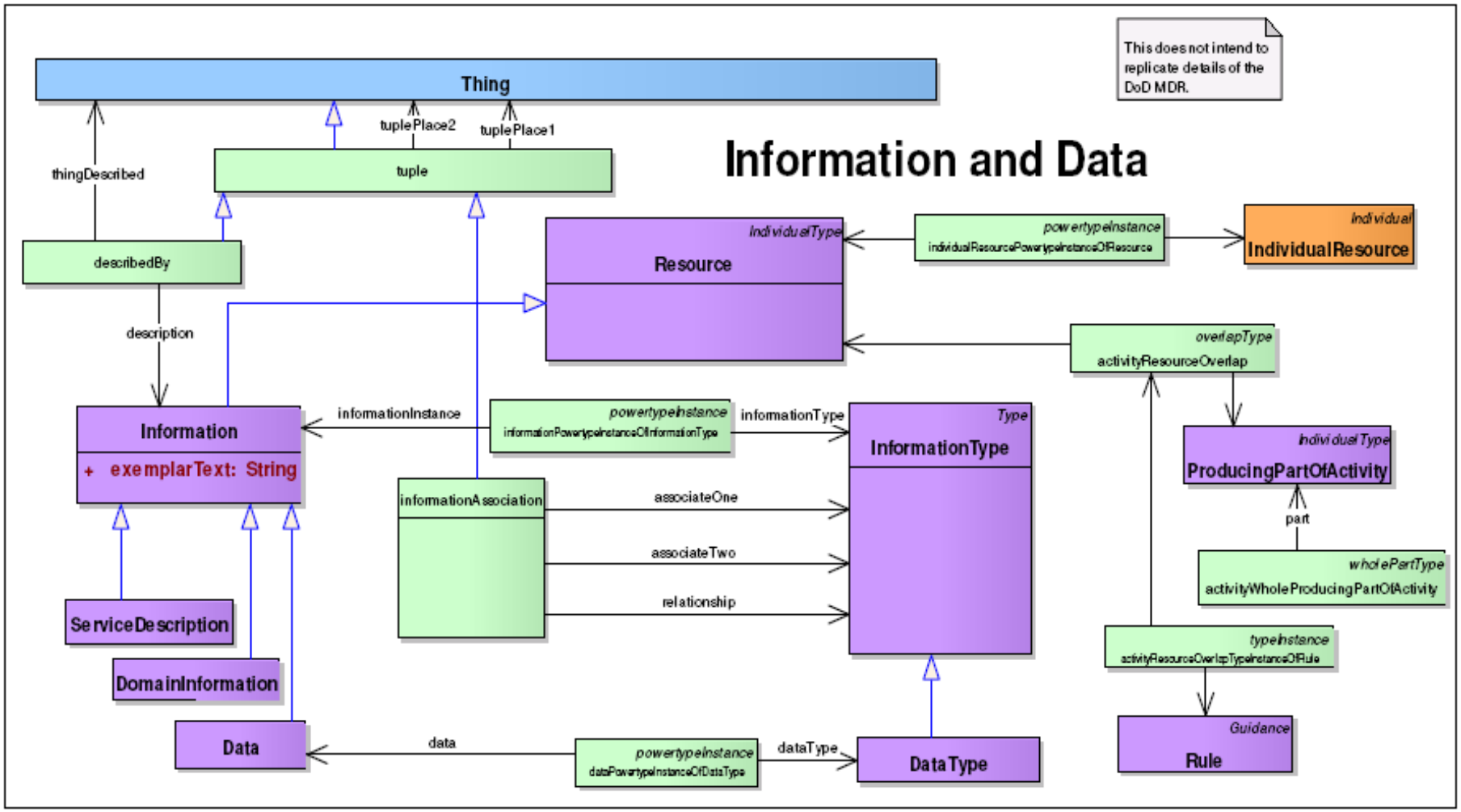
- Developed by an international group of computer scientists, engineers, mathematicians, and philosophers under defense sponsorship.
- See http://www.ideasgroup.org or http://en.wikipedia.org/wiki/IDEAS_Group



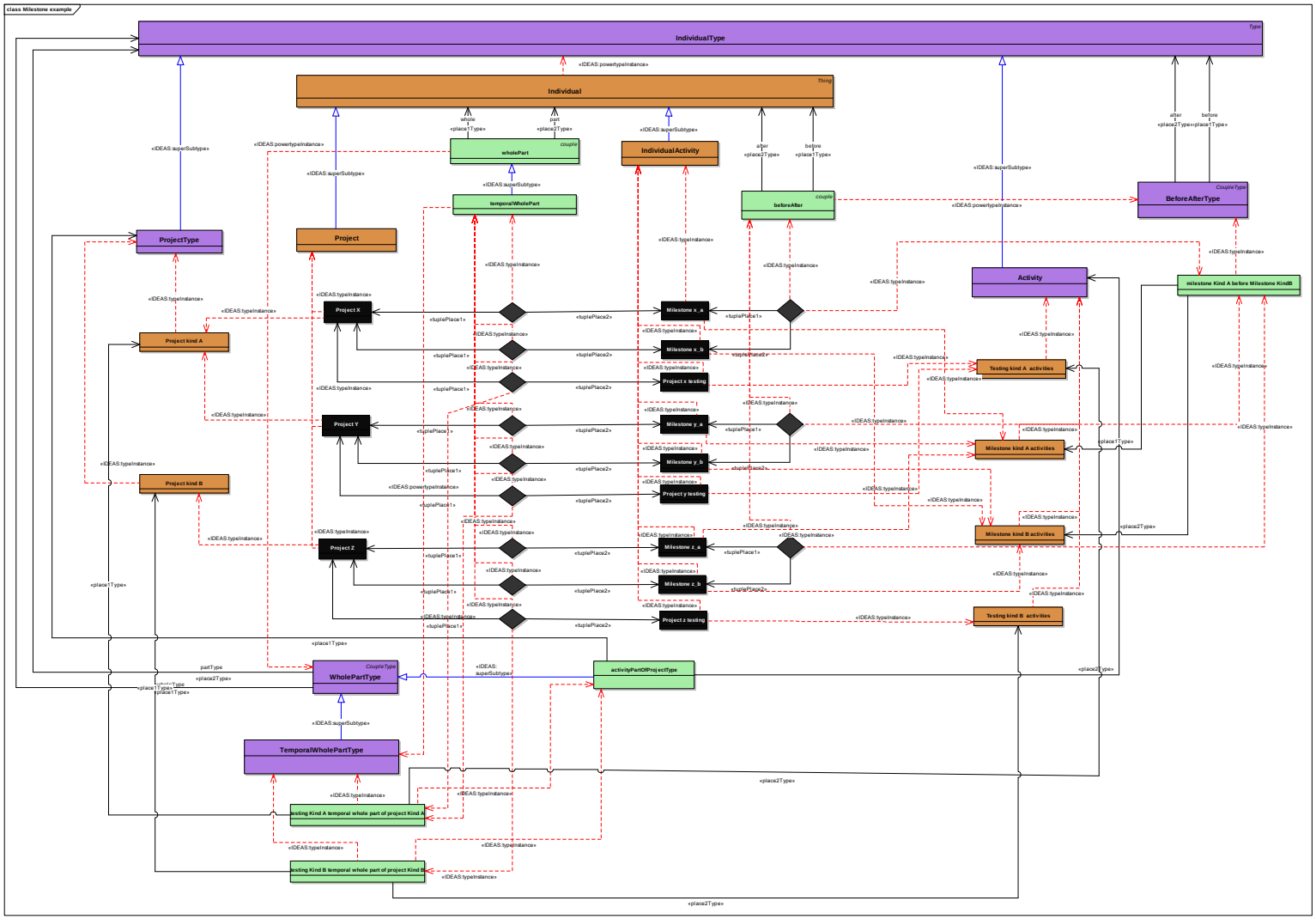
# Performer



# Information and Data



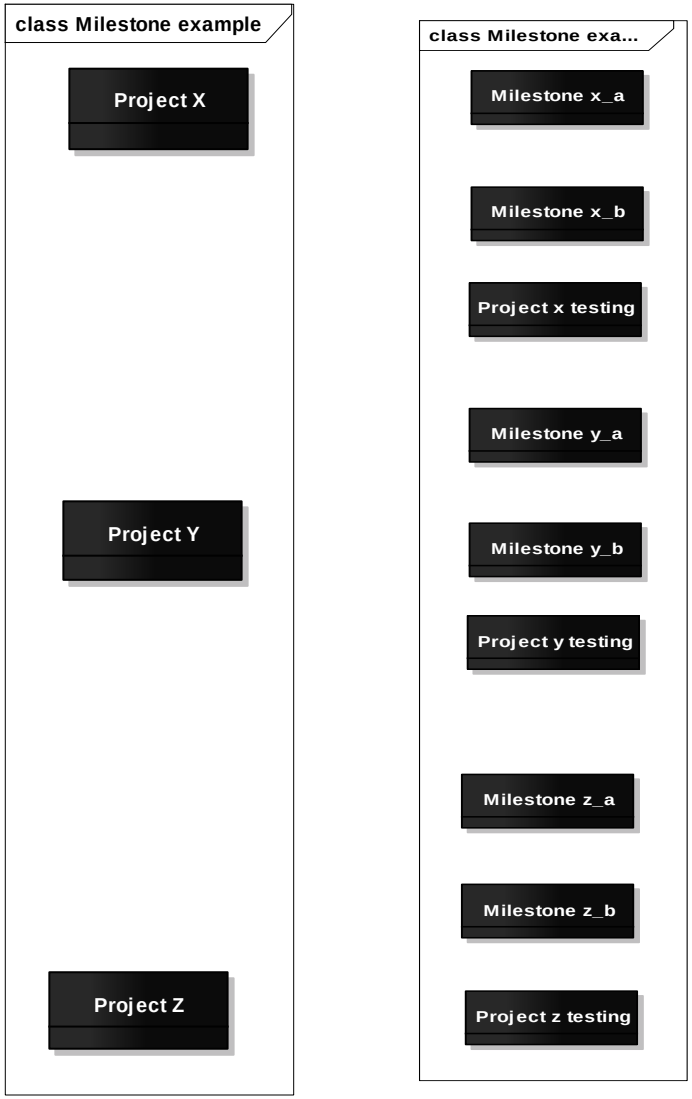
# An example of temporal element usage within DM2



# WHAT!!!

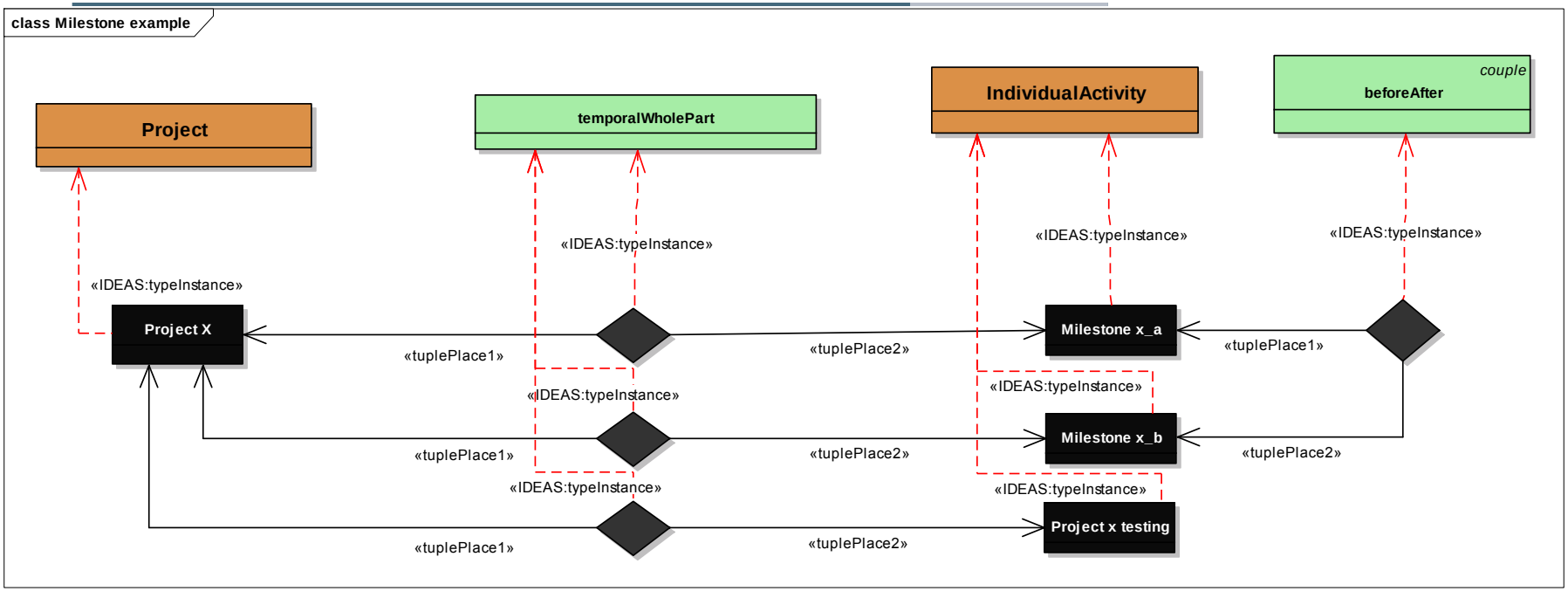
- The previous slide, apart from being unreadable looks extremely complicated, however:
  - It is actually almost completely an explicit DoDAF 2 PV-2 slide.
  - It contains almost all of the DoDAF defined necessary elements for a PV-2 slide (there are some missing mostly associated with measurements of various kinds).
  - It also contains some elements defined as optional for the view and some that a modeler is actually not allowed to use (notably Individual and IndividualType, these are used here since to exclude them would make it difficult to see where different elements point to).
  - The slide actually demonstrates a fairly large number of the temporal aspects of DoDAF 2 and it may therefore be of interest to look at them in slightly more detail.

# Projects and activities



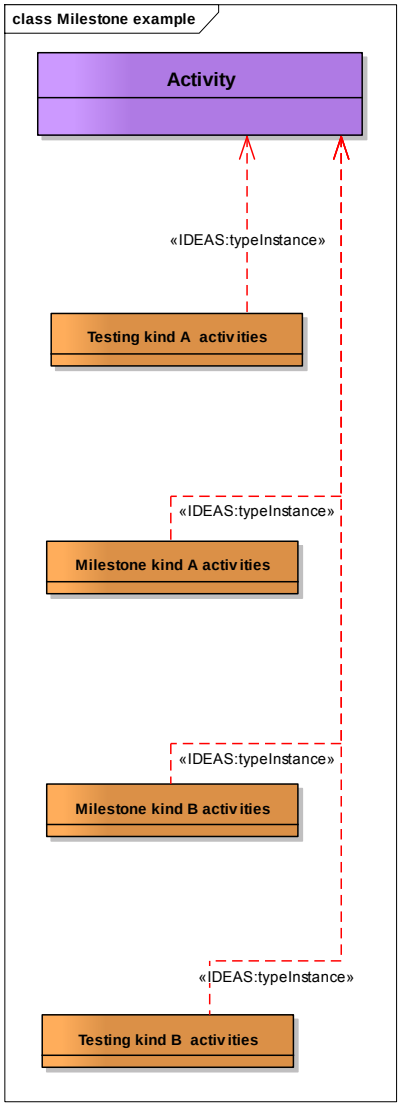

- A set of Individual projects are contained in the example model and a set of example activities.
- Since milestones are not a part of the DoDAF vocabulary activities have been chosen instead and there are a few different individual milestones as well as a completely different type of activity (testing) associated with each individual project.

