INCOSE
Healthcare
Working Group
5th Annual Systems Engineering in Healthcare Conference
May 1-2, 2019
Minneapolis, MN

INCOSE

Introducing Arcadia and Capella: deploying MBSE at large using an open source tool

Our journey today

Who are we

Technical foundations: Arcadia and Capella

MBSE Use Cases

Key enablers for MBSE adoption

MBSE deployment actions

Going further

Who are we?

Thales in a nutshell

N°1 worldwide


Payloads for telecom satellites


Air traffic management


Sonars


Data protection services


Driverless metro signalling

N°2 worldwide

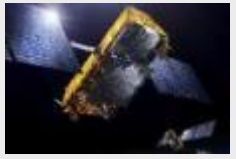

Rail signalling systems


In-flight entertainment and connectivity


Military tactical radiocommunications

N°3 worldwide


Commercial avionics


Civil satellites


Military surface radars

€14.9

billion
in revenues
(2017)

Thales in a nutshell

OBEO

- We create modeling tools for systems engineers, software engineers and domain modeling experts
- Open Source Software Vendor
- Global presence

The key enablers for MBSE adoption

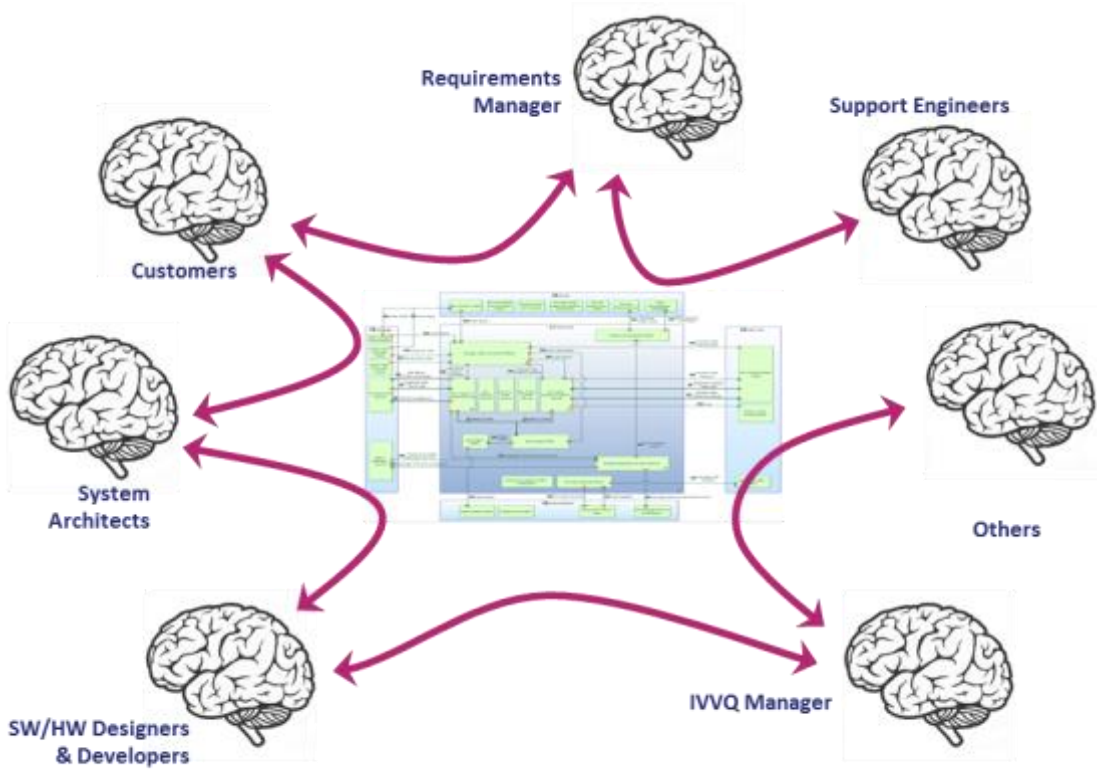
Why? - Company strategy: from equipment to global solution supplier

- (New) Systems
  - More complex missions, more constraints (safety, environmental impacts, weight, etc.)
  - More reliability, versatility and added value, less time to market
  - Increasing parts of SW
- (New) Systems Engineers / Architects roles
  - Plan, design, justify, integrate, validate, test, certify…
  - Frequently resolve complex issues while considering time, cost, risks.
  - Larger teams, with more development interdependencies

The case for a MBSE approach

- Systems rather than equipments
- Complex systems and products
- Highly competitive markets
- More demanding customers and stakeholders
- Multi-cultural and distributed engineering teams
- Heterogenous engineering workflows (top-down, bottom-up, legacy-based, iterative, …)

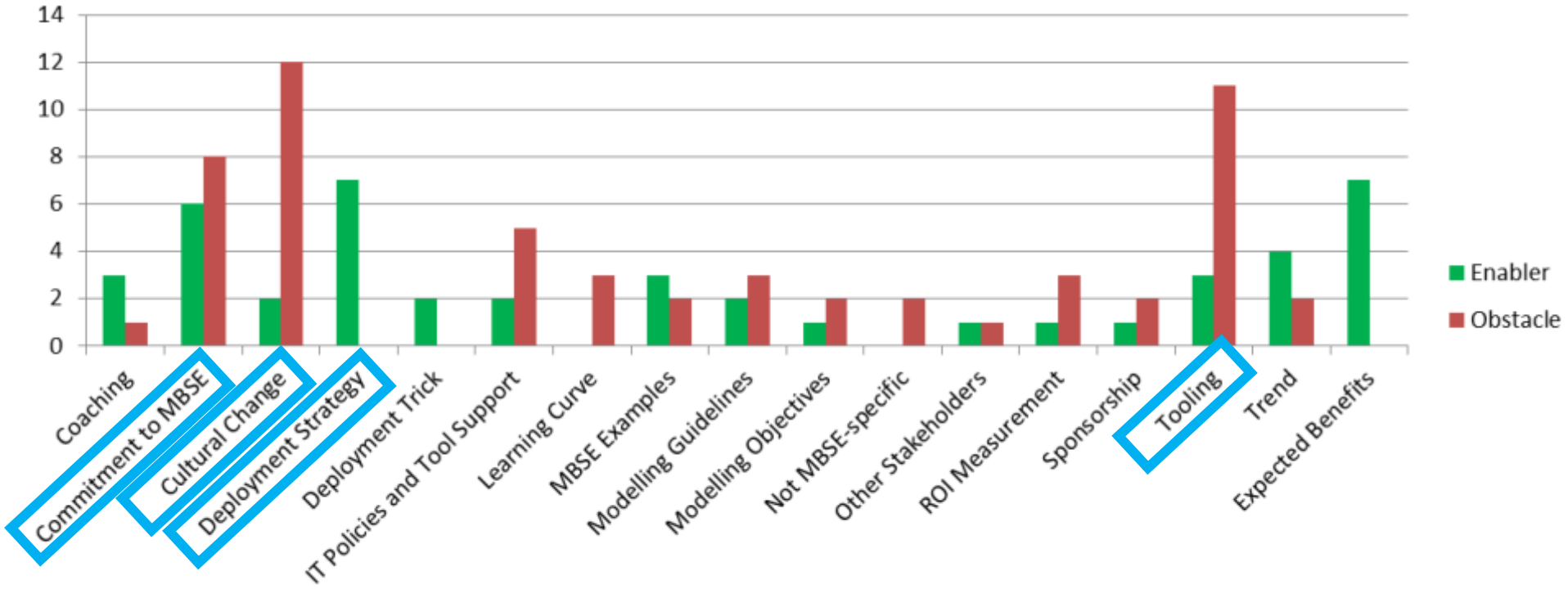
The case for a MBSE approach



- Methods and tools designed for (complex) systems engineering
- Common and simple modelling languages
- Efficient and customizable support for model-based workflows
- Support for knowledge capture and reuse
- Collaborative and distributed model development

Key enablers and obstacles for MBSE deployment

- Workshop held in INCOSE MBSE Symposium, Canberra, Australia, October 2014. 35 + 20 participants



