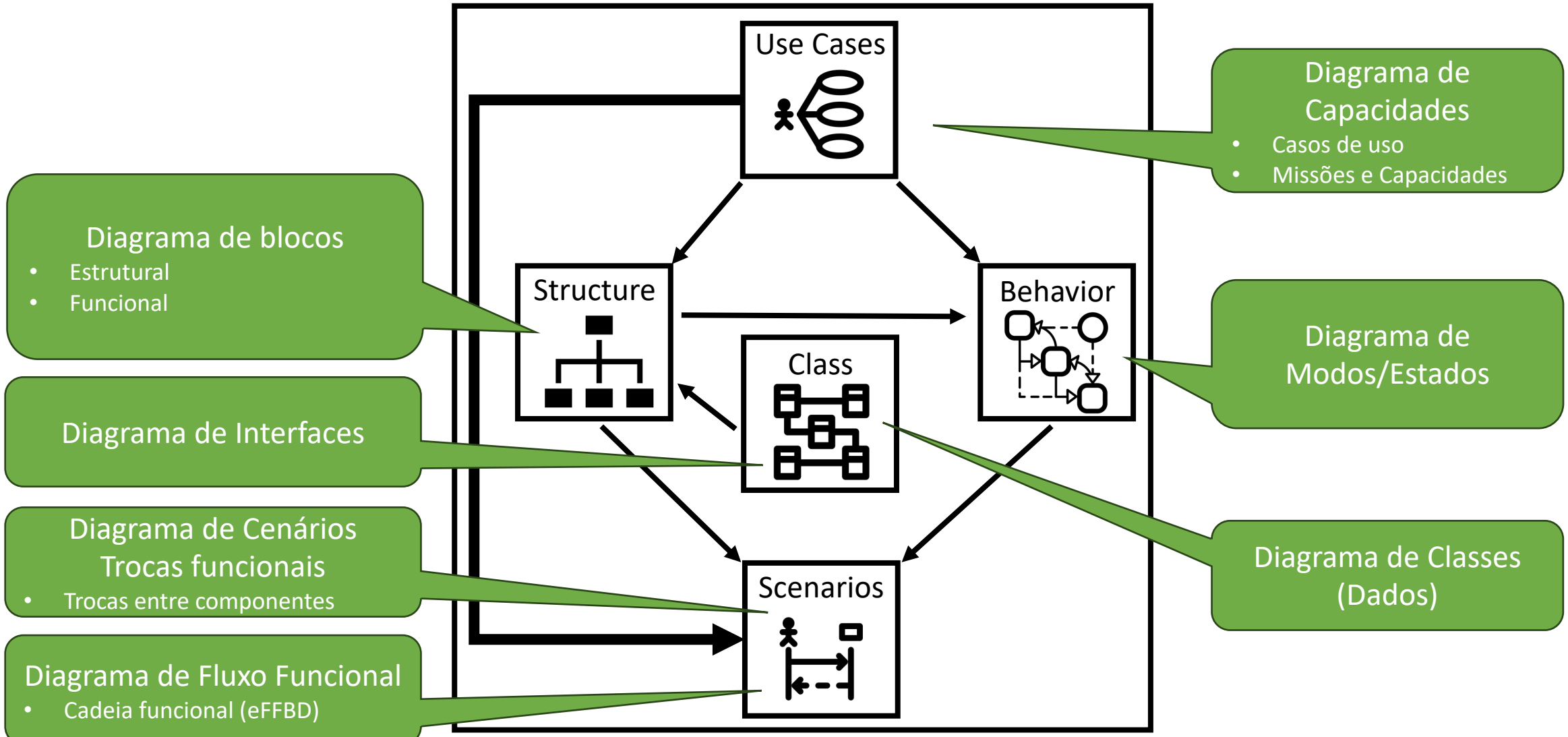
# REVISÃO


Use Cases
Structure
Behavior
Class
Scenarios
Diagrama de Capacidades
Casos de uso
Missões e Capacidades
Diagrama de blocos
Estrutural
Funcional
Diagrama de Modos/Estados
Diagrama de Interfaces
Diagrama de Cenários Trocas funcionais
Trocas entre componentes
Diagrama de Classes (Dados)
Diagrama de Fluxo Funcional
Cadeia funcional (eFFBD)

# REVISÃO

- Stakeholders
- CONOPs
  - *Validação*
- Requisitos
  - *Verificação*
- Ciclo de Vida
- Função
- Arquitetura
  - Funcional / Hierárquica
  - Coesão e Acoplamento
- Árvore de Produto / Decomposição do Sistema
- Dualidade Comportamento/Estrutura

**Motivador coorporativo: Sistema de transporte aéreo (pessoas/coisas) de curta/média distância em área urbana. Cada grupo vai fazer uma variação.**

**Grupo 01 – Transportador de carga (hub-hub)**
**Grupo 02 – Transportador de suprimento hospitalar.**
**Grupo 03 – Transportador de pessoas.**
**Grupo 04 – Transportador de pessoas em emergencia (Ambulância / bombeiro)**
**Grupo 05 – Transportador de carga (hub-cliente)**

**Todos os grupos vão compartilhar e harmonizar a gestão de trafego aéreo urbana (UAM - Urban Air Mobility)**
**Todos os grupos devem buscar os conceitos de tecnologias verdes.**

Aplicação e Retórica

| SEMANA | TEORIA | INDIVIDUAL | PESO | GRUPO | PESO |
|---|---|---|---|---|---|
| **9** | MBSE: Método Arcadia | AI-09 - Resumo sobre o documento da SAE contextualizado ao projeto | 5% | AG-09 - | 0% |
| *06-May* | | | | | |
| 10-May | Contexto: SAE | | | | |
| **10** | Retórica: **Análise do Contexto** | AI-10 - Re[illegible]bre o SORA contextu[illegible]projeto | 10% | AG-10 - Preparar a Análise de Contexto | 10% |
| *13-May* | | | | | |
| 17-May | | | | | |
| **11** | **Apresentação da Análise do Contexto (15min)** | AI-11 - R[illegible]re o CORUS contextual[illegible]projeto | 10% | AG-11 - Preparar a Intervenção Sistêmica | 10% |
| *20-May* | Retórica: **Intervenção Sistêmica** | | | | |
| 24-May | | | | | |
| **12*** | **Apresentação da Intervenção Sistêmica** | AI-12 - Resumo sobre a DCA-400-6 na parte de concepção contextualizado ao projeto. | 10% | AG-12 - Escrever os requisitos do nível de sistema e gerar o documento de requisitos. | 20% |
| *27-May* | Retórica: Representando requisitos | | | | |
| 31-May | M2DOC | | | | |
| **13** | **Apresentação dos Requisitos e Geração de Doc** | AI-13 - | 0% | AG-13 - Preparar a decomposição funcional e requisitos de subsistemas | 30% |
| *03-Jun* | Retórica: **Arquitetura Conceitual** | | | | |
| 07-Jun | | | | | |
| **14** | **Apresentação da Arquitetura Coneitual** | AI-14 - | 0% | AG-14 - Preparar a solução a ser construída, explicando como vai ser instrumento. | 30% |
| *10-Jun* | Retórica: **Arquitetura Concreta** | | | | |
| 14-Jun | | | | | |
| **15** | **Apresentação da Arquitetura Concreta** | AI-15 - | 0% | AG-15 - | 0% |
| *17-Jun* | Visita ao Laboratório e Encerramento do Curso e discussão sobre P2 | | | | |
| 21-Jun | | | | | |
| **16** | P2 | AI-16(P2) - Simulado ASEP (Pres/Consulta - sem chatGPT) | 70% | AG-16 - | 0% |
| *24-Jun* | | | | | |
| 28-Jun | | | | | |
| | | | **105%** | | **100%** |
| **EXAME** | **Grupo: Escrita de artigo (min 6pgs / max 10pgs) e Apresentação Gravada (max. 20min), relatando o case do seu grupo no padrão do SIGE.** | | | | **100%** |
| *01-Jul* | | | | | |
| *12-Jul* | | | | | |

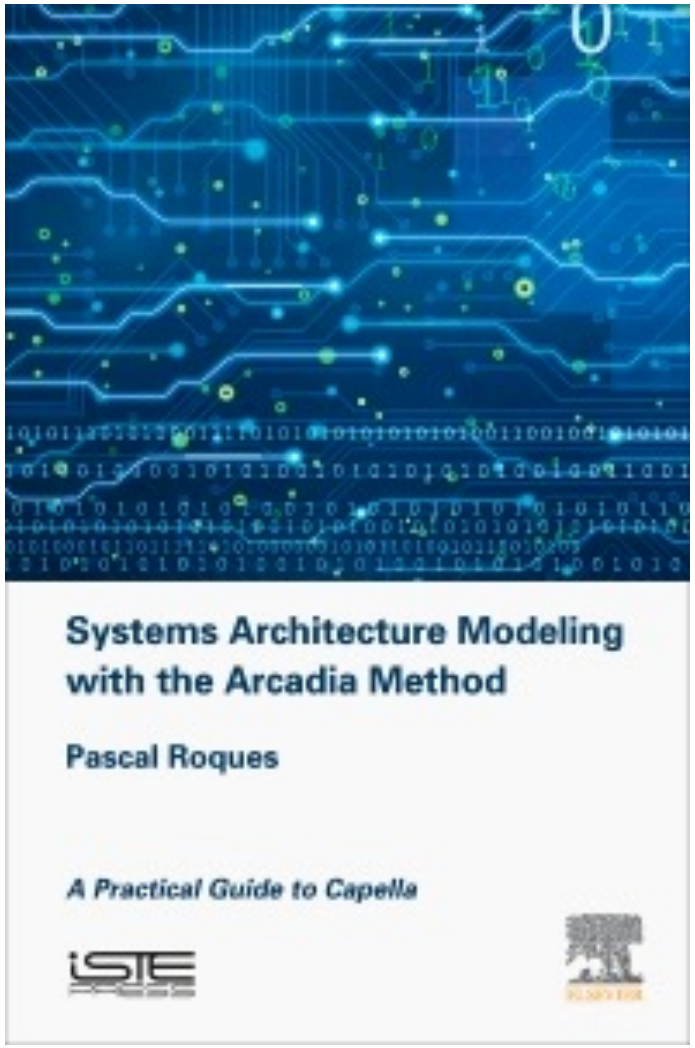
# Arcadia e Análise do Contexto

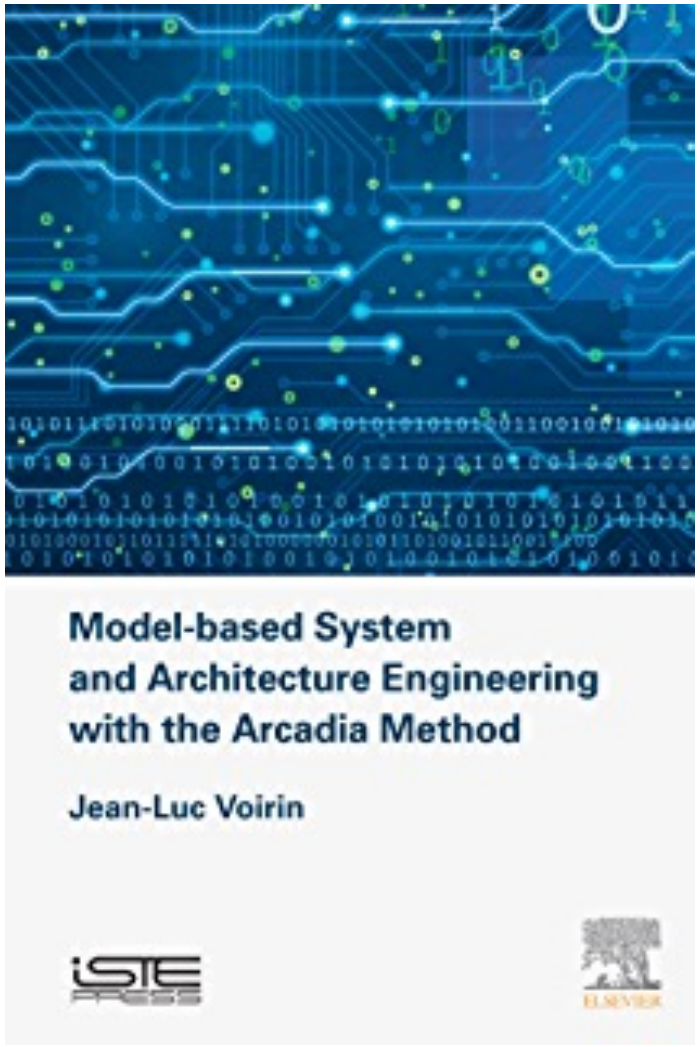
[2024]

Prof. Dr. Christopher S. Cerqueira

# MÉTODO ARCADIA

# PRINCIPAIS REFERÊNCIAS



- Os engenheiros de sistemas fazem uso de técnicas de modelagem há muito tempo.
- **A técnica de análise estruturada e projeto (Structured Analysis and Design Techniques SADT) e a análise estruturada em tempo real (Structured Analysis for Real Time SA/RT) são algumas das mais conhecidas e datam da década de 1980**.
- Existem muitas outras abordagens baseadas em redes de Petri ou máquinas de estados finitos.
- No entanto, também são **limitadas por sua abrangência e expressividade**, bem como pela **dificuldade em integrá-las a outros formalismos e exigências**.

- **Infelizmente, na prática, tem sido demonstrado que a filiação da linguagem SysML à UML muitas vezes leva a dificuldades em termos de compreensão e uso para engenheiros de sistemas que não são também cientistas da computação.**

- Esta é a razão que levou Thales a definir o método ARCADIA, estruturada por Jean-Luc Voirin, juntamente com seu formalismo subjacente, para suas próprias necessidades.

https://www.linkedin.com/in/jean-luc-voirin-8087a9155/


https://www.youtube.com/watch?v=NIFayQAueso

THALES

# O espírito de Arcadia e Capella em 8 minutos

Content: Stéphane Bonnet
Thales

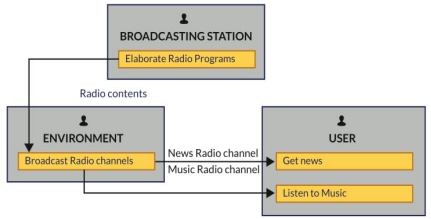
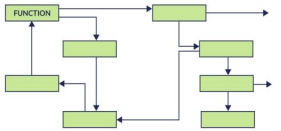
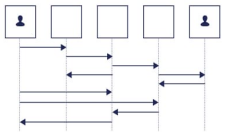
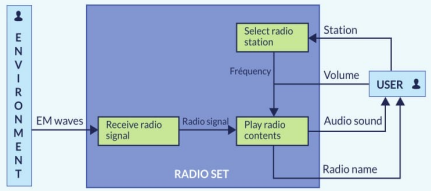
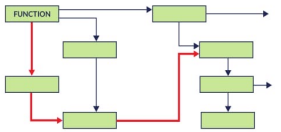
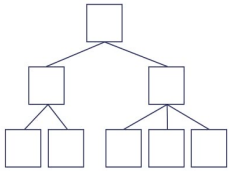
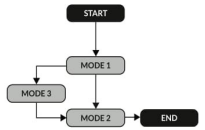
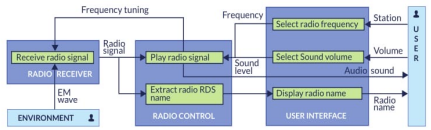
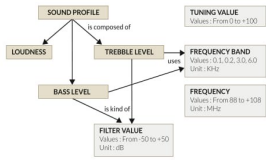
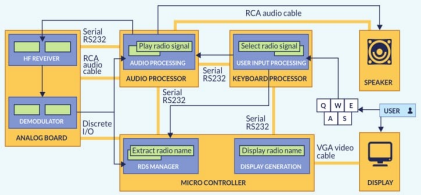
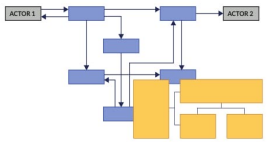
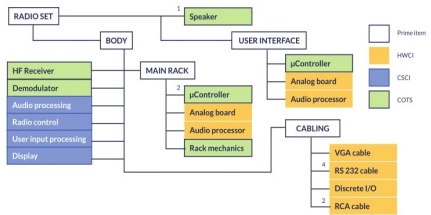
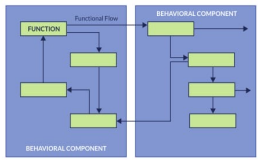
www.thalesgroup.com

https://www.youtube.com/watch?v=2fveJ7nwiuU

# PRINCÍPIOS FUNDADORES

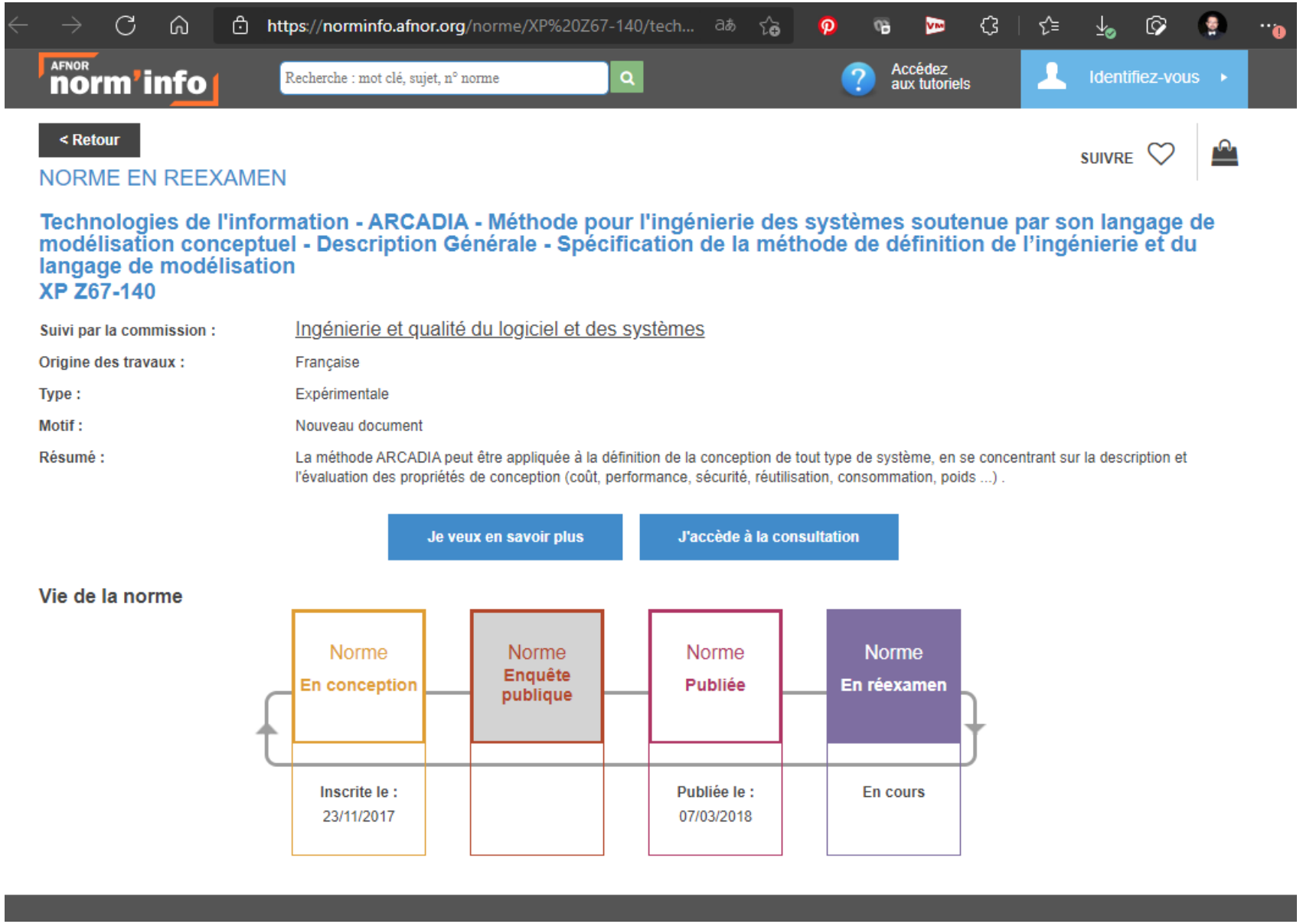
- Todas as partes interessadas em engenharia **compartilham a mesma metodologia, as mesmas informações, a mesma descrição da necessidade e do produto** na forma de um modelo compartilhado;
- Cada tipo especializado de engenharia (por exemplo, segurança, desempenho, custo e massa) é **formalizado como um "ponto de vista"** em relação aos requisitos a partir dos quais a arquitetura proposta é então verificada;
- As regras para **a verificação antecipada da arquitetura** são estabelecidas, a fim de verificar a arquitetura o mais rápido possível;
- A **co-engenharia entre os diferentes níveis de engenharia** é apoiada pela **elaboração conjunta de modelos**, sendo os modelos dos diferentes níveis e especialidades deduzidos/validados/vinculados um ao outro.


Architect
Specialty engineering: safety, perf, security, ...
Functions
Safety
Performance
Human Factors
Security
IMQ, Product Line, Cost...
ViewPoints
Product Line Manager, Etc.
Evaluation Rules
Sub-contractors
Solution Architecture

| | METHOD STEPS | TASKS | SAMPLE MODEL | CONCEPTS | DESCRIPTION MEANS |
|---|---|---|---|---|---|
| NEED | **Customer Operational Need Analysis**<br>What the users of the system need to accomplish | ✓ Define operational capabilities<br>✓ Perform an operational need analysis |  | - Operational capabilities<br>- Actors, operational entities<br>- Actor activities<br>- Interactions between activities & actors<br>- Information used in activities & interactions<br>- Operational processes chaining activities<br>- Scenarios for dynamic behaviour |  <br>**Dataflow**: functions, op. activities interactions & exchanges<br>**Scenarios**: actors, system, components interactions & exchanges<br>  |
| | **System/ SW/HW Need Analysis**<br>What the system has to accomplish for the Users | ✓ Perform a capability trade-off analysis<br>✓ Perform a functional and non-functional analysis<br>✓ Formalise and consolidate requirements |  | - Actors and system, capabilities<br>- Functions of system & actors<br>- Dataflow exchanges between functions<br>- Functional chains traversing dataflow<br>- Information used in functions & exchanges, data model<br>- Scenarios for dynamic behaviour<br>- Modes & states |  <br>**Functional chains, operational processes** through functions & op. activities<br>  <br>  <br>**Modes & states** of actors, system, components<br>**Breakdown** of functions & components |
| SOLUTIONS | **Logical Architecture Design**<br>How the system will work so as to fulfil expectations | ✓ Define architecture drivers and viewpoints<br>✓ Build candidate architectural breakdowns in components<br>✓ Select best compromise architecture |  | *SAME CONCEPTS, PLUS :*<br>- Components<br>- Component ports and interfaces<br>- Exchanges between components<br>- Function allocation to components<br>- Component interface justification by functional exchanges allocation |  <br>**Data model**: dataflow & scenario contents, definition & justification of interfaces |
| | **Physical Architecture Design**<br>How the system will be developed & built | ✓ Define architectural patterns<br>✓ Consider reuse of existing assets design a physical<br>✓ Design a physical reference architecture<br>✓ Validate and check it |  | *SAME CONCEPTS, PLUS :*<br>- Behavioural components refining logical ones, and implementing functional behaviour<br>- Implementation components supplying resources for behavioural components<br>- Physical links between implementation components |  <br>**Component wiring:** all kinds of components |
| | **Development Contracts**<br>What is expected from each designer/ sub-contractor | ✓ Define a components IVVQ strategy<br>✓ Define & enforce a PBS and component integration contract |  | - Configuration items tree<br>- Parts numbers, quantities<br>- Development contract (expected behaviour, interfaces, scenarios, resource consumption, non-functional properties…) |  <br>**Allocation** of op.activities to actors, of functions to components, of behav.components to impl.components, of dataflows to interfaces, of elements to configuration items |

# XP Z67-140 - ARCADIA

[Norme XP Z67-140 (afnor.org)]