# Temporal parts and before after



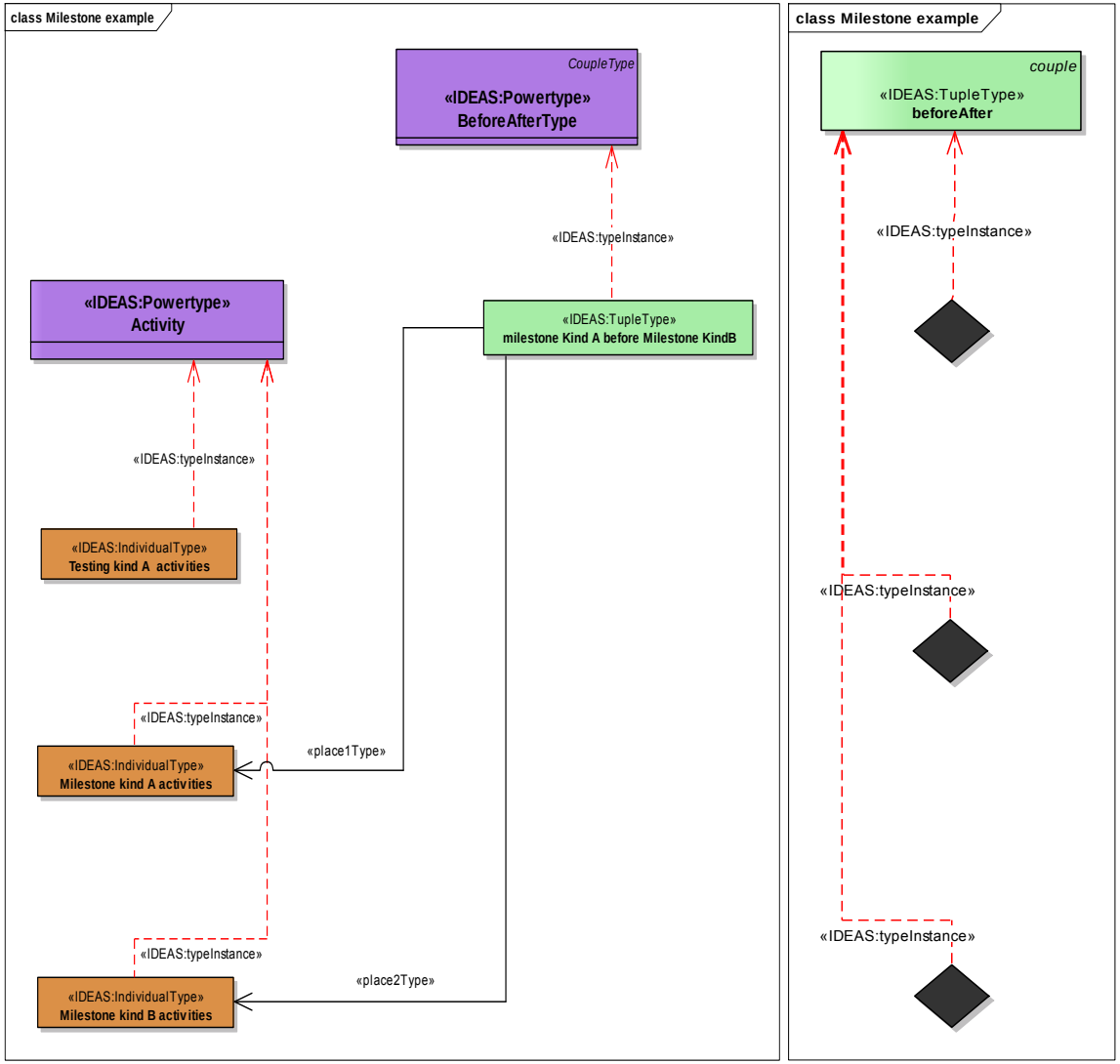

- The above shows project X with three different individual activities. Two of these are milestones and one is a testing activity.
- All three activities are temporal parts of the X project and before after is used to indicate that milestone a is before milestone b, note that there is no indication of the time interval in between.

# Activity handling




- Activity in DoDAF 2 is the set of all subsets of the set of all individual activities and therefore the four sets defined here are instances of the Activity subset.
- Testing Kind A activities contain: Project x testing and Project y testing.
- Testing Kind B activities contain: Project z testing
- Milestone Kind A activities contain: Milestone x_a, Milestone y_a and Milestone z_a
- Milestone Kind B activities contain: Milestone x_b, Milestone y_b and Milestone z_b

# BeforeAfterType

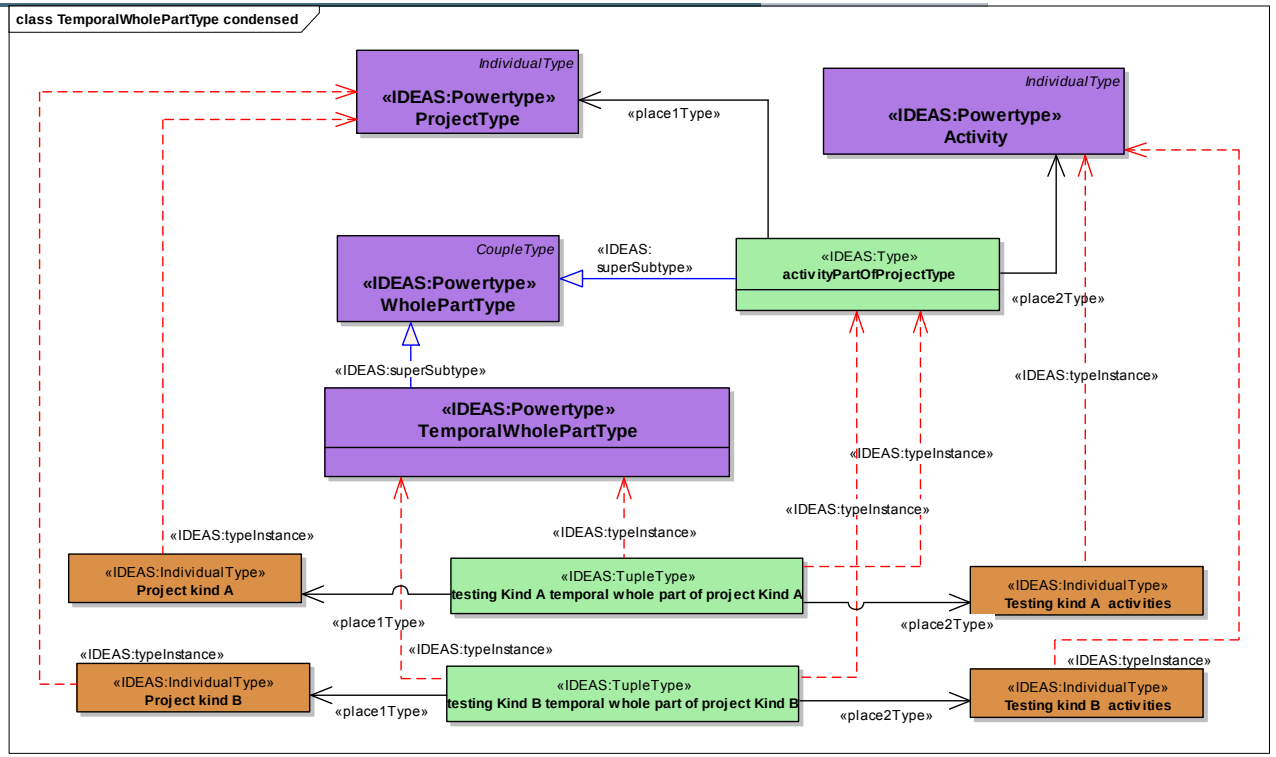






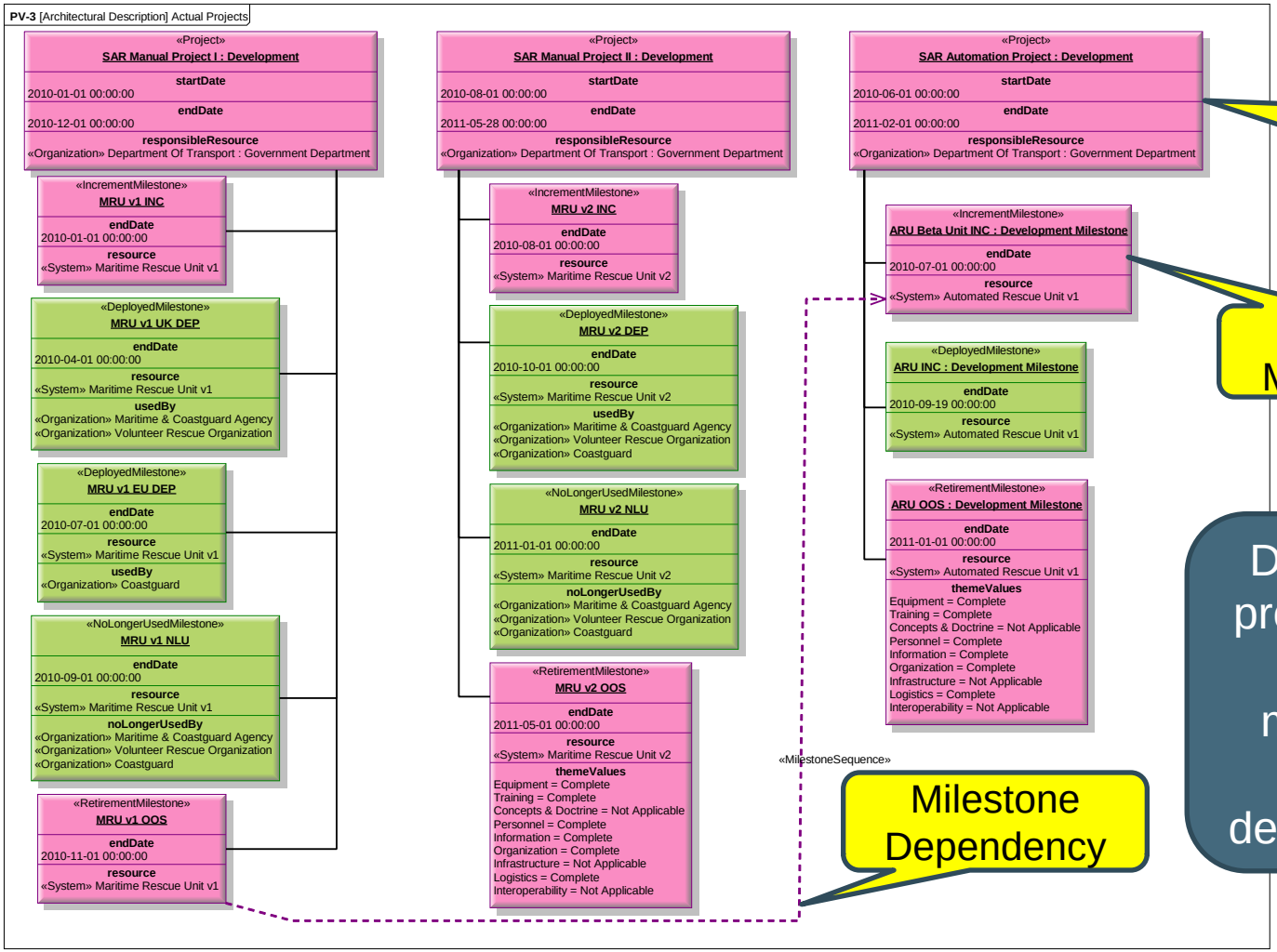
- Since all instances within Milestone Kind A activities occur (i.e. end) before all instances within Milestone Kind B activities an instance of BeforeAfterType can be created in the form of the element *milestone Kind A before Milestone Kind B.*
- This element contains all of the before after relationships defined in the example.

# TemporalWholePartType




- As was shown previously, the testing activities can be combined into two distinct subsets that are instances of Activity (since it contains all possible subsets).
- This also means that instances of TemporalWholePartType can be created that contain the relationships that deal with temporal whole parts for testing Kind A and testing kind B.
- These in turn are instances of the DM2 element activityPartOfProjectType.

# Project View in UPDM



# The Unified Profile for DoDAF and MODAF (UPDM)

- UPDM is a standardized way of expressing DoDAF and MODAF artefacts using UML and SysML
  - UPDM is *NOT* a new Architectural Framework
  - UPDM is not a methodology or a process
  - UPDM implements DoDAF 2.0, MODAF & NAF
- UPDM was developed by members of the OMG with help from industry and government domain experts.
- UPDM is a DoD mandated standard and has been implemented by multiple tool vendors.

# Time and Architecture

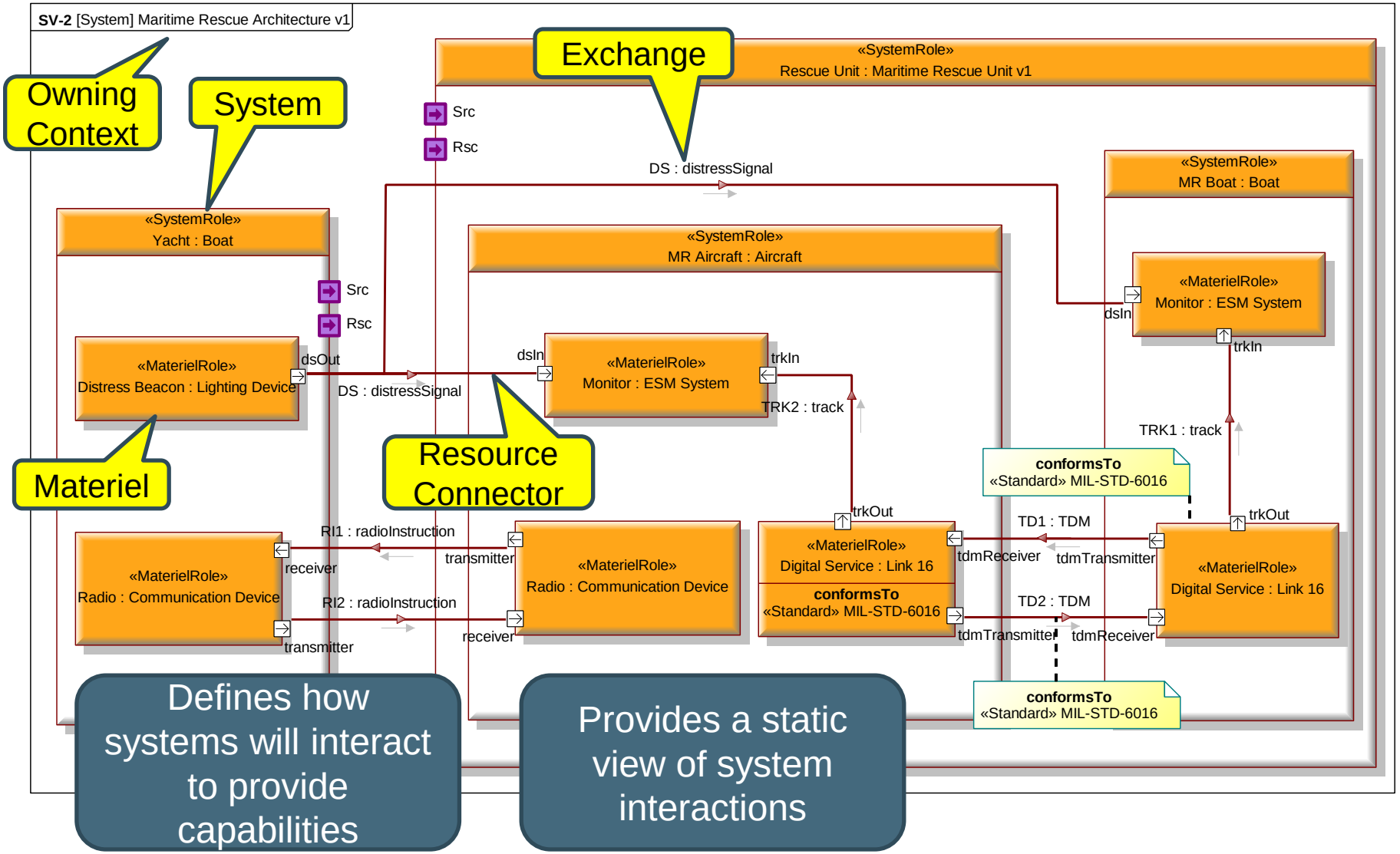
# Time in DoDAF

- Sequence of events
- System changes over time
- Use of a system changing over time
- Different systems supporting a capability over time
- A system supporting different capabilities at different phases of its lifecycle
- System states showing time dependent behavior
- Time dependent activity sequences
- Modeling processing time, latency, transport time etc.
- Scheduling deployment of systems over time
- Personnel deployment and competency assessment
- Data Lifecycles
- Integrating system acquisition cost, deployment cost etc. to show total cost of ownership.
- Modeling product variants
- Showing cost vs. time vs. capability
- Etc.