Key enablers for MBSE deployment

Good foundations – Language, Method &Tool

Mobilize the organization towards the deployment of MBSE

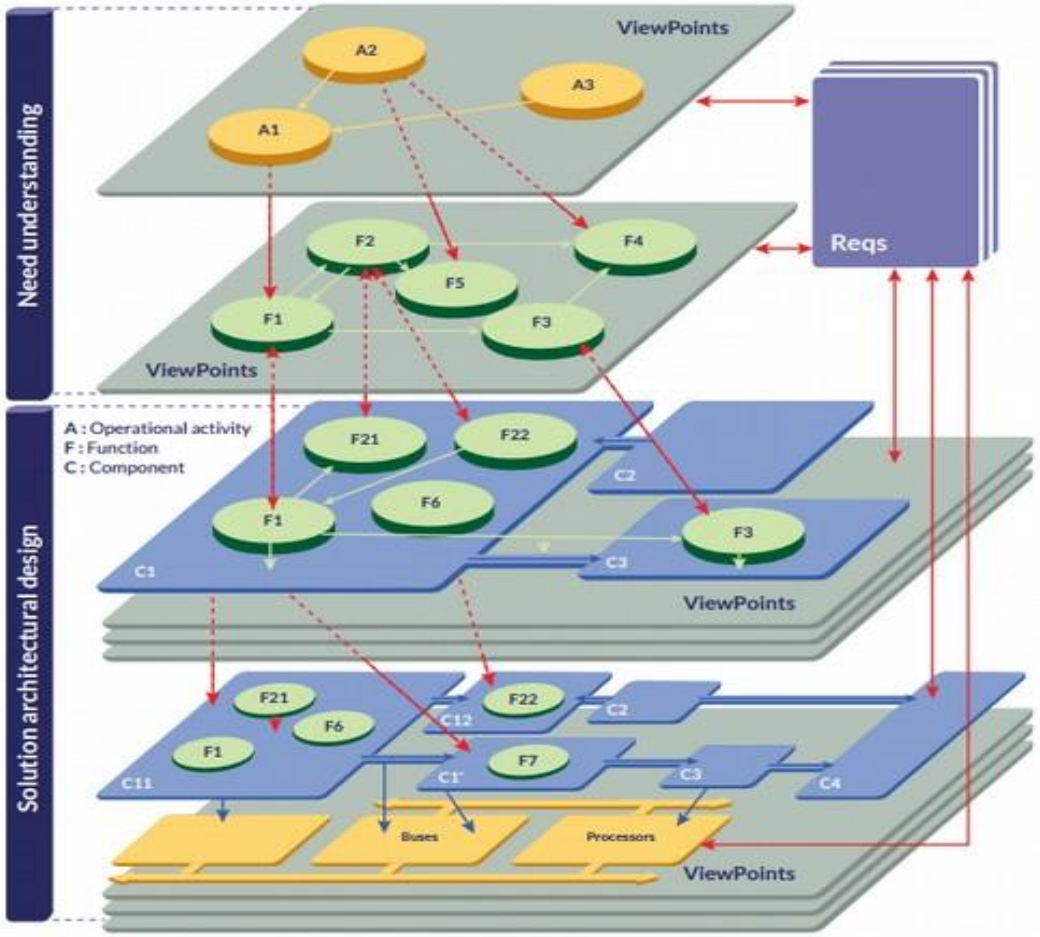
Key enablers for MBSE deployment

Good foundations – Language, Method &Tool

Mobilize the organization towards the deployment of MBSE

Technical Foundations: Arcadia Method and Language

ARCADIA

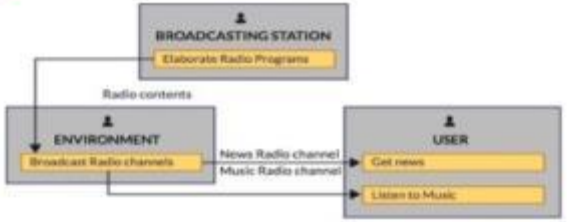
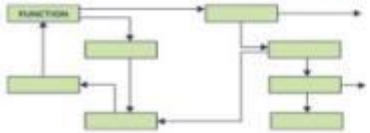
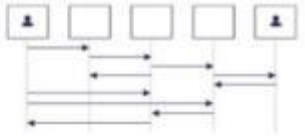
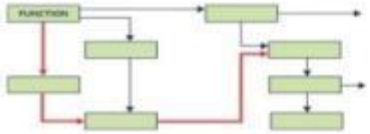
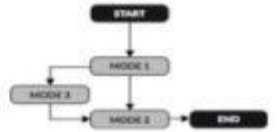
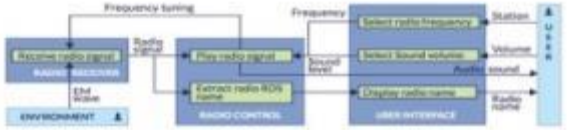
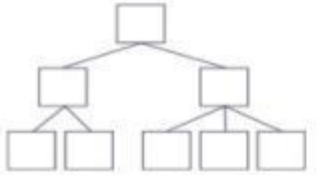
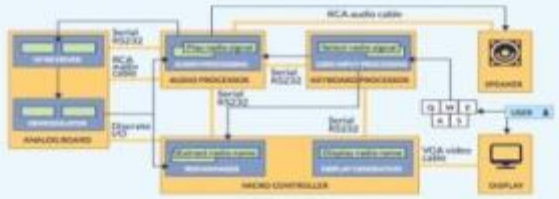
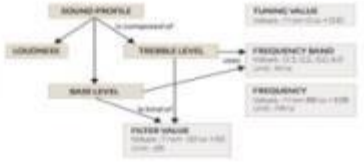
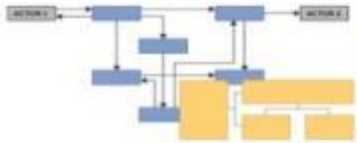
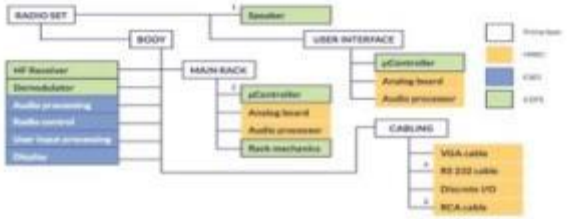
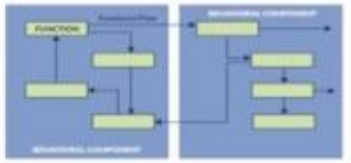


