

The background of the slide is composed of a grid of hexagons. Each hexagon contains a different digital or abstract pattern in shades of blue, white, and grey. Some hexagons show network-like structures with dots and lines, while others have geometric or pixelated designs. The overall effect is a modern, tech-oriented aesthetic.

# Digital Modeling Training: Systems Viewpoint Session 3

7 Aug 2023

Nataliya Shevchenko

Software Engineering Institute  
Carnegie Mellon University  
Pittsburgh, PA 15213

Copyright 2023 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at [permission@sei.cmu.edu](mailto:permission@sei.cmu.edu).

DM23-0865

# Digital Modeling Training Outline

Section 1: SysML Behavior Diagrams

Section 2: Systems Viewpoint Overview

Section 3: Demo in the model

Digital Modeling Training

# Section 1: SysML Behavior Diagrams

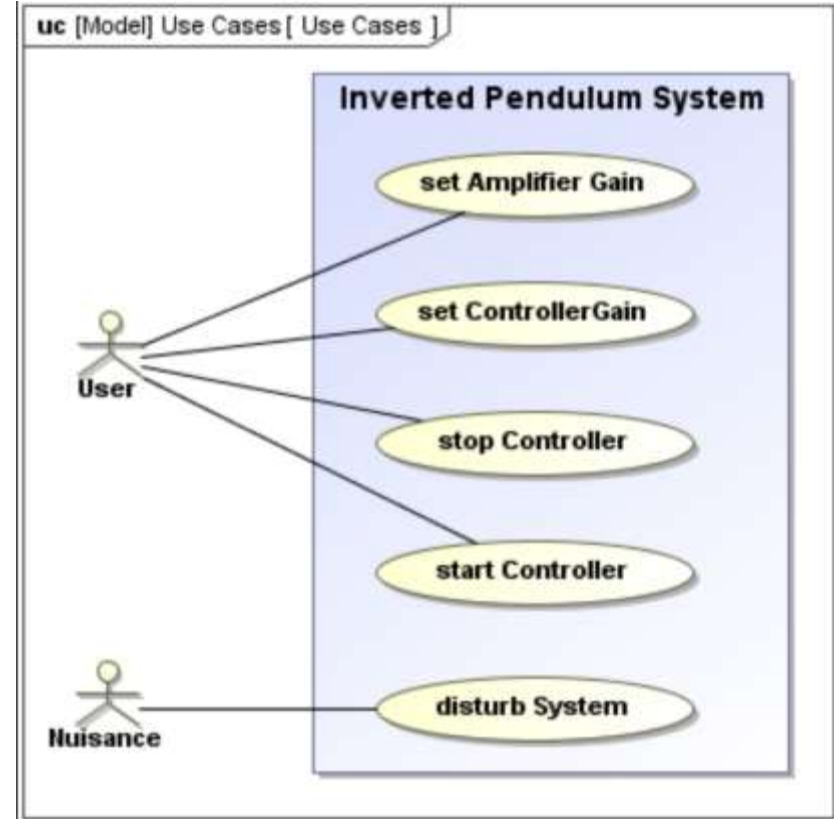


# DoDAF/SysML/UML Behavior Diagrams Types

1. Use Case Diagram (not in DoDAF)
2. Activity Diagram
3. State Machine Diagram
4. Sequence Diagram

# SysML Use Case Diagram (not in DoDAF)

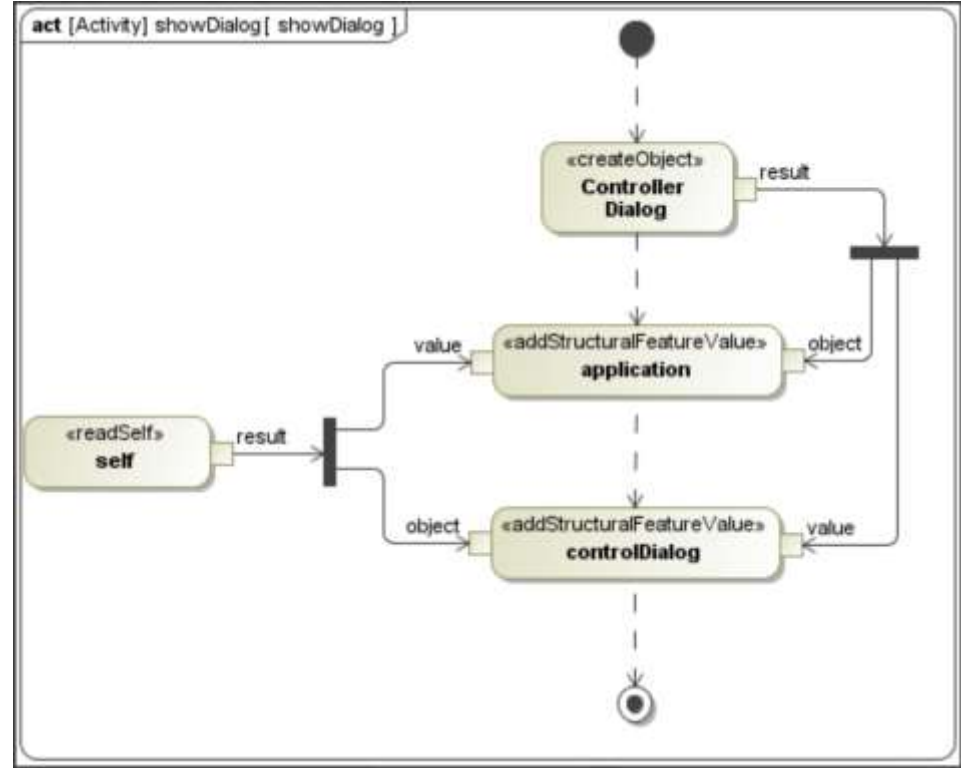
A Use Case Diagram describes the usage of a system. The associations between actors and use cases represent the communications that occur between the actors and the subjects to accomplish the functionalities associated with the use cases. The subject of a use case can be represented through a system boundary. Use Cases can be used as a tool for elicitation of requirements.