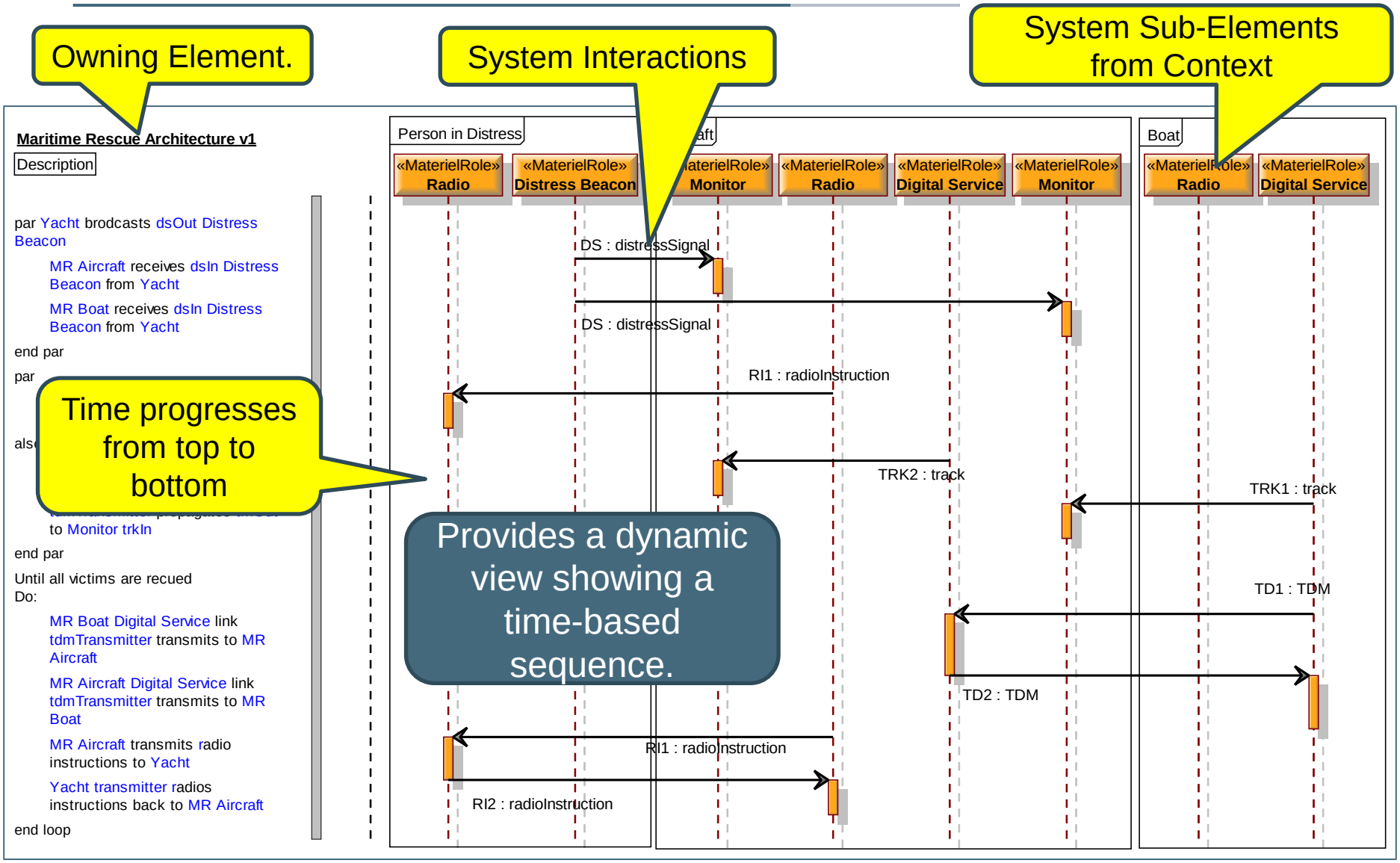
# Sequence of Events

- SV-10c Resource Event Trace Description
- Shows interactions in time order that have been created on SV-1 and SV-2 diagrams.
  - Dynamic (SV-10c) vs. Static View (SV-1, SV-2)
- Shows how resources interact
  - Exchange of information between resources
  - Elements on the diagram are parts of the owning element
  - Sequence of exchanges
  - Time
- Interaction between Systems Node Roles, Organization Roles, Post Roles, System Roles and Software Roles
- Also shows behavioral interactions such as events and operations
- Similar to OV-6c View

# SV-2: Resource Interaction Specification



# SV-10c: System Event Trace Description



# Modeling processing time, latency, transport time etc.

- Timing information can be added to enhance analysis
  - Processing duration
  - Transmission delay
  - Latency
  - Etc.
- Simulation can verify timing and behavior

# SV-10c: System Event Trace Description
# With added timing information

**Maritime Rescue Architecture v1**

Description

par Yacht brodcasts dsOut Distress Beacon
- MR Aircraft receives dsIn Distress Beacon from Yacht
- MR Boat receives dsIn Distress Beacon from Yacht
- Process Message

end par

Process Message

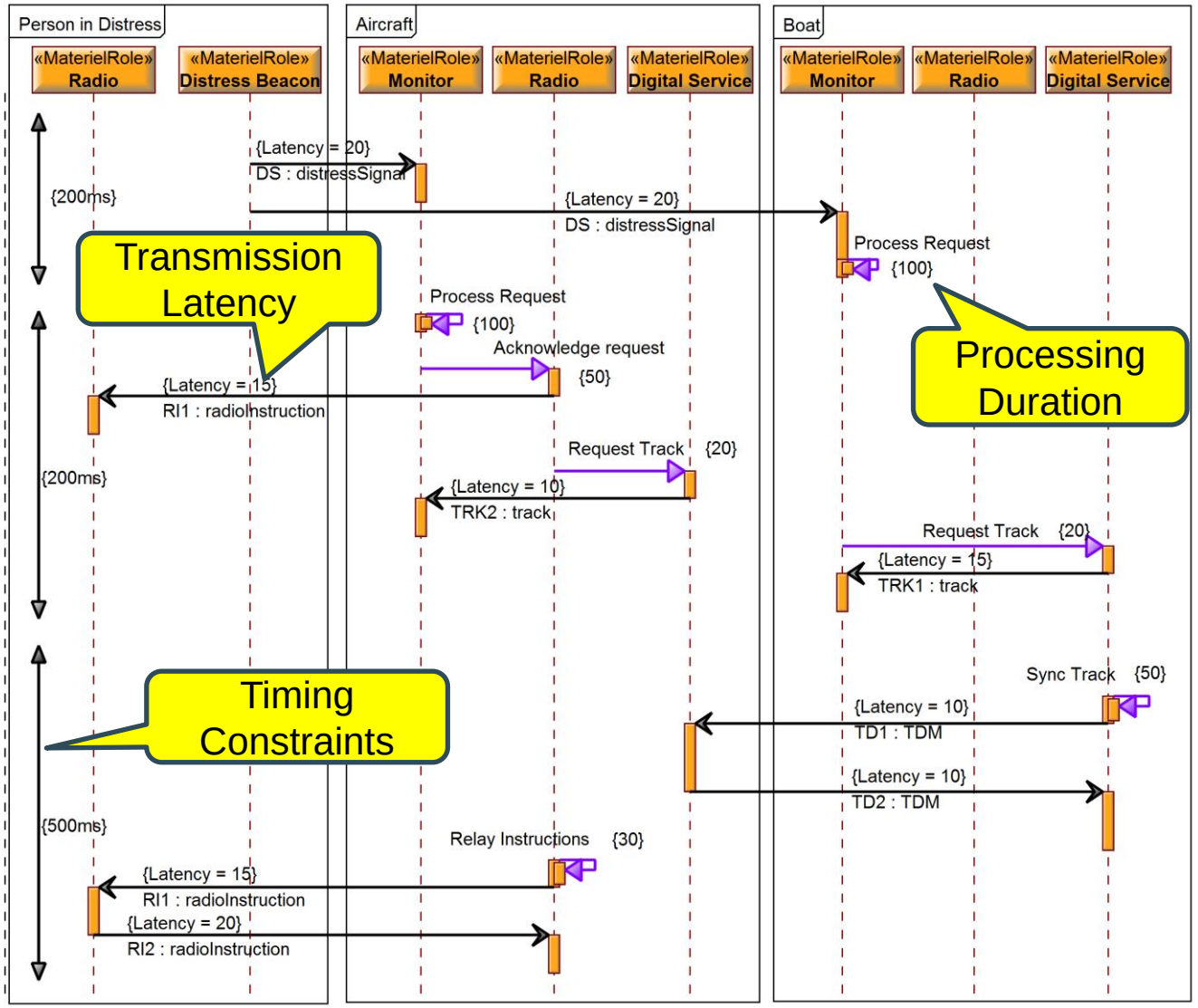
par
- ACK
- MR Aircraft transmits radio instructions to Yacht

also par
- Track
- tdmTransmitter propagates trkOut to Monitor trkIn
- Request Track
- tdmTransmitter propagates trkOut to Monitor trkIn

end par

Until all victims are recued
Do:
- Track Sync
- MR Boat Digital Service link tdmTransmitter transmits to MR Aircraft
- MR Aircraft Digital Service link tdmTransmitter transmits to MR Boat
- Relay Instructions
- MR Aircraft transmits radio instructions to Yacht
- Yacht transmitter radios instructions back to MR Aircraft

end loop



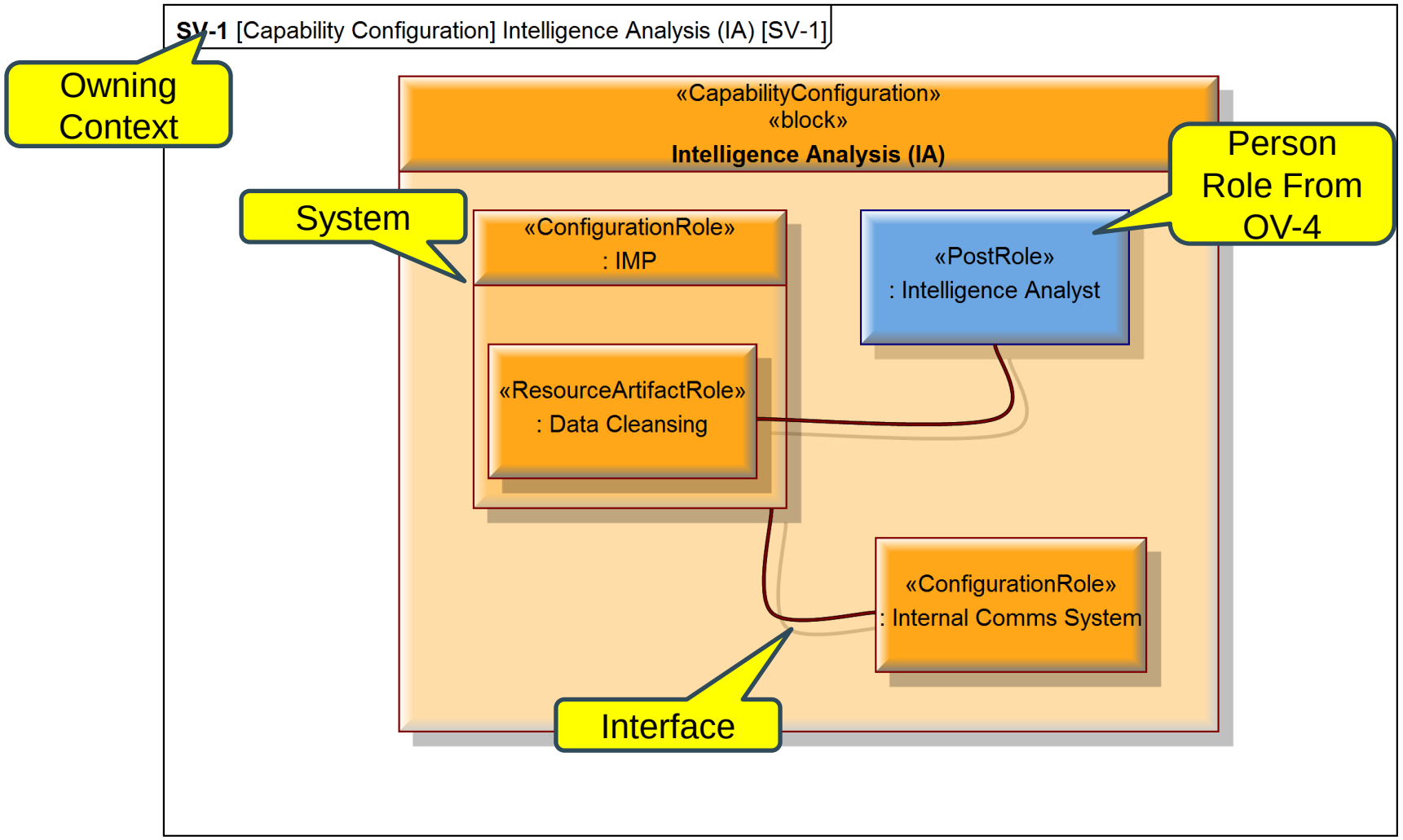
# Systems Changing over Time

- The paradox of the Ship of Theseus – Plutarch
  - If you take all the parts of a system (Theseus' ship) and replace them, is it the same ship?
- Abe Lincoln's axe
  - Lincoln was well known for his ability with an axe, and axes associated with his life are held in various museums.
  - Are they all "Abe Lincoln's Axe"?
- Systems change over time
  - System lifecycle of design, manufacture, deployment, maintenance, retirement
  - Changes for mission-based configurations
  - Changes due to maintenance
  - Etc.

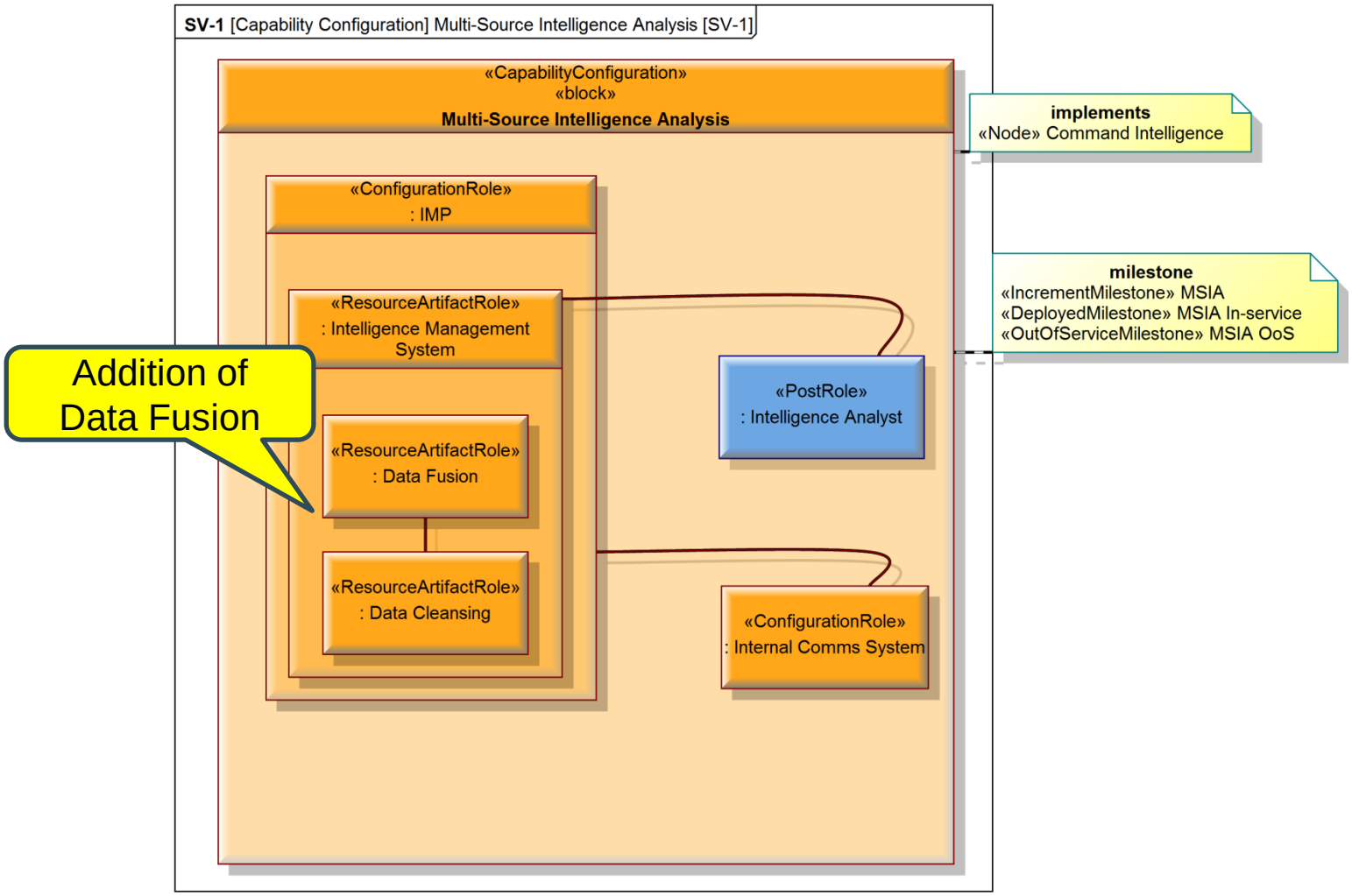
# System Changes over Time

- SV-1/SV-2 Resource interaction specification are used to define system structure
- Shows how resources (systems, roles, posts and organizations) interact
- Created from systems, system nodes and organizations
- Defined system configurations can be linked to project deployments