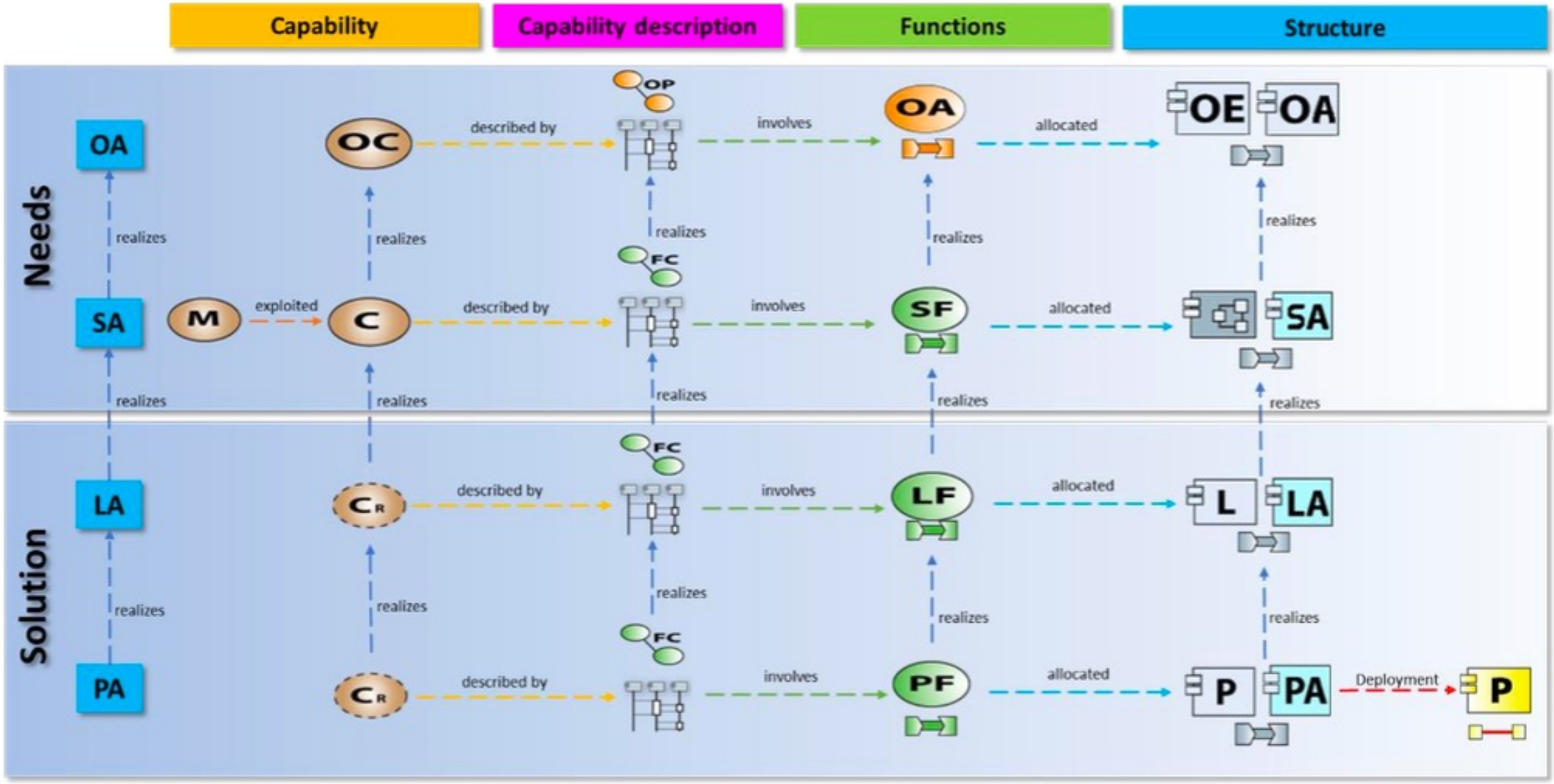



Figure 2.3: Arcadia ontology traceability

https://www.slideshare.net/HelderCastro3/mbse-with-arcadia-methodpdf-256664096

| Arcadia layer | Requirements | Capability | Capability description | Functional | Structure | Modes and States | Data | Interfaces |
|---|---|---|---|---|---|---|---|---|
| | R-OA | OA1 | OA2 | OA3 | OA4 | M&S-OA5 | D-OA6 | I-OA7 |
| Operational Analysis | Capture stakeholder requirements | Define Operational Capabilities | Define processes and scenarios | Define Operational Activities and interactions | Capture Operational Entities and Actors. Allocate Operational Activities to Operational Actors, Entities | Define operational modes and states | Define operational data model | Define interfaces and describe interfaces scenarios |
| | R-SA | SA1 | SA2 | SA3 | SA4 | M&S-SA5 | D-SA6 | I-SA7 |
| System Analysis | Derive Stakeholder requirements and capture System requirements | Define System Missions and System Capabilities | Define Functional Chains and Scenarios. | Define System Functions. Define Functional Exchanges and components | Allocate System Functions to System and Actors | Define system modes and states | Define system data model | Define interfaces and describe interfaces scenarios Enrich Logical Scenarios. |
| | R-LA | LA1 | LA2 | LA3 | LA4 | M&S-LA5 | D-LA6 | I-LA7 |
| Logical Architecture | Derive system requirements and Capture components requirements | Transition Capabilities Realization from system layer | Define Functional Chains and scenarios | Derive System Functions and define Logical Functions. Define Functional Exchanges and components. | Allocate Logical Functions to Logical Components | Define logical components modes and states | Define logical data model | Delegate System Interfaces and create Logical Interfaces. Enrich Logical Scenarios. |
| | R-PA | PA1 | PA2 | PA3 | PA4 | M&S-PA5 | D-PA6 | I-PA7 |
| Physical Architecture | Derive logical requirements and capture physical requirements | Transition Capabilities Realization from logical layer | Define Functional Chains, Scenarios, and Physical Path | Derive Logical Functions and define Physical Functions. Define Functional Exchanges and components. | Define Physical Nodes and refine Behavioural Physical Components. Allocate Behavioural Components. | Define physical nodes modes and states | Define physical data model | Delegate Logical Interfaces and create Physical Interface. Enrich Physical Scenarios. |

Table 3.2: Arcadia matrix activities

https://www.slideshare.net/HelderCastro3/mbse-with-arcadia-methodpdf-256664096

| Arcadia layer | Requirements | Capability | Capability description | Functional | Structural | Modes and States | Data | Interfaces |
|---|---|---|---|---|---|---|---|---|
| Operational Analysis | **R-OA**<br>No dedicated diagram | **OA1**<br>[OCB] Operational Capabilities | **OA2**<br>[OAS] Operational Activity Scenario<br>[OPD] Operational Process Scenario<br>[OES] Operational Entity Scenario | **OA3**<br>[OABD] Operational Activity Breakdown Diagram<br>[OAIB] Operational Activity Interaction Blank | **OA4**<br>[OEBD] Operational Entities Blank Diagram<br>[ORB] Operational Roles Blank<br>[OAB] Operational Architecture Blank | **M&S-OA5**<br>[MSM] Modes and States | **D-OA6**<br>[CDB] Class Diagram | **I-OA7**<br>[IDB] Interface Definition Blank<br>[CEI] Component External Interfaces<br>[IS] Interface Scenario<br>[CDI] Component Detailed Interface |
| System Analysis | **R-SA**<br>No dedicated diagram | **SA1**<br>[MCB] Mission and Capabilities Blank<br>[CC] Contextual Capability | **SA2**<br>[FS] System Functional Scenario<br>[ES] System Entity Scenario<br>[SFCD] System Functional Chain Description | **SA3**<br>[SFBD] System Functional Breakdown Diagram<br>[SDFB] System Data Flow Blank | **SA4**<br>[CSA] Contextual System Actor<br>[SAB] System Architecture Blank | **M&S-SA5**<br>[MSM] Modes and States | **D-SA6**<br>[CDB] Class Diagram | **I-SA7**<br>[IDB] Interface Definition Blank<br>[CEI] Component External Interfaces<br>[IS] Interface Scenario<br>[CDI] Component Detailed Interface |
| Logical Architecture | **R-LA**<br>No dedicated diagram | **LA1**<br>[CRB] Capabilities Realization Blank<br>[CRI] Contextual Capability Realization Involvement | **LA2**<br>[FS] Logical Functional Scenario<br>[ES] Logical Entity Scenario<br>[LFCD] Logical Functional Chain Description | **LA3**<br>[LFBD] Logical Functional Breakdown Diagram<br>[LDFB] Logical Data Flow Blank | **LA4**<br>[LCBD] Logical Component Breakdown Diagram<br>[LAB] Logical Architecture Blank | **M&S-LA5**<br>[MSM] Modes and States | **D-LA6**<br>[CDB] Class Diagram | **I-LA7**<br>[IDB] Interface Definition Blank<br>[CEI] Component External Interfaces<br>[IS] Interface Scenario<br>[CDI] Component Detailed Interface |
| Physical Architecture | **R-PA**<br>No dedicated diagram | **PA1**<br>[CRB] Capabilities Realization Blank<br>[CRI] Contextual Capability Realization Involvement | **PA2**<br>[FS] Physical Functional Scenario<br>[ES] Physical Entity Scenario<br>[PFCD] Physical Functional Chain Description | **PA3**<br>[PFBD] Physical Functional Breakdown Diagram<br>[PDFB] Physical Data Flow Blank | **PA4**<br>[PCBD] Physical Component Breakdown Diagram<br>[PAB] Physical Architecture Blank | **M&S-PA5**<br>[MSM] Modes and States | **D-PA6**<br>[CDB] Class Diagram | **I-PA7**<br>[IDB] Interface Definition Blank<br>[CEI] Component External Interfaces<br>[IS] Interface Scenario<br>[CDI] Component Detailed Interface |

Table 3.3: Arcadia diagrams matrix



https://www.slideshare.net/HelderCastro3/mbse-with-arcadia-methodpdf-256664096

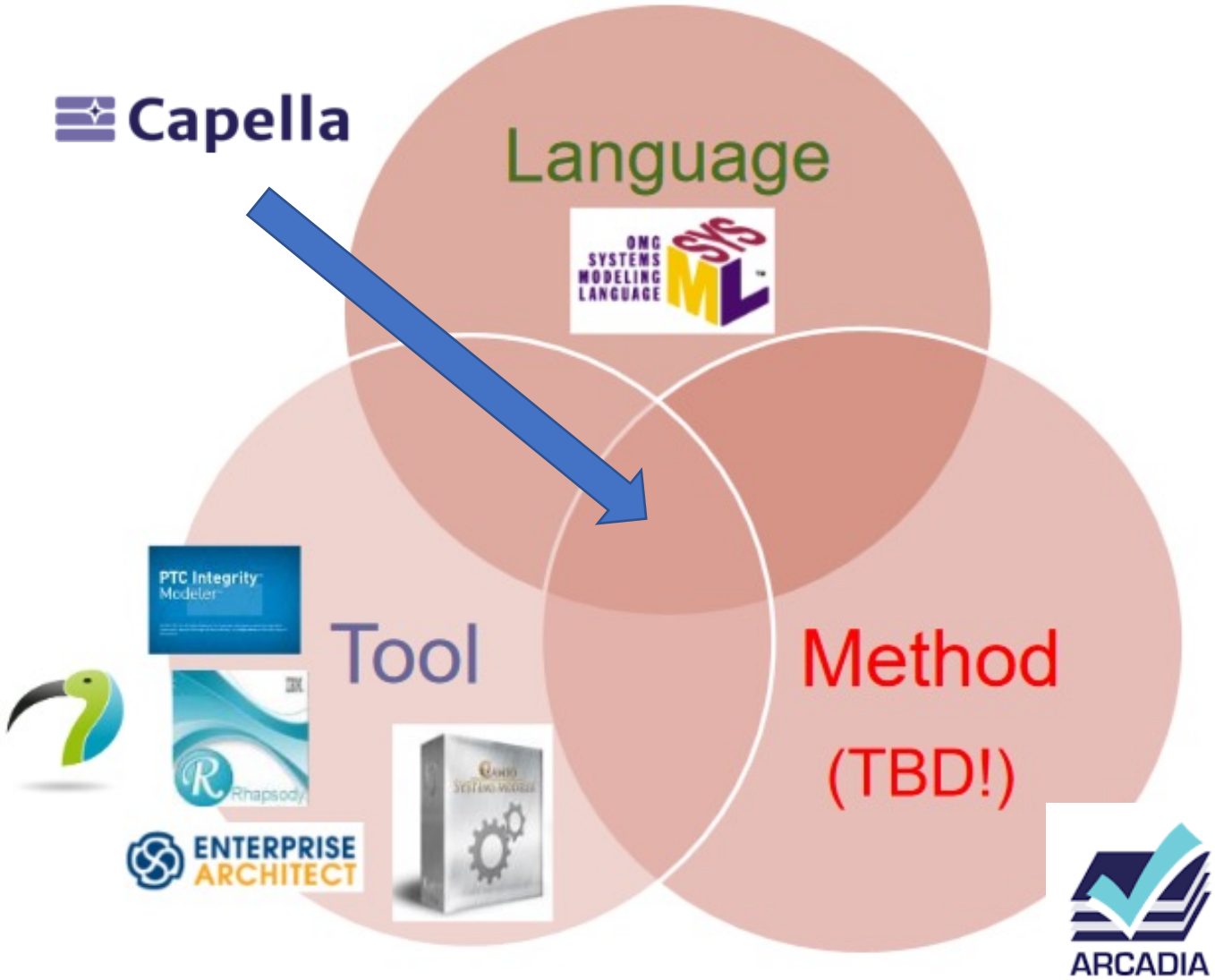

https://www.eclipse.org/capella/adopters.html

# ARCADIA / CAPELLA

- **Thales**

- Capella é uma camada acima do Eclipse IDE
- Sirius
- Eclipse Modelling Framework (EMF)

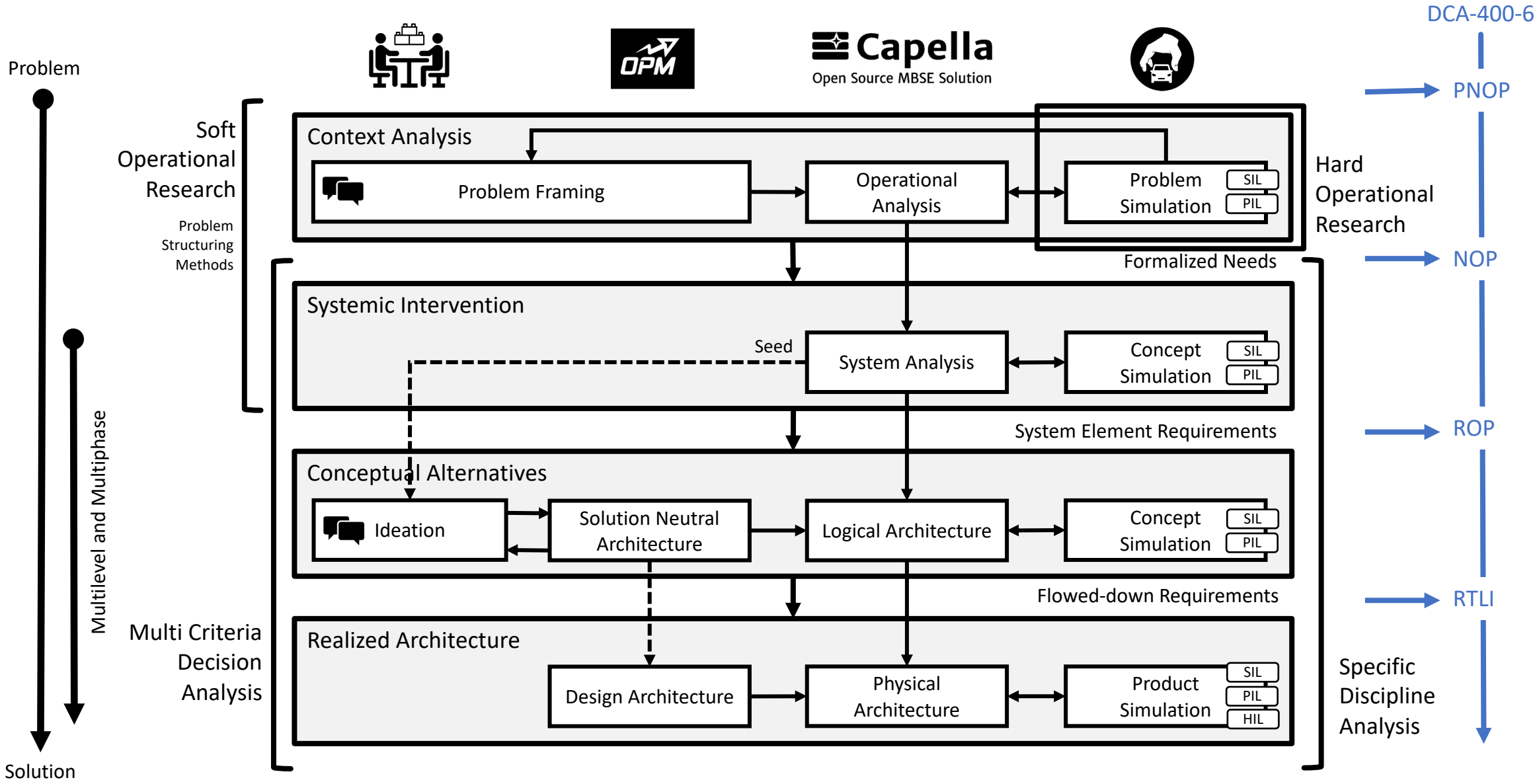
- Versão Atual: 6.1
  - https://www.eclipse.org/capella/



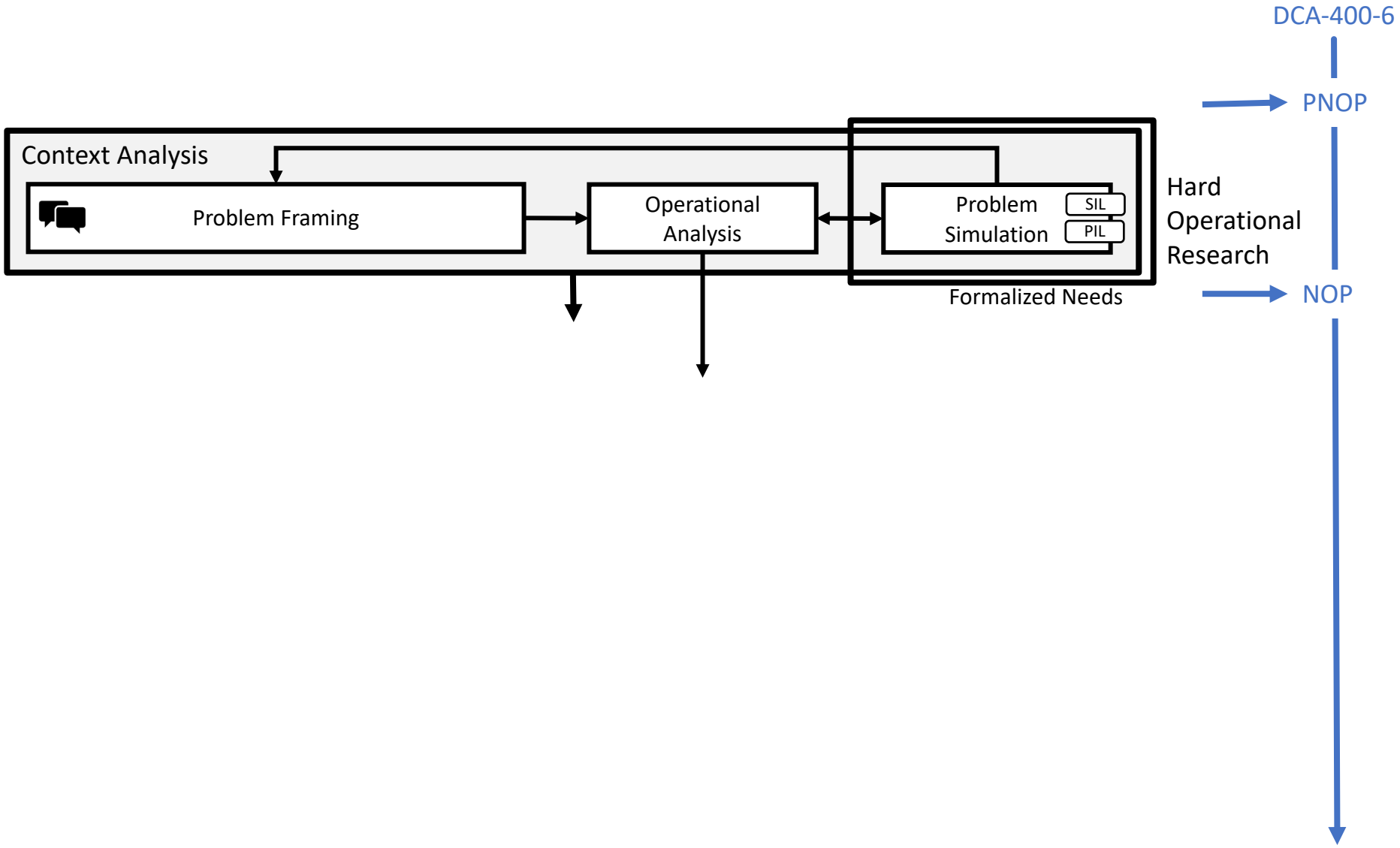
https://youtu.be/nv8IOg_xVMs

# EXEMPLO UP

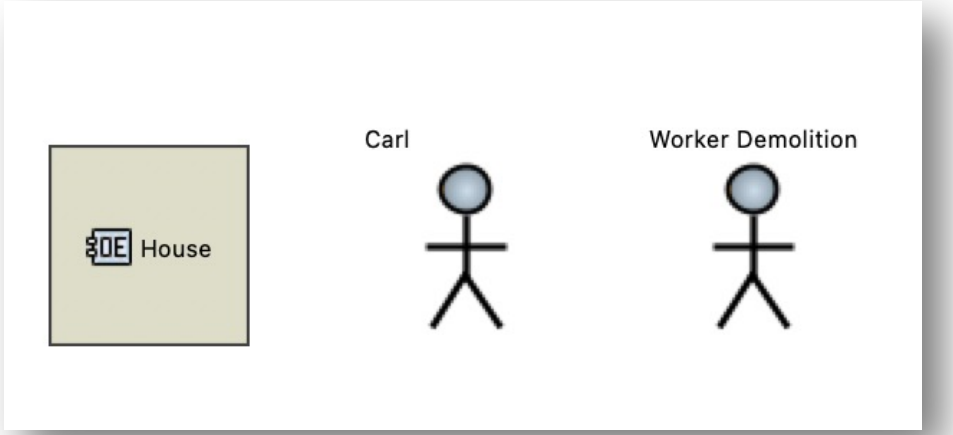
# MMMF


Problem
Solution
Multilevel and Multiphase
Soft Operational Research
Problem Structuring Methods
Multi Criteria Decision Analysis
Capella
Open Source MBSE Solution
Context Analysis
Problem Framing
Operational Analysis
Problem Simulation
SIL
PIL
Hard Operational Research
Formalized Needs
Systemic Intervention
Seed
System Analysis
Concept Simulation
System Element Requirements
Conceptual Alternatives
Ideation
Solution Neutral Architecture
Logical Architecture
Flowed-down Requirements
Realized Architecture
Design Architecture
Physical Architecture
Product Simulation
HIL
Specific Discipline Analysis
DCA-400-6
PNOP
NOP
ROP
RTLI

# CONTEXT ANALYSIS


DCA-400-6
PNOP
Context Analysis
Problem Framing
Operational Analysis
Problem Simulation
SIL
PIL
Hard Operational Research
Formalized Needs
NOP

# MODELLING THE ACTORS/ENTITIES OF WHAT IS HAPPENING NOW (AS IS)




Symbols

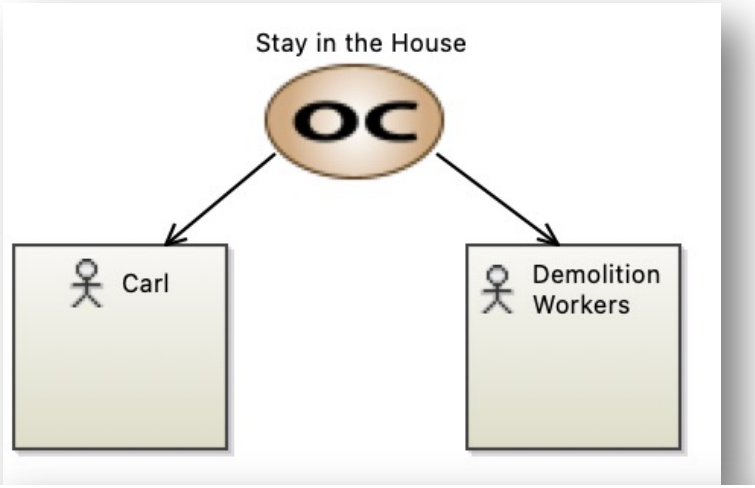

Use context images



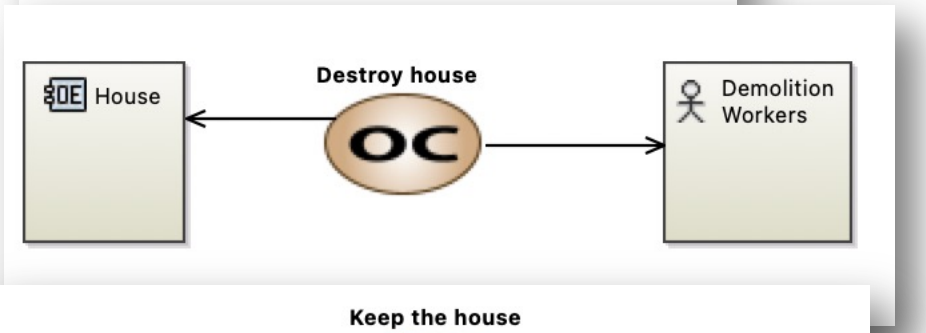
Capella>>Operational Analysis>>Operational Entity Breakdown Diagrams

# MAP WHAT IS HAPPENING

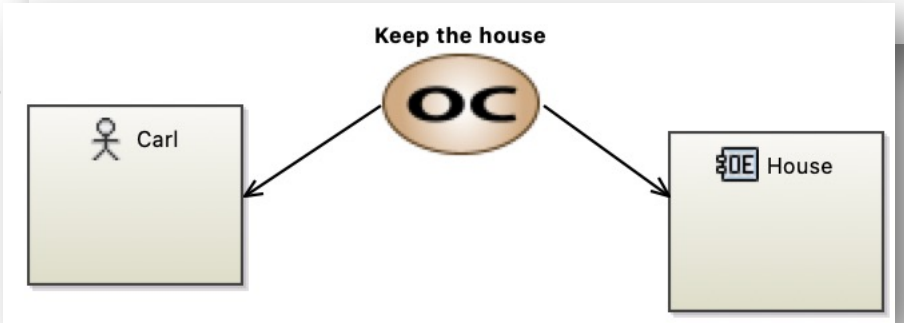
1.