Source: <https://docs.nomagic.com/display/SYSMLP190/SysML+Use+Case+Diagram>

# SysML/UML Activity Diagram

The activity diagram describes **control**, **input**, and **output flows** among actions. It represents the system business and operational workflows. It captures actions and displays their results. Though Activity diagrams are often classified alongside interaction diagrams, they actually **focus on the flows driven by internal processes (as opposed to external events)**.

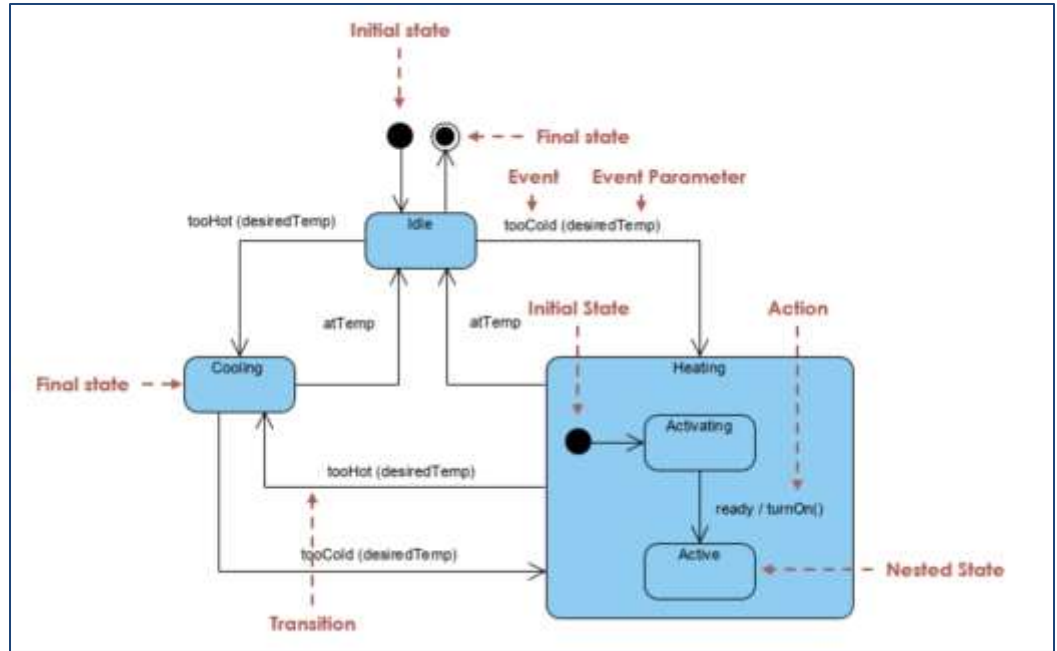


Source: <https://docs.nomagic.com/display/SYSMLP190/SysML+Activity+Diagram>

# UML State Machine Diagram

A State Machine is a behavior that specifies the **sequences of states** an object goes through during its lifetime **in response to events**, together with its **responses to those events**.

State Machine Diagram consists of **states**, **transitions**, **events**, and **activities**. They are especially important in modeling the behavior of an interface, class, or collaboration.



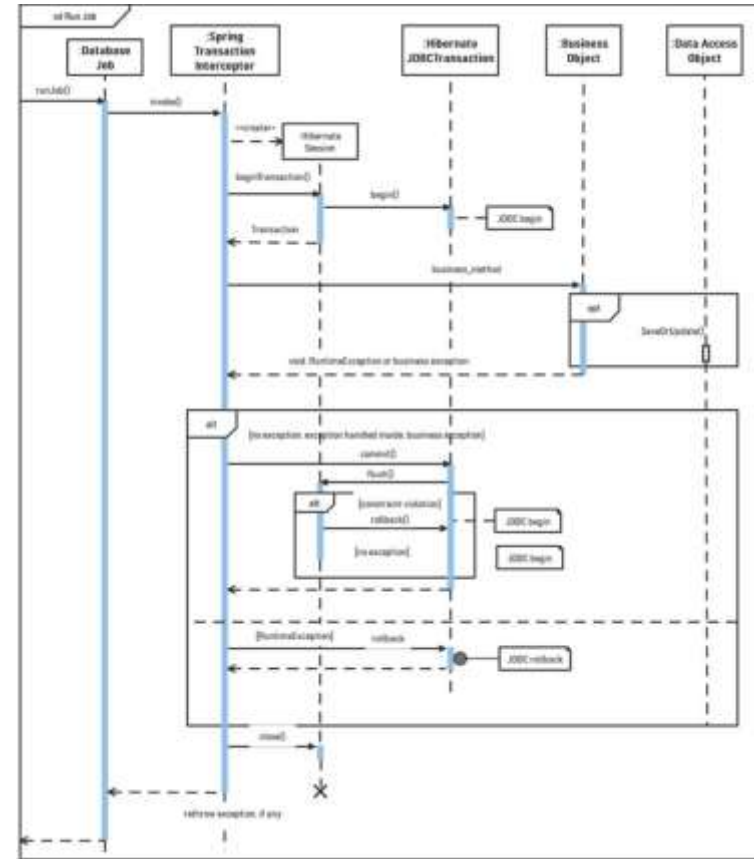
Source: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/about-state-diagrams/>



# UML Sequence Diagram

The Sequence diagram is a type of **Interaction diagram** that focuses on the **Message** interchange between various **Lifelines**.