Operational Analysis
What the users of the system need to accomplish

Functional & Non Functional Need
What the system has to accomplish for the users

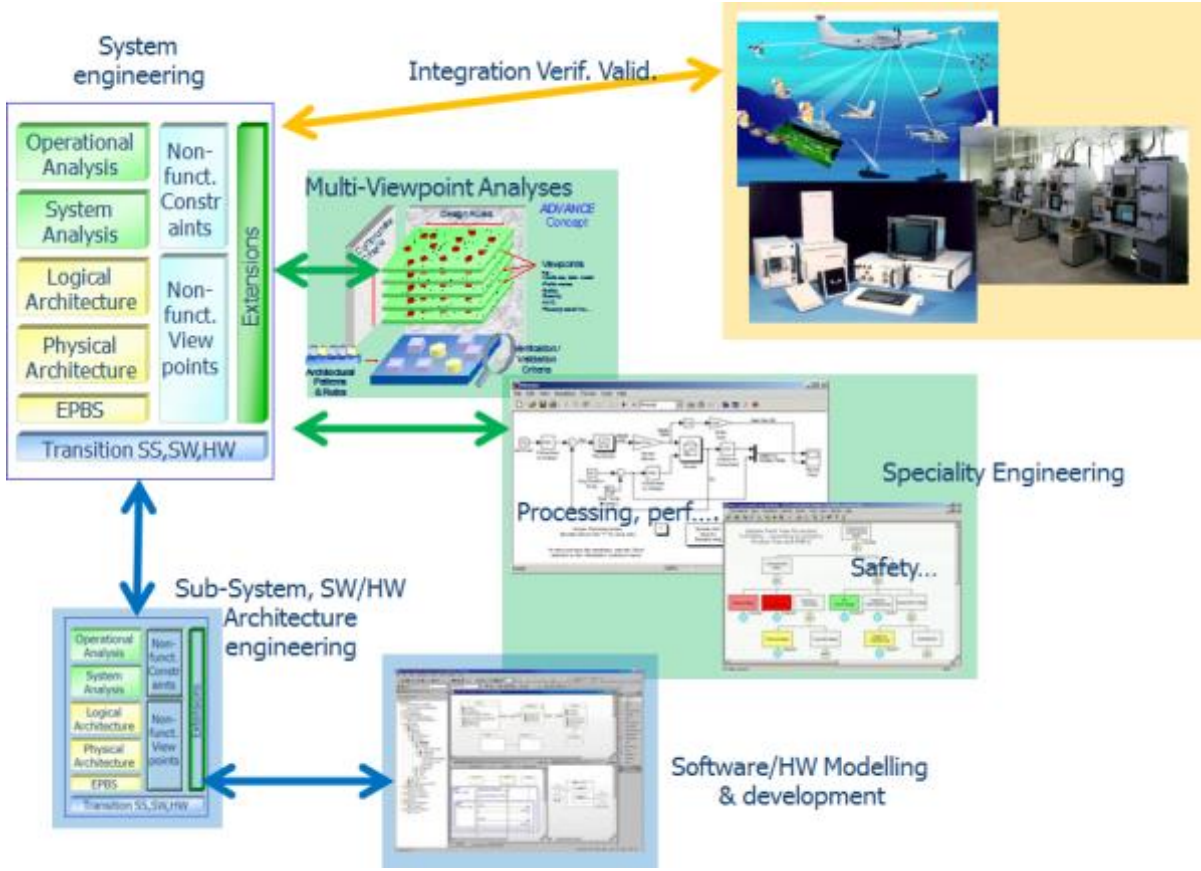
Logical Architecture
How the system will work to fulfill expectations

Physical Architecture
How the system will be developed and built

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

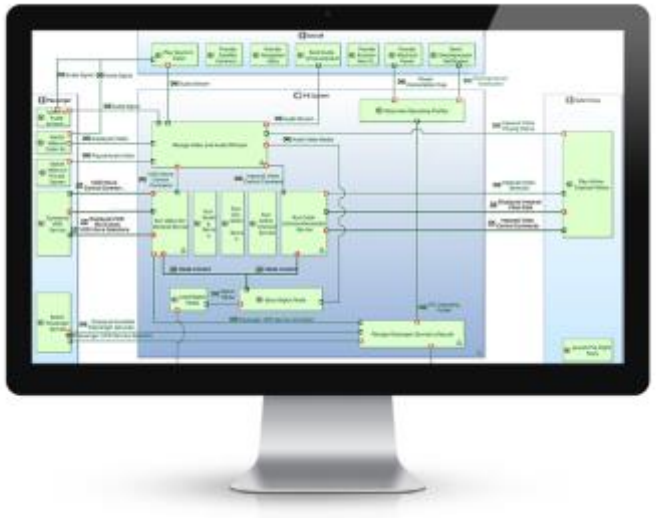
Technical Foundations: Method and Language

- Comprehensive coverage of engineering and architecture design
  - From requirements to integration and validation
  - From complex system to equipment and subsystem, SW, HW
  - From functional to non-functional constraints and engineering specialties



Arcadia main objectives

- Better understand the customer need
- Define and share the solution among stakeholders
- Secure SYS/SW/HW engineering, prepare subcontracting
- Early evaluate and justify architectural design
- Prepare and master V&V

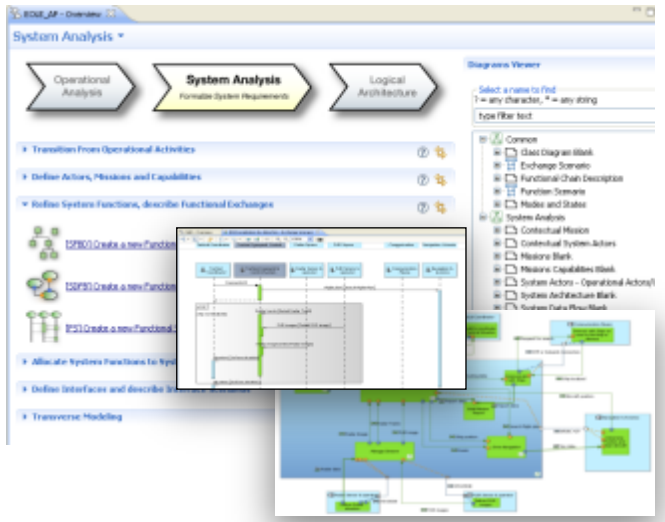


Technical Foundations:

Capella

Extensibility
New diagrams, new layers, M2 extensions, etc.

Edition Tools
Layered diagrams, Tables, Editors



Model Monitoring
Progress, metrics

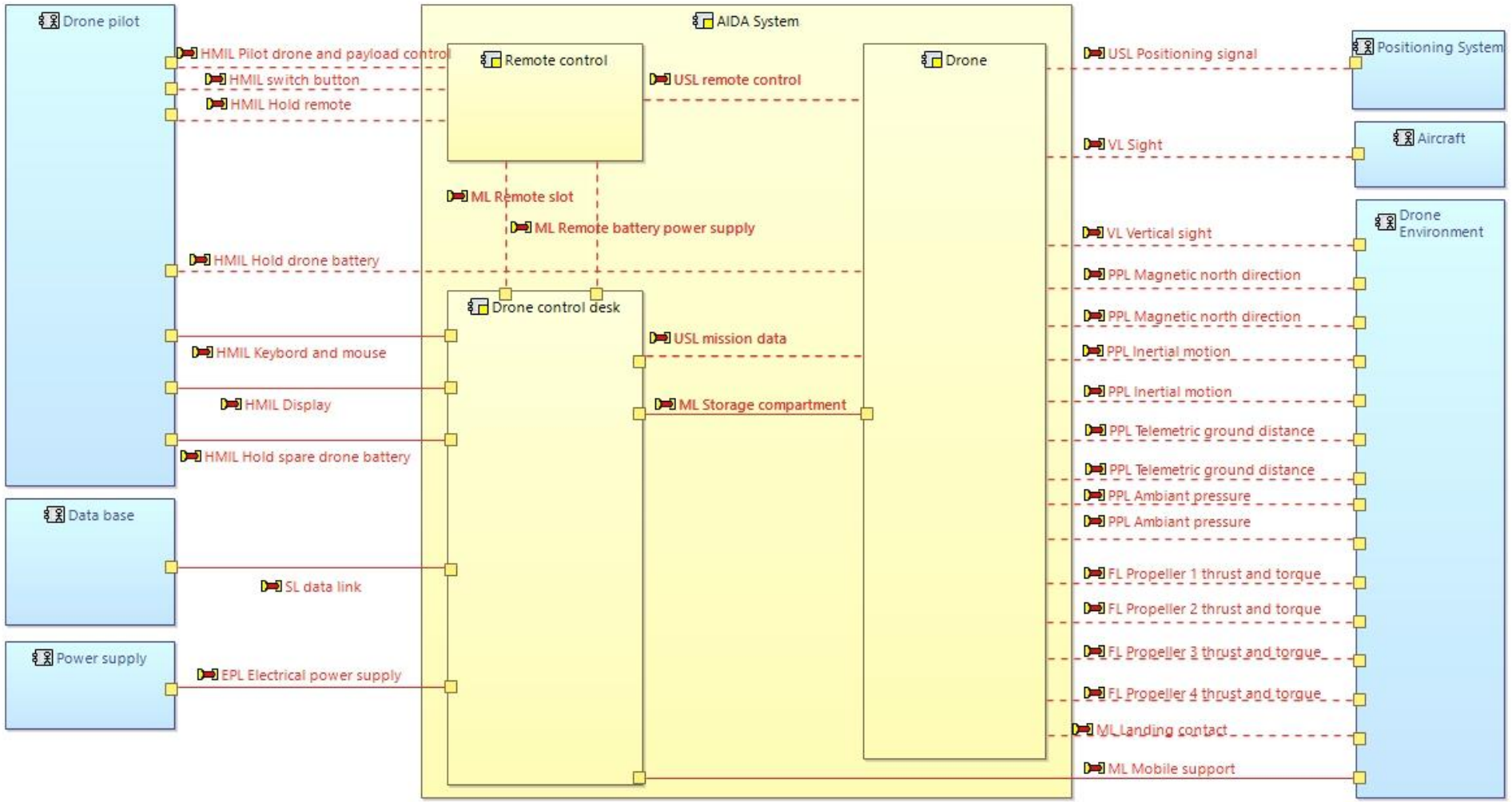
Embedded Methodological Guide

Modularity & Reuse
Librairies, Patterns, Coupling with PLE tools, etc.