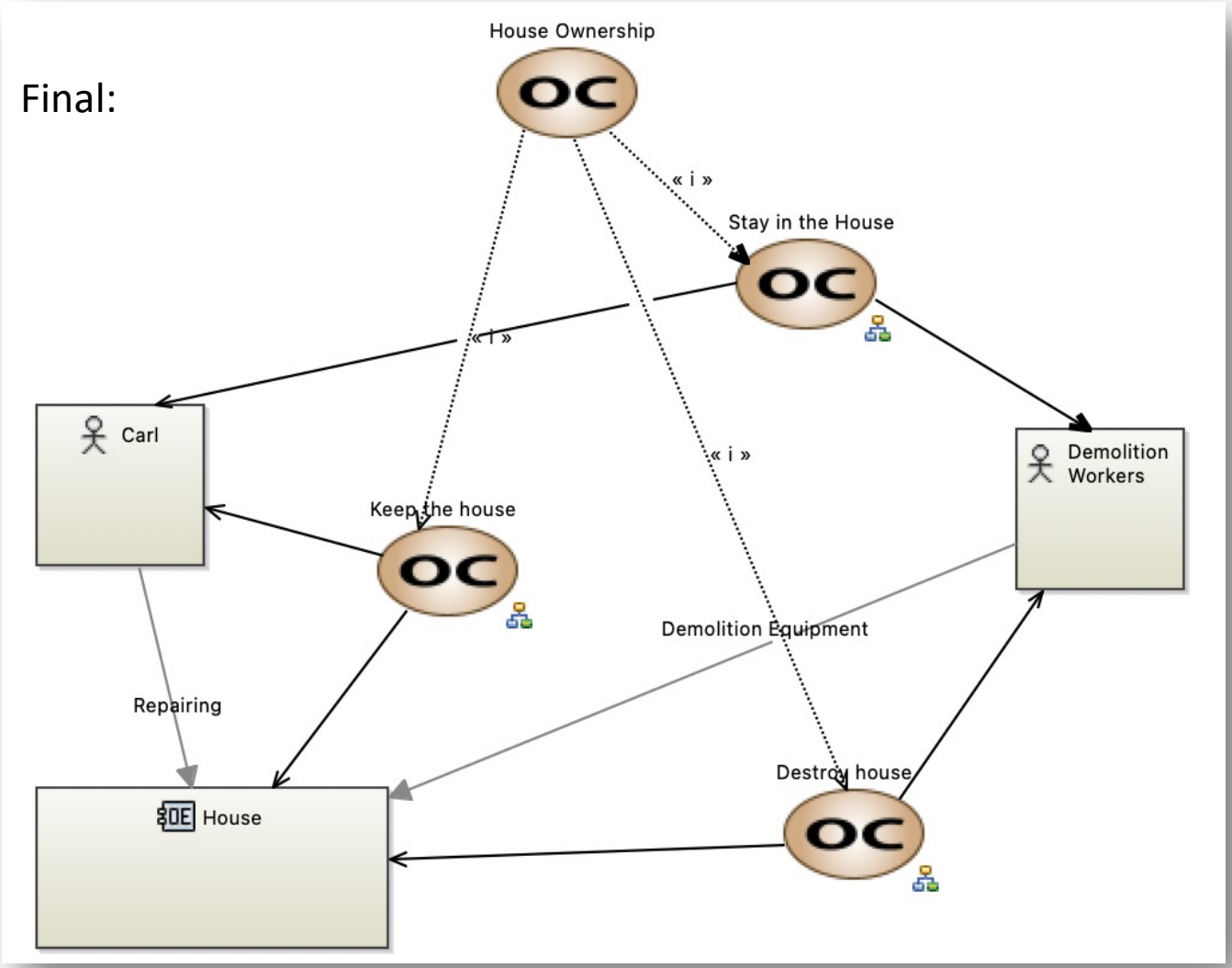

2.




3.

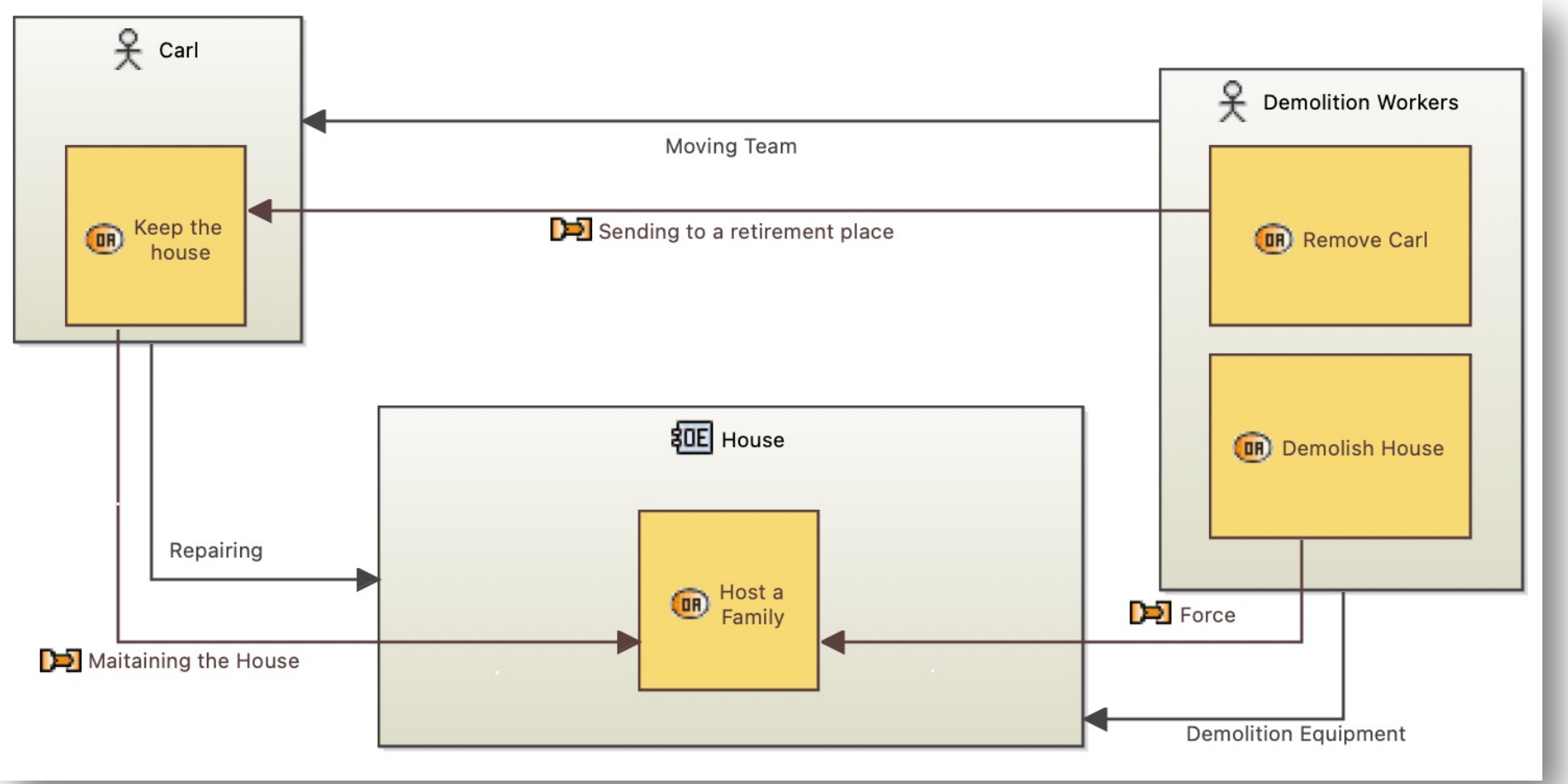



Final:



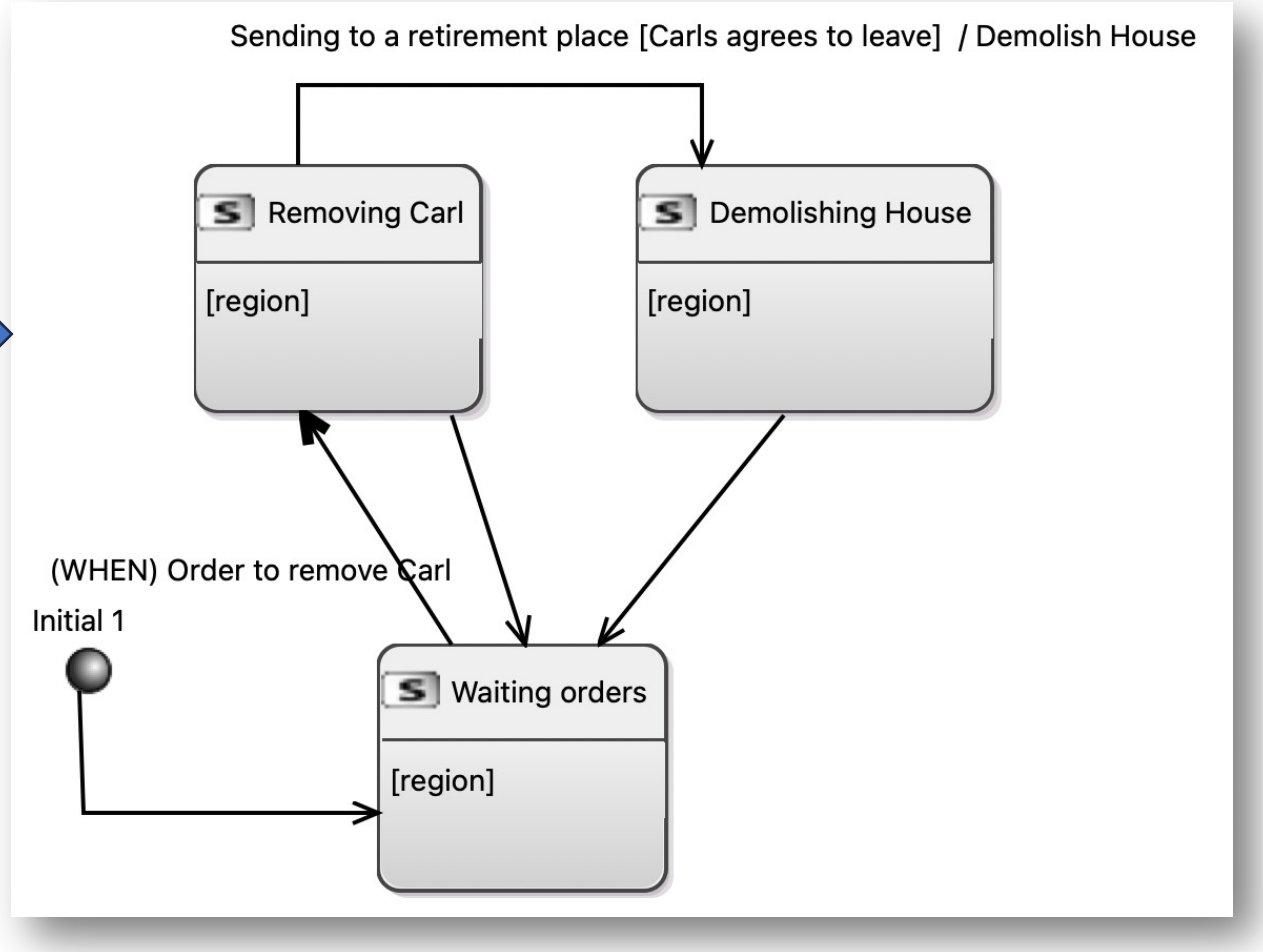

Capella>>Operational Analysis>>Operational Capabilities Diagrams

# EACH STAKEHOLDER (ACTOR/ENTITY) DO SOMETHING (ACTIVITY) AND RELATES TO EACH OTHER (INTERACTION)




Capella>>Operational Analysis>>Operational Architecture Diagram

# DESCRIBE THE STAKEHOLDERS' BEHAVIORS



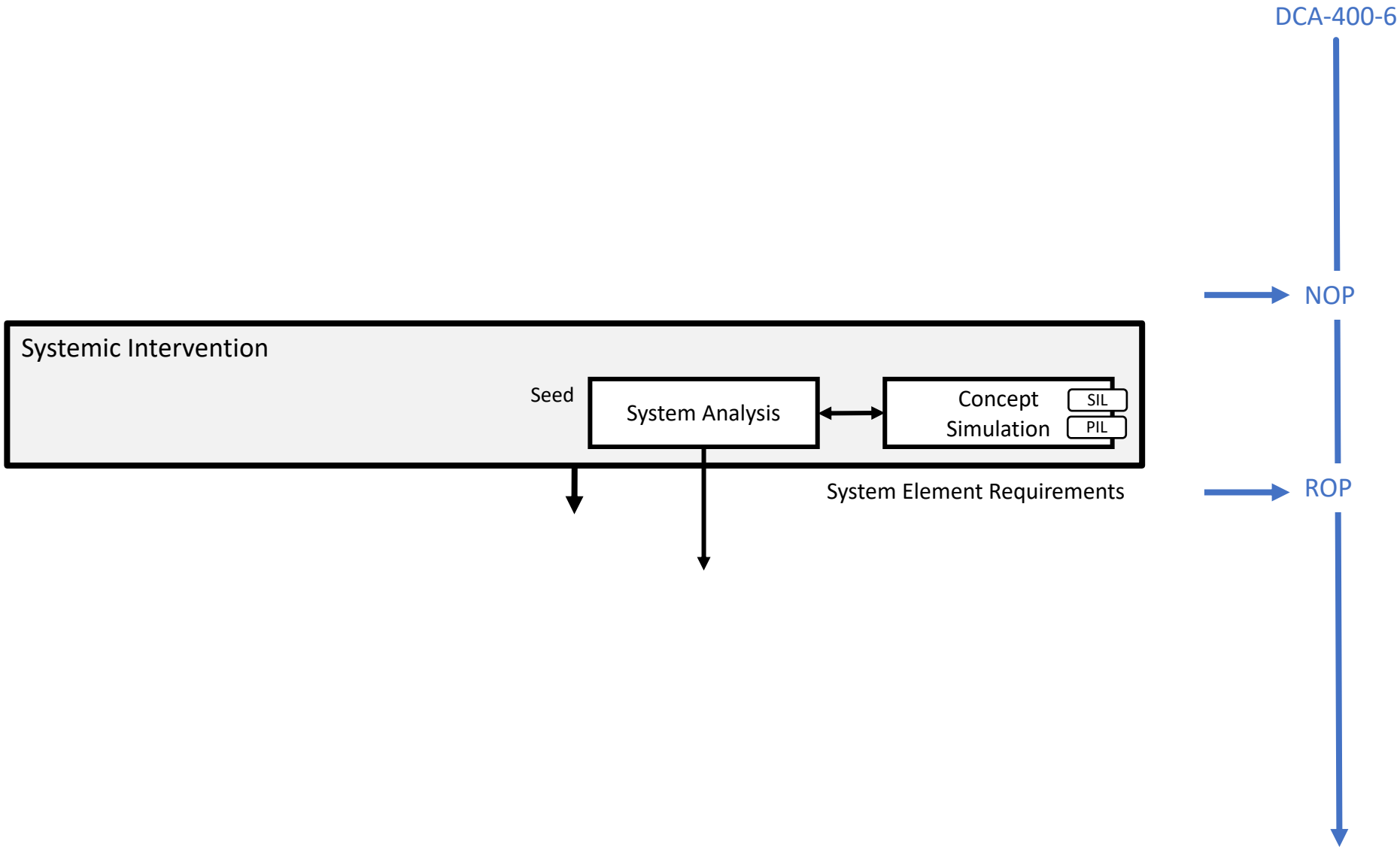

Capella>>Operational Analysis>>StateMachine Diagram

# WHAT DO WE NEED TO FINISH WITH IT?

- Needs mapped: What the users of the system need to accomplish
  - Mission Requirements
  - User Requirements
- Maybe not all the stakeholders opinion/needs are going to be “relevant”. It is a matter of analysis and priorization of the organization.
- One thing: this is the problem domain..... So your systems ***DOES NOT EXIST.***

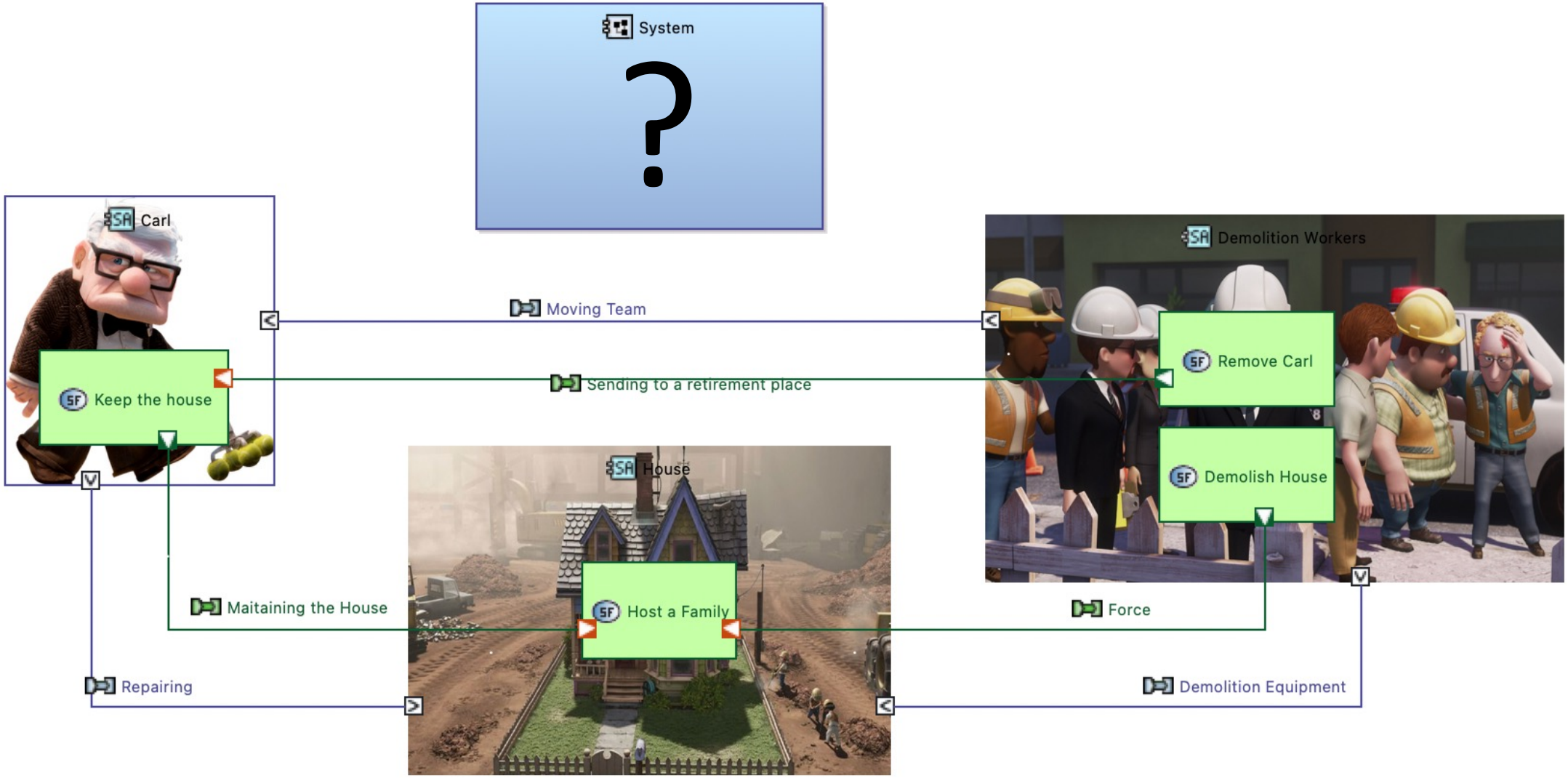
# SYSTEM INTERVENTION

What the system has to accomplish for the users


DCA-400-6
NOP
ROP
Systemic Intervention
Seed
System Analysis
Concept
Simulation
SIL
PIL
System Element Requirements

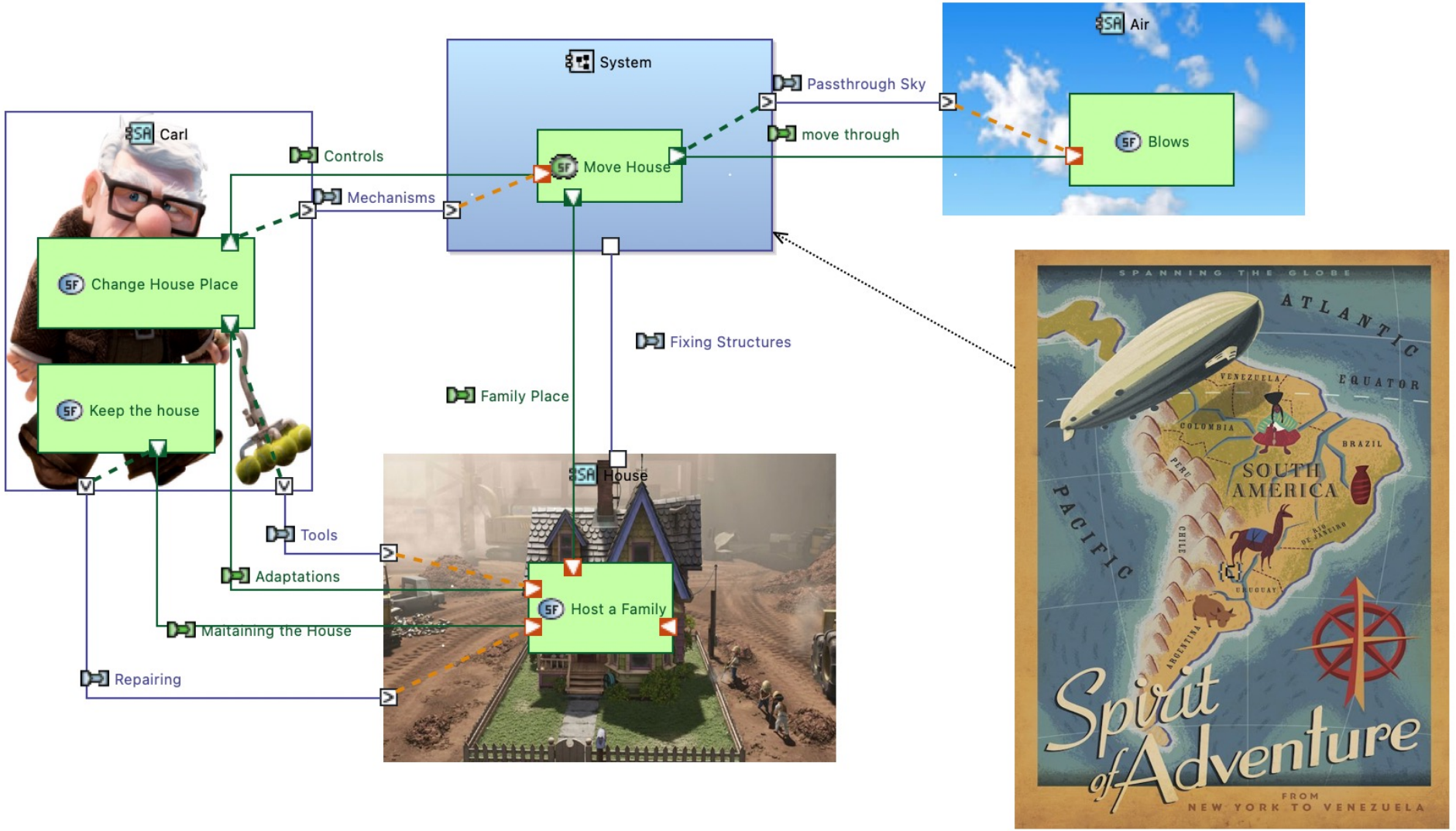

SPANNING THE GLOBE
ATLANTIC
EQUATOR
VENEZUELA
COLOMBIA
BRAZIL
PERU
SOUTH AMERICA
PACIFIC
CHILE
RIO DE JANEIRO
URUGUAY
ARGENTINA
Spirit of Adventure
FROM
NEW YORK TO VENEZUELA