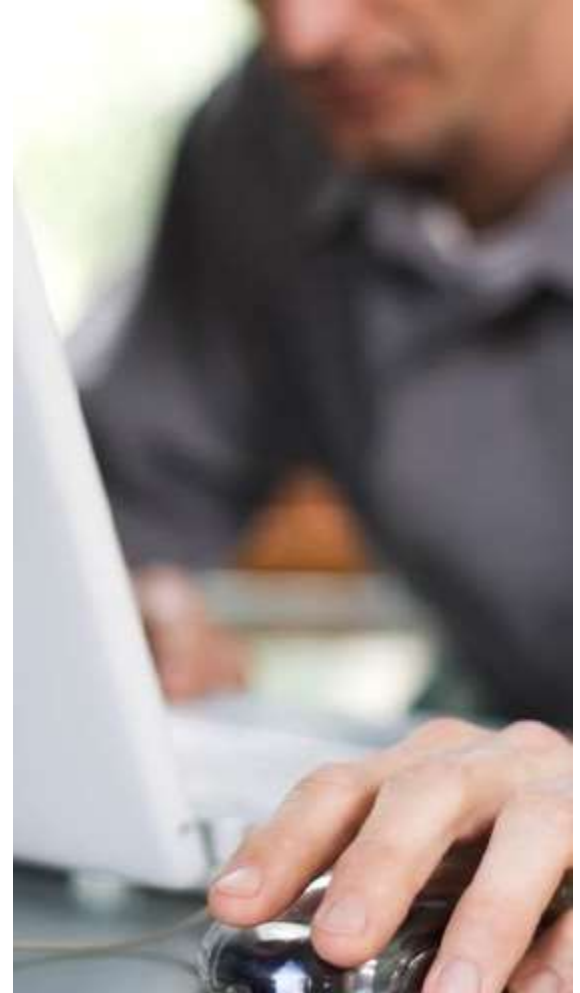
A Sequence diagram shows the interaction information with an emphasis on the **time sequence**. The diagram has two dimensions: the **vertical axis** represents **time**. And the **horizontal axis** represents the **participating objects**.



Source: <https://docs.nomagic.com/display/MD190/Sequence+diagram>, <https://www.edrawmax.com/article/sequence-diagram-example.html>

Digital Modeling Training

## Section 2: Systems Viewpoint Overview



# Systems Viewpoint

“The DoDAF-described Models within the Systems Viewpoint describes systems and interconnections providing for, or supporting, DoD functions.

DoD functions include both **warfighting** and **business functions**.

The Systems Models associate **systems** resources to the **operational** and **capability** requirements.

These systems resources support the **operational activities** and facilitate the **exchange of information**.

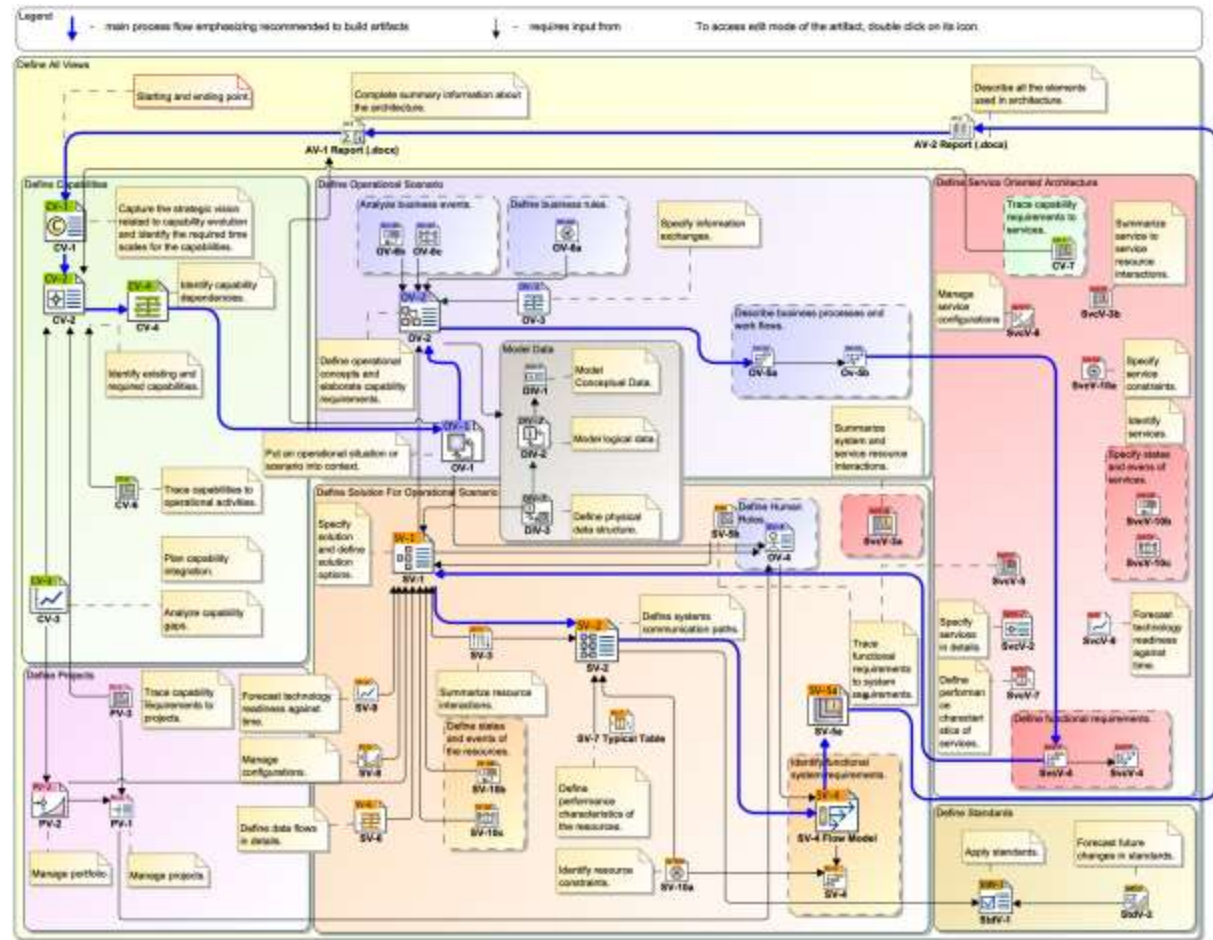
The Systems DoDAF-described Models are available for support of legacy systems.