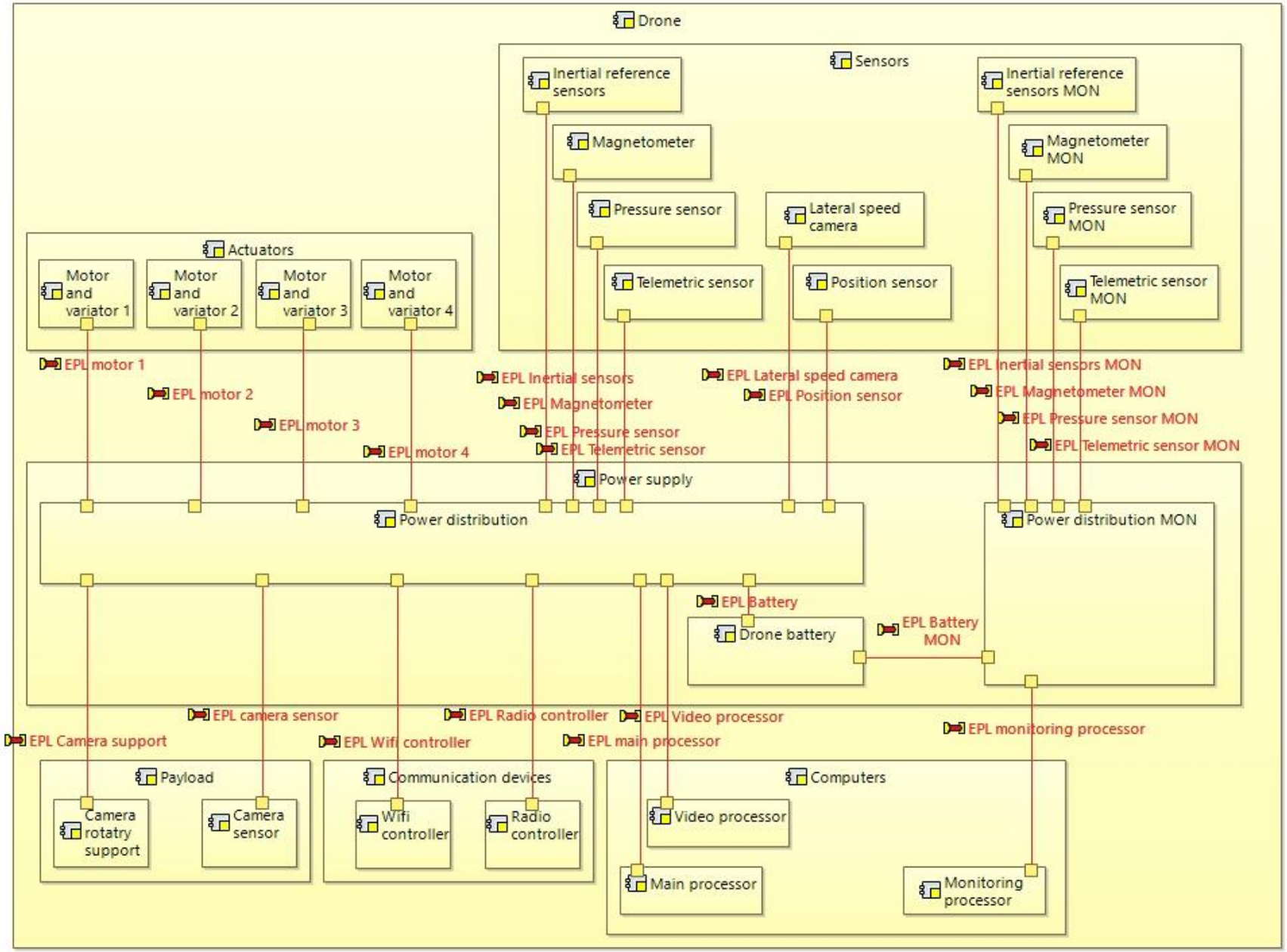
Iterative Transition Tools
Traceability, Generation

Model Analysis
Semantic browser, Model check, etc.

Some Architecture Diagram examples



This diagram is used for: {C} Diagram for Architecture design | {C} Diagram for Safety Analysis | {C} Diagram for Cosimulation

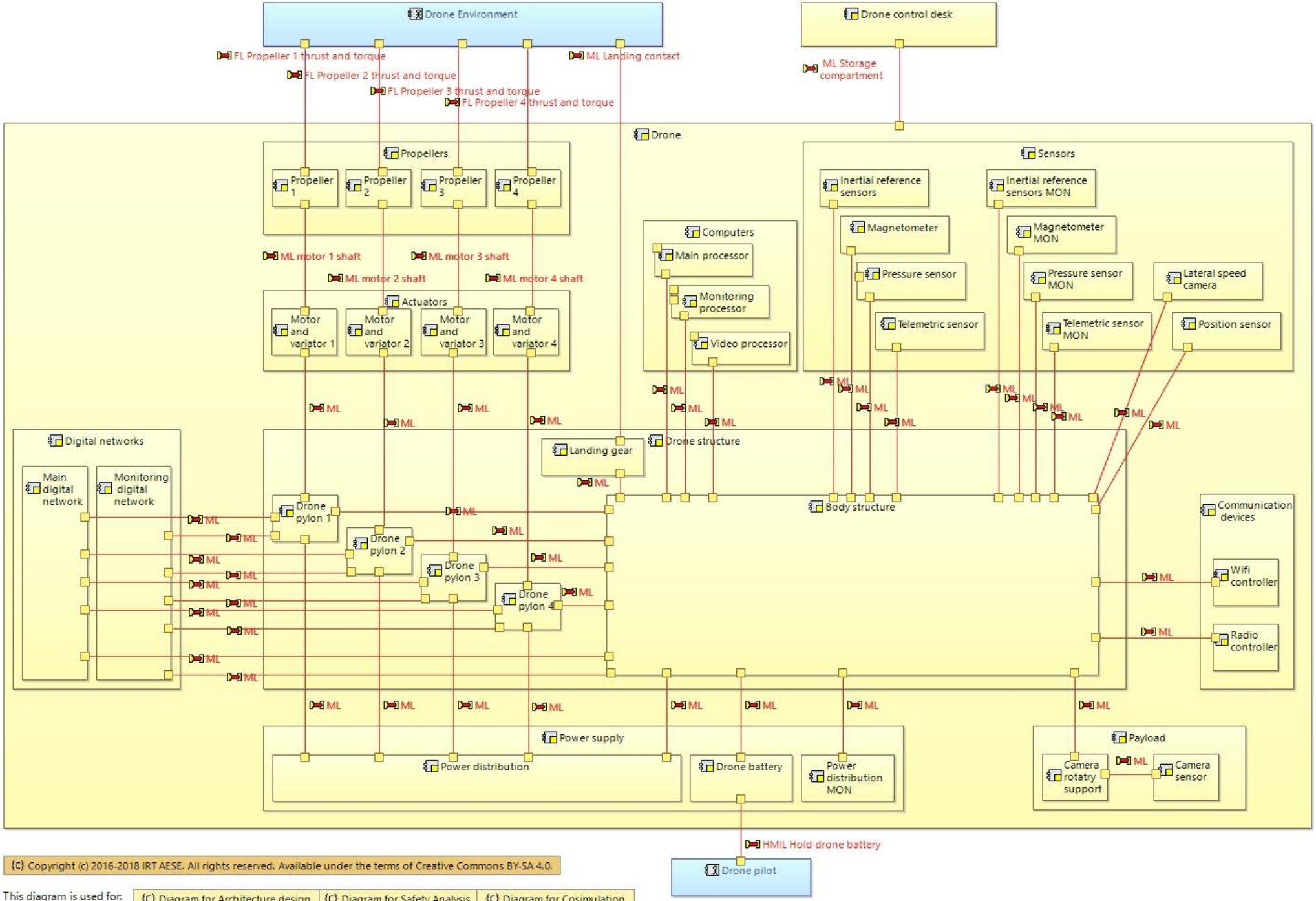


This diagram is used for: {C} Diagram for Architecture design | {C} Diagram for Safety Analysis | {C} Diagram for Cosimulation