# WELL.. WHAT DO THE SYSTEM MUST DO?!




Capella>>System Analysis>>System Architecture Diagram

# WELL... CARL WANTS TO MOVE THE HOUSE





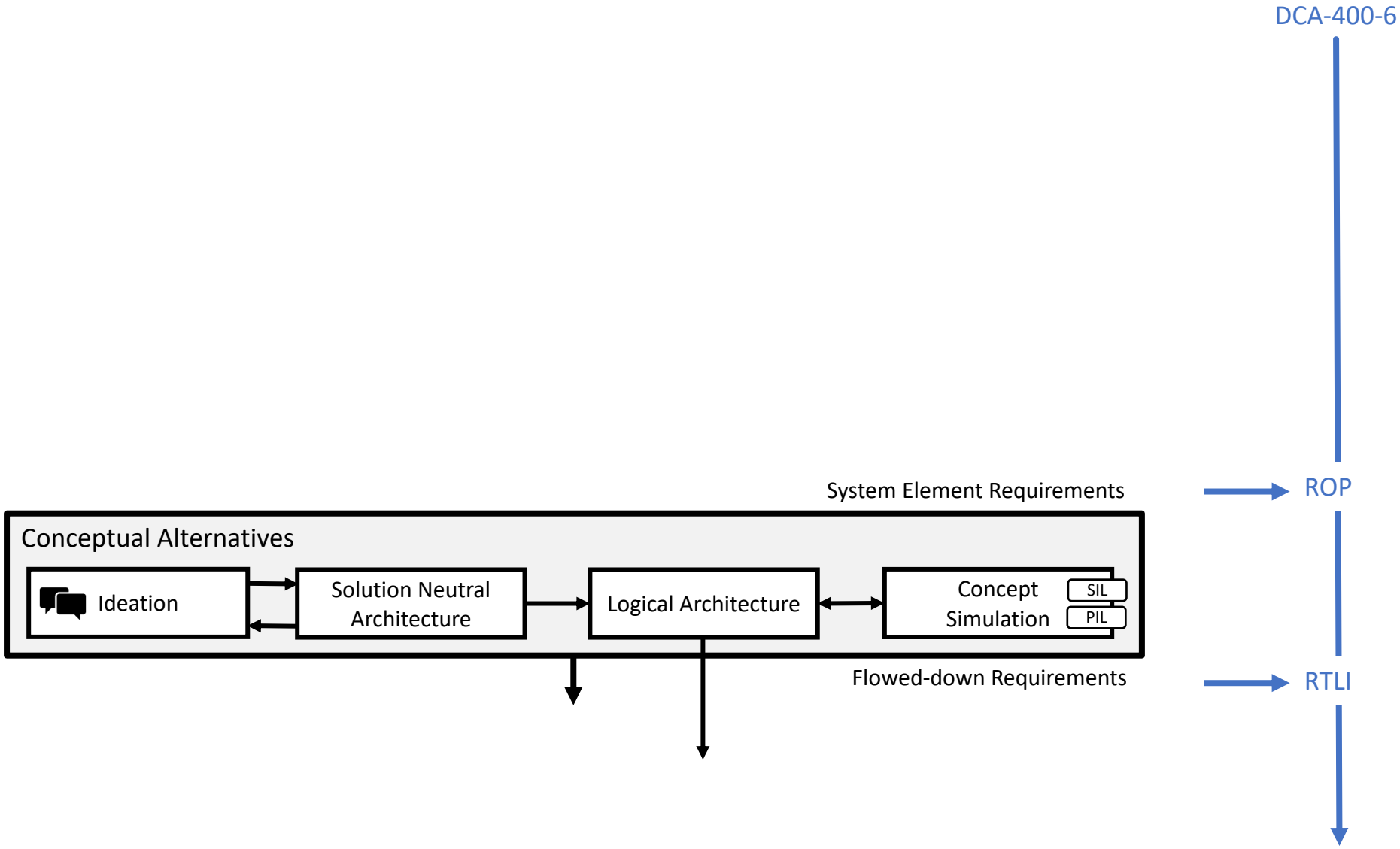

Capella>>System Analysis>>System Architecture Diagram

# WHAT DO WE NEED TO FINISH WITH IT?

- Requirements mapped: What the system has to accomplish for the users
  - System Requirements
- Remember that requirements are on the problem domain → does not carry solution on it.
  - The system must receive 24V /// and not /// The Li-Po Battery must provide 24V to the System.
- One thing: The System is a black box... We can not see inside only the frontier functions (interface/external functions) – such functions are what emerges!!! (emergent properties)

# CONCEPTUAL ARCHITECTURE

How the system will work to fulfill the expectations


DCA-400-6
System Element Requirements
ROP
Conceptual Alternatives
Ideation
Solution Neutral Architecture
Logical Architecture
Concept Simulation
SIL
PIL
Flowed-down Requirements
RTLI


ZEPPELIN

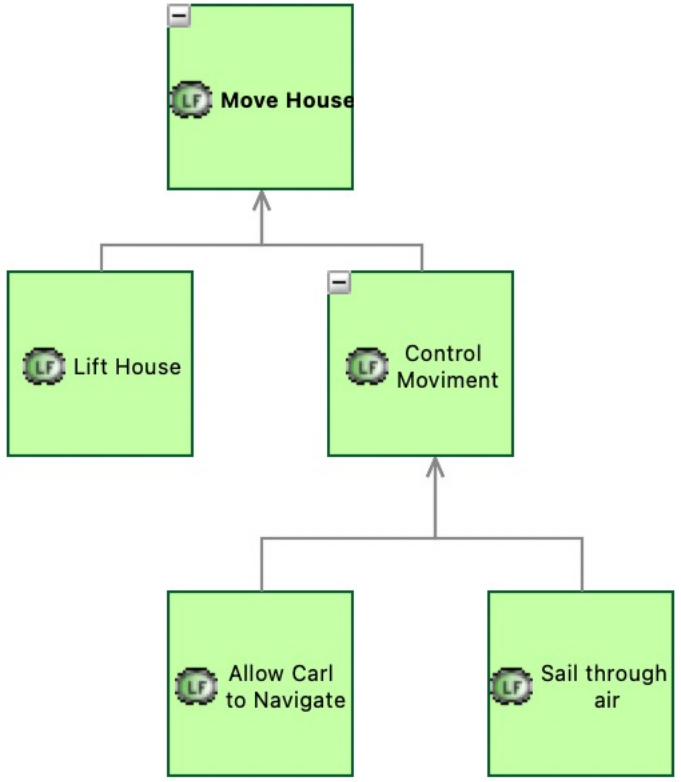

ZEPPELIN


LED ZEPPELIN

# HUMMM

- Even though the joke with Led Zeppelin is a good one... And I could not avoid to make it... ☺
- It is more a balloon than a zeppelin.

# WELL.. THE MAIN FUNCTION WAS: MOVE HOUSE



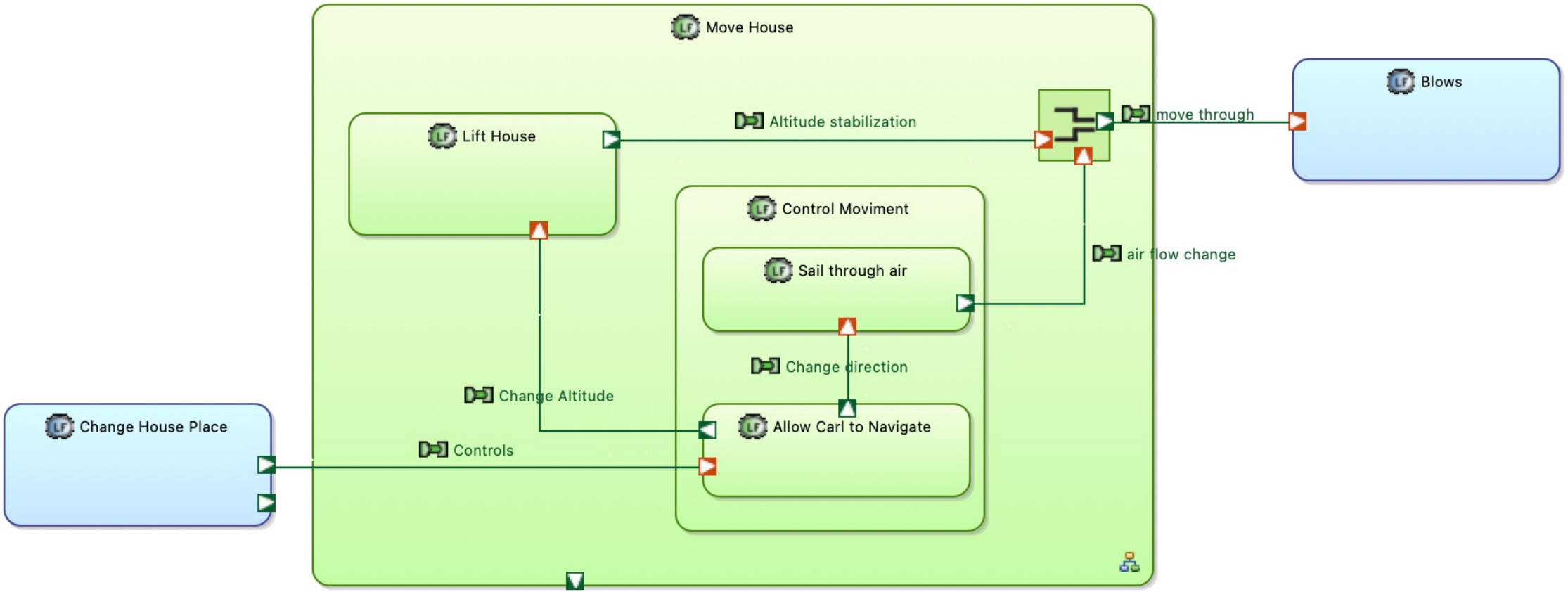

We can decompose the functions in subfunctions.

Only leaf functions must be used.



Capella>>Logical Architecture>>Logical Function Breakdown Diagram

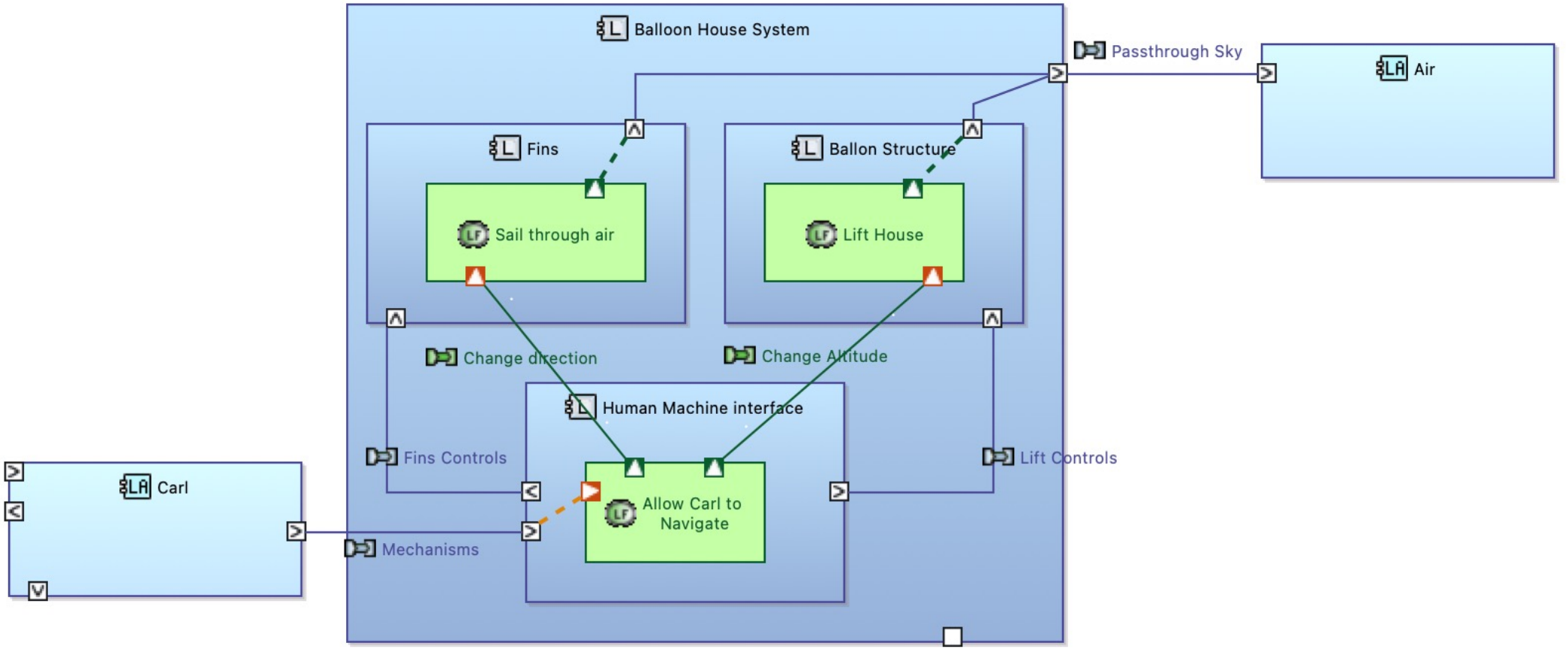
# THE FUNCTIONS MIGHT HAVE ITS OWN ARCHITECTURE: FUNCTIONAL ARCHITECTURE






Capella>>Logical Architecture>>Logical Function Data Flow Diagram

# WE CAN CONCEPTUALLY SPLIT FUNCTIONS INTO A REFERENCE ARCHITECTURE OF THE AIMING SOLUTION






Capella>>Logical Architecture>>Logical Architecture Diagram

# WE COULD HAVE DECIDED A CONOPS TO THIS SOLUTION CONCEPT

# WHAT DO WE NEED TO FINISH WITH IT?

- Requirements mapped: How the system will work to fulfill the expectations
  - Subsystem Requirements (or any decomposition part of it)
- We have a functional architecture spread through a desired architecture.
  - We can plan verifications, transitions, integrations, operations, and everything.
  - Here is the place to ask for functions that will have a technological solution on the next step.
- One thing: The System is now a white box... We can see inside and design the desired (at least requested) architecture.

# CONCRETE ARCHITECTURE

How the system will be built

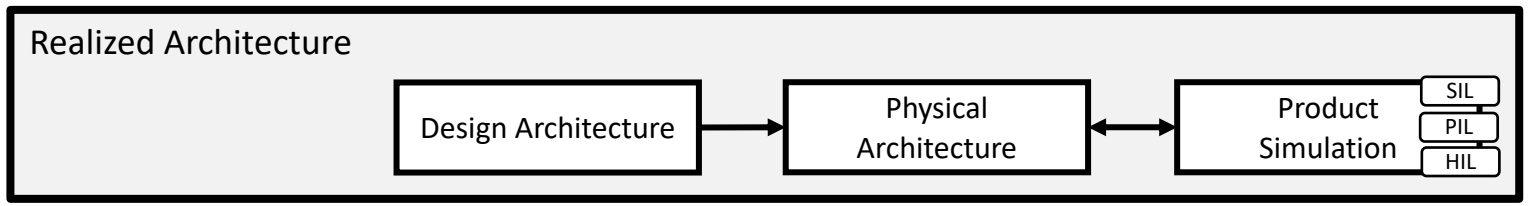
DCA-400-6

PNOP

NOP

ROP

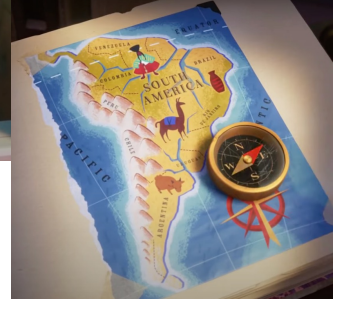
RTLI



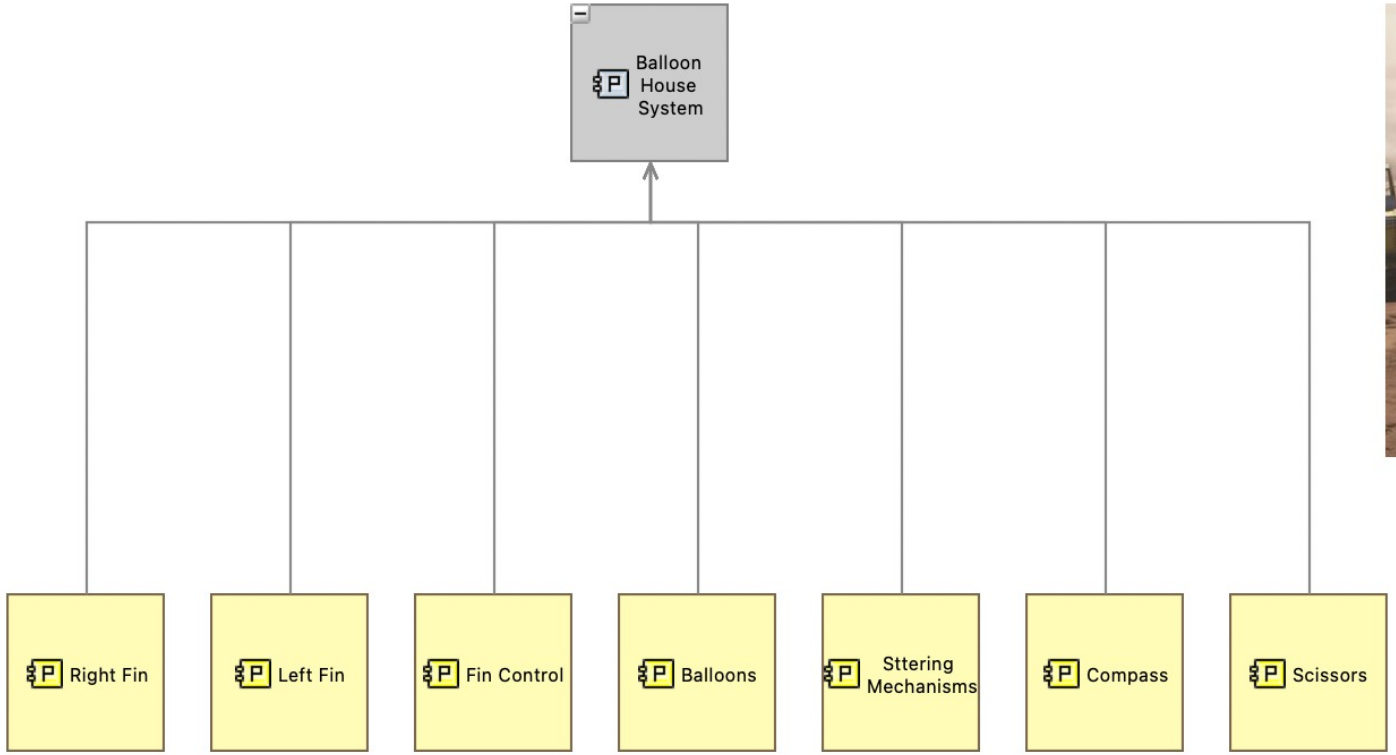
# SO OK… FINAL STEP IS SPECIFY WHAT IS GOING TO BE BUILT

- He had the “things” that were feasible, pre-existing in the house and easily acquirable.
  - To lift: balloons
  - To steer: some house tools
  - To sail: towels, blankets
  - To navigate: compass
  - To adjust altitude: cut the balloon strings


W
N
E


SOUTH AMERICA
PACIFIC

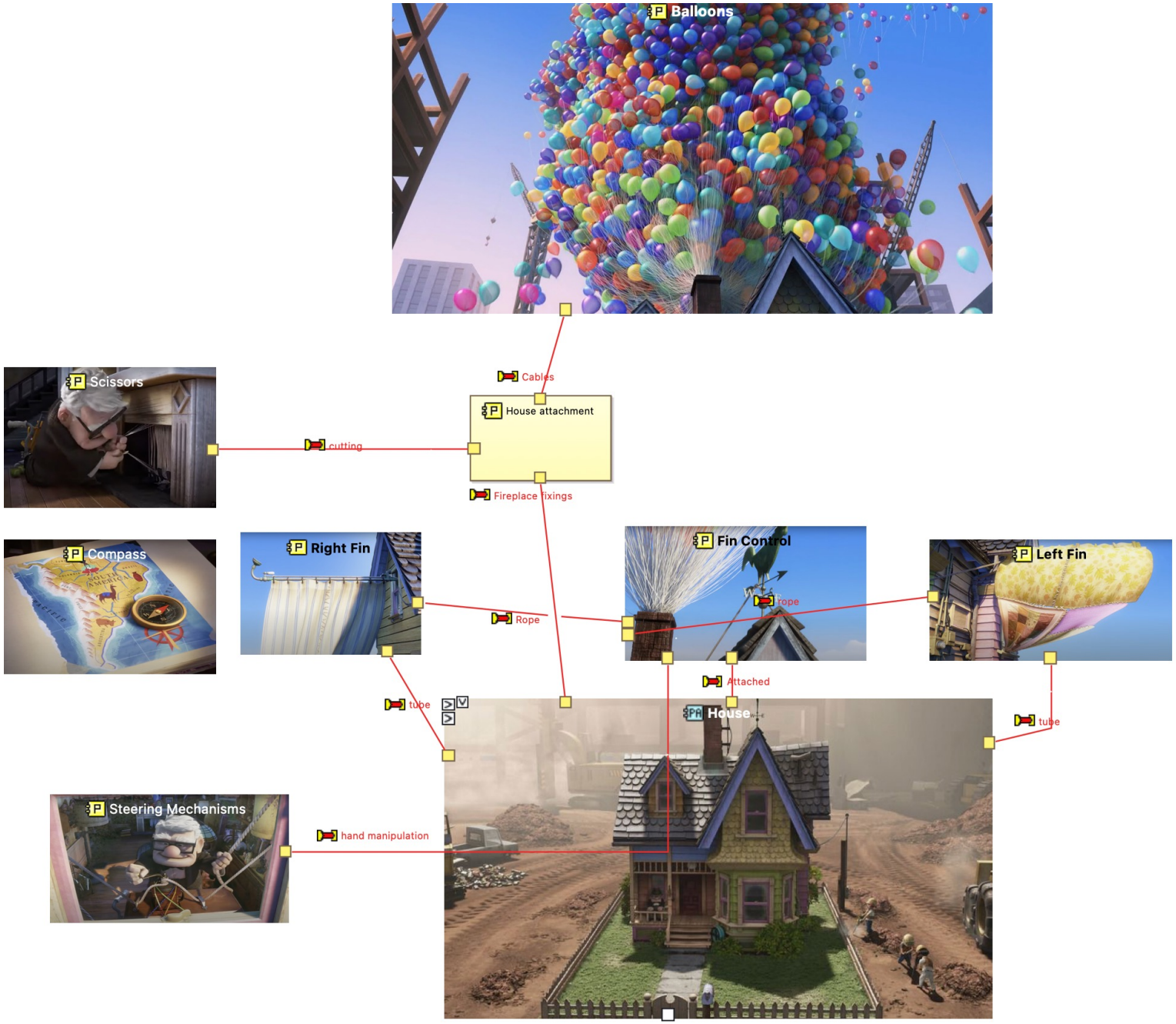
# POINT OUT THE TECHNOLOGICAL CHOICES TO BUILT THE BALLOON HOUSE SYSTEM







Capella>>Physical Architecture>>Physical Component Breakdown Diagram




# TO BE BUILT MODEL



Capella>>Physical Architecture>>Physical Architecture Diagram

# WHAT DO WE NEED TO FINISH WITH IT?

- Specifications to the development/acquisition/building process
  - Would go to every details necessary to build the system.
- We have a concrete architecture ( do not be confused by the word physical – does not need to be “physical”… can be a process, software, information, so on)
- Usually in the Phase 0 / Pre-A of the Space System Lifecycle it is designed a feasibility architecture with co-engineering (in Concurrent Engineering Labs). This Architecture would be born in this phase and iterated/adapted through the next life cycle phases.

# SYSTEM DELIVERED:

# REQUIREMENTS IN CAPELLA

https://www.slideshare.net/Obeo_corp/capella-webinar-writing-perfect-textual-requirements

# requirement [ ri-**kwahy*uh*** **r**-m*uh* nt ] SHOW IPA

SEE SYNONYMS FOR *requirement* ON THESAURUS.COM

*noun*

1 that which is required; a thing demanded or obligatory:
*One of the requirements of the job is accuracy.*

2 an act or instance of requiring.