As architectures are updated, they should transition from Systems to Services and utilize the models within the Services Viewpoint.”

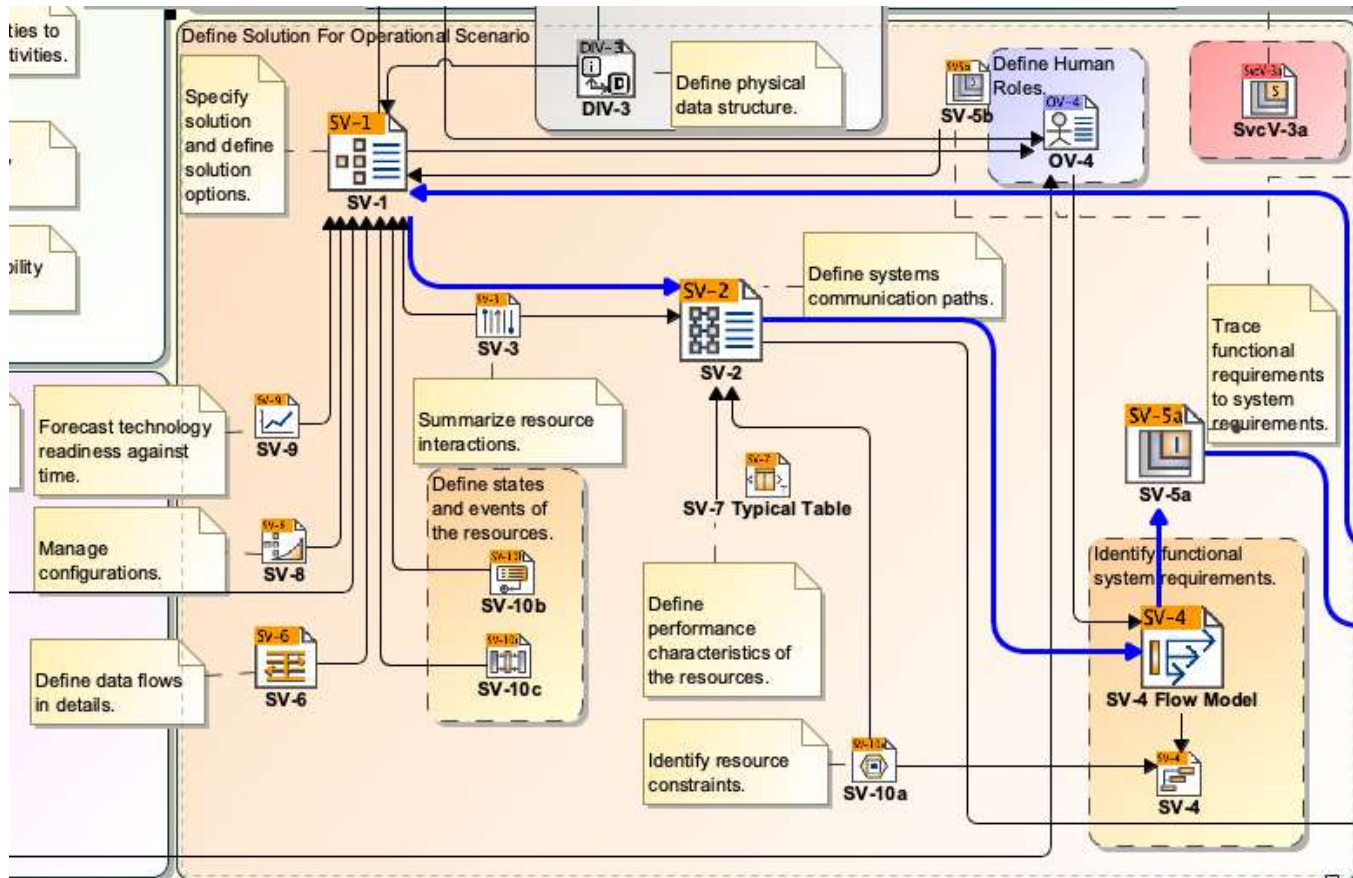
[DoDAF V2.0 Volume II]