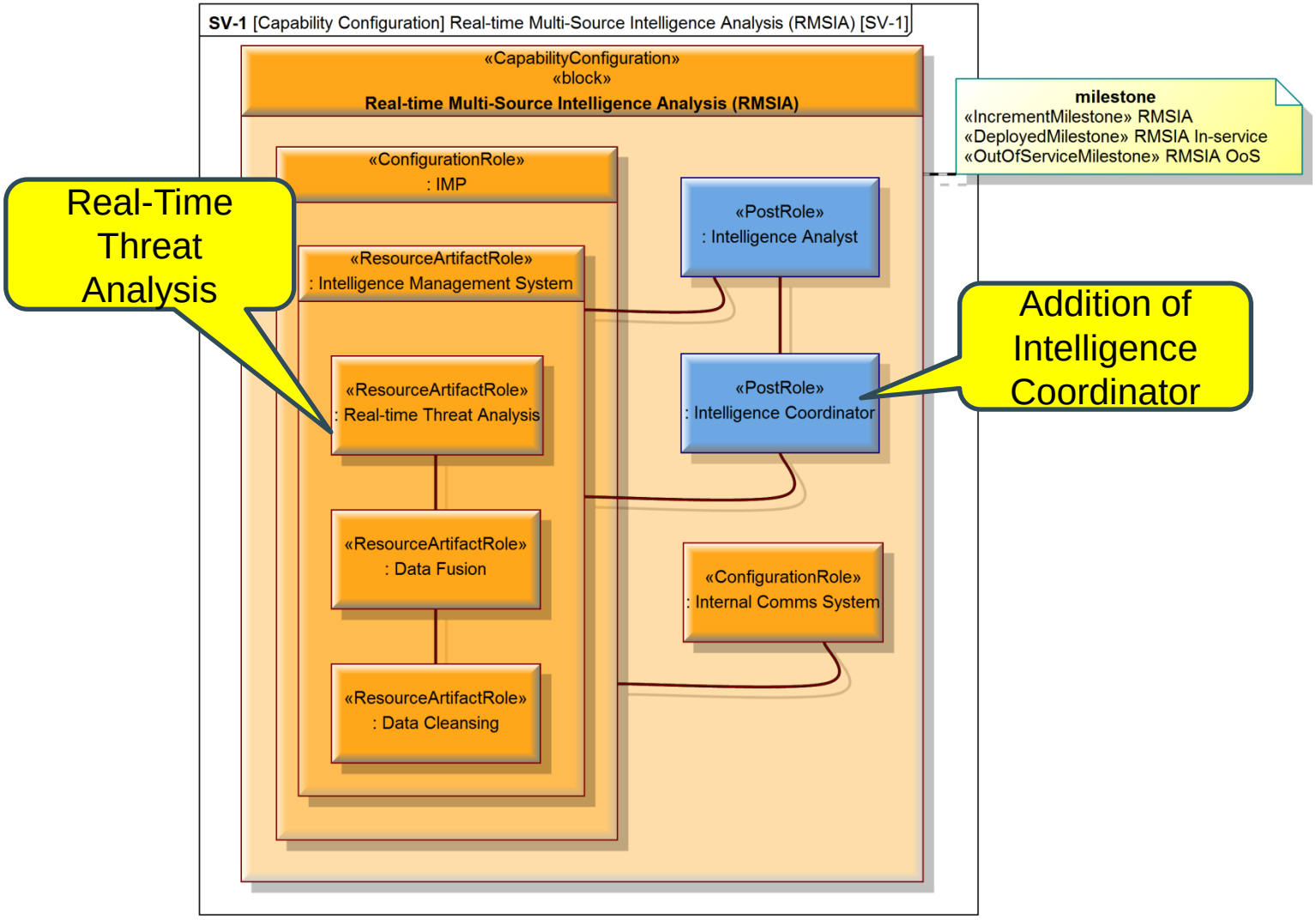
# SV-1: Resource Interaction Specification Version 1: Intelligence Analysis



# SV-1: Resource Interaction Specification
# V2: Multi-Source Intelligence Analysis



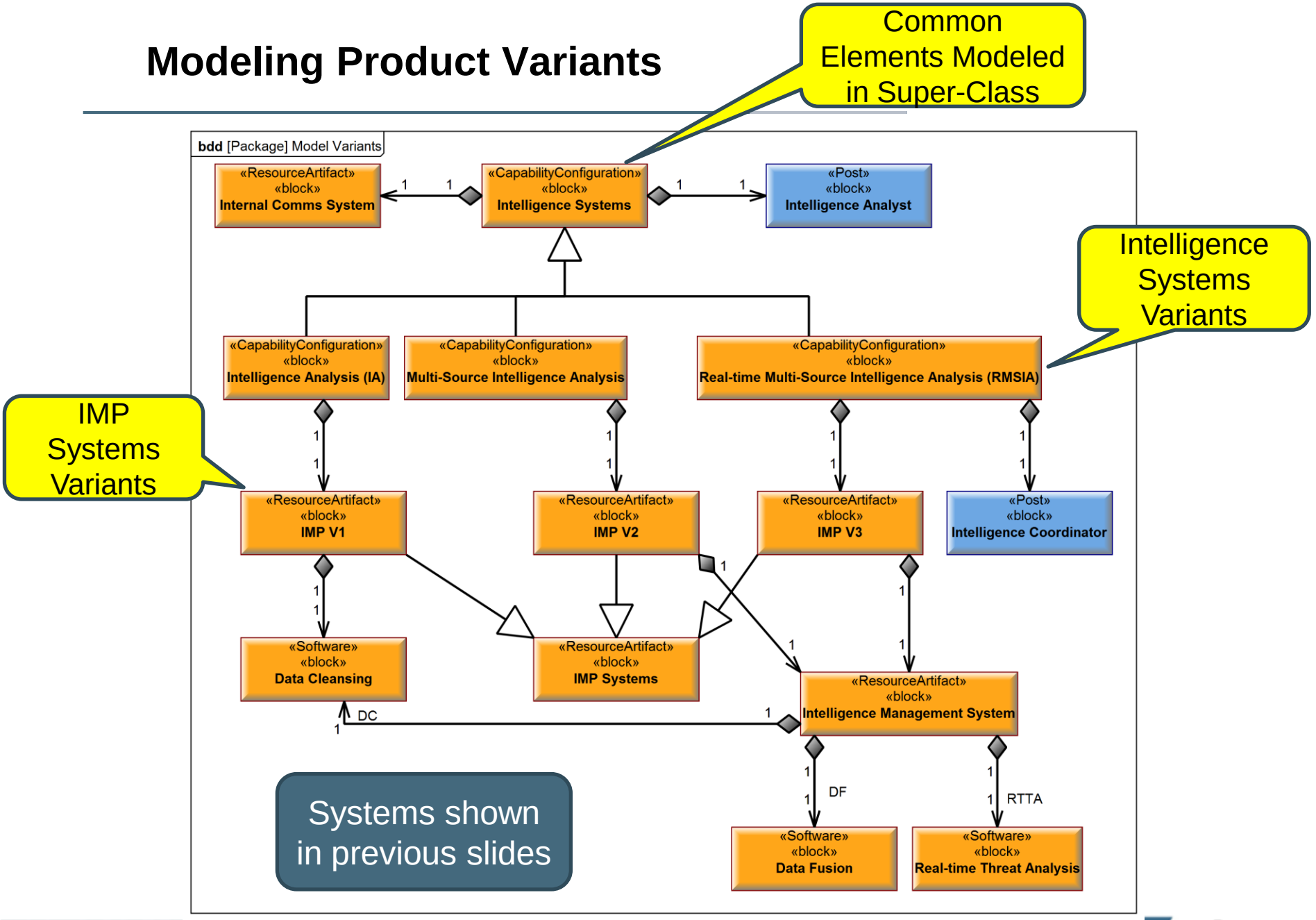
# SV-1: Resource Interaction Specification



# Modeling product variants

- No Replication of Work:
  - Common elements defined once and reused
    - Parts
    - Interfaces
    - Functions
- Similar model elements should be modeled only once, in the family model
  - Enforced re-use
- Maturing standards for variability modeling
- Compatibility with all views defined in SysML
  - Requirements, Structure, Behavior, Parametrics
  - Variability Methods are more then a combinatory re-use of structural components

# Modeling Product Variants



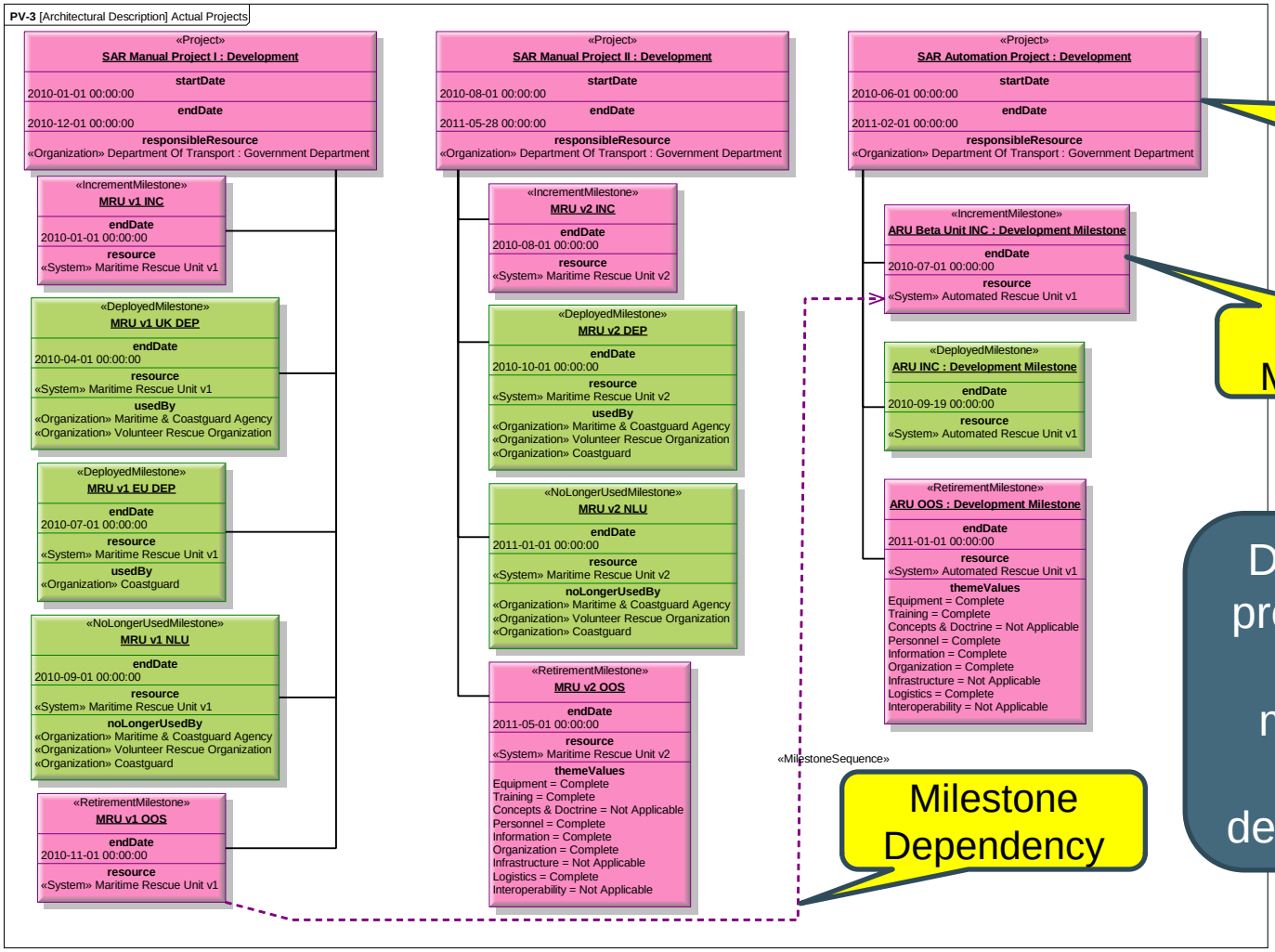
# Project Views: Scheduling deployment of systems over time

- Contains information about programs, projects, portfolios, or initiatives and relating that information to capabilities and other programs, projects, portfolios, or initiatives.
- PV-1: Project Portfolio Relationships
  - It describes the dependency relationships between the organizations and projects and the organizational structures needed to manage a portfolio of projects.
- PV-2: Project Timelines
  - A timeline perspective on programs or projects, with the key milestones and interdependencies.
- PV-3: Project to Capability Mapping
  - A mapping of programs and projects to capabilities to show how the specific projects and program elements help to achieve a capability.

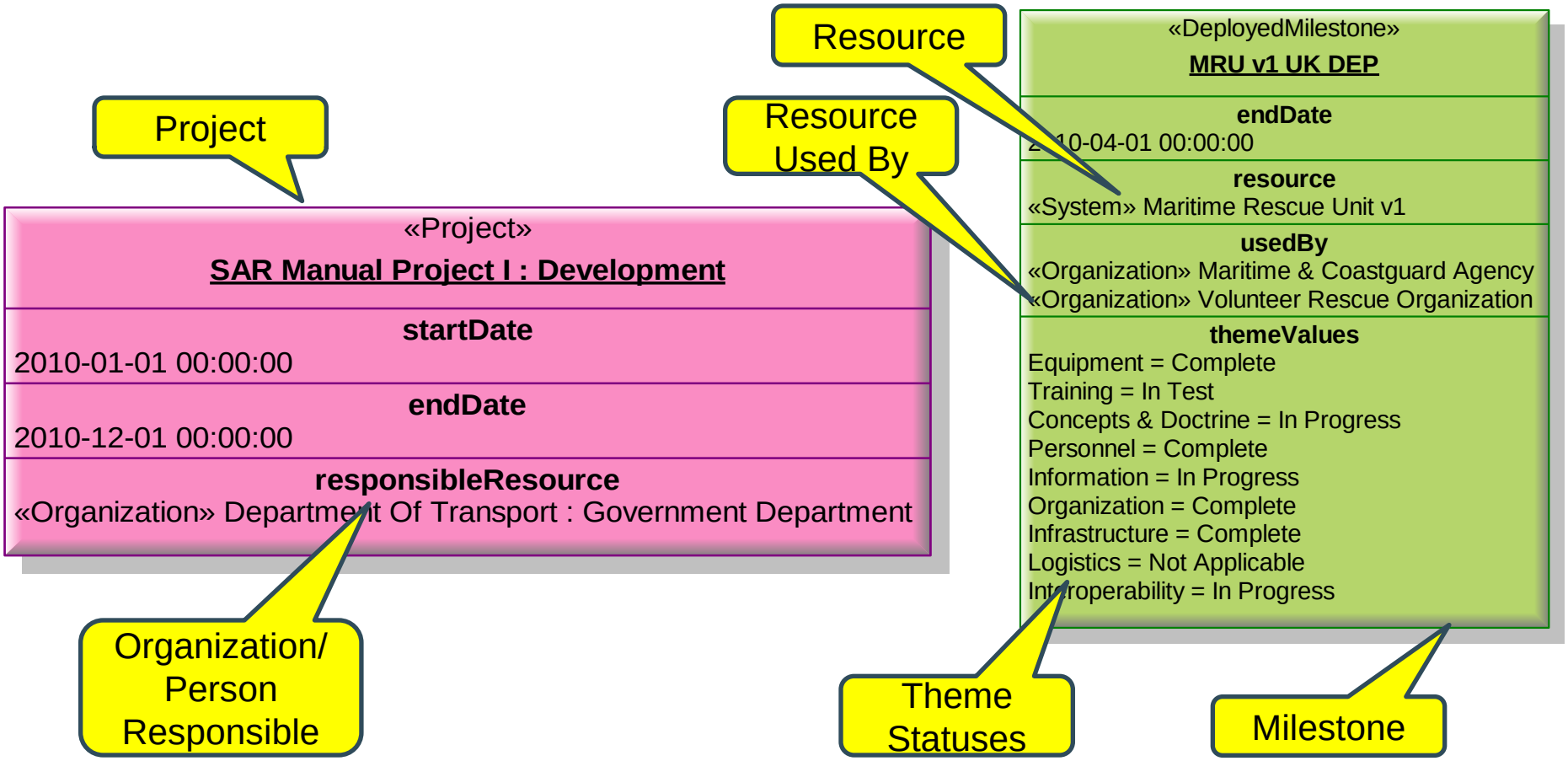
# Scheduling deployment of systems over time

- A PV-1 Actual Projects defines Projects, Actual Project Milestones, Deployed Milestones, Increment Milestones, No Longer Used Milestones, Retirement Milestones, Project Sequences and Milestone Sequences.
- Links are created between the Project and its actual Milestones.
- Milestone sequences link Milestones
- Project sequences link Projects
- Specify responsible resources (people, organizations) for projects
- Specify resources to be deployed etc., and organizations that use the resource for milestones.