3 a need or necessity:
*to meet the requirements of daily life.*

DEFINITIONS

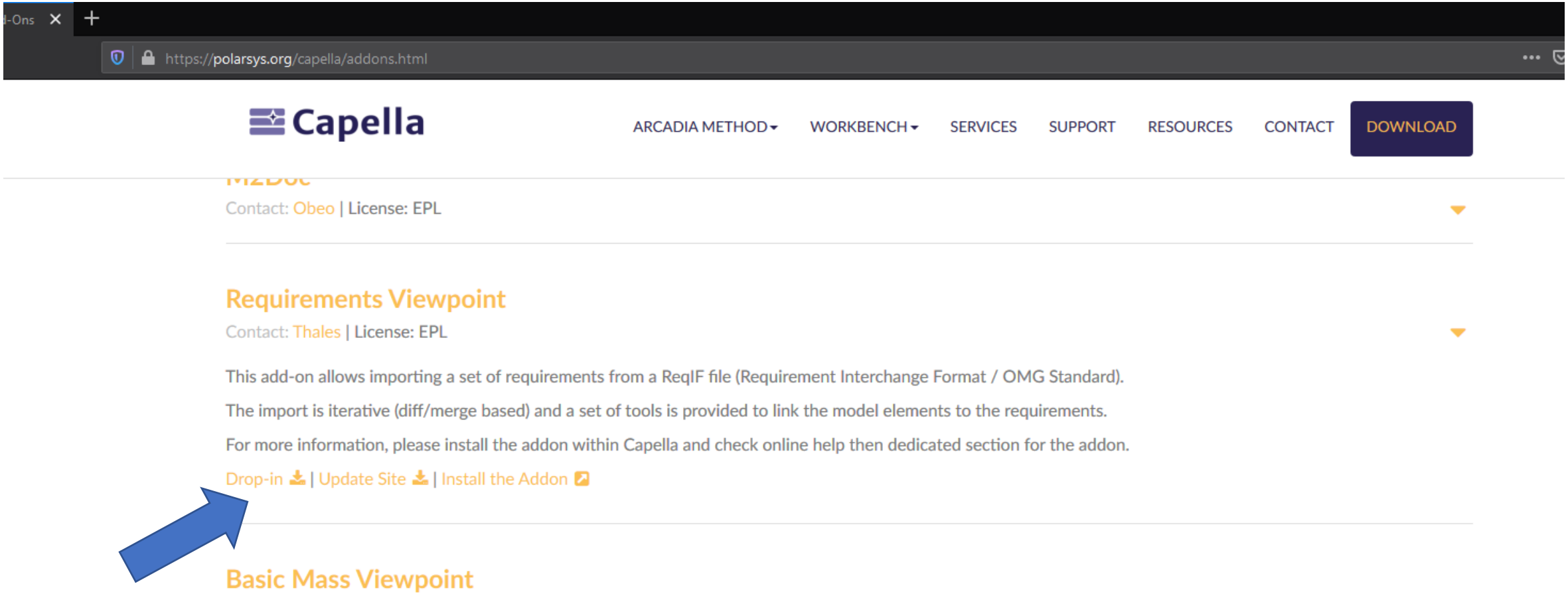
# REQUIREMENTS ARE ESSENTIAL TO:

- To show **results the user want** from the system.
- To show **traceability back to sources** and the history of changes.
- To show what the **organization needs**.
- To show what the **system must do**.
- To form a basis for the **design and design optimization**.
- To enable a **logical approach to change management**
- To partition the **work out to contractors**.
- To act as a **foundation for testing and payment**.
- To **test the system or any of its parts** during development.
- To **communicate** the basis about the system in non-technical terms to all participants.

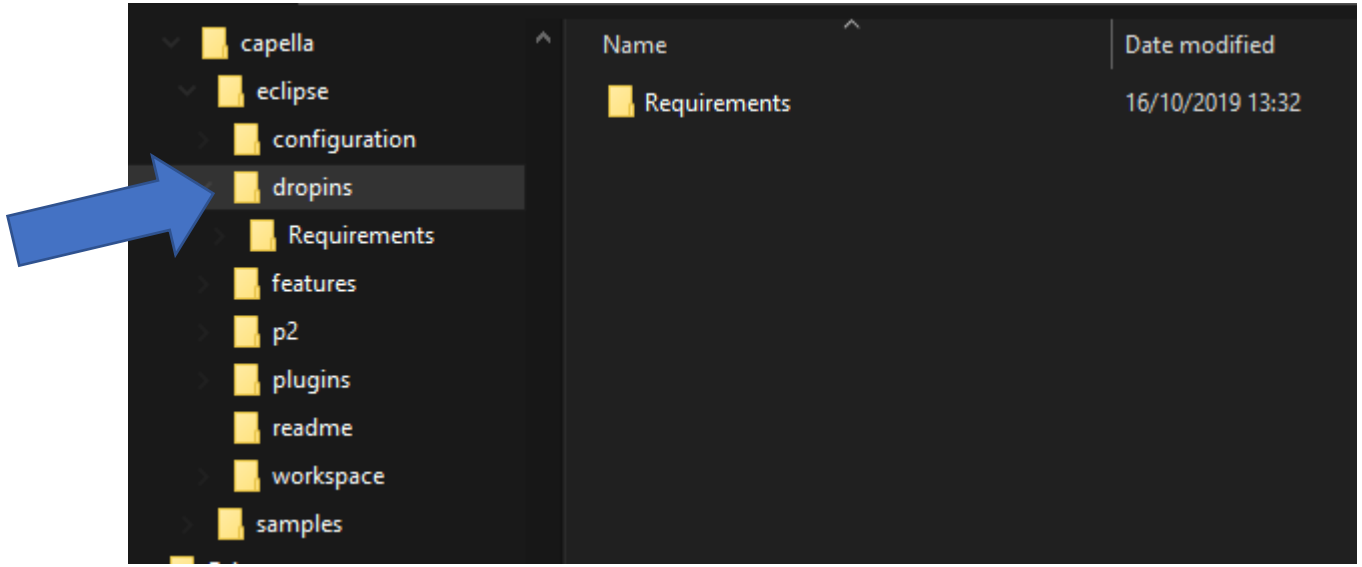
# TO DO OR NOT TO DO

- **Functional Requirements** describe what the system should do and **Non-functional Requirements** place constraints on how these functional requirements are implemented.

# [IF NOT INSTALLED] ADD THE REQ ADDON

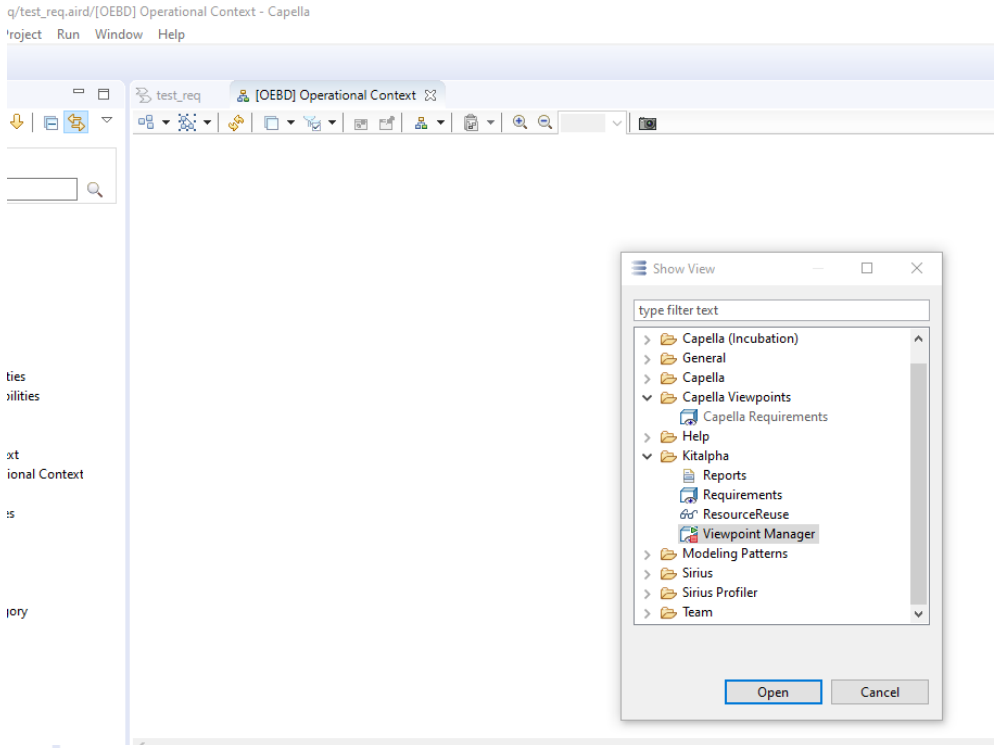


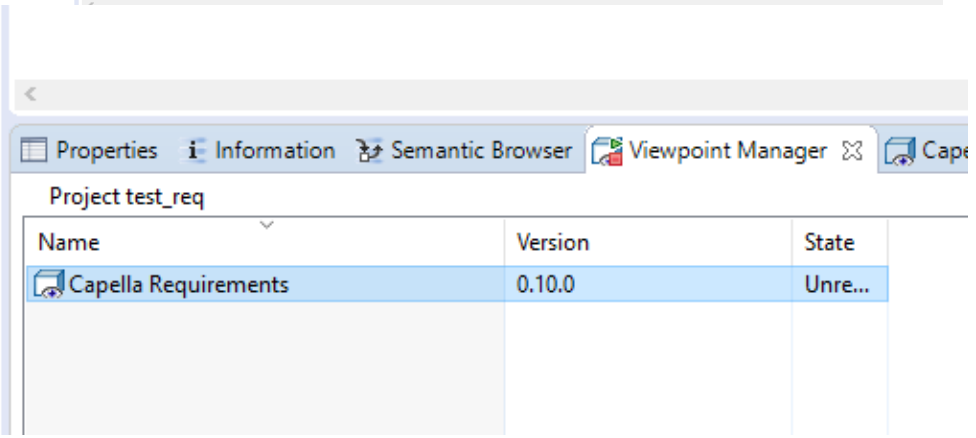
# [IF NOT INSTALLED] UNZIP IN DROPIN FOLDER



# [IF NOT INSTALLED] LAST STEPS

- Start Capella
- Open the **Viewpoint Manager** view using **Window** menu then **Show View** and **Other…**
- Select **Viewpoint Manager** in **Kitalpha** directory and press **OK**
- The **Viewpoint Manager** view is displayed
- The viewpoints available in the platform are listed in this view.
- If using Capella version < 1.0.x
  - Right-click on the name of a viewpoint and select **Start** in order to start the viewpoint
- If using Capella version > 1.0.x
  - Select any model element (diagram element, element in the project explorer) related to your project
  - Right-click on the name of a viewpoint and select **Reference** in order to start the viewpoint

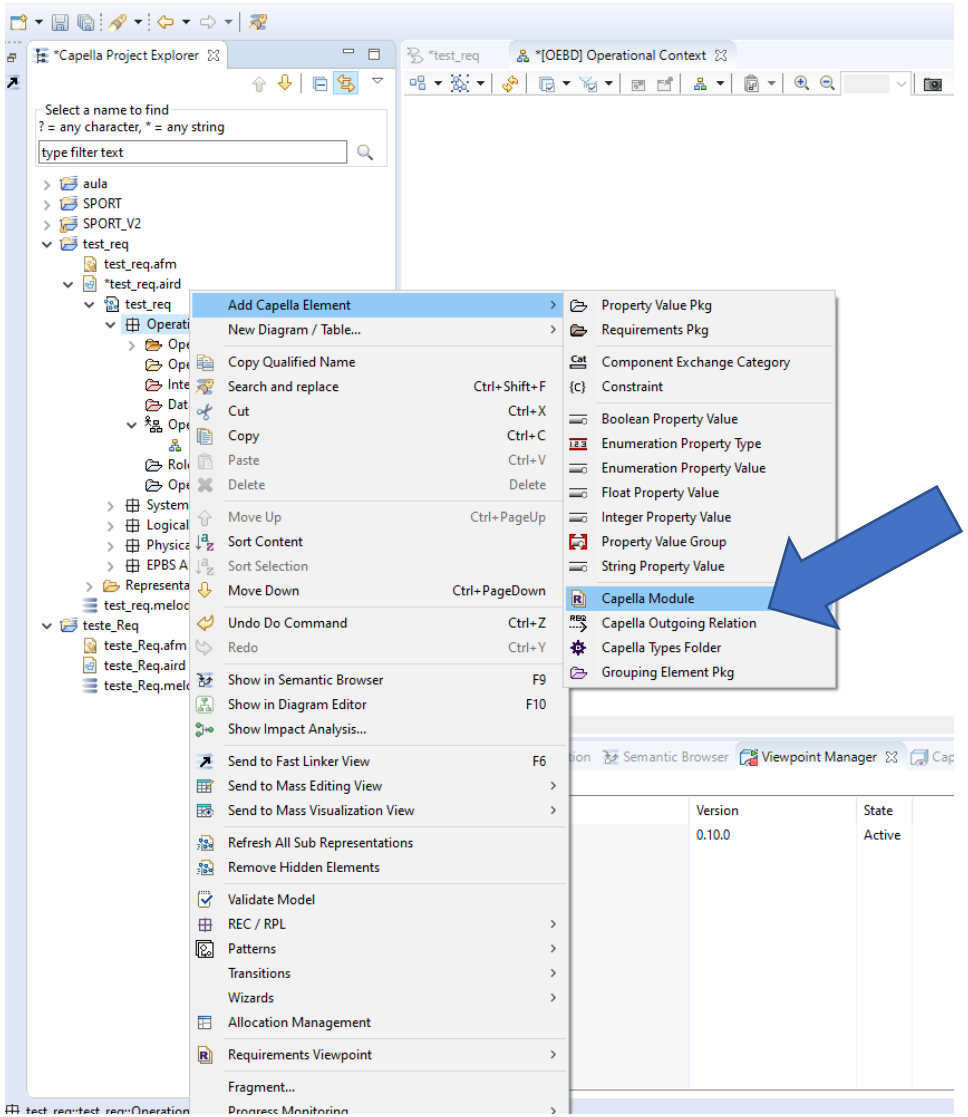


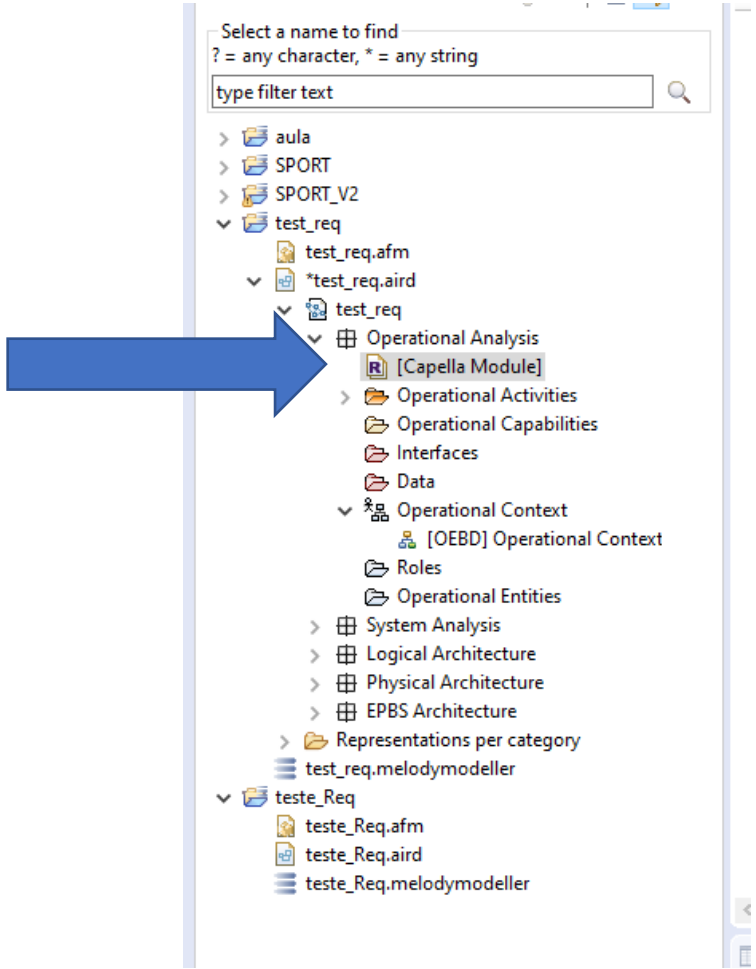



# CAN BE USED IN MULTIPLE LAYER

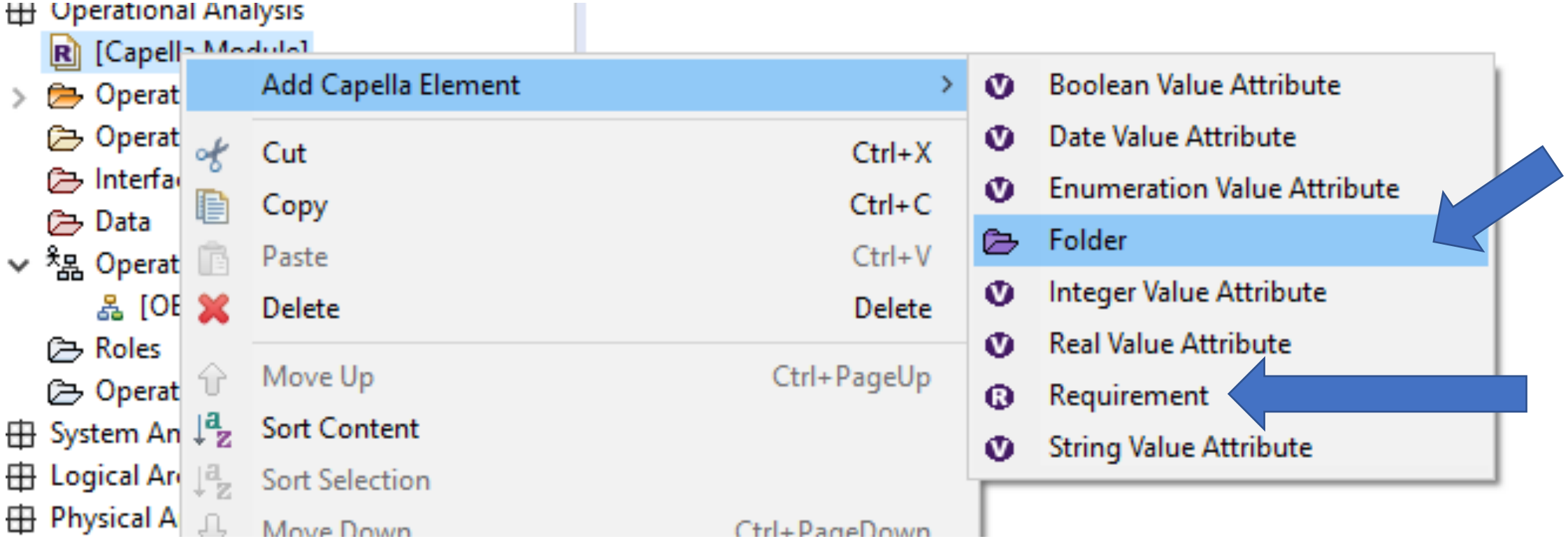
- Operational Analysis Requirements
- System Analysis Requirements
- Logical Architecture Requirements
- Physical Architecture Requirements
- EPBS Architecture Requirements

# ADD A CAPELLA MODULE IN THE LAYER



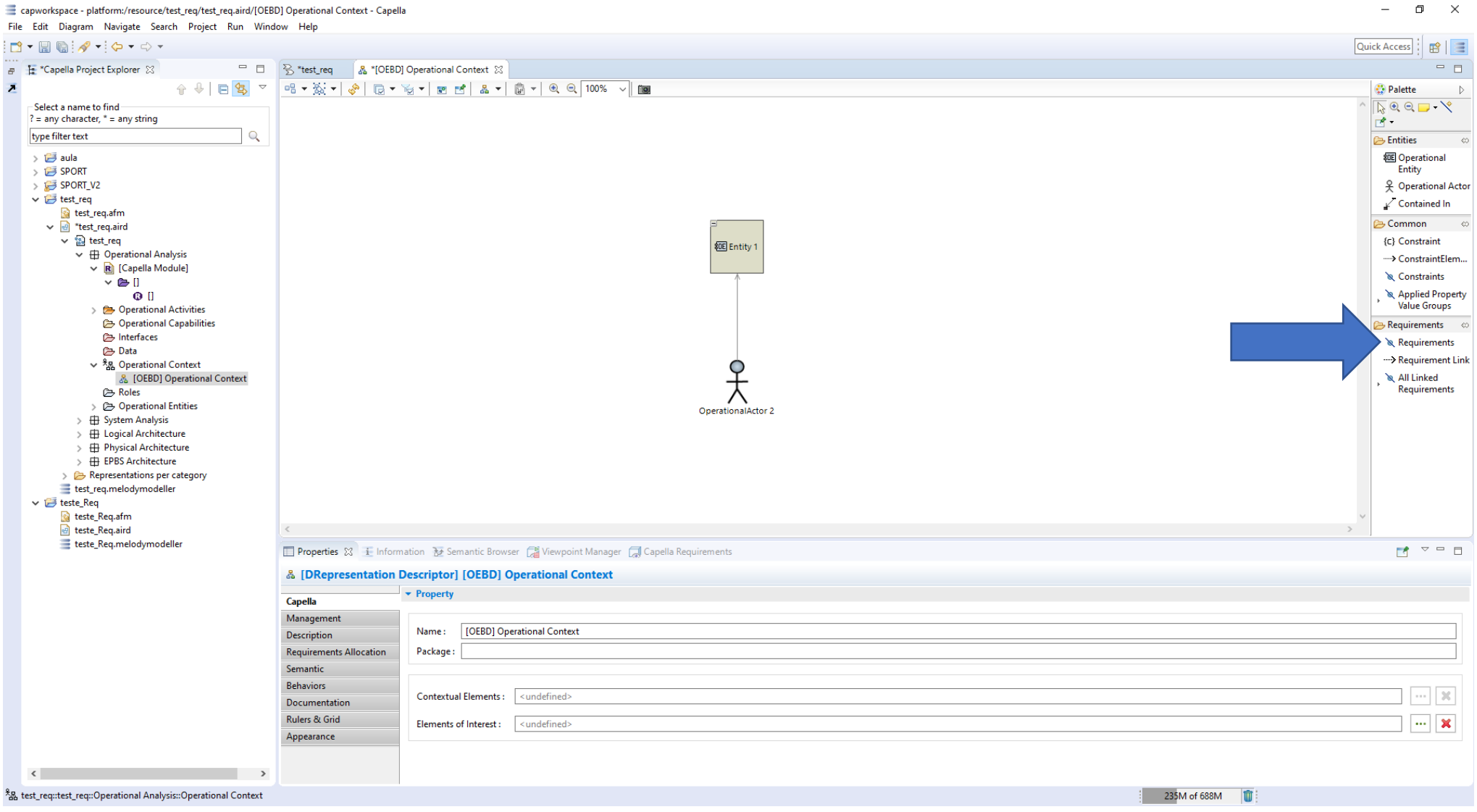
# CREATE A REQUIREMENT FOLDER & REQUIREMENT



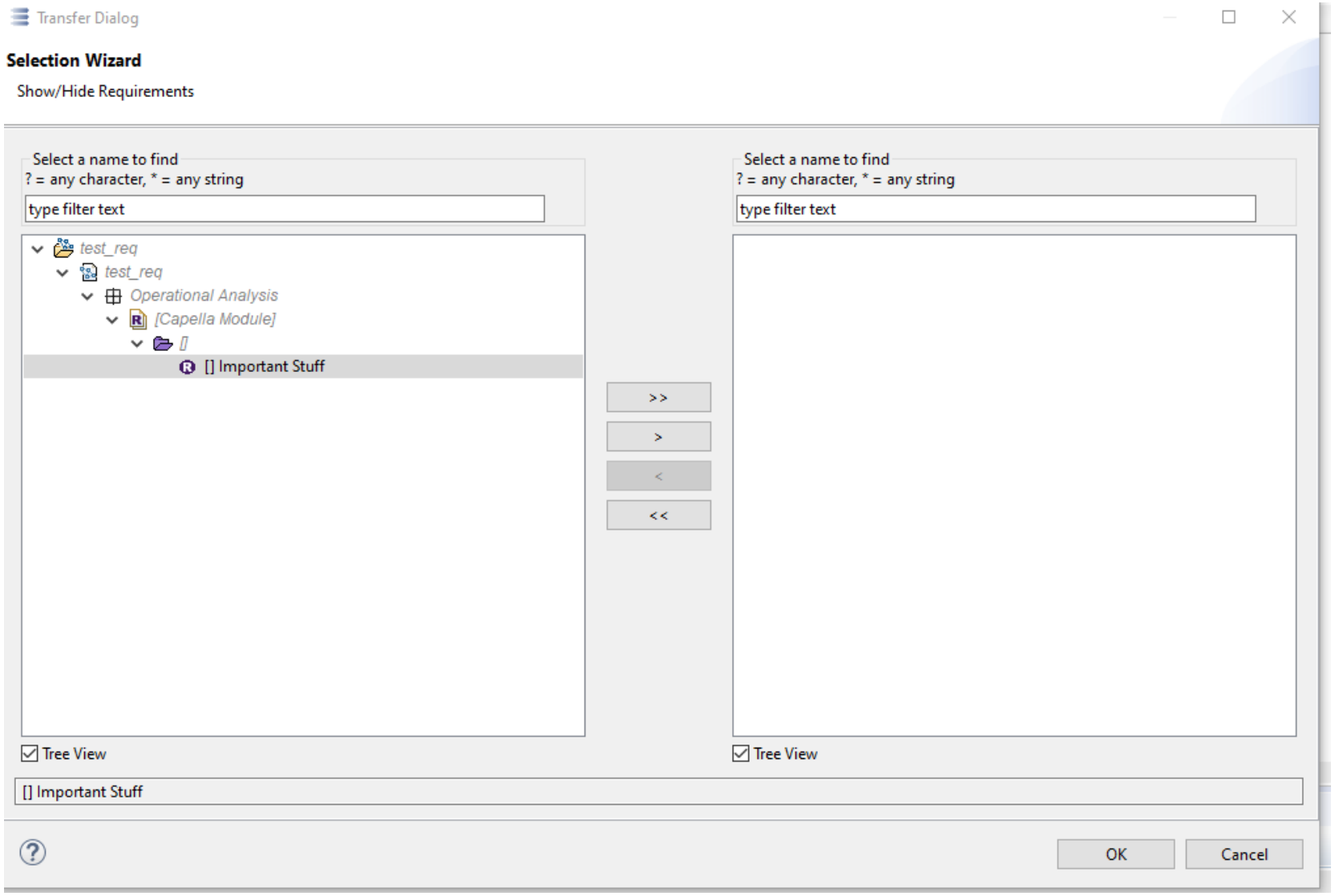

The only way to create requirements is through the Project Explorer.