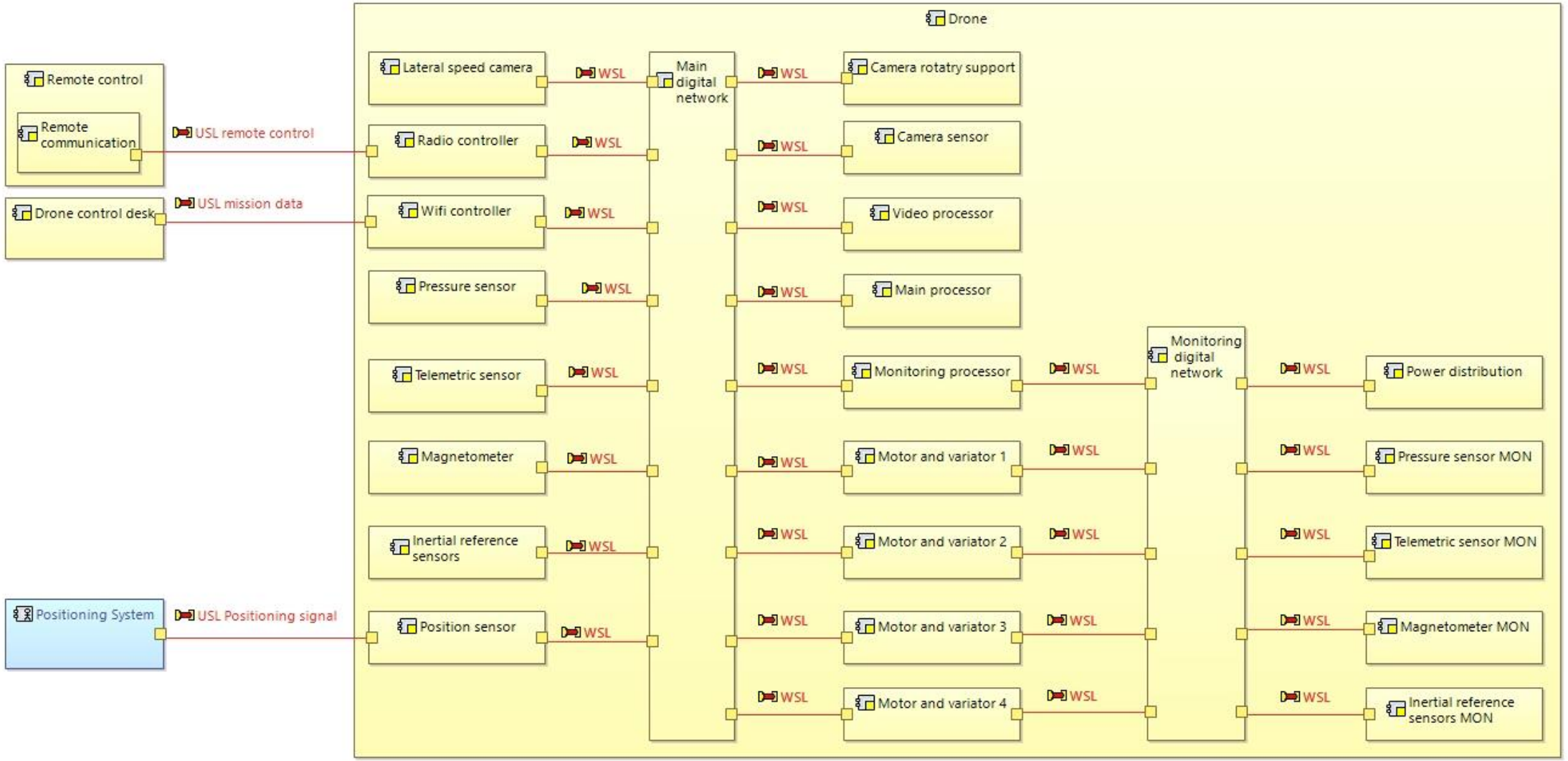
INCOSE Healthcare Working Group

5th Annual Systems Engineering in Healthcare Conference

May 1-2, 2019
Minneapolis, MN



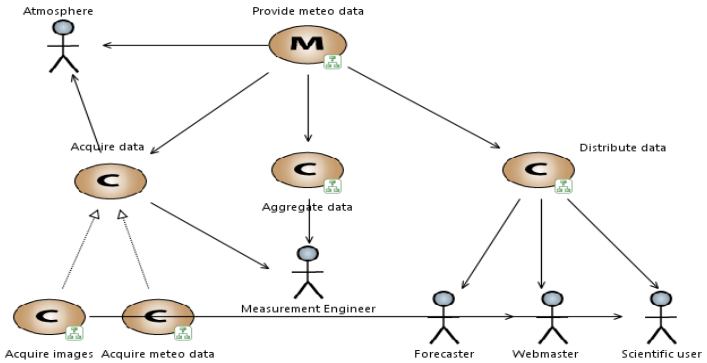
This diagram is used for: {c} Diagram for Architecture design | {c} Diagram for Safety Analysis | {c} Diagram for Cosimulation



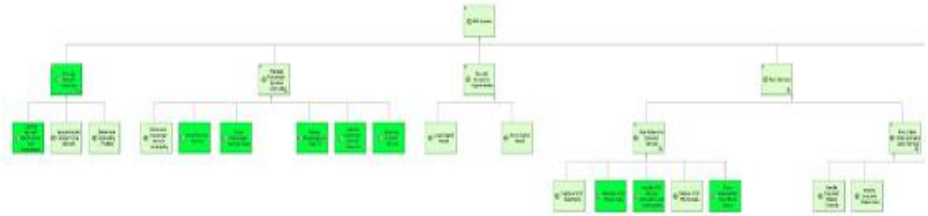
This diagram is used for: {C} Diagram for Architecture design | {C} Diagram for Safety Analysis | {C} Diagram for Cosimulation

How is Capella different ?