Source: <https://docs.nomagic.com/display/UPDM2P190SP4/Systems+viewpoint>

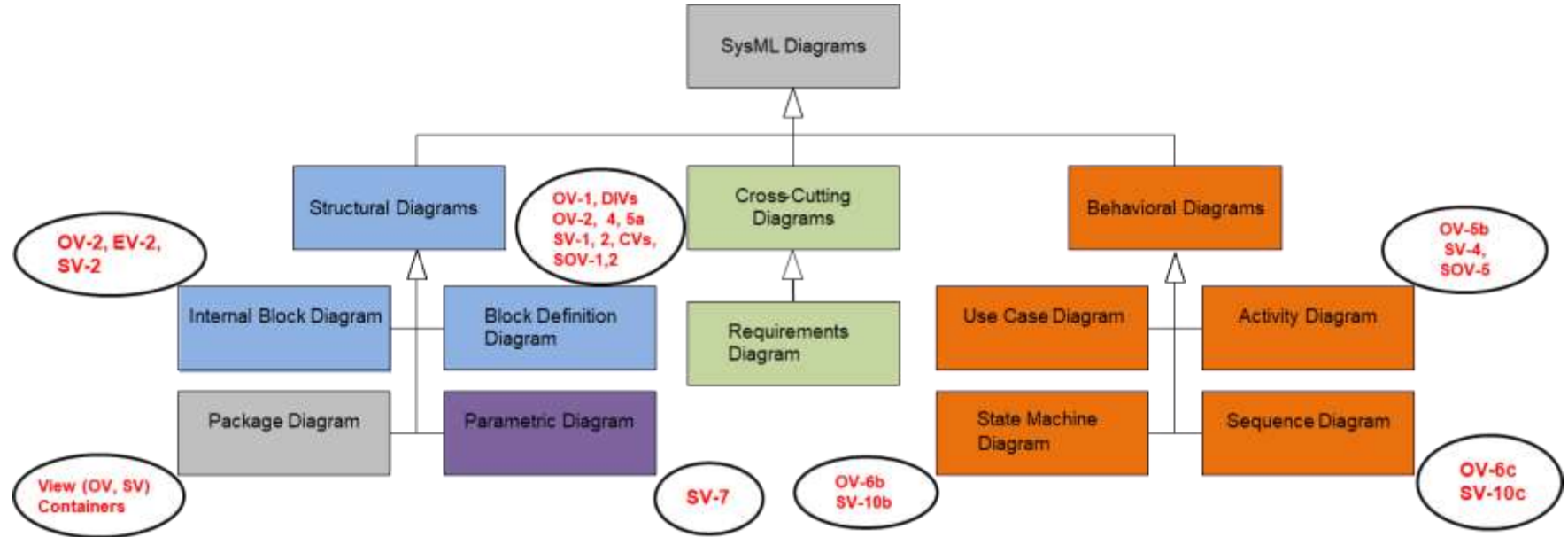
# Process Guide



# Process Guide – Systems Viewpoint



# DoDAF to SysML Mapping



Source: "Model Based Systems Engineering and Systems Modeling Language", DoDAF Plenary, January 5, 2012

# SV-1 Systems Interface Description

The SV-1 addresses the composition and interaction of Systems.

For DoDAF V2.0, the SV-1 incorporates the human elements as types of Performers - Organizations and Personnel Types.

The SV-1 links together the operational and systems architecture models by depicting how Resources are structured and interact to realize the logical architecture specified in an OV-2 Operational Resource Flow Description.

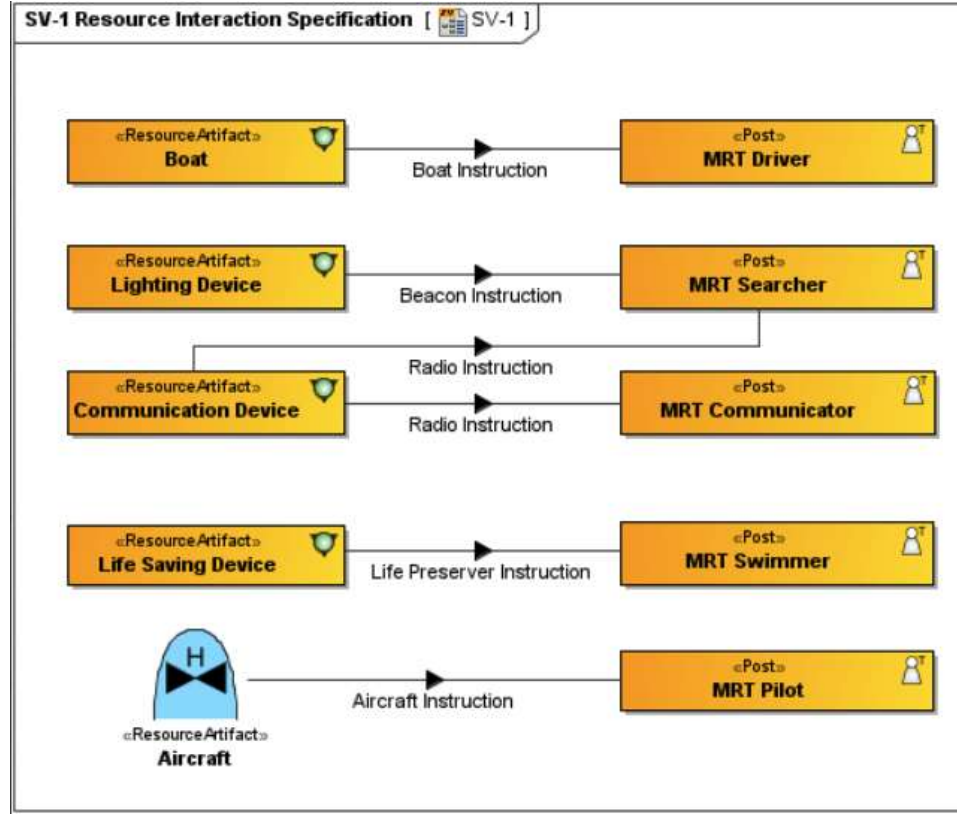
SV-1 can be represented using:

- SV-1 diagram is based on the UML Class diagram.
- SV-1 diagram is based on the UML Composite Structure diagram.
- UML Class diagram.
- UML Composite Structure Diagram.
- SysML Block Definition Diagram.
- SysML Internal Block diagram.

Source: <https://docs.nomagic.com/display/UPDM2P190SP4/SV-1+Systems+Interface+Description>



# SV-1 Diagram



Source: <https://docs.nomagic.com/display/UPDM2P1905P4/SV-1+Systems+Interface+Description>



# SV-2 Systems Resource Flow Description

SV-2 specifies the System Resource Flows between Systems and may also list the protocol stacks used in connections.