[good side] Capella <could> connects to Doors ($$$) to import requirements.

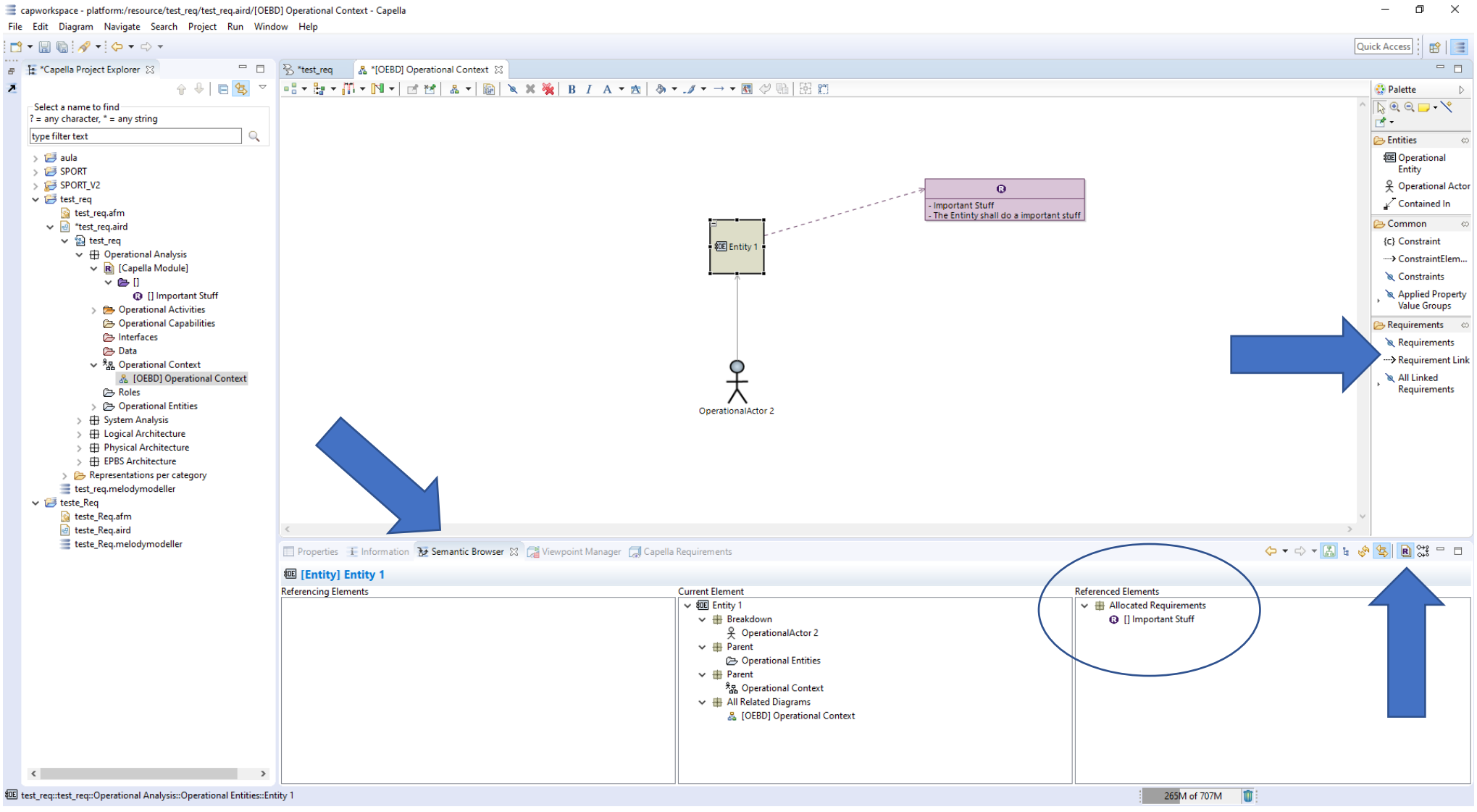
# REQUIREMENTS CAN BE USED IN ANY VIEW



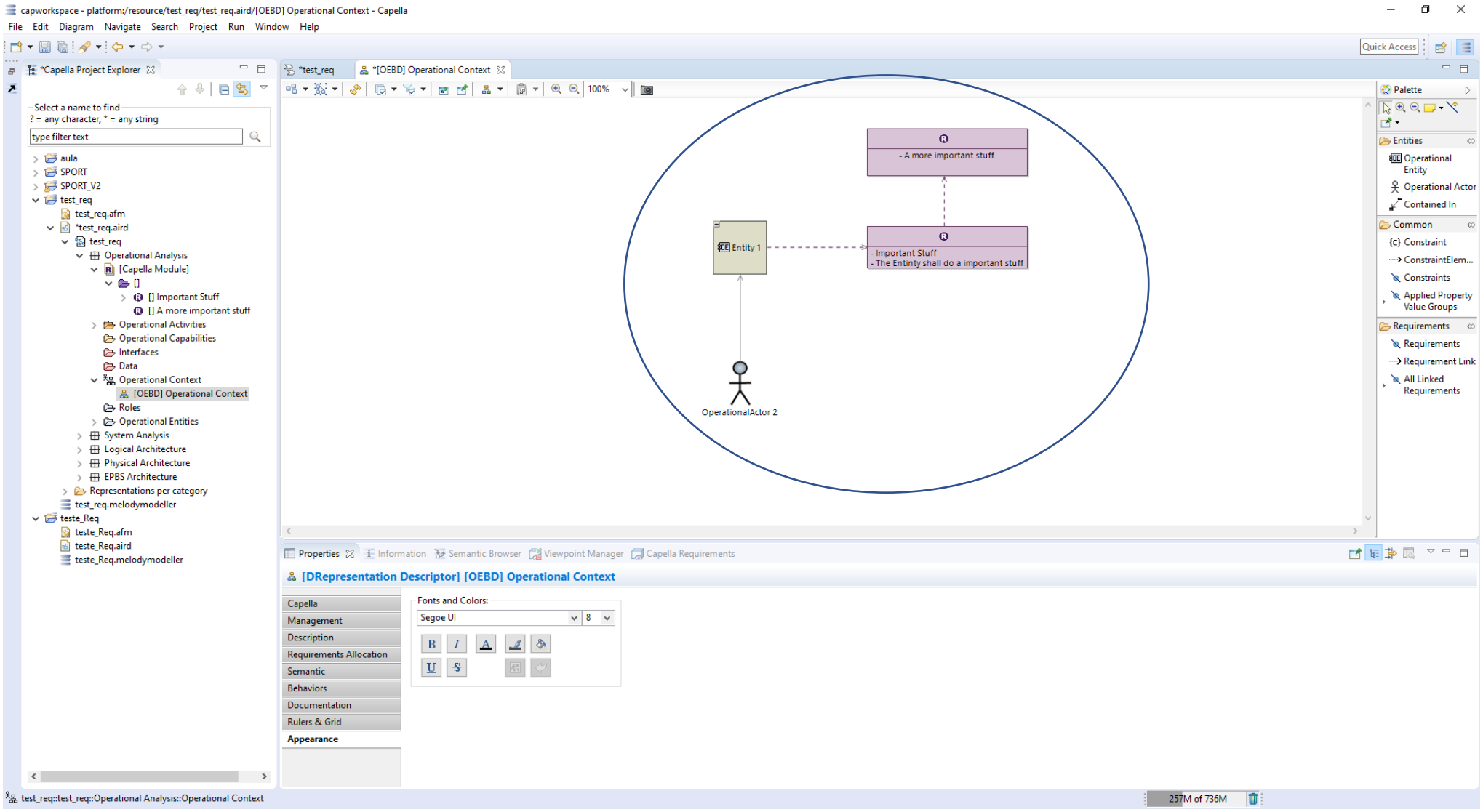
# SELECT THE REQUIREMENTS THAT WANT TO USE IN THE VIEW.



# ADD A LINK / CHECK RELATIONS

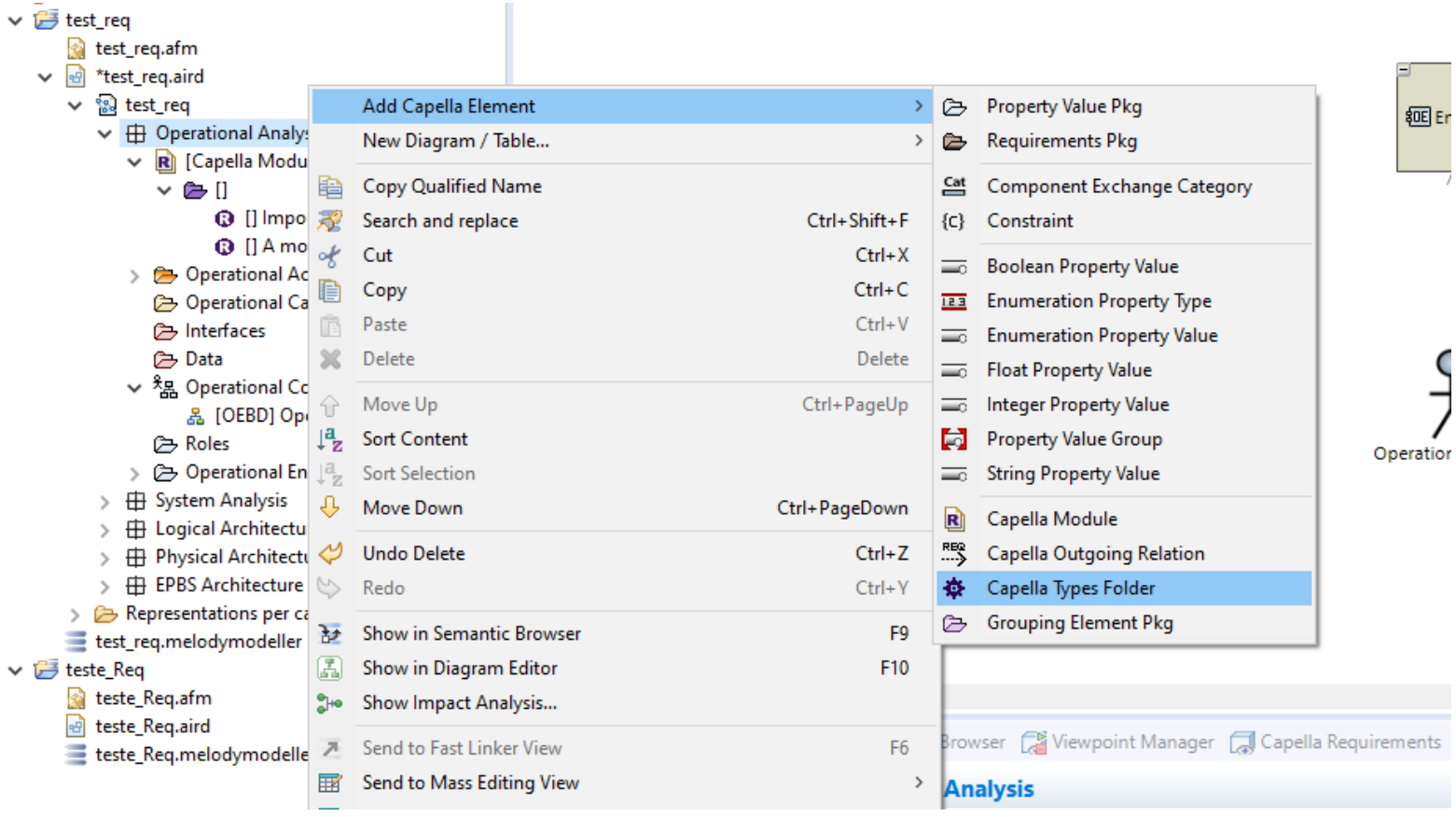


# REQUIREMENT TREES



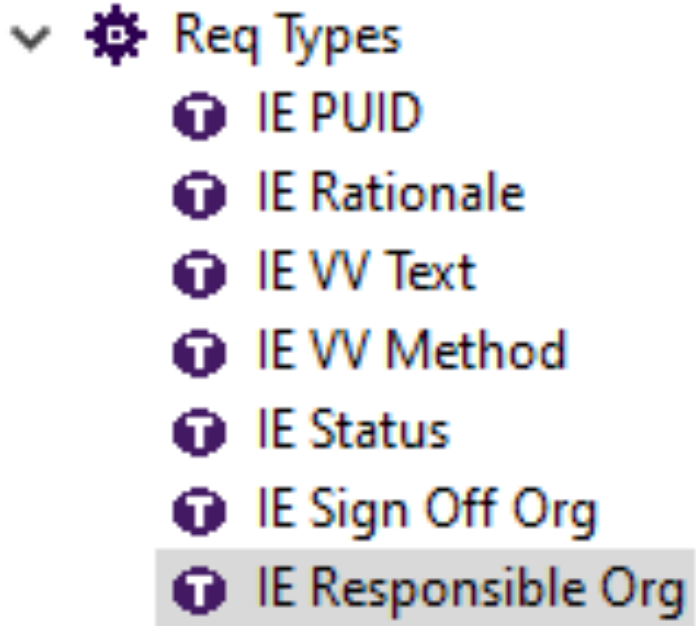
# ADD REQUIREMENT METADATA

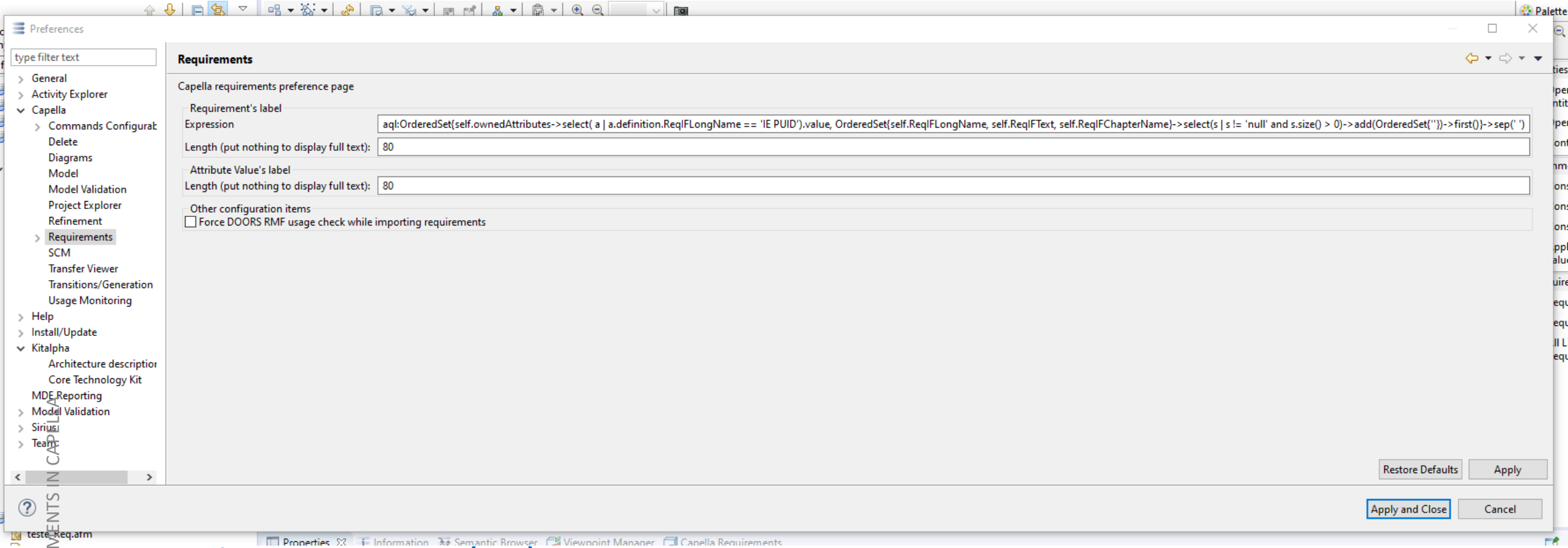
- It is required to create a new Type
- Create a Capella Types Folder → Rename Req Types



# REQUIREMENT DATA TYPE DEFINITIONS

- IE PUID (Requirement ID – name comes from DOORS)
- IE Rationale
- IE Verification Text
- IE Verification Method Expected
- IE Requirement Status
- IE Sign off Org
- IE Responsible Org



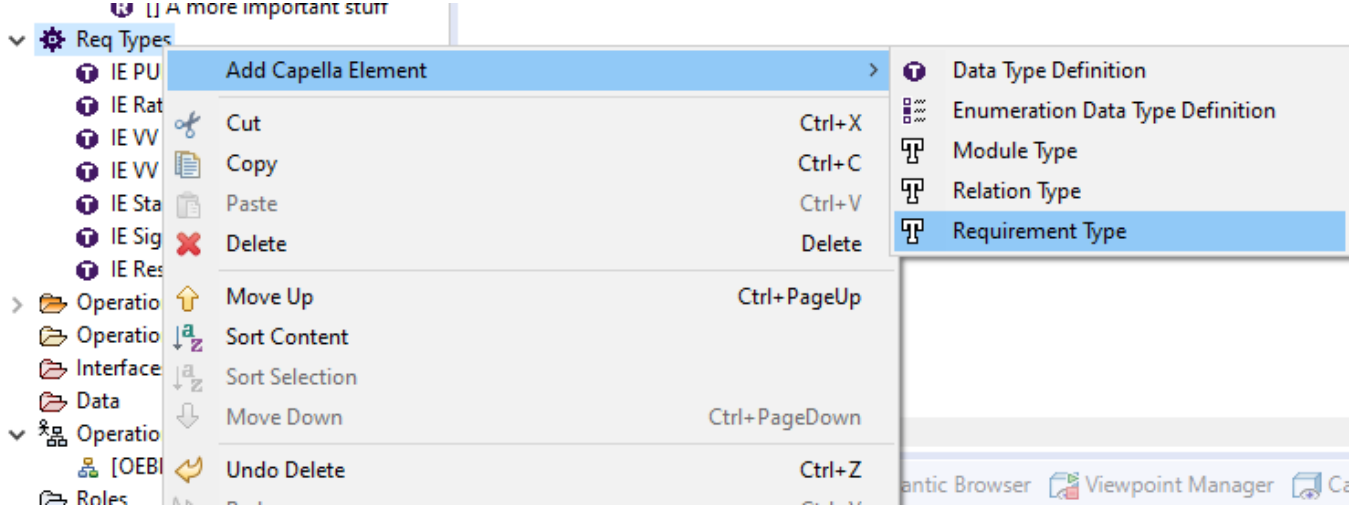


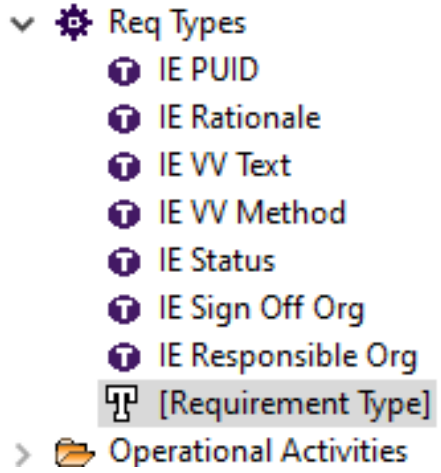


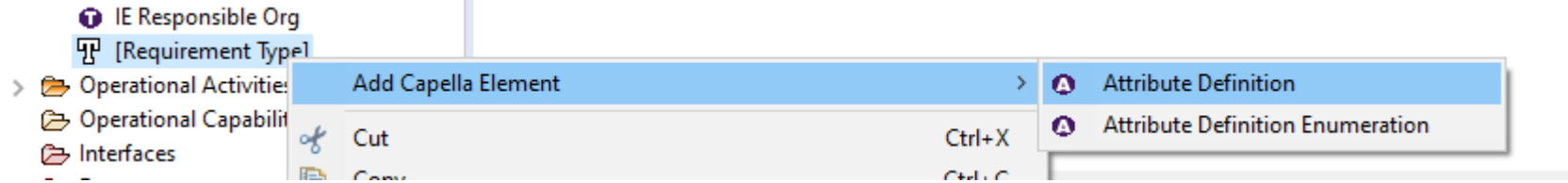

## Annotation Query Language (AQL)

- aql:OrderedSet{self.ownedAttributes->select( a | a.definition.ReqIFLongName == '**IE PUID**').value, OrderedSet{self.ReqIFLongName, self.ReqIFText, self.ReqIFChapterName}->select(s | s != 'null' and s.size() > 0)->add(OrderedSet{''})->first()}->sep(' ')