# Scheduling deployment of systems over time PV-1



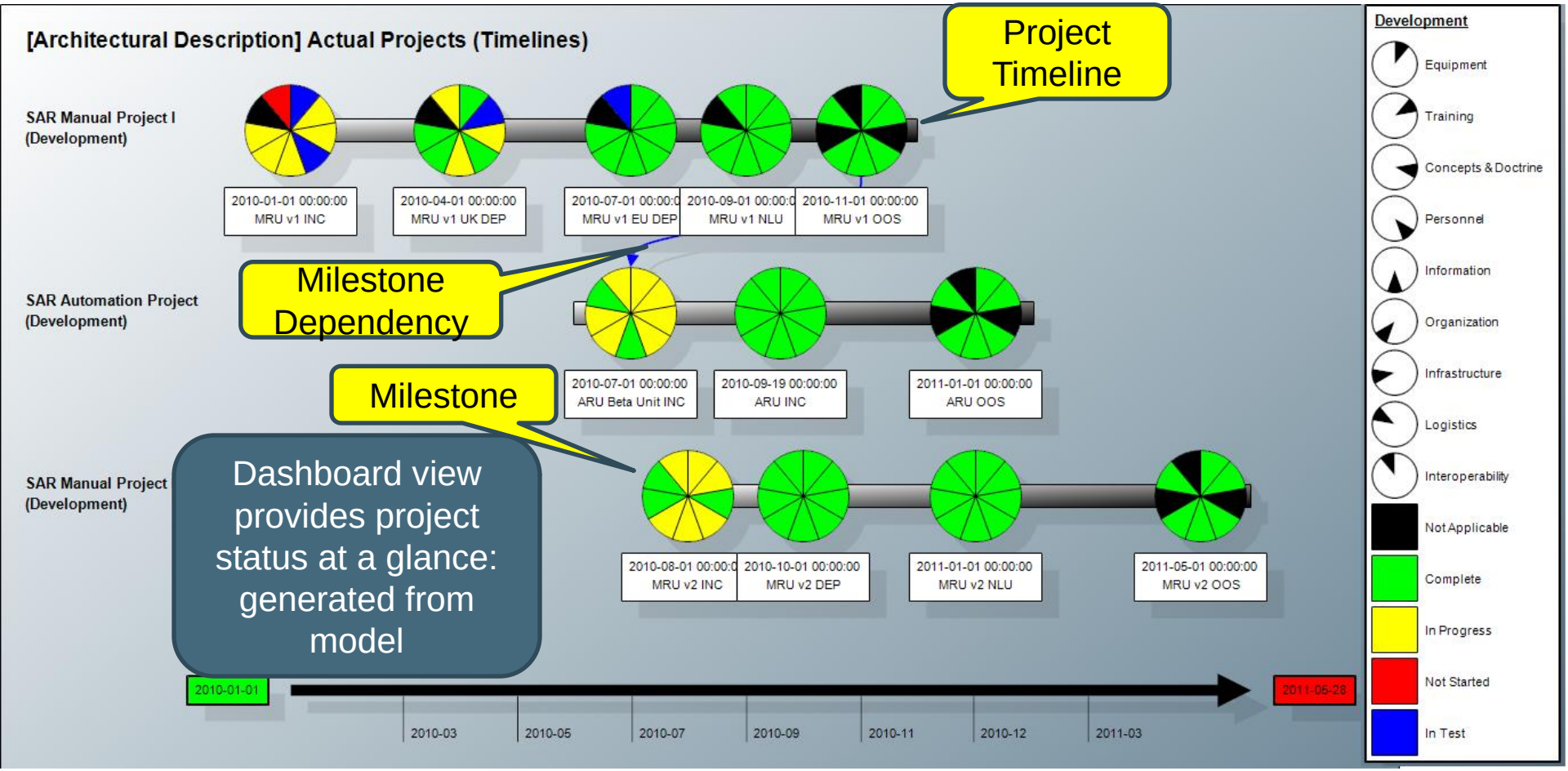
# Scheduling deployment of systems over time PV-1 Detail



# Scheduling deployment of systems over time
# PV-2 Project Timelines

- A PV-2 provides a timeline perspective on projects and their relationship to Systems.
  - There is no set format for this view. Each tool has implemented the PV-2 diagram in its own way.
- The PV-2 is populated using live data from the model.
- It contains project timelines, project milestones statuses, and project and milestone sequences.

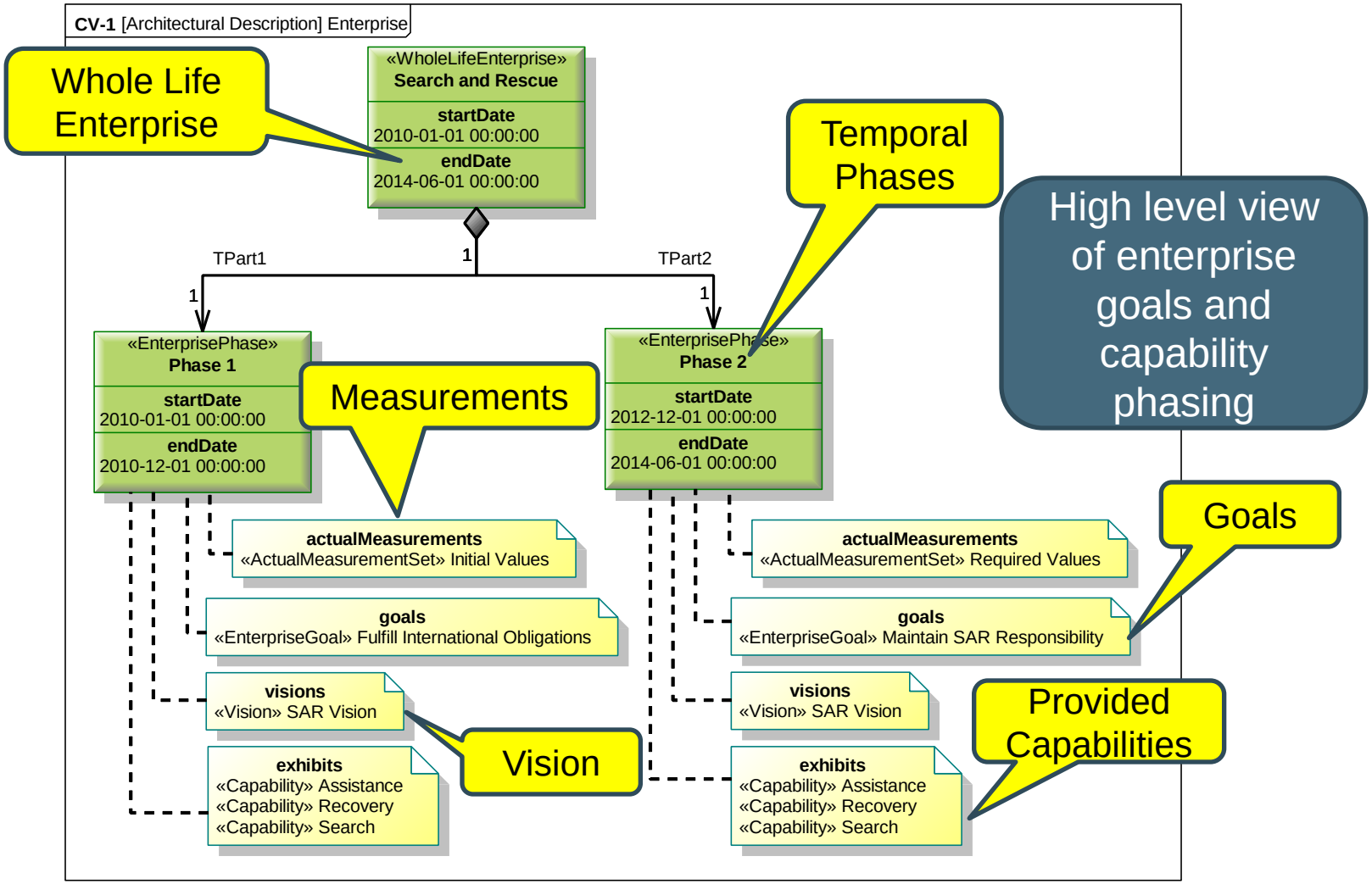
# Scheduling deployment of systems over time
# PV-2 Project Timelines



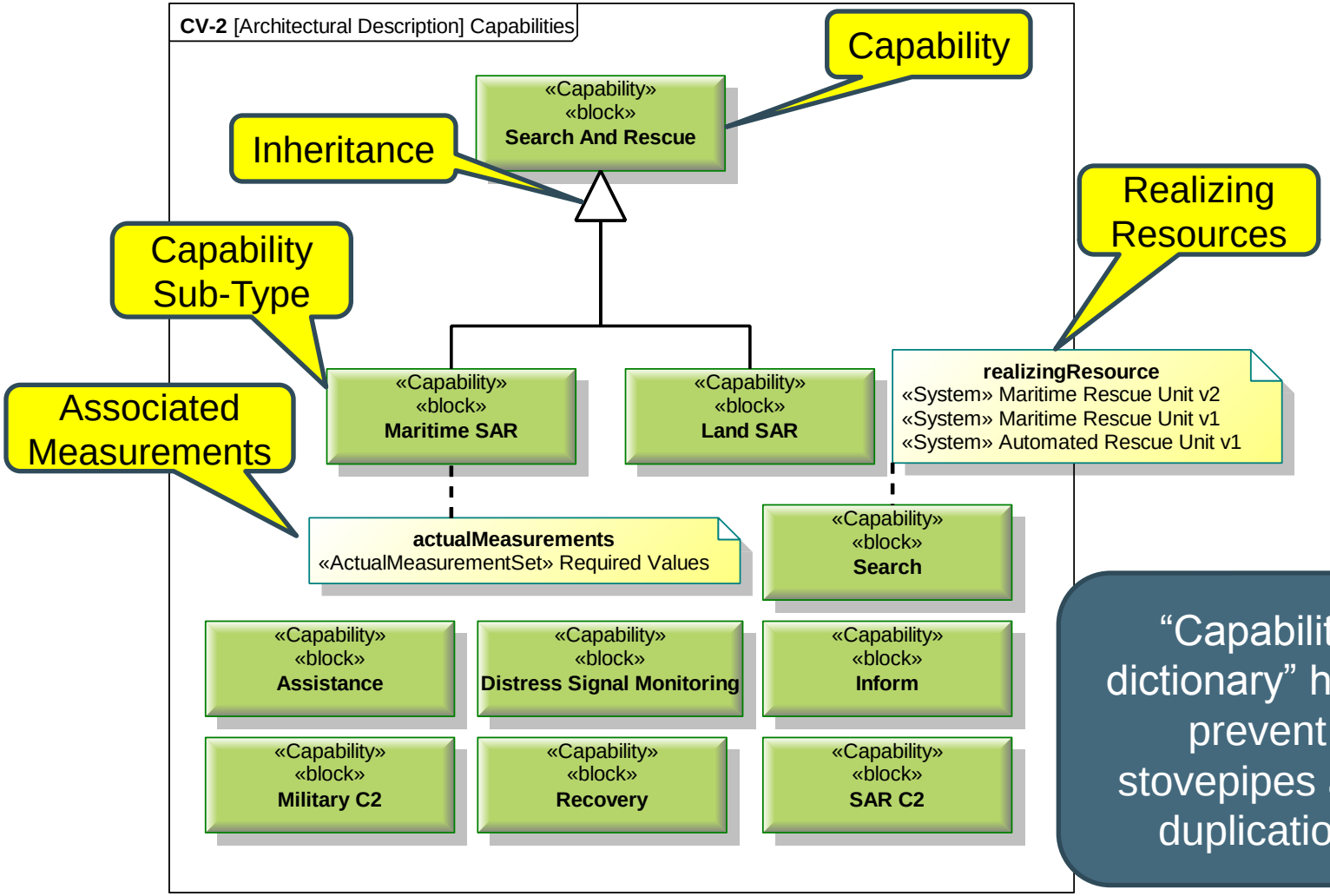
# Different systems support the same capability over time (1) Capability Views

- Defines capabilities, capability dependencies and relationships
- Provides high-level view of expected capability within each time frame
- Shows the intersection between capabilities, systems, system metrics, etc. over time
- Used for:
  - Capability overlap/gap analysis
  - Identification of high-level capability issues

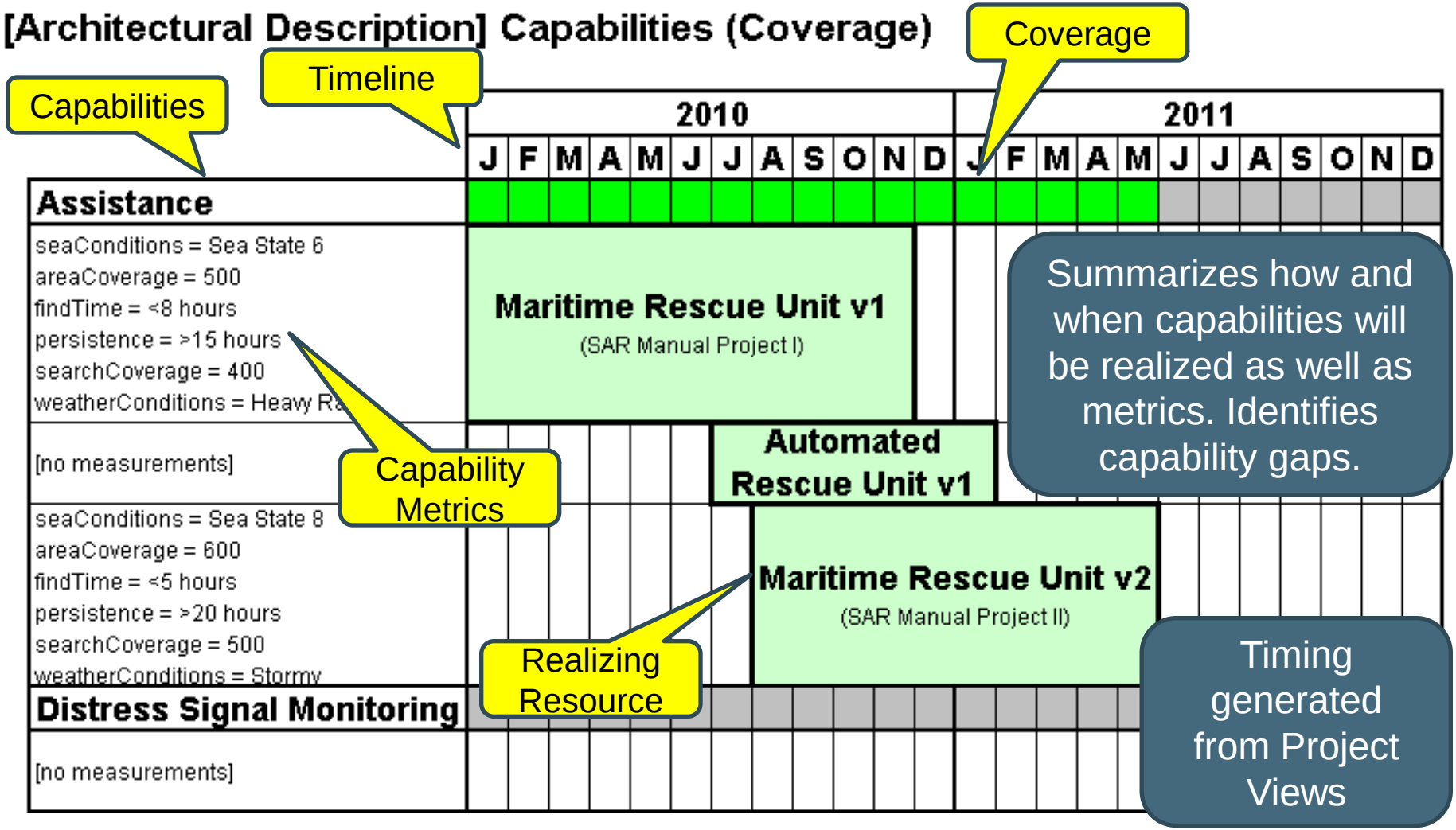
# CV-1 Capability Vision



# CV-2 Capability Taxonomy



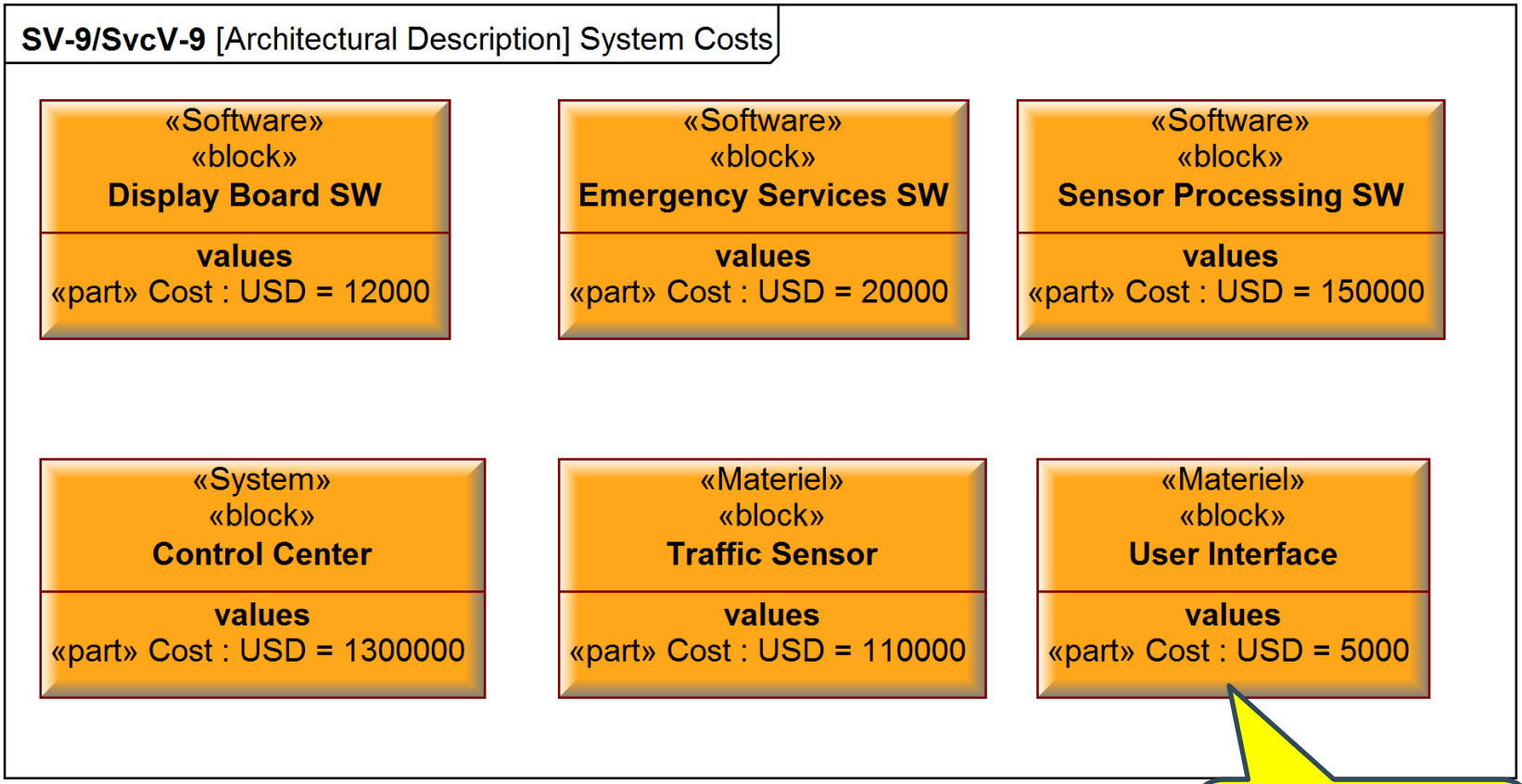
# CV-3 Capability Phasing (Fragment)



# Total Cost and Cost Over Time

- DoDAF/SysML provide a means to add value properties to model elements
  - Size, Weight, Power, Cost, etc.
- A system is an aggregation of its parts.
  - Therefore, the total number of each type of part can be calculated
  - If the cost of each part is known, the total cost of the system can be calculated
  - Running costs can also be defined for the system for sub-systems as well as people
  - Project views provide a means to show system deployment schedules, therefore running costs can be calculated as well.