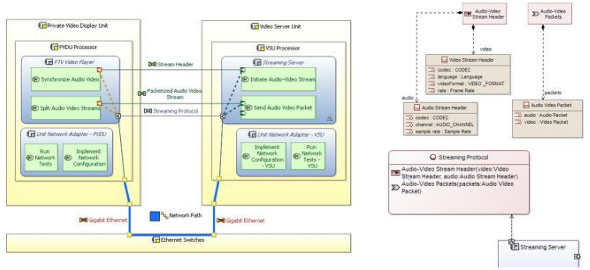
Instances first



Functionnal Analysis support

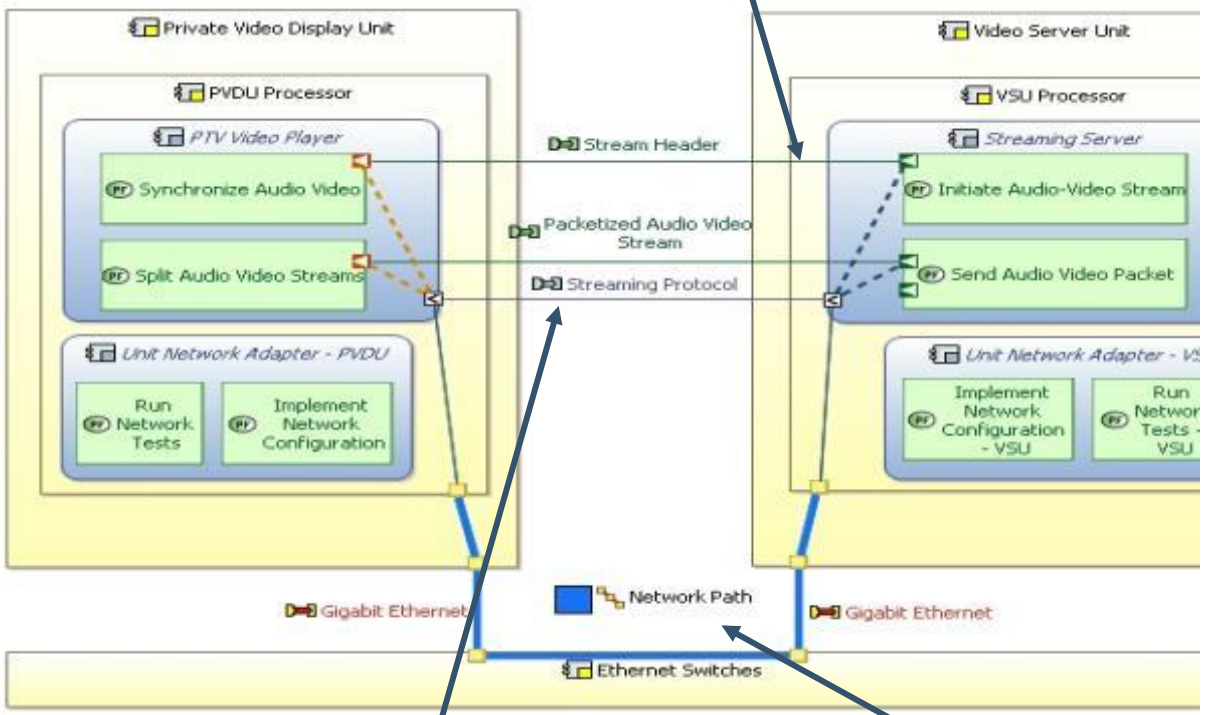


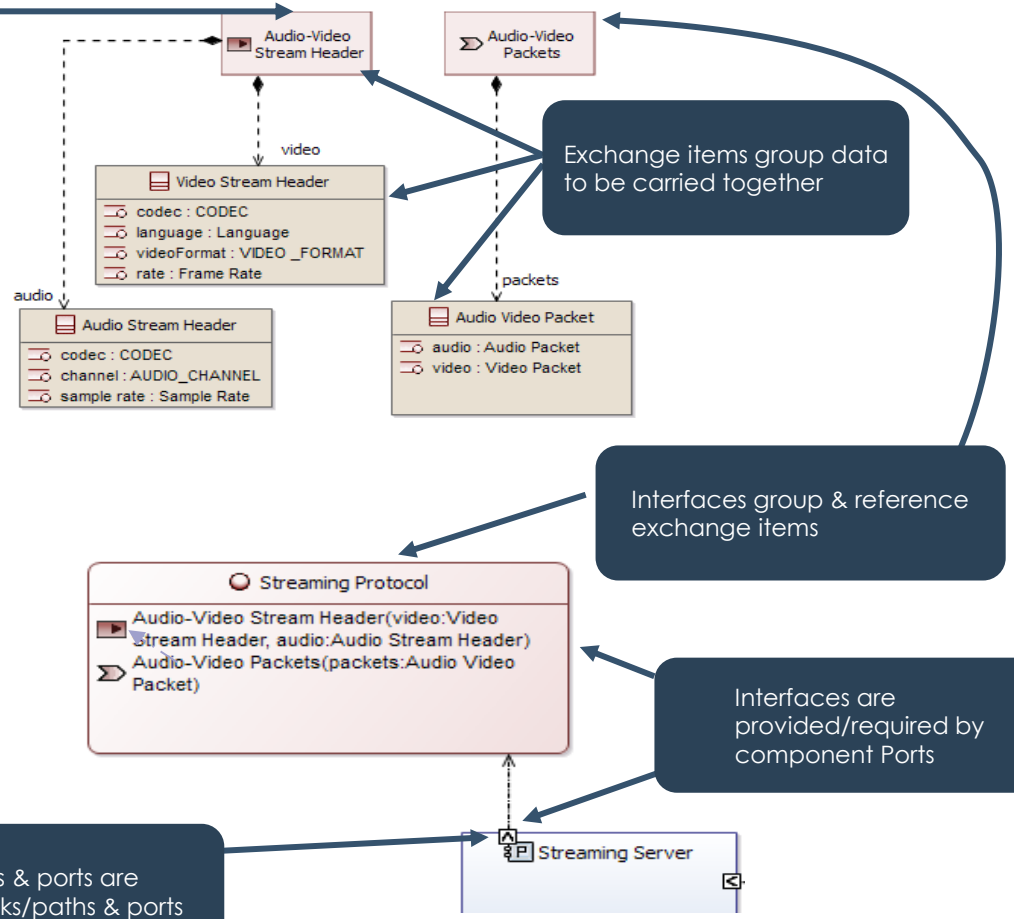
Functions, Interfaces and components integration



Functional analysis, structures and interfaces

Functional exchanges & ports carry exchange items





Exchange items group data to be carried together

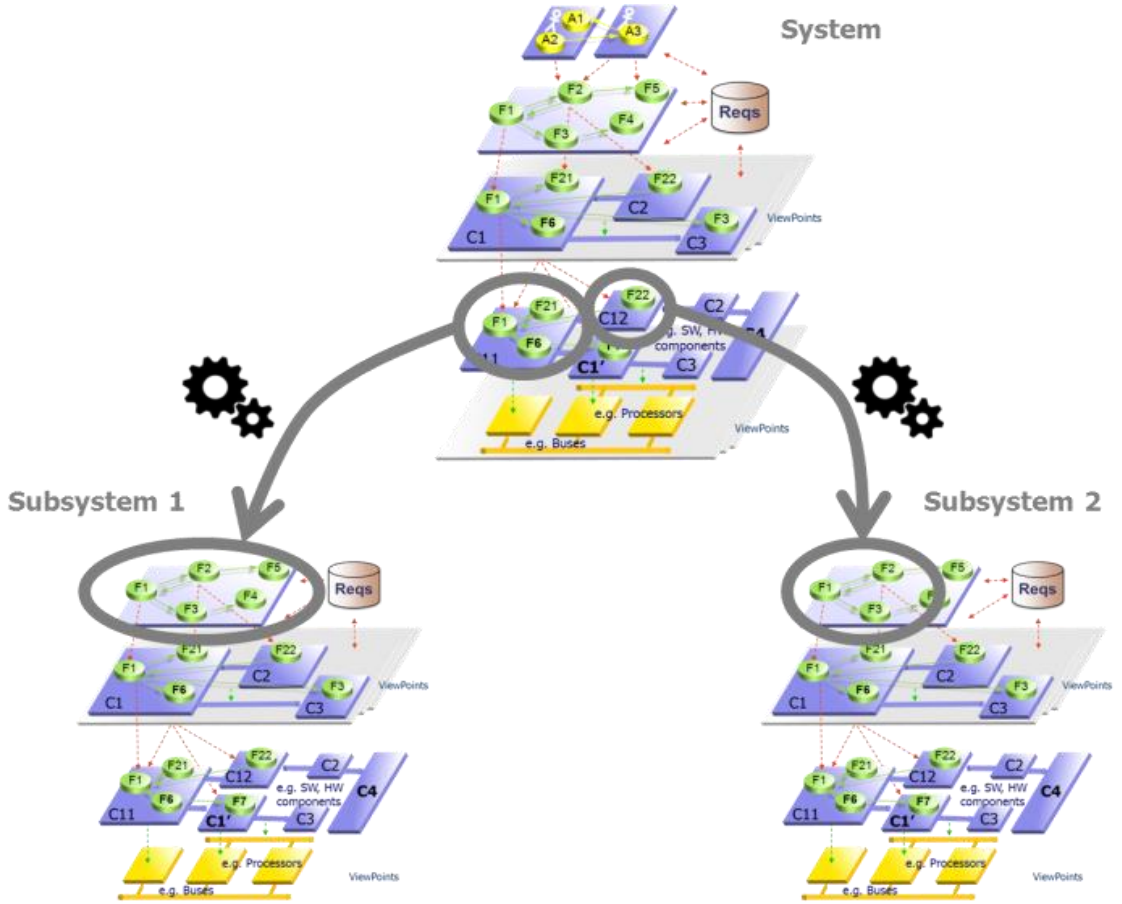
Interfaces group & reference exchange items