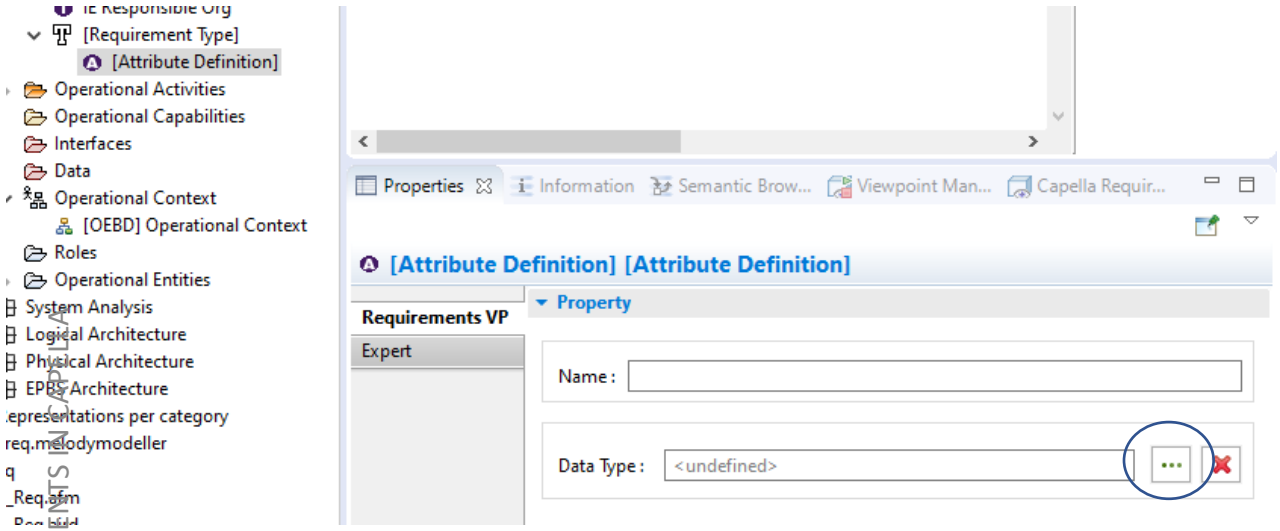
# CREATE THE REQUIREMENT TYPE THAT INCLUDE THE DATA TYPES AS ATTRIBUTES

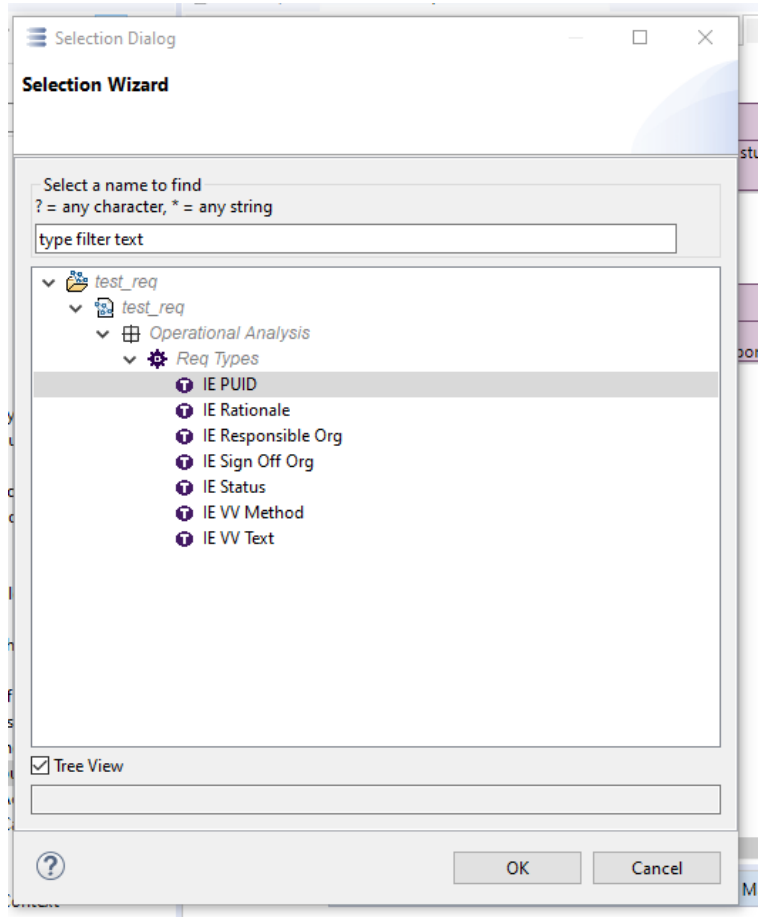


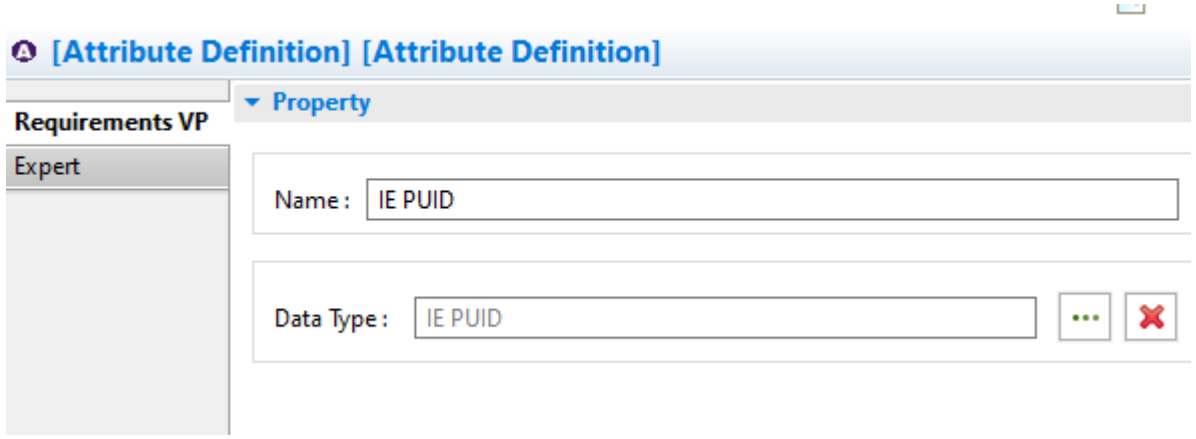



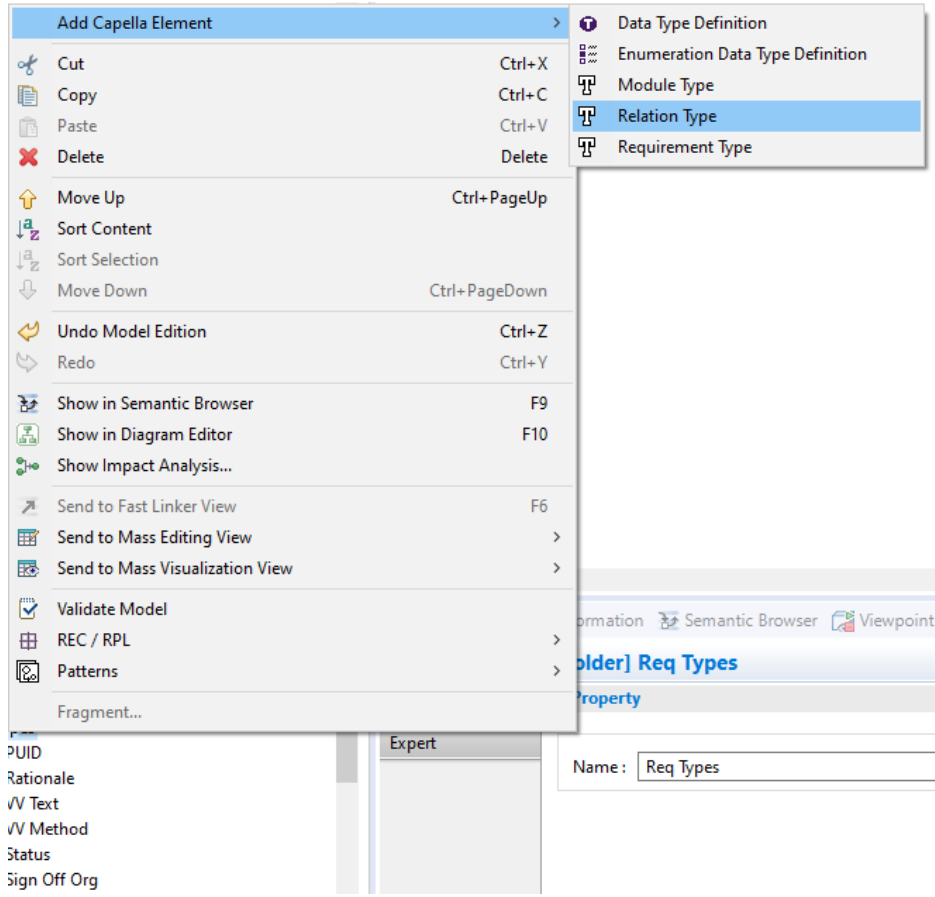
# CONFIGURE THE ATTRIBUTE

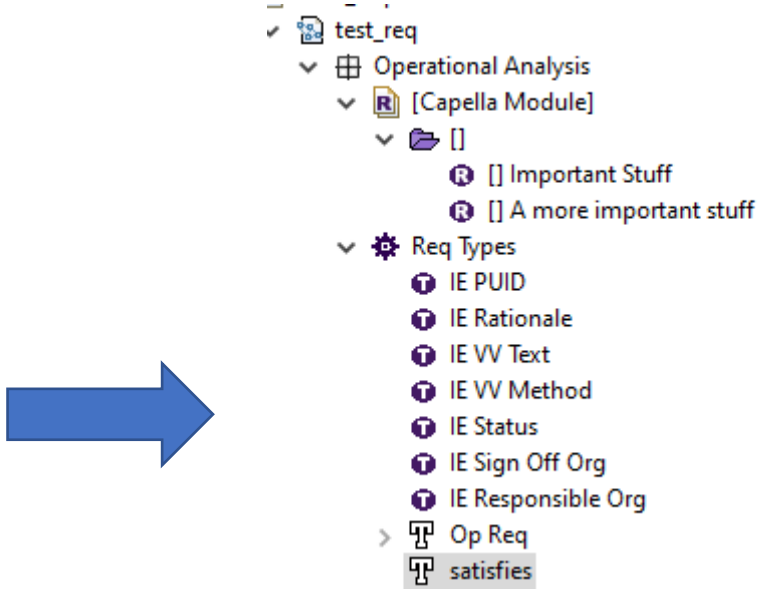


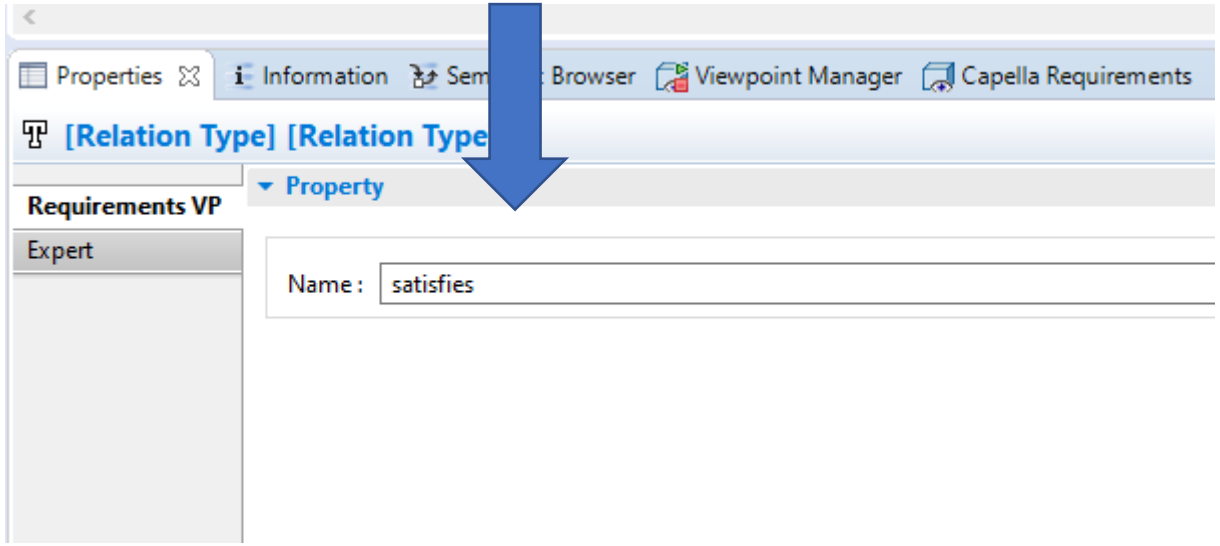



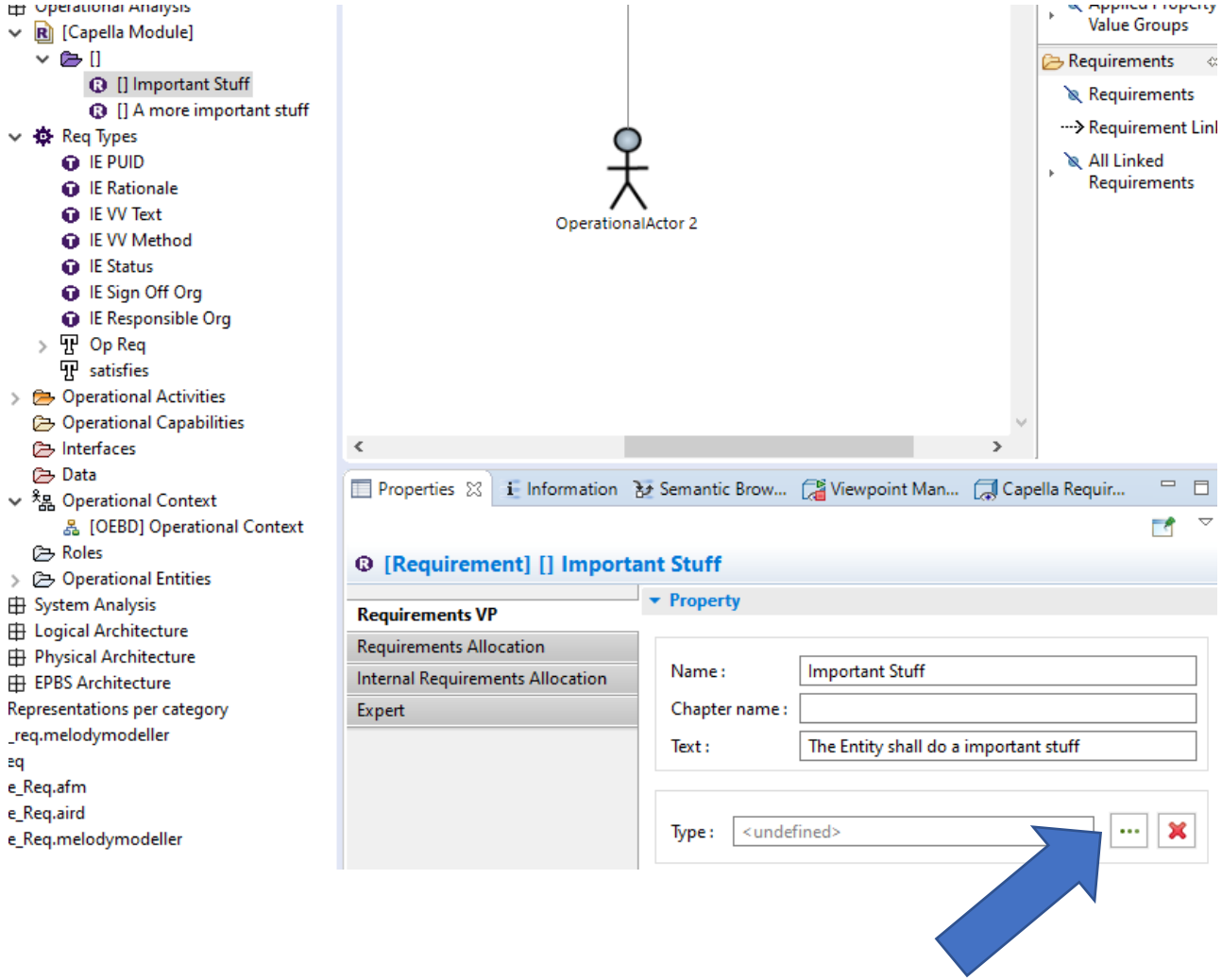
# ADD RELATION METADATA

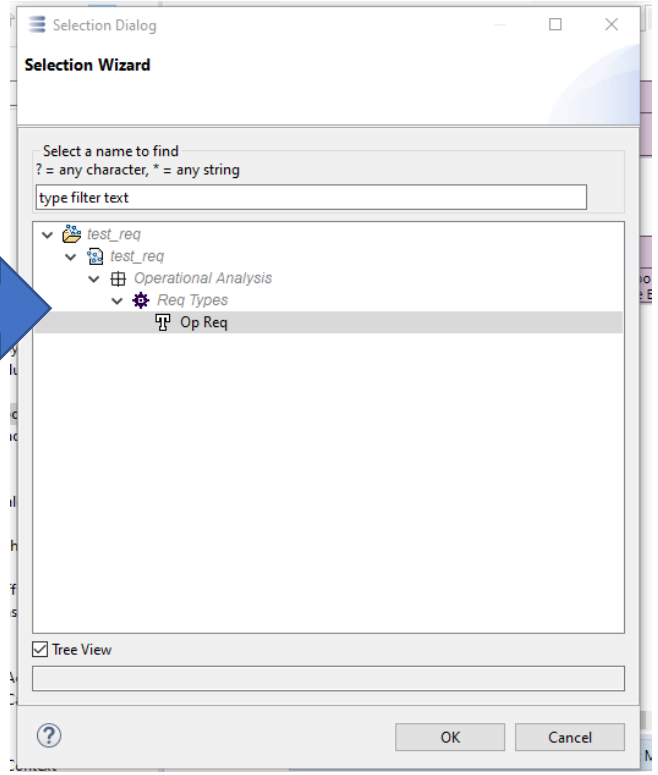


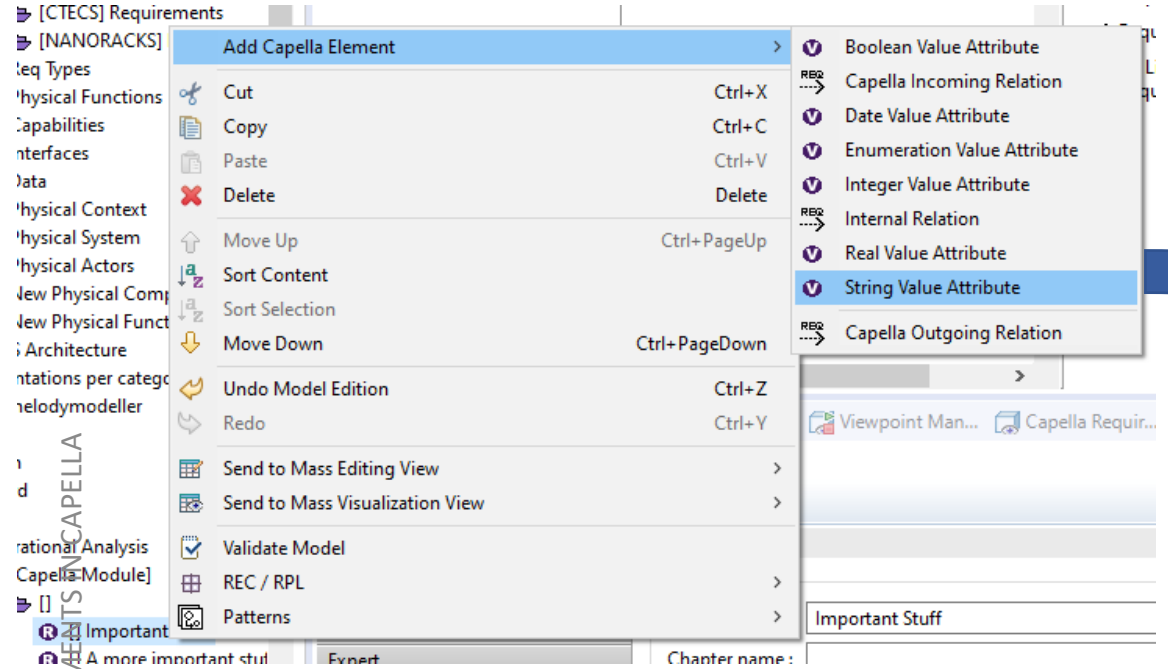



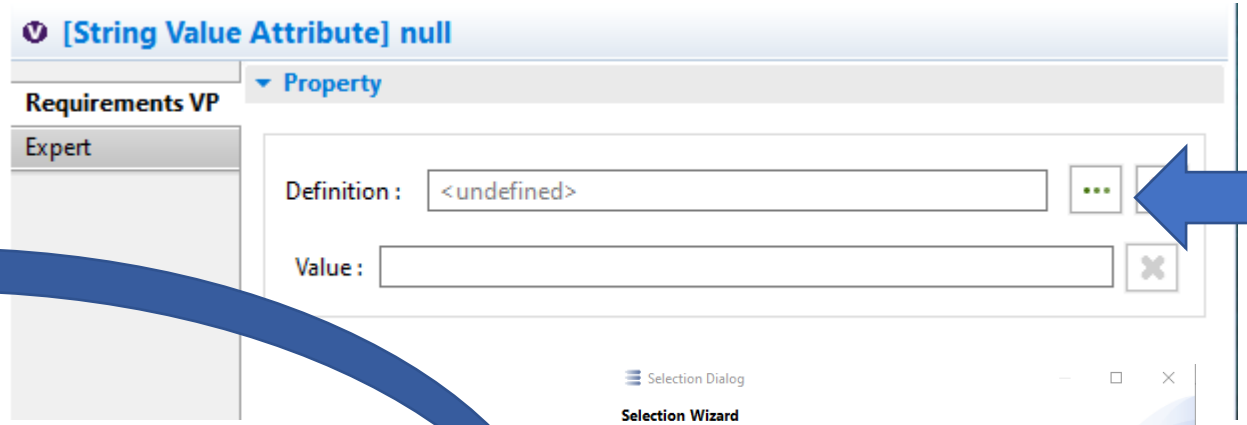
# APPLY TO THE REQUIREMENT SET

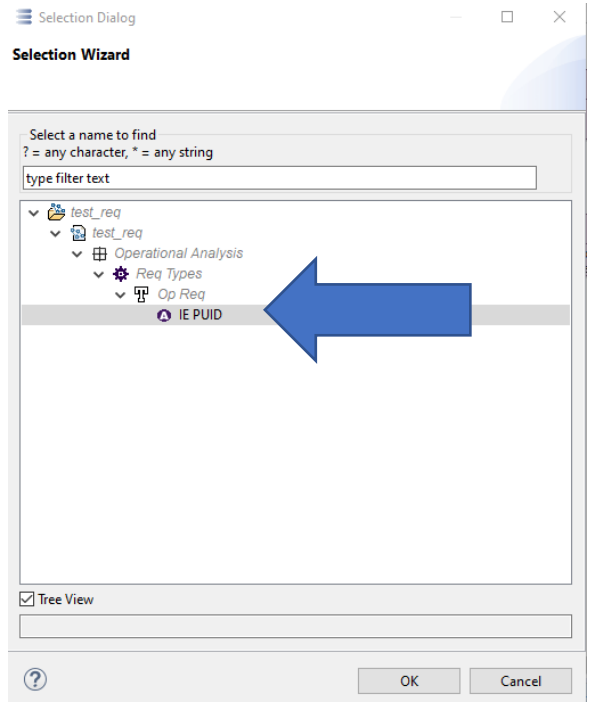


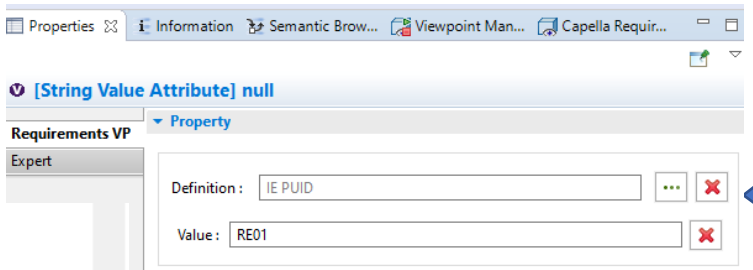



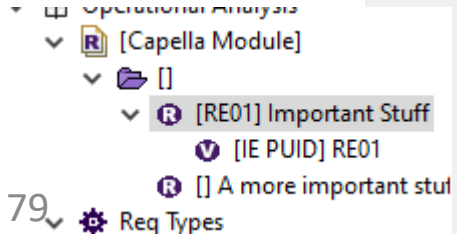
# CREATING THE ATTRIBUTES




REQUIREMENTS IN CAPELLA

# EACH LAYER HAS A “DEFAULT” REQ RELATION TABLE

- Operational:
  - Activities X Requirements
- System:
  - System Function X Requirements
- Logical
  - Logical Functions x Requirements
  - Logical Component x Requirements
  - Logical Architecture Requirement Refinements
- Physical
  - Physical Functions x Requirements
  - Physical Component x Requirements
- EPBS
  - Configuration Itens x Requirements
  - EPBS Requirement Refinements

# EVERYTHING IS WRITTEN IN XMI



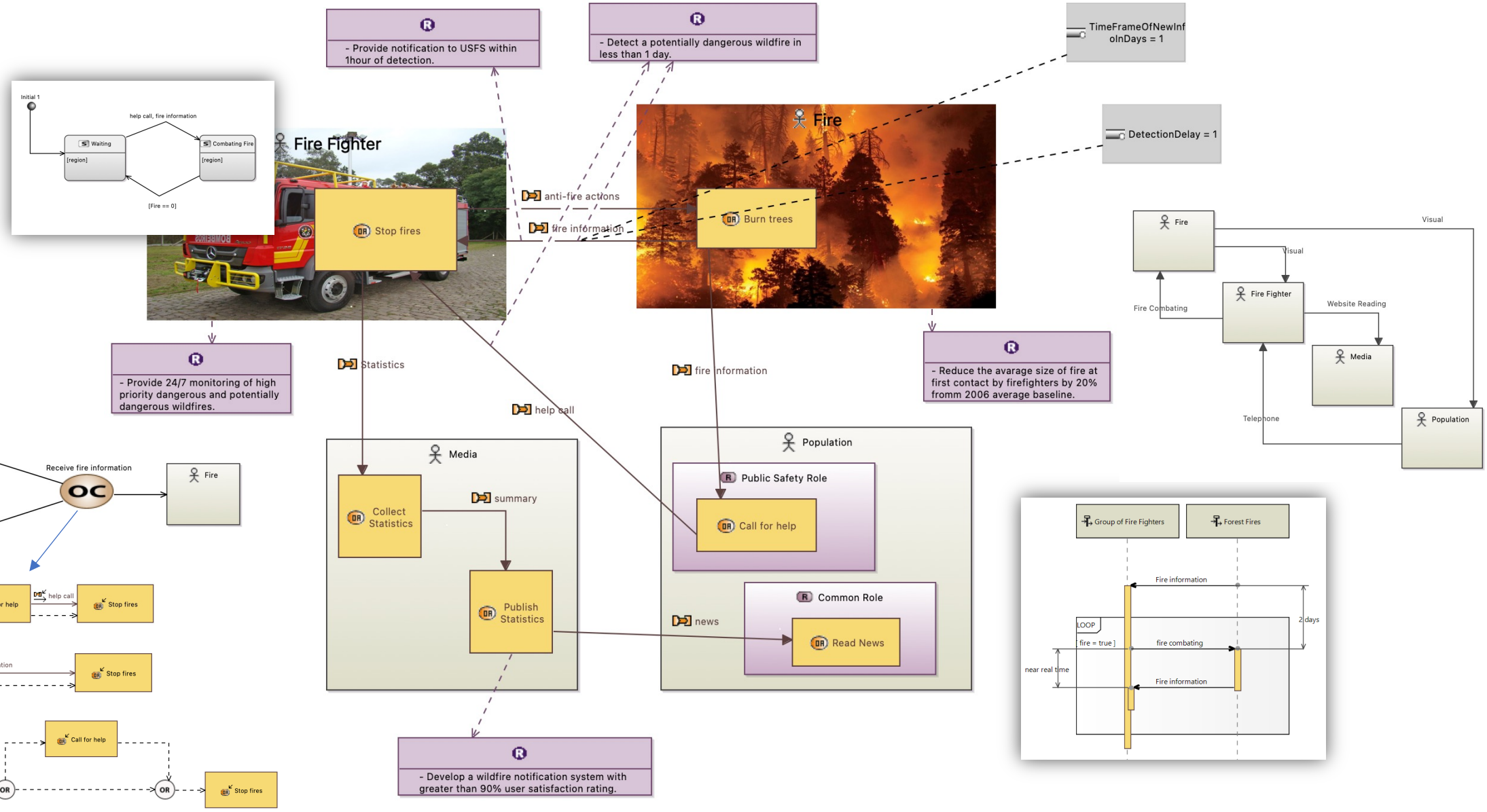
```
181  <ownedDiagramElements xmi:type="diagram:DNodeList" uid="_t5WE8PZlEem5oexdVu1e8g" name="RE01" incomingEdges="_t5dZsPZlEem5oexdVu1e8g">
182    <target xmi:type="Requirements:Requirement" href="test_req.melodymodeller#192c0814-e2aa-40f2-b5b9-09f1c3abf731"/>
183    <semanticElements xmi:type="Requirements:Requirement" href="test_req.melodymodeller#192c0814-e2aa-40f2-b5b9-09f1c3abf731"/>
184    <arrangeConstraints>KEEP_LOCATION</arrangeConstraints>
185    <arrangeConstraints>KEEP_SIZE</arrangeConstraints>
186    <arrangeConstraints>KEEP_RATIO</arrangeConstraints>
187    <ownedStyle xmi:type="diagram:FlatContainerStyle" uid="_t5WsAPZlEem5oexdVu1e8g" borderSize="1" borderSizeComputationExpression="1" borderColor="114,73,110" bac
188      <description xmi:type="style:FlatContainerStyleDescription" href="platform:/plugin/org.polarsys.capella.vp.requirements.design/description/CapellaRequirement
189    </ownedStyle>
190    <actualMapping xmi:type="description_1:ContainerMapping" href="platform:/plugin/org.polarsys.capella.vp.requirements.design/description/CapellaRequirements.ode
191    <ownedElements xmi:type="diagram:DNodeListElement" uid="_t5bkgPZlEem5oexdVu1e8g" name="- Important Stuff&#xA;- The Entity shall do a important stuff">
192      <target xmi:type="Requirements:Requirement" href="test_req.melodymodeller#192c0814-e2aa-40f2-b5b9-09f1c3abf731"/>
193      <semanticElements xmi:type="Requirements:Requirement" href="test_req.melodymodeller#192c0814-e2aa-40f2-b5b9-09f1c3abf731"/>
194      <ownedStyle xmi:type="diagram:Square" uid="_t5bkgfZlEem5oexdVu1e8g" showIcon="false" labelPosition="node">
195        <description xmi:type="style:SquareDescription" href="platform:/plugin/org.polarsys.capella.vp.requirements.design/description/CapellaRequirements.od
196      </ownedStyle>
197      <actualMapping xmi:type="description_1:NodeMapping" href="platform:/plugin/org.polarsys.capella.vp.requirements.design/description/CapellaRequirements.od
198    </ownedElements>
199  </ownedDiagramElements>
200  <ownedDiagramElements xmi:type="diagram:DEdge" uid="_t5dZsPZlEem5oexdVu1e8g" sourceNode="_nwDn8PZkEem5oexdVu1e8g" targetNode="_t5WE8PZlEem5oexdVu1e8g">
```