# System View Showing Individual System Costs



# Summary of System Costs

| Name | Quantity | Cost (in USD) | | | | |
|---|---|---|---|---|---|---|
| | | Estimate | No Margin | Margin (%) | With Margin | Budget |
| Traffic Context (Autoville Traffic Management Architecture::System View | n/a | 50000000 | 51799000 | 12.00% | 58014880 | 600000000 |
| Control Center (Autoville Traffic Management Architecture::System Vie | 1 | 1000000 | 1649000 | 20.00% | 1978800 | 1300000 |
| Control Room (Autoville Traffic Management Architecture::System Vie | 1 | 0 | 649000 | 10.06% | 714300 | 782000 |
| Control Room Operator (Autoville Traffic Management Architecture:: | 2 | 0 | 0 | 0.00% | 0 | 0 |
| Control System (Autoville Traffic Management Architecture::System | 1 | 0 | 645000 | 10.06% | 709900 | 777000 |
| Display Board SW (Autoville Traffic Management Architecture::Syst | 1 | 10000 | 10000 | 10.00% | 11000 | 12000 |
| Emergency Services SW (Autoville Traffic Management Architecture | 1 | 15000 | 15000 | 10.00% | 16500 | 20000 |
| Sensor Processing SW (Autoville Traffic Management Architecture:: | 1 | 100000 | 100000 | 10.00% | 110000 | 150000 |
| Traffic Control SW (Autoville Traffic Management Architecture::Syst | 1 | 100000 | 100000 | 10.00% | 110000 | 120000 |
| Traffic Data Archive SW (Autoville Traffic Management Architecture | 1 | 10000 | 10000 | 10.00% | 11000 | 15000 |
| Traffic Display SW (Autoville Traffic Management Architecture::Syst | 1 | 100000 | 100000 | 10.00% | 110000 | 110000 |
| Traffic Event SW (Autoville Traffic Management Architecture::Syste | 1 | 120000 | 120000 | 10.00% | 132000 | 130000 |
| Traffic Flow Calculation SW (Autoville Traffic Management Architec | 1 | 50000 | 50000 | 10.00% | 55000 | 60000 |
| Traffic Prediction SW (Autoville Traffic Management Architecture::S | 1 | 0 | 0 | 0.00% | 0 | 0 |
| Traffic Report Generation SW (Autoville Traffic Management Archit | 1 | 20000 | 20000 | 12.00% | 22400 | 25000 |
| Traffic Signal SW (Autoville Traffic Management Architecture::Syste | 1 | 20000 | 20000 | 10.00% | 22000 | 25000 |
| Video Processing SW (Autoville Traffic Management Architecture::S | 1 | 100000 | 100000 | 10.00% | 110000 | 110000 |
| Web Presence SW (Autoville Traffic Management Architecture::Syst | 1 | 0 | 0 | 0.00% | 0 | 0 |
| Weather Processing SW (Autoville Traffic Management Architecture | 1 | 0 | 0 | 0.00% | 0 | 0 |
| User Interface (Autoville Traffic Management Architecture::System V | 2 | 2000 | 4000 | 10.00% | 4400 | 5000 |
| City Planning (Autoville Traffic Management Architecture::System View | 1 | 0 | | | | 0 |
| Emergency Services (Autoville Traffic Management Architecture::Syster | 1 | 0 | | | | 0 |
| Event Venue (Autoville Traffic Management Architecture::System View | 1 | 0 | | | | 0 |
| Internet (Autoville Traffic Management Architecture::System Views::Re | 1 | 0 | | | | 0 |
| Mass Media (Autoville Traffic Management Architecture::System Views | 1 | 0 | | | | 0 |
| Road Maintenance (Autoville Traffic Management Architecture::System | 1 | 0 | 0 | 0.00% | 0 | 0 |

Generated report from system configuration.

# Modeling Cost vs. Time vs. Capability (1)

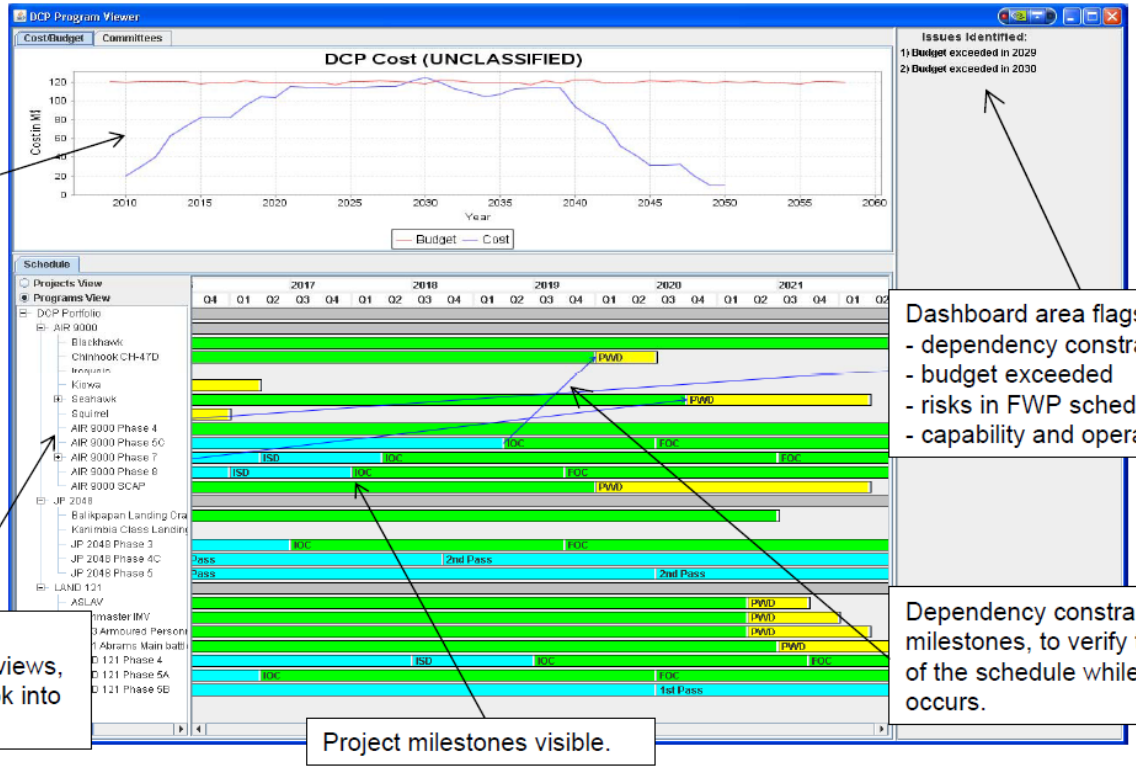
- Capability Goals
  - How do individual projects contribute to capability outcomes?
  - Over what timeframes are the capability outcomes realized?
- Maintenance of Capability
  - When a platform or piece of equipment reaches its planned withdrawal date will there be a new platform/piece of equipment available to replace it?
- Capability Over Time
  - What capability is available at any given point in time?
  - What platforms/equipment/organizations provide the capability?
- Tolerance
  - How much can I move things before it starts to have an impact?
- Committees and Approvals Schedule
  - What impact do changes have on committees and approvals?

Architecture-based Decision Support for Capability Development
Gary Bulluss and Kevin O'Shea
Joint Decision Support Centre, Joint Operations Division
Defence Science and Technology Organisation

# Showing cost vs. time vs. capability (2)

UNCLASSIFIED

## Custom Visualisation




Cost vs Budget visualisation and analysis.

Dashboard area flags all issues:
- dependency constraints broken
- budget exceeded
- risks in FWP schedule
- capability and operational issues

Structured view of projects. High level views, with the ability to look into lower-level details.

Dependency constraints between milestones, to verify the achievability of the schedule while shuffling occurs.

Project milestones visible.

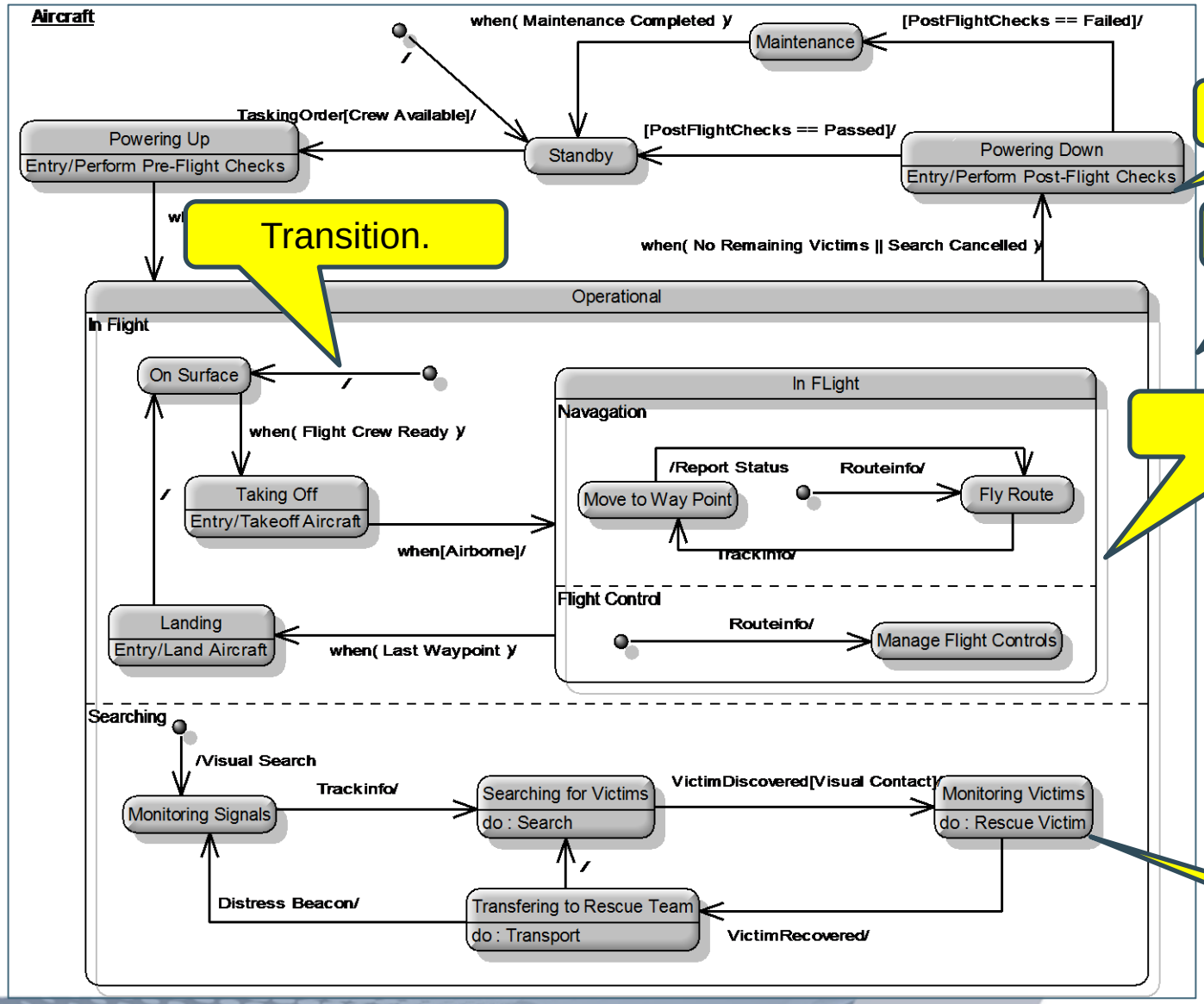
Note: Project names are from the public DCP. Schedules and costings are fictitious.

UNCLASSIFIED

# System states showing time dependent behavior SV-10b Resource State Transition Description

- Shows states within a resource
  - Shows resource state changes
- One diagram per resource (System nodes and systems)
- Shows events passed to and from resources

# SV-10b: System State Transition Description




System Function

Transition.

Owning Element.

Sub-State

State transitions occur based on events and time.

States.