Interfaces are provided/required by component Ports

Functional exchanges & ports are allocated to component exchanges & ports

Component exchanges & ports are allocated to Physical links/paths & ports

Multiple engineering levels: System to Subsystem transitions



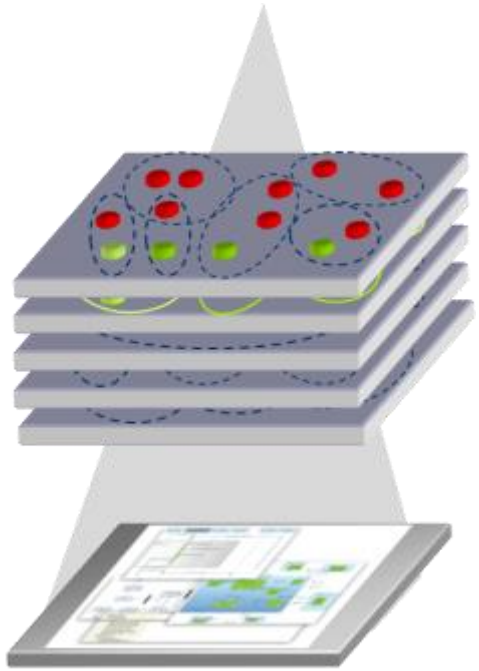
Architecture early evaluation

Performance

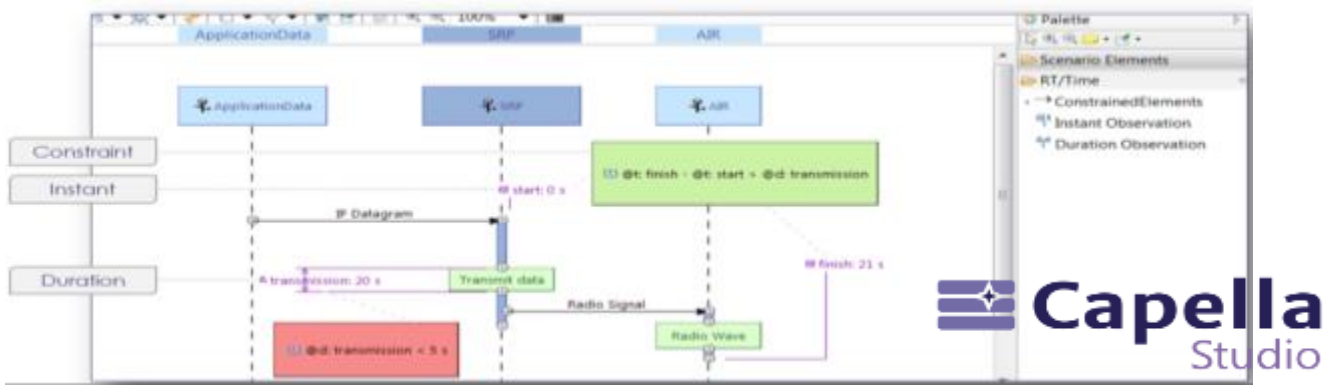
Mass

Safety

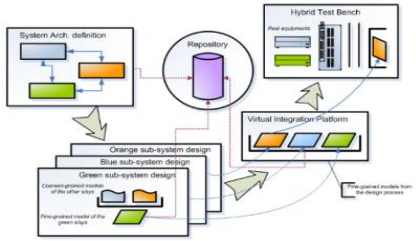
...



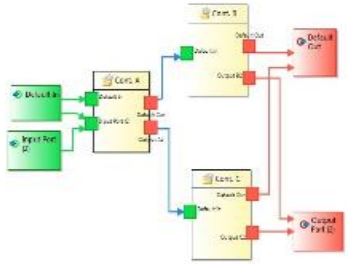
Autonomous viewpoints



Birectional coupling with specialty tools



Citrus simulation env.