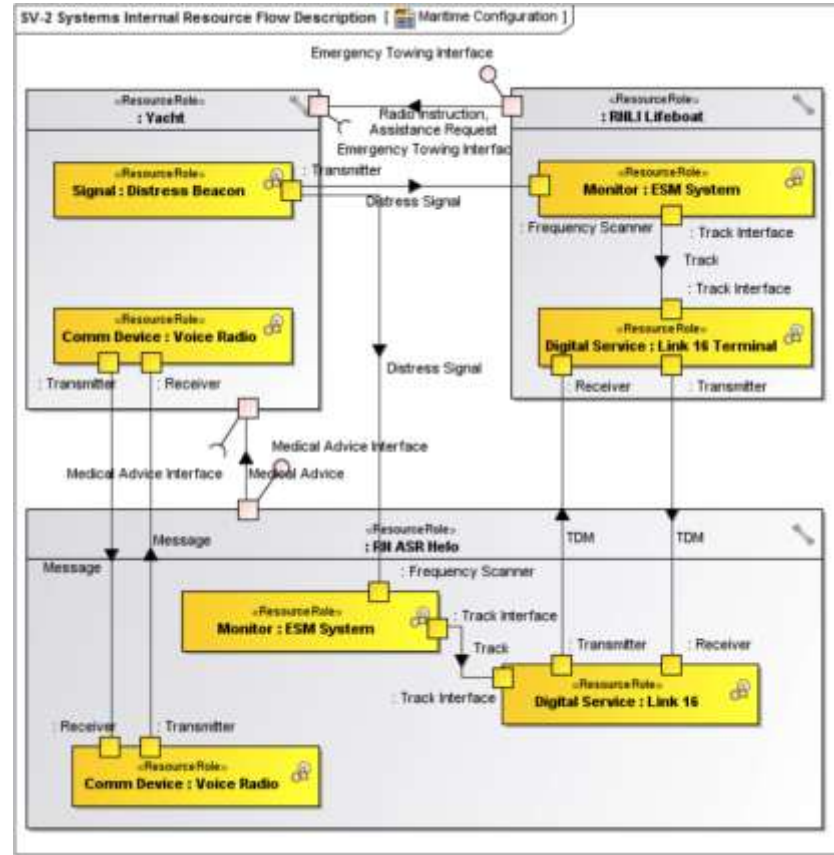
SV-2 DoDAF-described Model is used to give a precise specification of a connection between Systems. This may be an existing connection or a specification for a connection that is to be made.

SV-2 can be represented using:

- SV-2 diagram is based on the UML Class diagram.
- SV-2 diagram is based on the UML Composite Structure diagram.
- UML Class Diagram.
- UML Composite Structure Diagram.
- SysML Block Definition Diagram.
- SysML Internal Block Diagram.

Source: <https://docs.nomagic.com/display/UPDM2P1905P4/SV-2+Systems+Internal+Resource+Flow+Description>

# SV-2 Diagram



Source: <https://docs.nomagic.com/display/UPDM2P190SP4/SV-2+Systems+Internal+Resource+Flow+Description>

# SV-3 Systems-Systems Matrix

SV-3 enables a quick overview of all the system resource interactions specified in one or more SV-1 Systems Interface Description models.

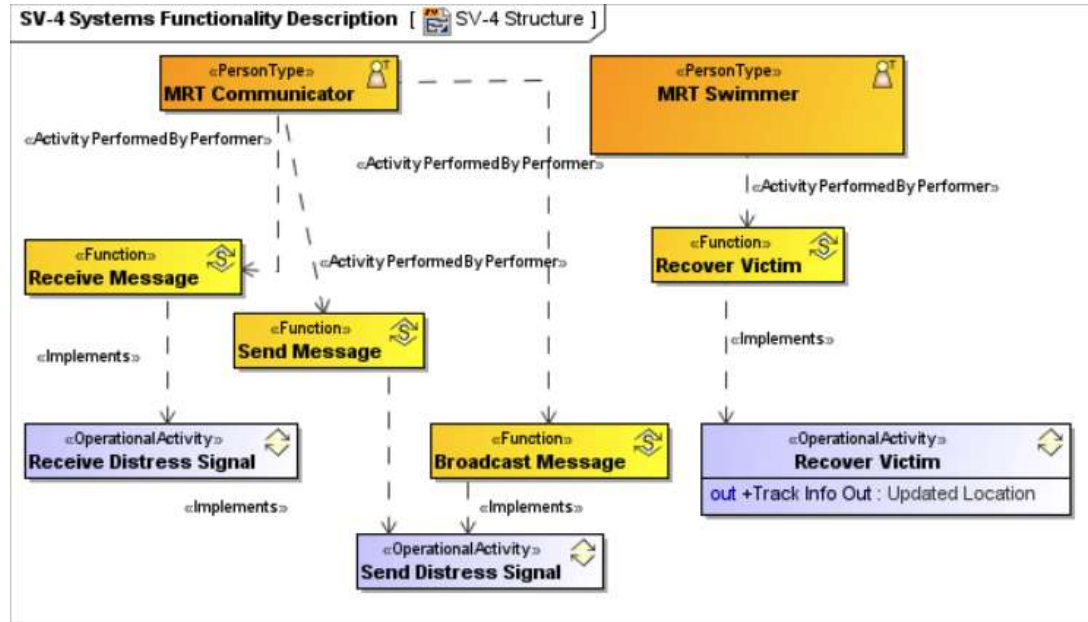
	Distress Beacon	ESM System	Frequency Scanner	Link 16	Link 16 Terminal	RN ASR Helo	RNLI Lifeboat	Voice Radio	Yacht
Distress Beacon									
ESM System									
Frequency Scanner									
Link 16									
Link 16 Terminal									
RN ASR Helo									
RNLI Lifeboat									
Voice Radio									
Yacht									

Source: <https://docs.nomagic.com/display/UPDM2P1905P4/SV-3+Systems-Systems+Matrix>

# SV-4 Systems Function Description

The SV-4 addresses human and system functionality.