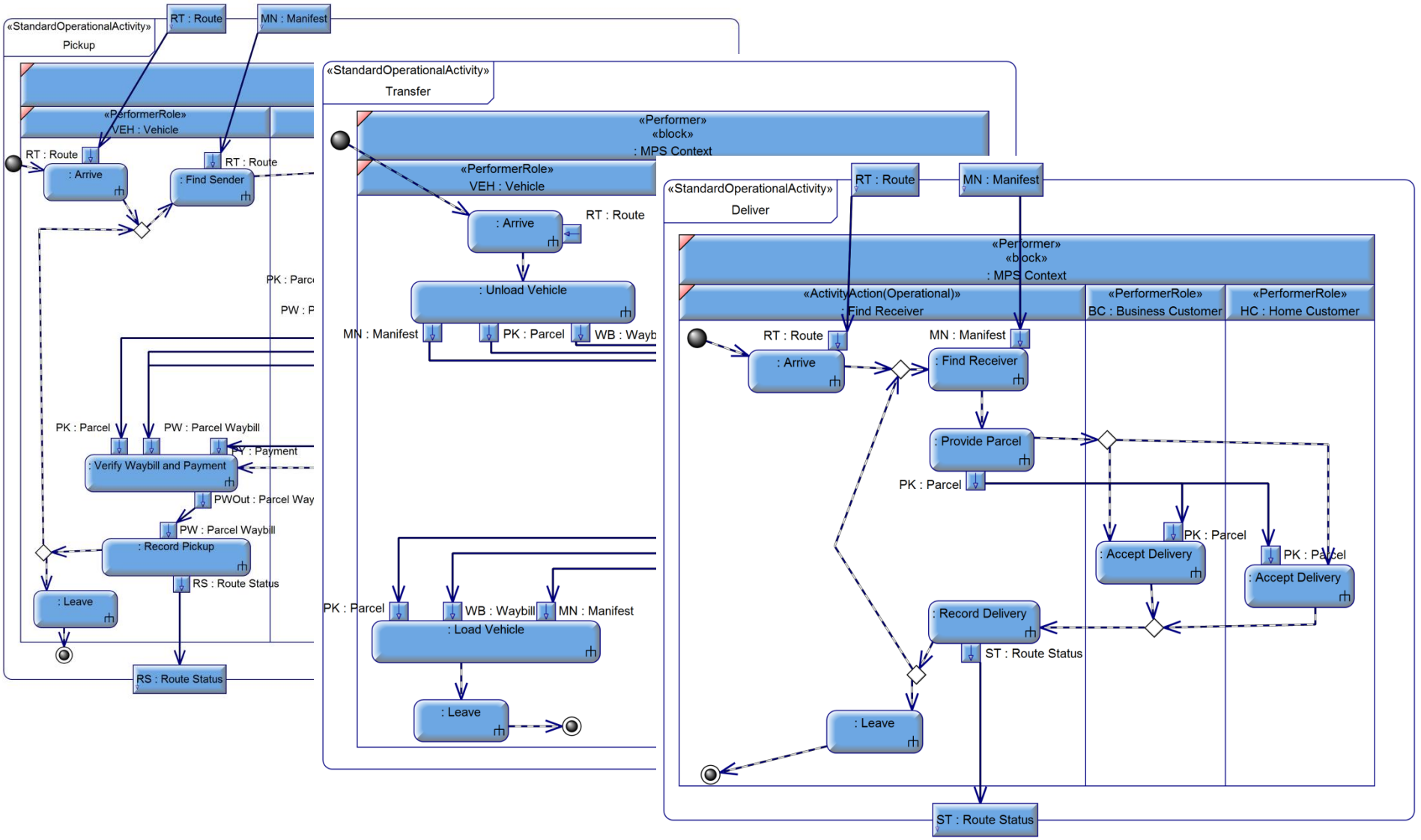


atego

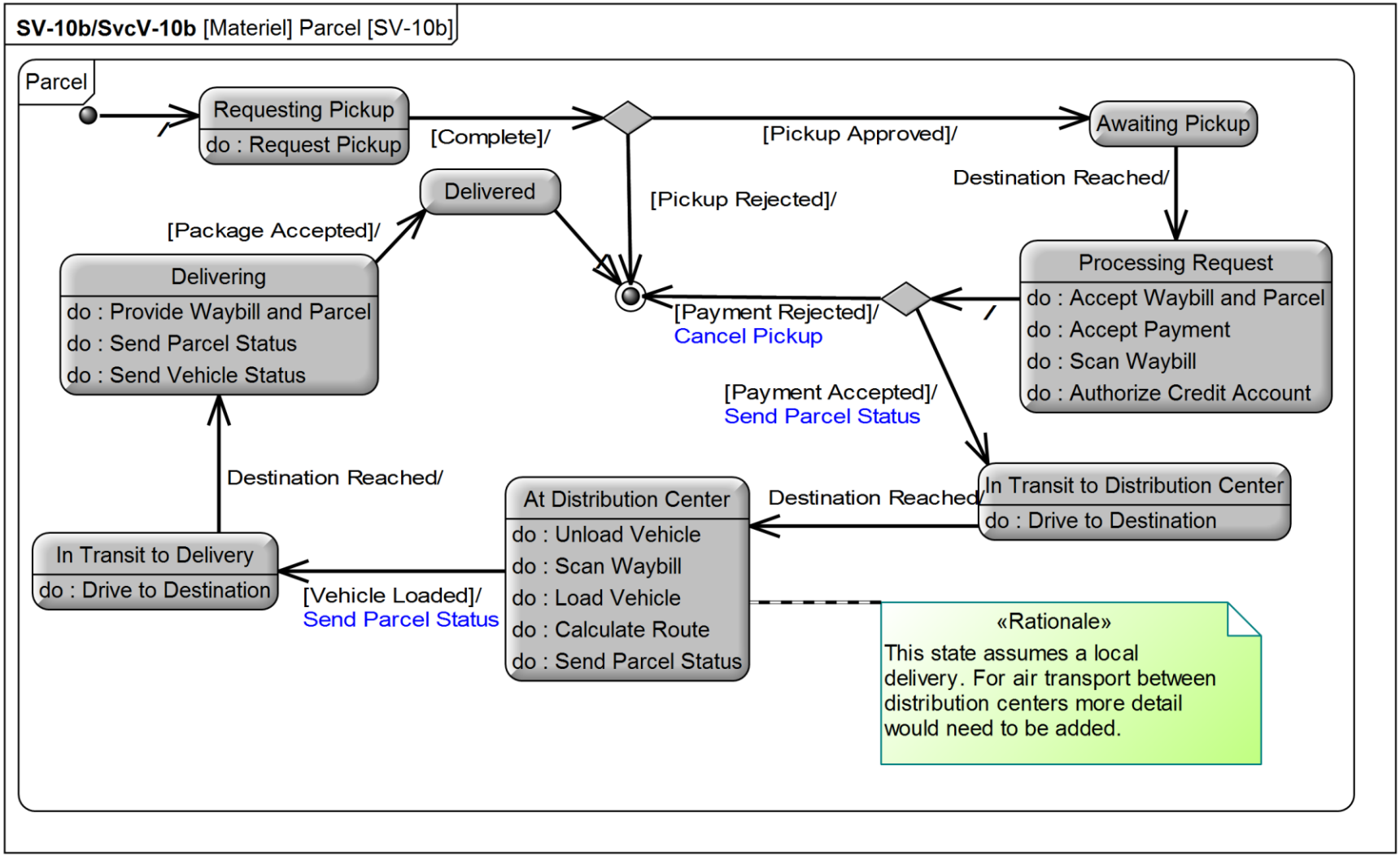
# Data Lifecycles

- Data changes over time
  - Data transformation (Activities)
  - Data Fusion (Activities/Data modeling)
  - Data lifecycle (State Machines)
  - Case management (State machines/Activities/Sequence)
  - Data flow (Activities)
  - Data sequencing (Sequence/Structure)
  - Etc.

# Data management lifecycles – Data Processing



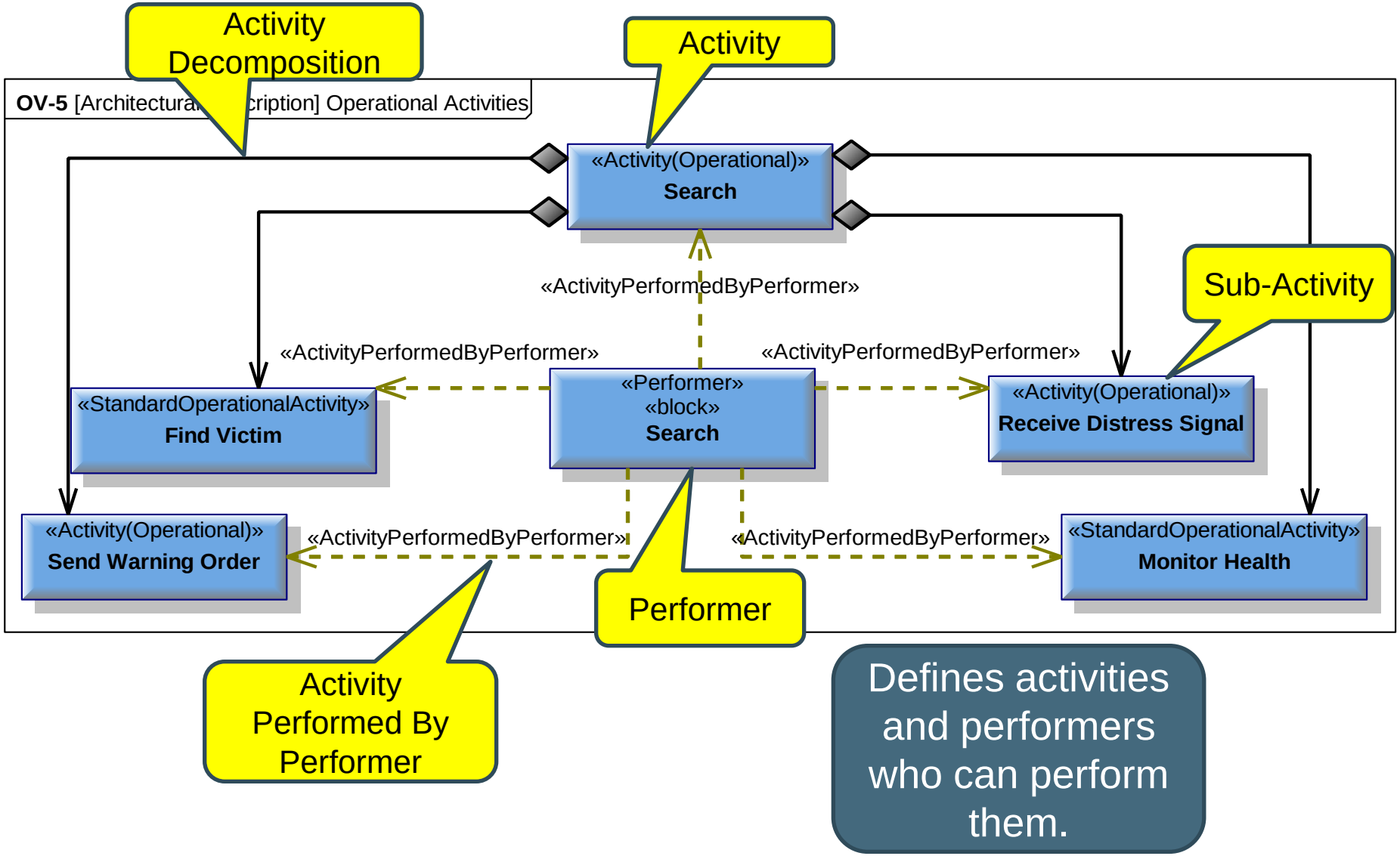
# Data management lifecycles – State Based



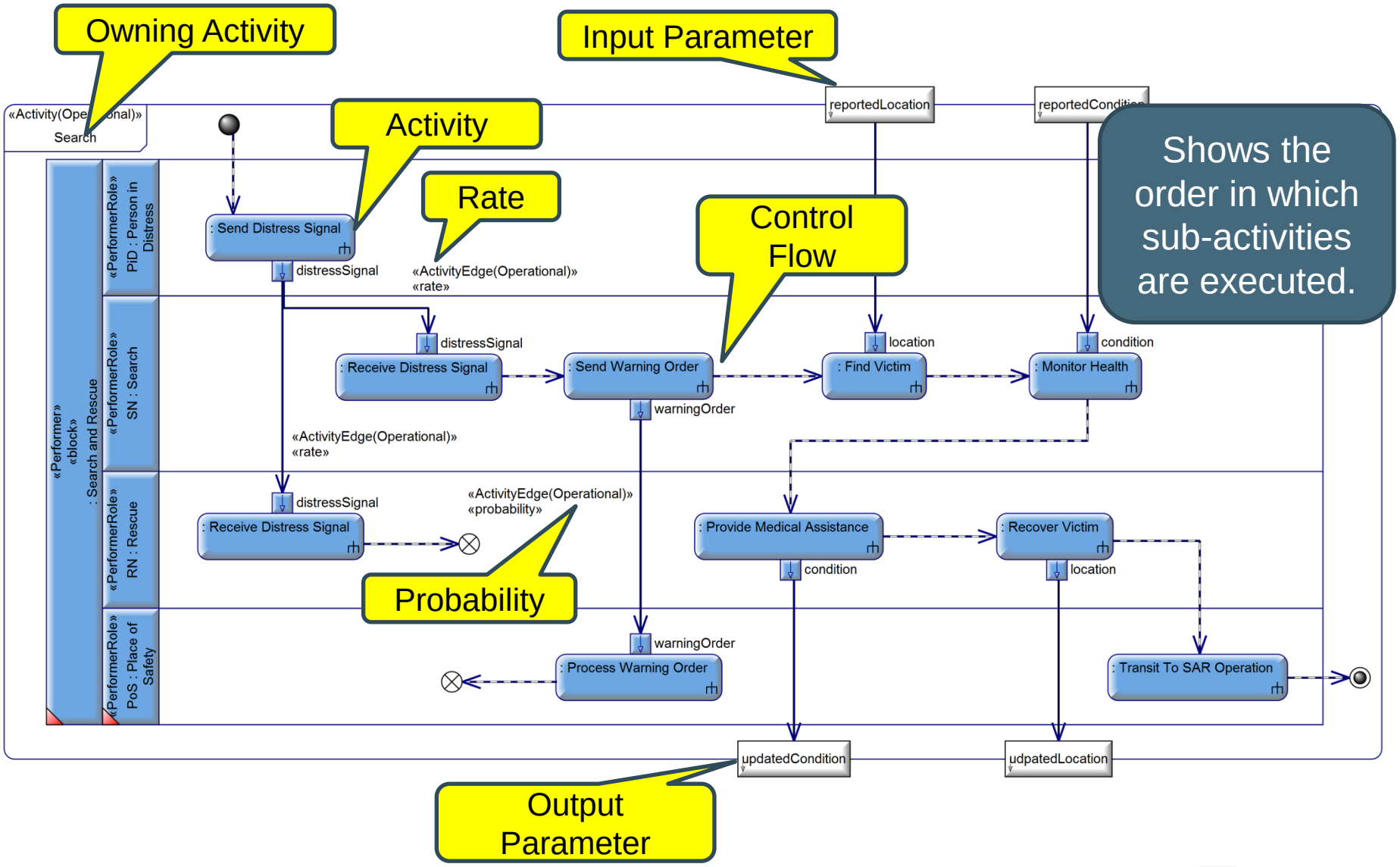
# Time Dependent Activity Sequences
# OV-5 Operational activity model

- Shows operational activities and their relationships
- Class diagrams
  - Activity hierarchies
  - Performing nodes
  - Created from an Architectural Description
- Activity diagrams
  - Activity sequence execution order
  - Swimlanes
  - Events
  - Interactions
  - Created from an operational activity
- Created from an Architectural Description