# ANÁLISE DO CONTEXTO

# IDENTIFICAÇÃO DOS STAKEHOLDERS

- Levantar quem são
- O que querem
- Quais as mudanças desejadas na situação atual
- Capturar as métricas de sucesso (MoEs)
- Levantar

# ANÁLISE OPERACIONAL

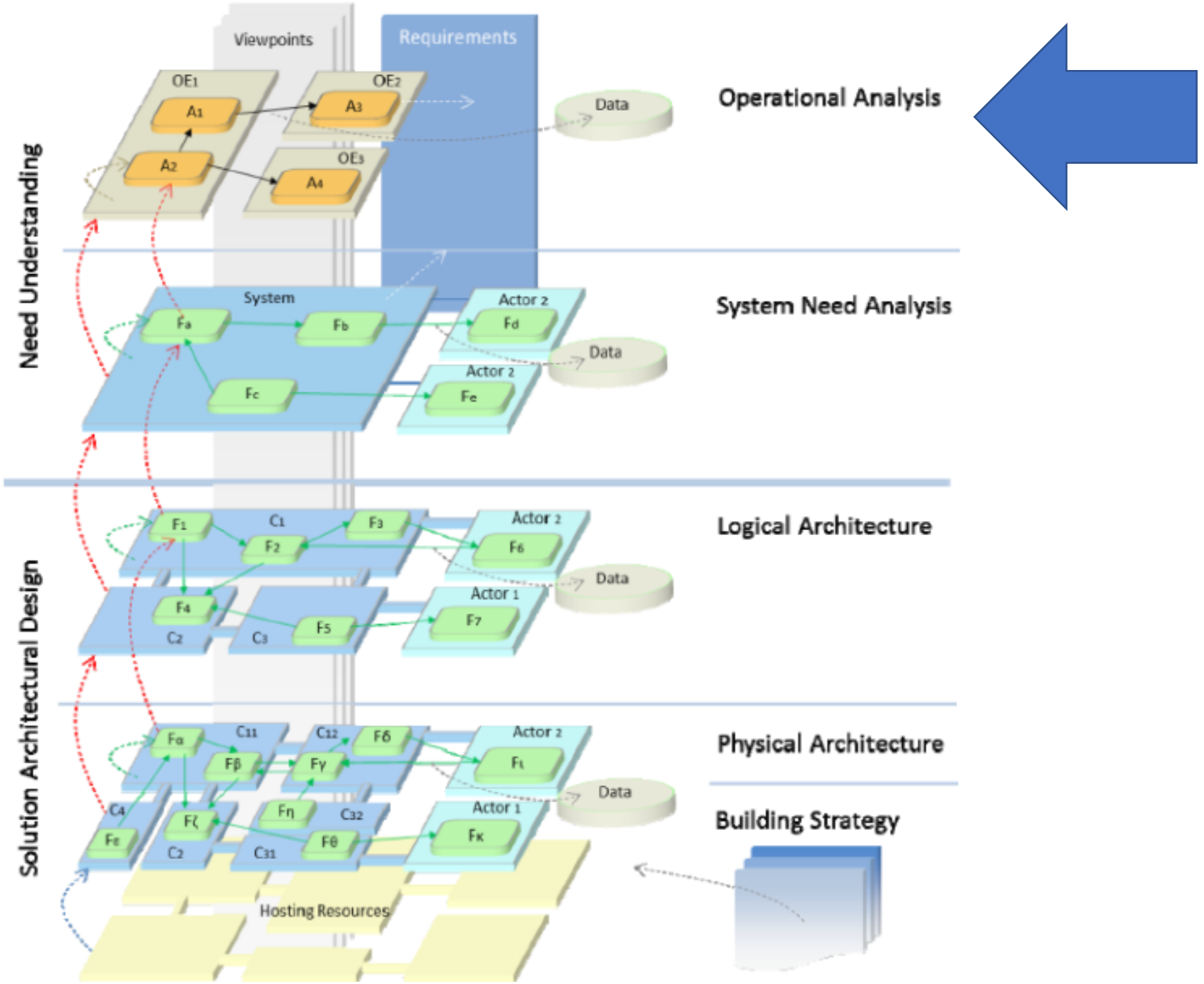


# FAB: PUBLICAÇÃO DA NOP

- Descrever os stakeholders (OMs)
- Descrever o conjunto de documentos originadores
- Estruturar as propostas de necessidades
- Descreve a situação atual com a mudança que precisa existir.
- Rastrear o desejo de mudança com a arquitetura da situação atual
- Justificar conjunto de necessidades.
  - Isento de solução

# ARCADIA – ANÁLISE OPERACIONAL

1h


Viewpoints
Requirements
OE1
OE2
OE3
A1
A2
A3
A4
Data
Need Understanding
Operational Analysis
System
Fa
Fb
Fc
Actor 2
Fd
Fe
Data
System Need Analysis
F1
C1
F2
F3
Actor 2
F6
Data
F4
C2
C3
F5
Actor 1
F7
Logical Architecture
Fα
C11
Fβ
C12
Fγ
Fδ
Actor 2
Fι
Data
C4
Fε
Fζ
C2
Fη
C31
C32
Fθ
Actor 1
Fκ
Physical Architecture
Building Strategy
Hosting Resources
Solution Architectural Design

# O QUE É ANÁLISE OPERACIONAL (OA)

# ANÁLISE OPERACIONAL

***"O que os usuários do sistema devem alcançar"***

***"O que os usuários do sistema precisam realizar"***

- Essa perspectiva analisa os usuários operacionais, identificando os atores que podem interagir com o sistema, seus objetivos, atividades, restrições e as interações entre eles.

OA **NÃO deve mencionar o sistema**, *para não barrar alternativas potencialmente interessantes para alcançar a satisfação das necessidades dos clientes.*

- EXEMPLO - Suponha que a necessidade do cliente é poder ter um espelho em uma parede.

*Se essa necessidade é traduzida muito rapidamente em "como pendurar um espelho?"*

- Isso exclui prematuramente outras possibilidades (como o uso de cola, por exemplo),
- E, também, critérios que ajudariam a orientar o processo para a solução certa (como a necessidade ou não de poder desmontar o espelho depois).

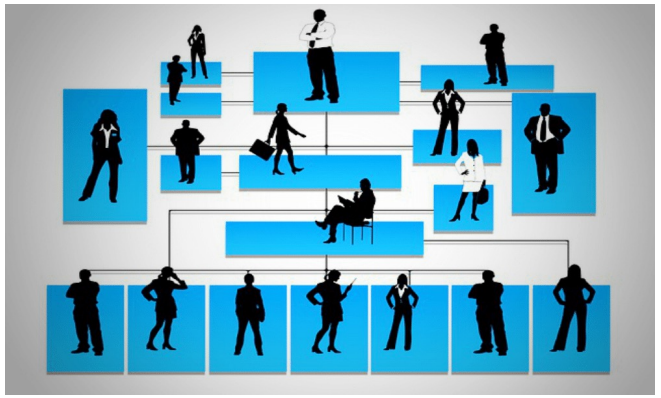
# MÉTODO ARCADIA – RESUMO DA ANÁLISE OPERACIONAL

| | |
|---|---|
| **Definir missões e capacidades operacionais necessárias** | determinar as missões dos futuros usuários do sistema e do ambiente – ou, mais geralmente, suas motivações, expectativas, metas, objetivos, intenções, etc., bem como as capacidades necessárias para assumir essas missões. |
| **Realizar análise de necessidades operacionais** | captar as condições para a realização de uma missão previamente identificada, e as condições para a implementação das capacidades associadas, principalmente através das atividades e interações dos principais atores que contribuem para a mesma. |

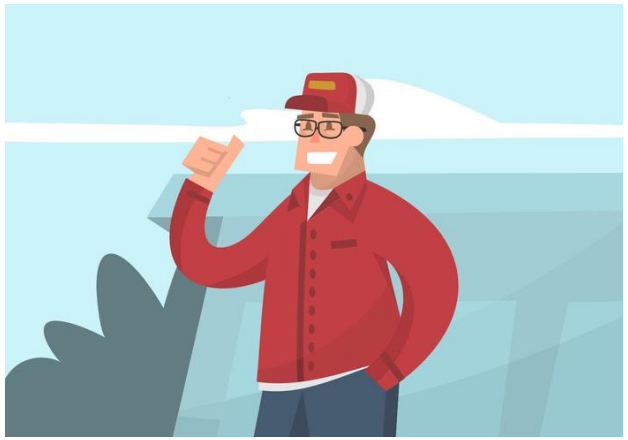
# VOCABULÁRIO DO OA

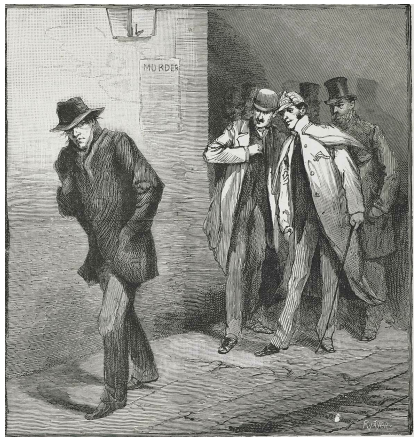
- **Capacidade Operacional (Operational Capability)**: **capacidade de fornecer um "serviço"** que leve a um objetivo operacional a ser atingido (*por exemplo, Fornecer previsões meteorológicas, etc.*); - Objetivos de alto nível


- **Entidade Operacional (Operational Entity)**: entidade **pertencente ao mundo real** (organização, sistema existente, etc.) cuja função é **interagir com o sistema em estudo ou com seus usuários** (por exemplo, Tripulação, Navio, etc.);



- **Ator Operacional (Operational Actor)**: caso particular de **uma entidade operacional (humana) não decomponível** (por exemplo, Piloto, etc.);



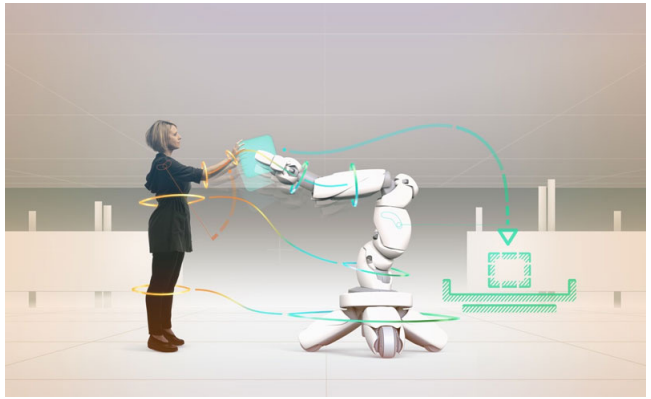


- **Atividade Operacional (Operational Activity)**: **ação realizada por uma entidade operacional**, que pode precisar usar o futuro sistema para fazê-lo (por exemplo, detectar uma ameaça, coletar dados meteorológicos, etc.);

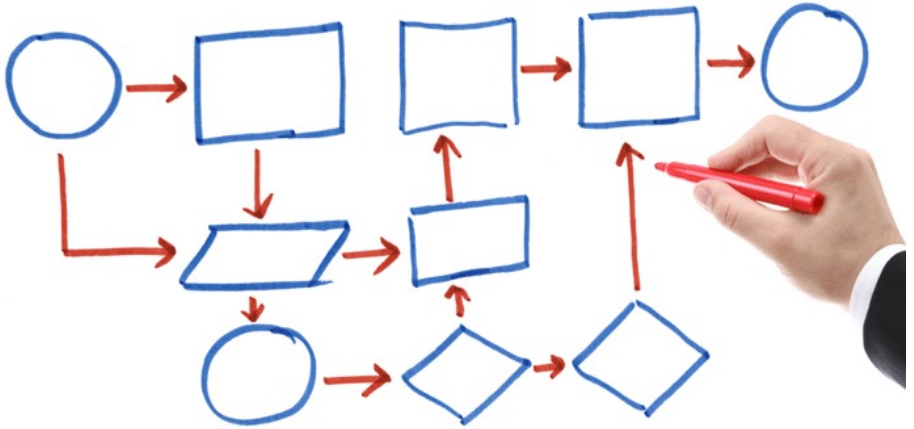
- **Interação Operacional (Operational Interaction): intercâmbio de matéria/energia/informação** unidirecional entre atividades operacionais (por exemplo, dados meteorológicos, etc.);


- **Meio de interação (Communication mean)** – Conexão entre os stakeholders (entidades e atores), relações que permitem que as interações acontecerem.

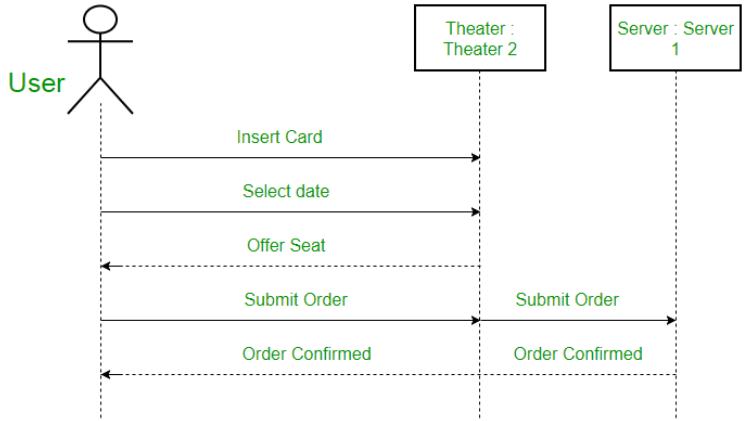


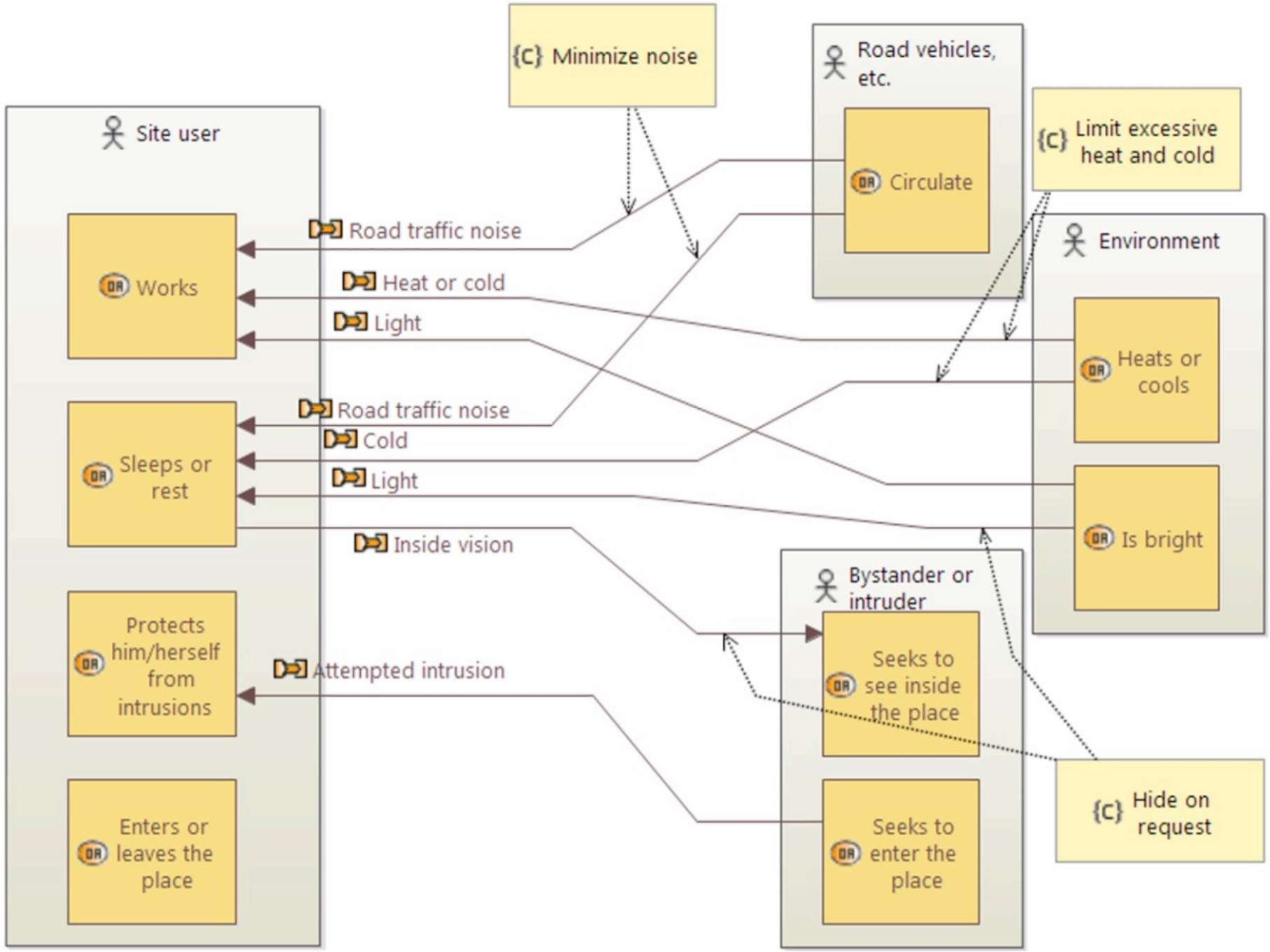


- **Processo (Operational Process)**: **série de atividades e de interações** que contribuem para uma capacidade operacional.

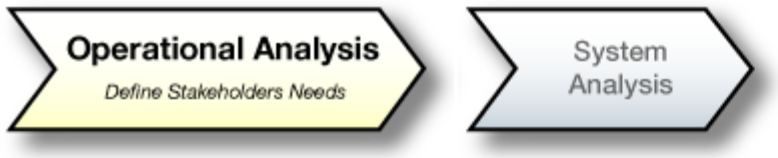



- **Cenário Operacional (Operational Scenario)**: cenário que descreve o **comportamento de entidades e/ou atividades operacionais no contexto de uma capacidade operacional**.




{C} Minimize noise
Road vehicles, etc.
Site user
Circulate
{C} Limit excessive heat and cold
Environment
Road traffic noise
Works
Heat or cold
Light
Heats or cools
Road traffic noise
Cold
Sleeps or rest
Light
Is bright
Inside vision
Bystander or intruder
Protects him/herself from intrusions
Attempted intrusion
Seeks to see inside the place
{C} Hide on request
Enters or leaves the place
Seeks to enter the place

# DIAGRAMAS




**Define Operational Entities and Capabilities**


[OEBD] Create a new Operational Entity Breakdown diagram


[OCB] Create a new Operational Capabilities diagram

Identificar o domínio operacional: quem são os atores e entidades, quais são suas finalidades? As atividades dão uma visão global sobre os objetivos operacionais do negócio.

**Define Operational Activities and describe Interactions**


[OABD] Create a new Operational Activity Breakdown diagram


[OAIB] Create a new Operational Activity Interaction diagram

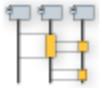

[OAS] Create a new Operational Activity Scenario

Detalhar a divisão da atividades operacionais, descrever as interações entre entidades e modelar os processos.

**Allocate Operational Activities to Operational Actors, Entities or Roles**

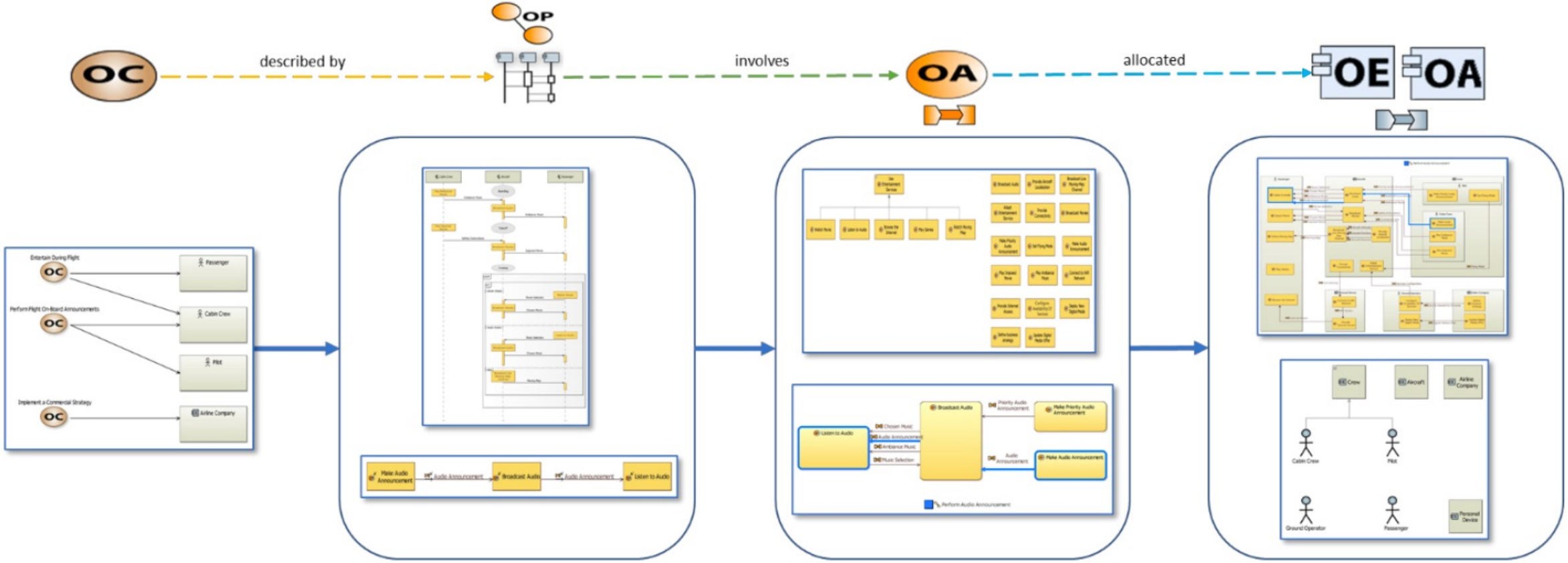

[OAB] Create a new Operational Architecture diagram


[ORB] Create a new Operational Role diagram


[OES] Create a new Operational Entity Scenario

Os atores e entidades operacionais são responsáveis pela execução das atividades operacionais. Gerenciar alocações e deduzir meios de comunicação entre entidades.
Criar cenários para ilustrar interações entre os atores operacionais e as entidades

**Transverse Modeling**




| Capability diagram | Capability description diagrams | Functional diagrams | Structural diagrams |
|---|---|---|---|
| [OCB] Operational Capabilities | [OAS] Operational Activity Scenario<br>[OPD] Operational Process Scenario<br>[OES] Operational Entity Scenario | [OABD] Operational Activity Breakdown Diagram<br>[OAIB] Operational Activity Interaction Blank | [OEBD] Operational Entities Blank Diagram<br>[ORB] Operational Roles Blank<br>[OAB] Operational Architecture Blank |

Figure 4.10: Operational Analysis model elements and diagrams traceability

https://www.slideshare.net/HelderCastro3/mbse-with-arcadia-methodpdf-256664096

# CONSIDERAÇÕES FINAIS

# ATIVIDADES

- AI-09 - Resumo sobre o documento da SAE sobre Engenharia de Sistemas na área aeronáutica

- AG-10 - Preparar a Análise de Contexto
  - Apresentação na próxima aula.