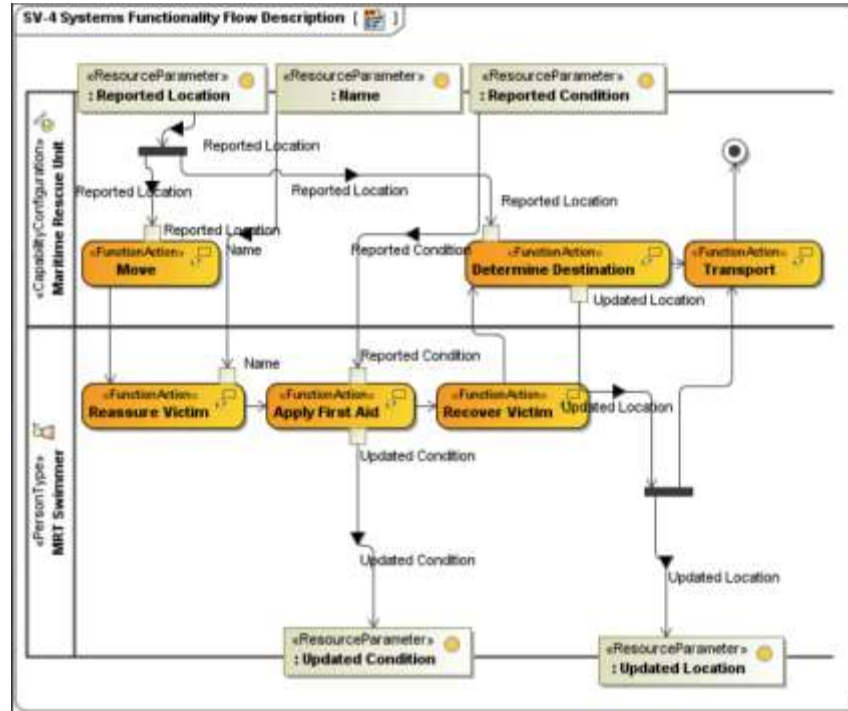
A SV-4 diagram for System Function hierarchies. This diagram is based on the UML Class diagram.



Source: <https://docs.nomagic.com/display/UPDM2P190SP4/SV-4+Systems+Functionality+Description>

# SV-4 Systems Function Flow Description

An SV-4 diagram for System Function flows. This diagram is based on the UML Activity diagram.



Source: <https://docs.nomagic.com/display/UPDM2P1905P4/SV-4+Systems+Functionality+Description>

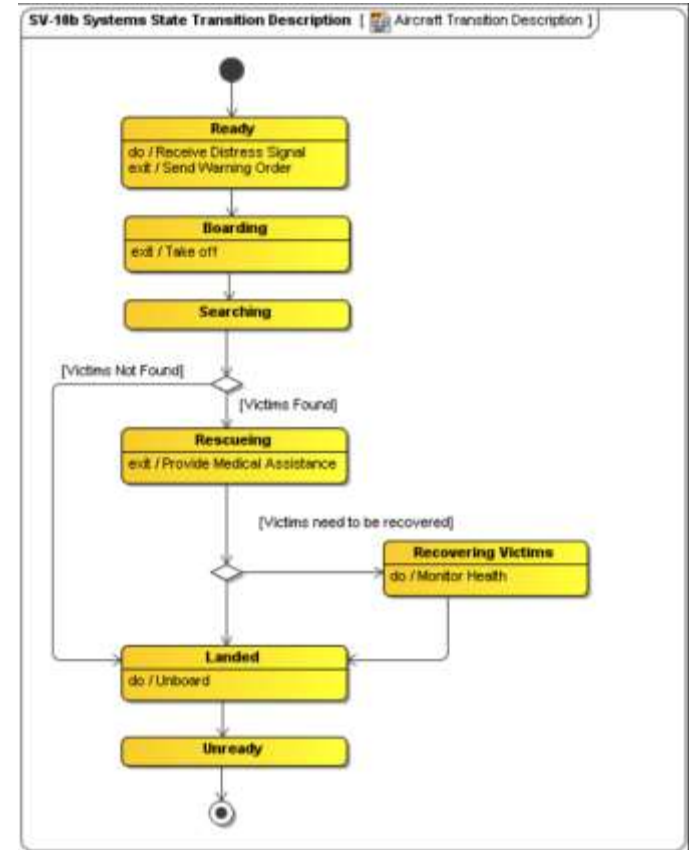
# SV-10b Systems State Transition Description

The SV-10b is a graphical method of describing a resource (or system function) response to various events by changing its state.

The diagram basically represents the sets of events to which the resources in the Activities respond (by taking an action to move to a new state) as a function of its current state.

Each transition specifies an event and an action.

SV-10b can be represented using a UML State Machine diagram.



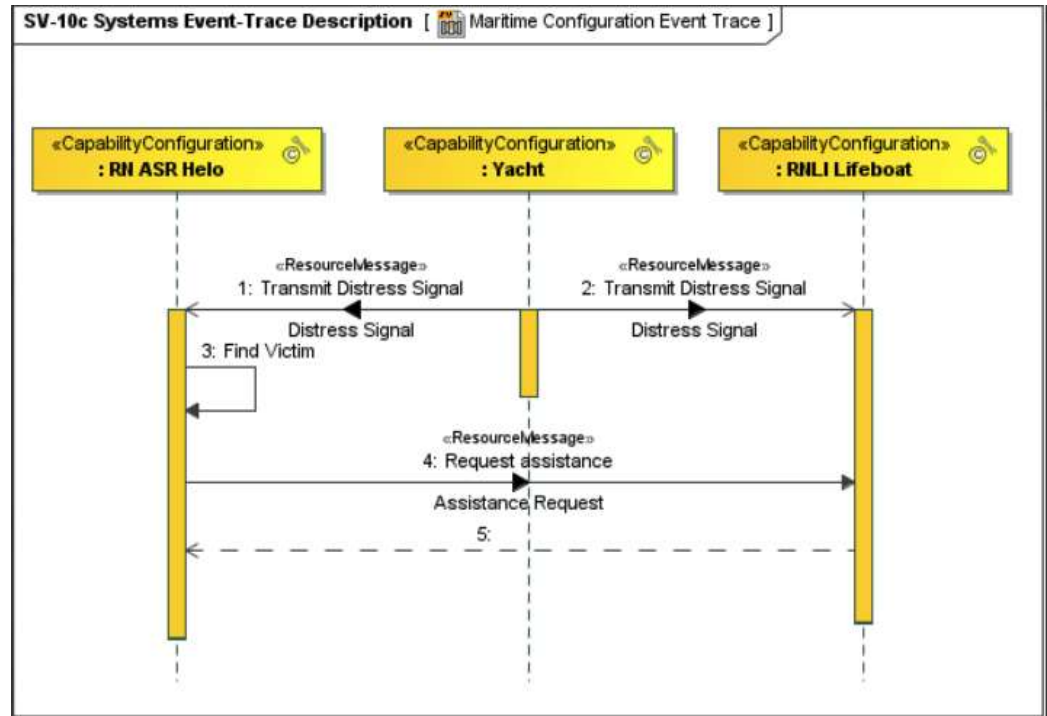
Source: <https://docs.nomagic.com/display/UPDM2P190SP4/SV-10b+Systems+State+Transition+Description>