# OV-5 Activity Hierarchy Diagram



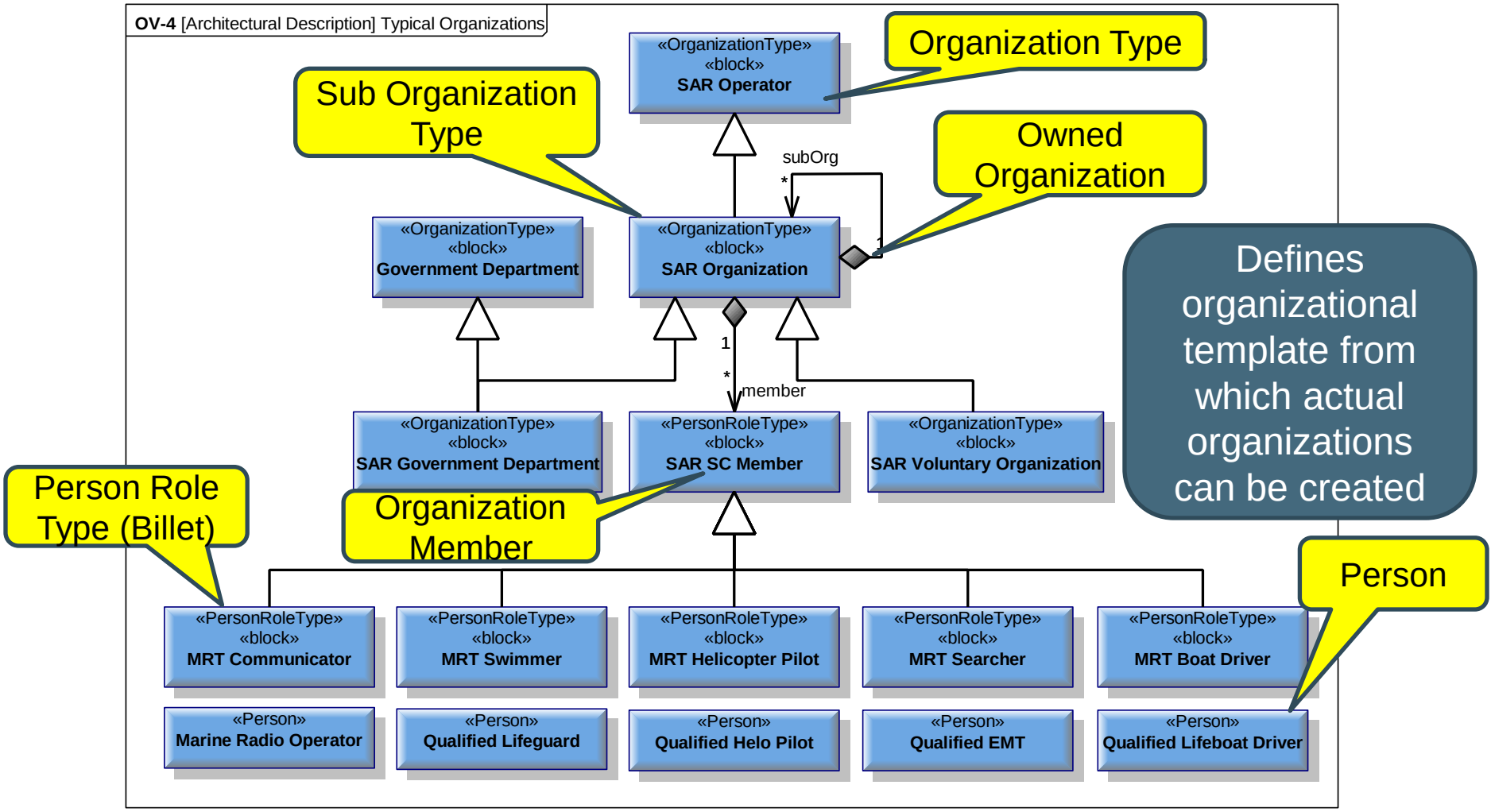
# OV-5 Search Activity Diagram



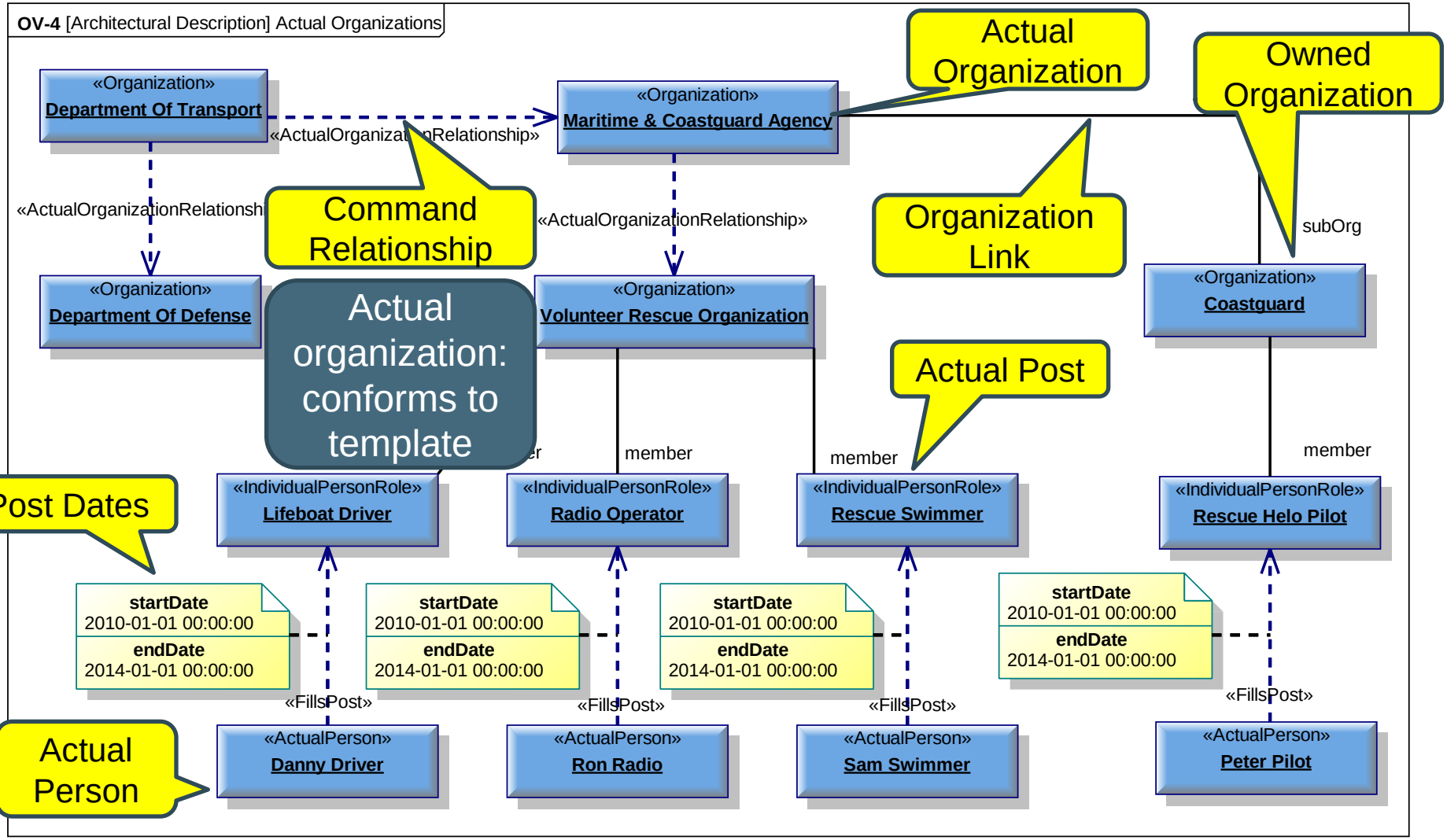
# Personnel deployment and competency assessment

- DoDAF 2.0 allows the definition of
  - People
  - Competencies/Skills
  - Competency Forecasts
  - Job Specifications
  - Competency Forecasts
  - Job Allocations
  - Job deployment report
  - Etc.

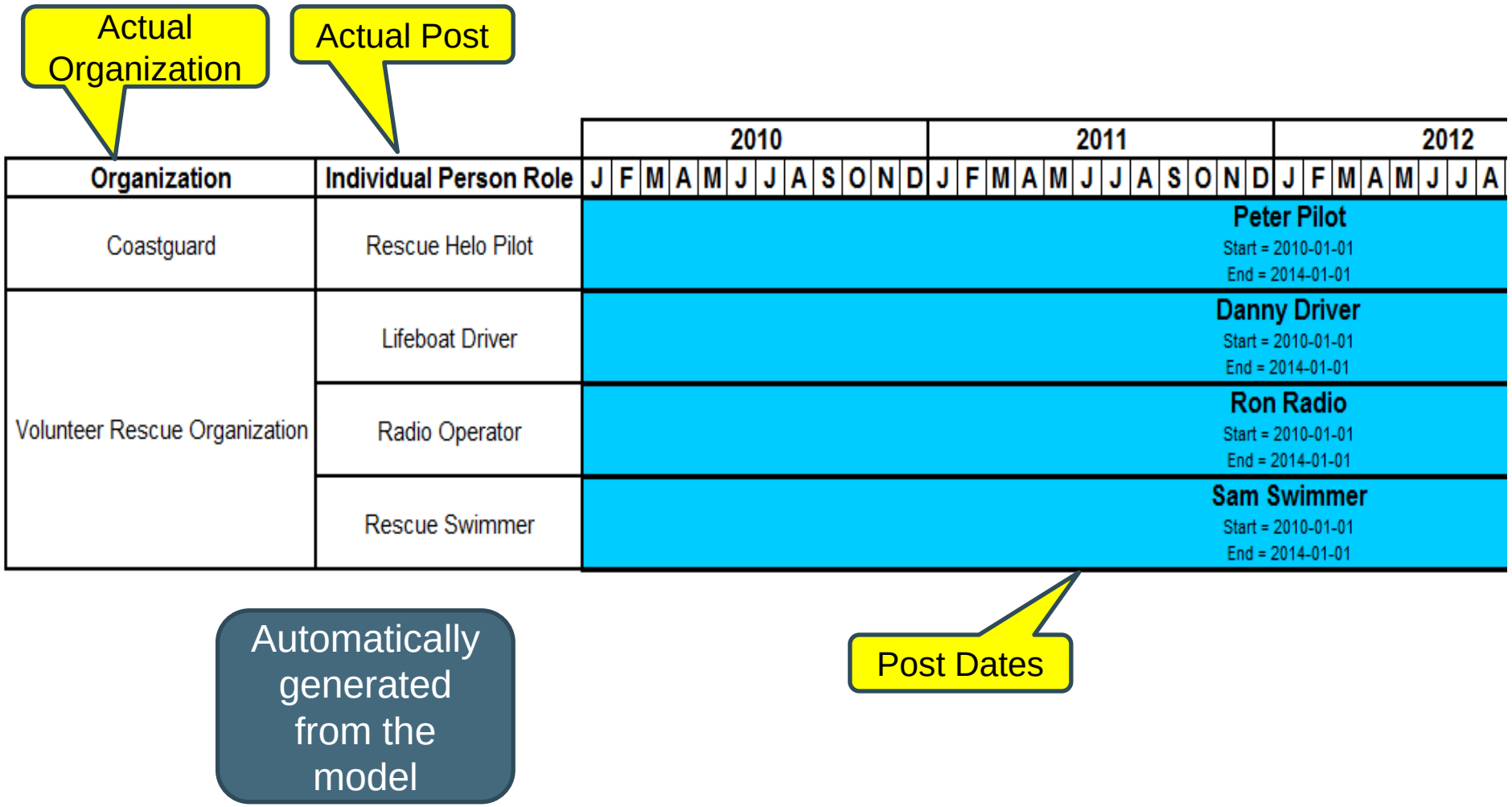
# OV-4 Organizational Template



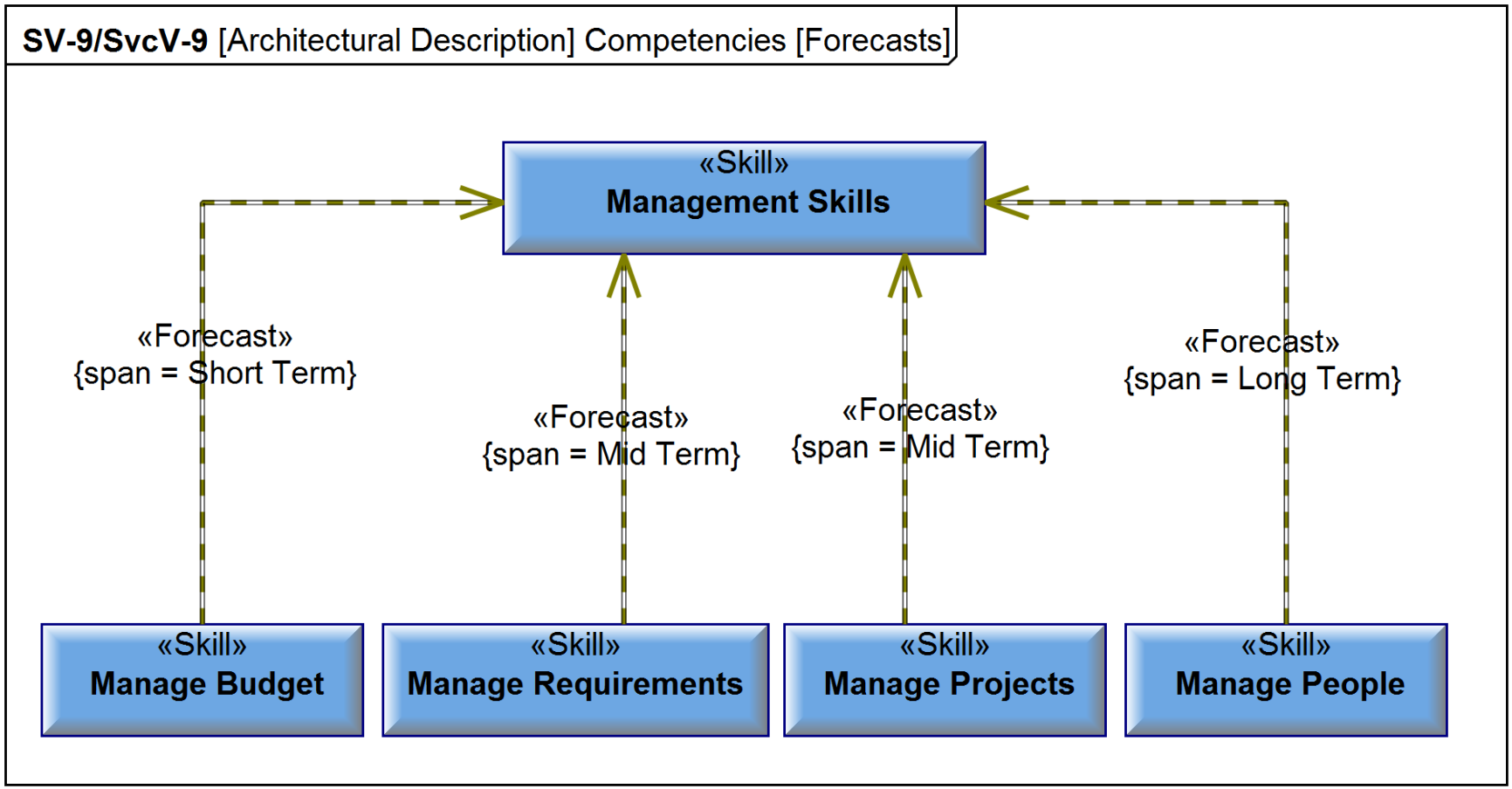
# Personnel deployment and competency assessment
# OV-4 Actual Organizations



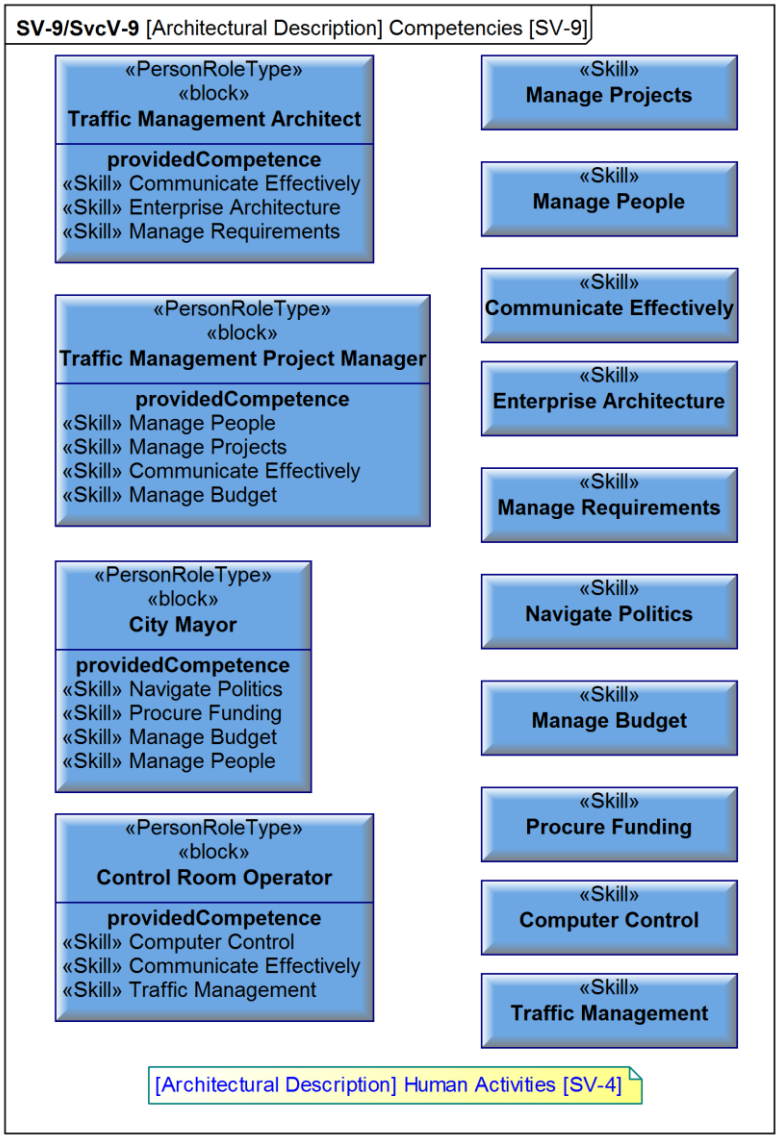
# Personnel deployment and competency assessment
# OV-4: Actual Organizations Report

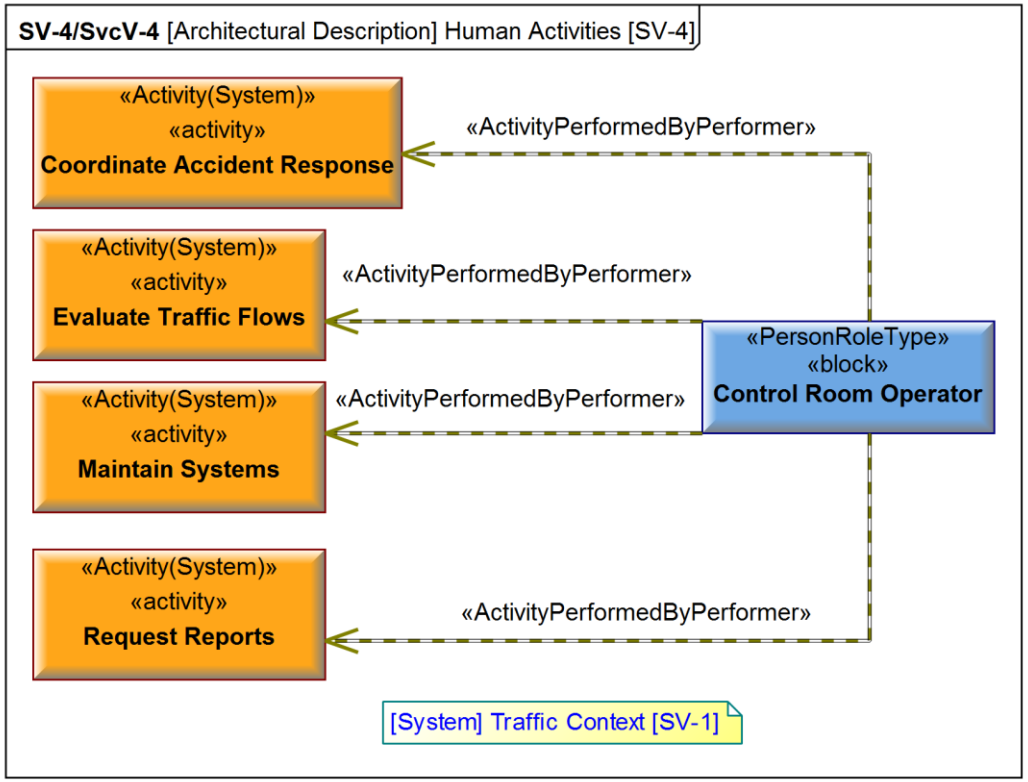


# Personnel deployment and competency assessment
# Competency Forecasts



# Personnel deployment and competency assessment
# Definition and Assignment of Competencies and Tasks



# Conclusion and Summary

- DoDAF has provided a significant improvement in the definition of time concepts
- Time is distributed throughout the architecture in the various views
- Interconnections between the views is essential to make use of these concepts.

# Questions, Comments, Discussion