# SV-10c Systems Event-Based Description

The SV-10c provides a **time-ordered** examination of the interactions between functional resources.

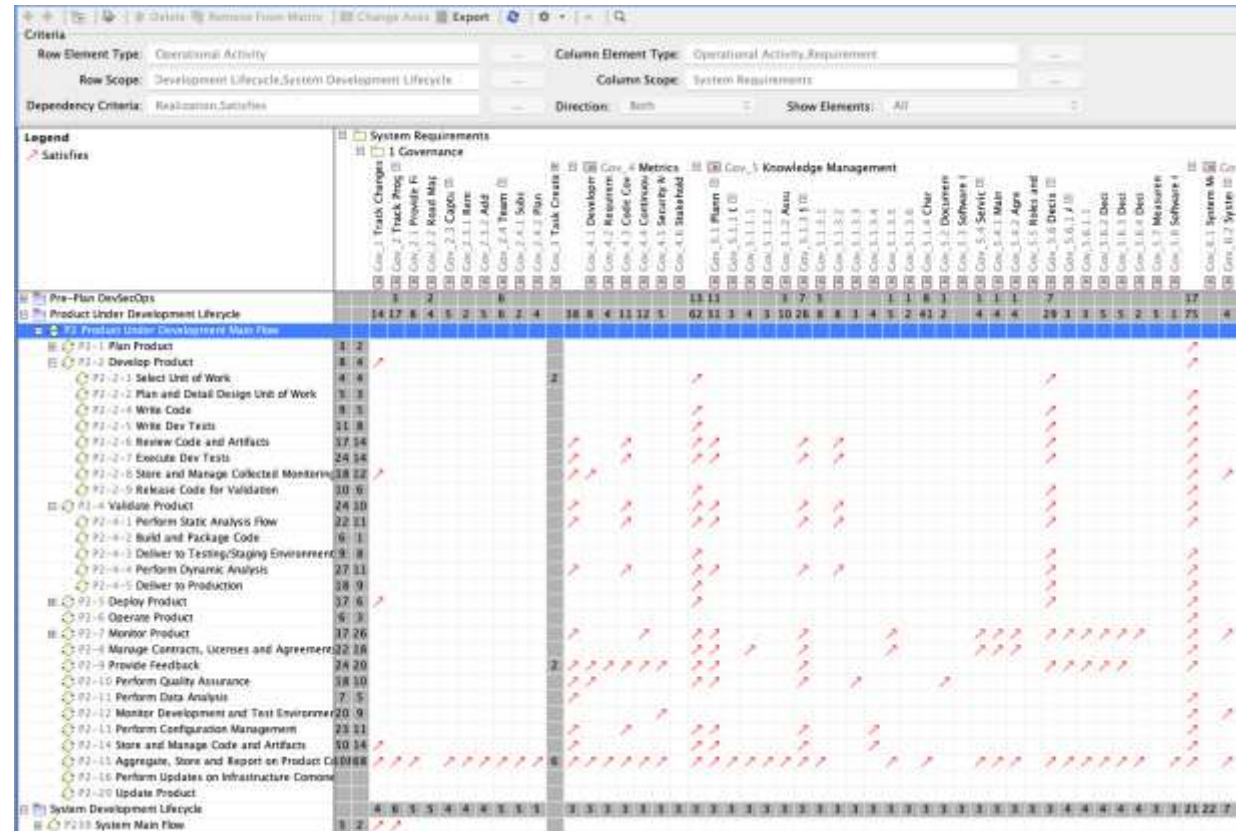
Each event-trace diagram should have an accompanying description that defines the particular scenario or situation.

SV-10c can be represented using a UML Sequence Diagram.



Source: <https://docs.nomagic.com/display/UPDM2P1905P4/SV-10c+Systems+Event-Trace+Description>

# Requirements Traceability





# Block Interactions

← → 📄 📁 🗑️ Delete 🗑️ Remove From Matrix 📄 Change Axes 📄 Export 🔄 ⚙️ ⏏️ 🔍

**Criteria**

Row Element Type: Block ... Column Element Type: Block

Row Scope: Blocks {xy} ... Column Scope: Blocks

Dependency Criteria: Item Flow ... Direction: Both ⬆️ ⬆️ Show

**Legend**

↗️ Item Flow

☐ Blocks

Coffee Container  
Coffee Machine  
Controller  
Water Heater  
Water Tank

☐ Blocks	2	2		
☐ Coffee Container	2	↗️		
☐ Coffee Machine				
☐ Controller	2	↗️		
☐ Water Heater				
☐ Water Tank				

# Behavior Allocation

**Criteria**

Row Element Type:  ... Column Element Type:   
Row Scope:  (xy) ... Column Scope:   
Dependency Criteria:  ... Direction:  Show Elements

**Legend**  
 Allocate

		Behavior	Check Water(context)	Initiate Check Water	Initiate Start(context)	Push Button
<b>Blocks</b>			2	2	2	
Coffee Container						
Coffee Machine						
Controller	4					
Water Heater						
Water Tank	2					

Digital Modeling Training

## Section 3: Demo in the model