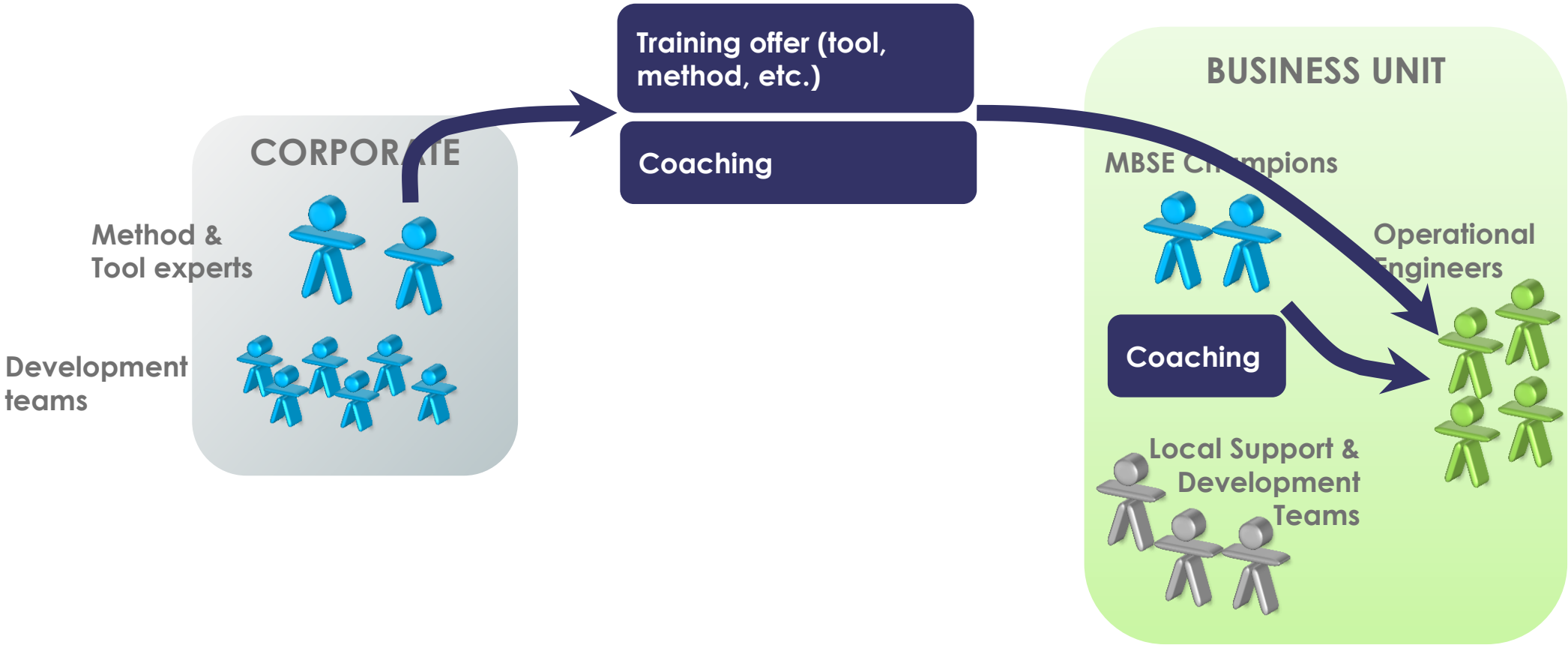
All4Tec Safety Architect

Key enablers for MBSE deployment

Good foundations – Language, Method &Tool

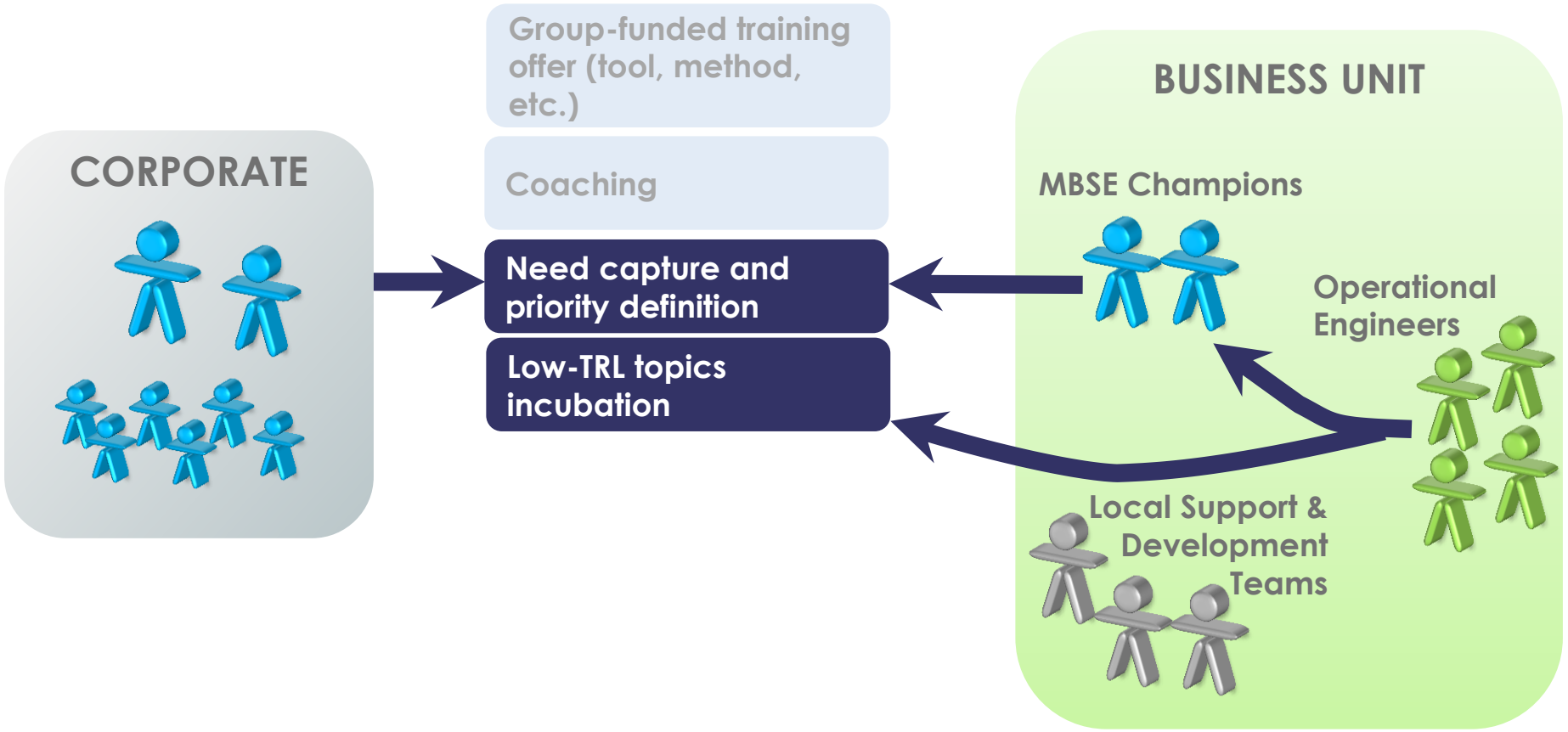
Mobilize the organization towards the deployment of MBSE

MBSE deployment actions

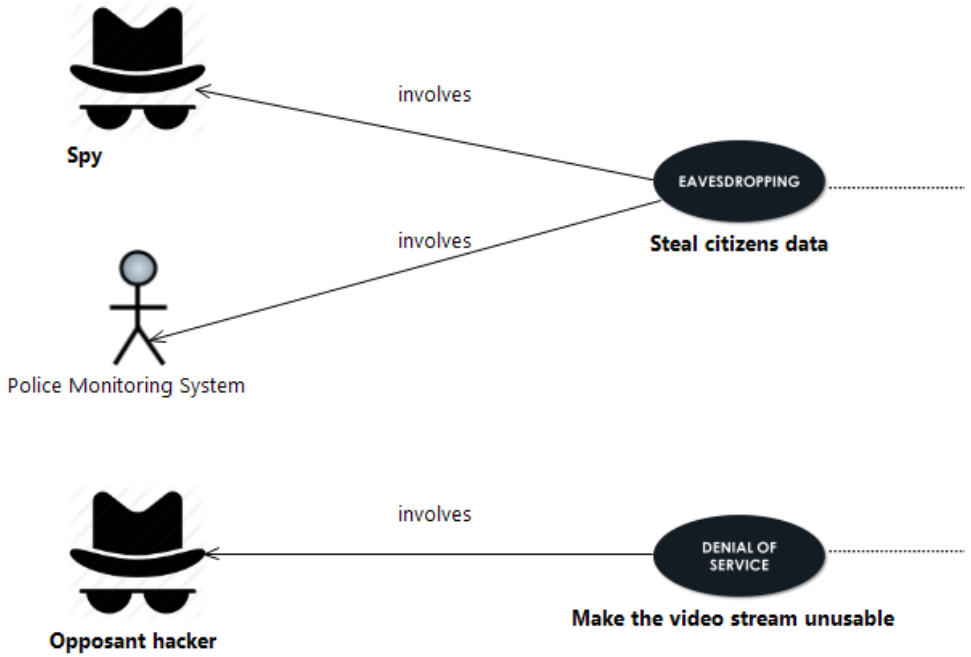


Focus on MBSE Coaching: Coaching types

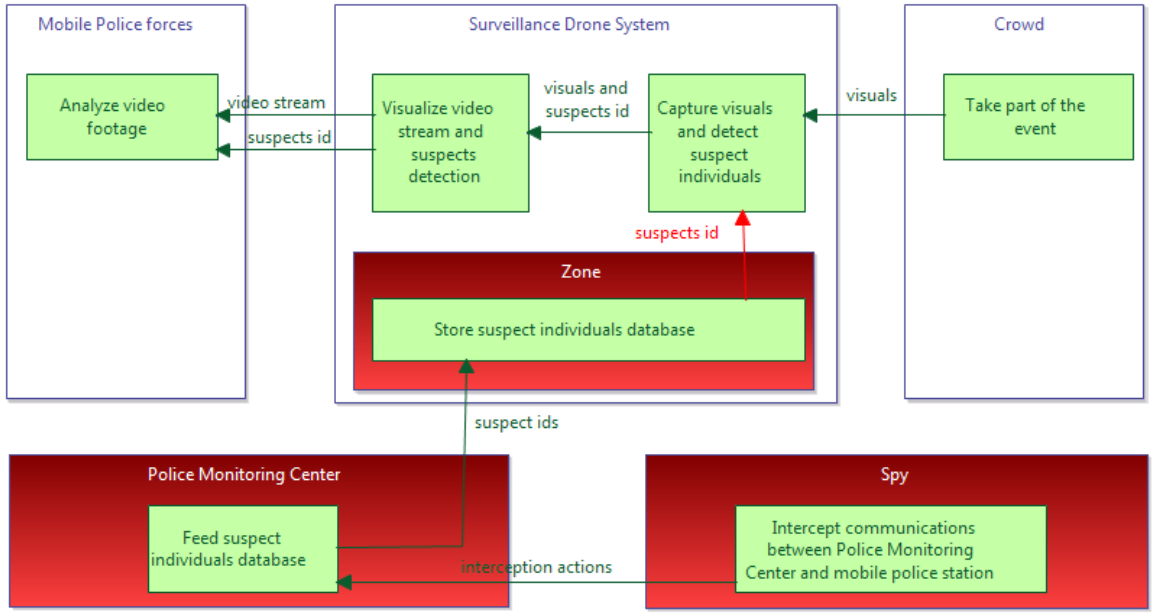
MBSE deployment actions



Focus on low TRL topics: Model-Based Cyber Security

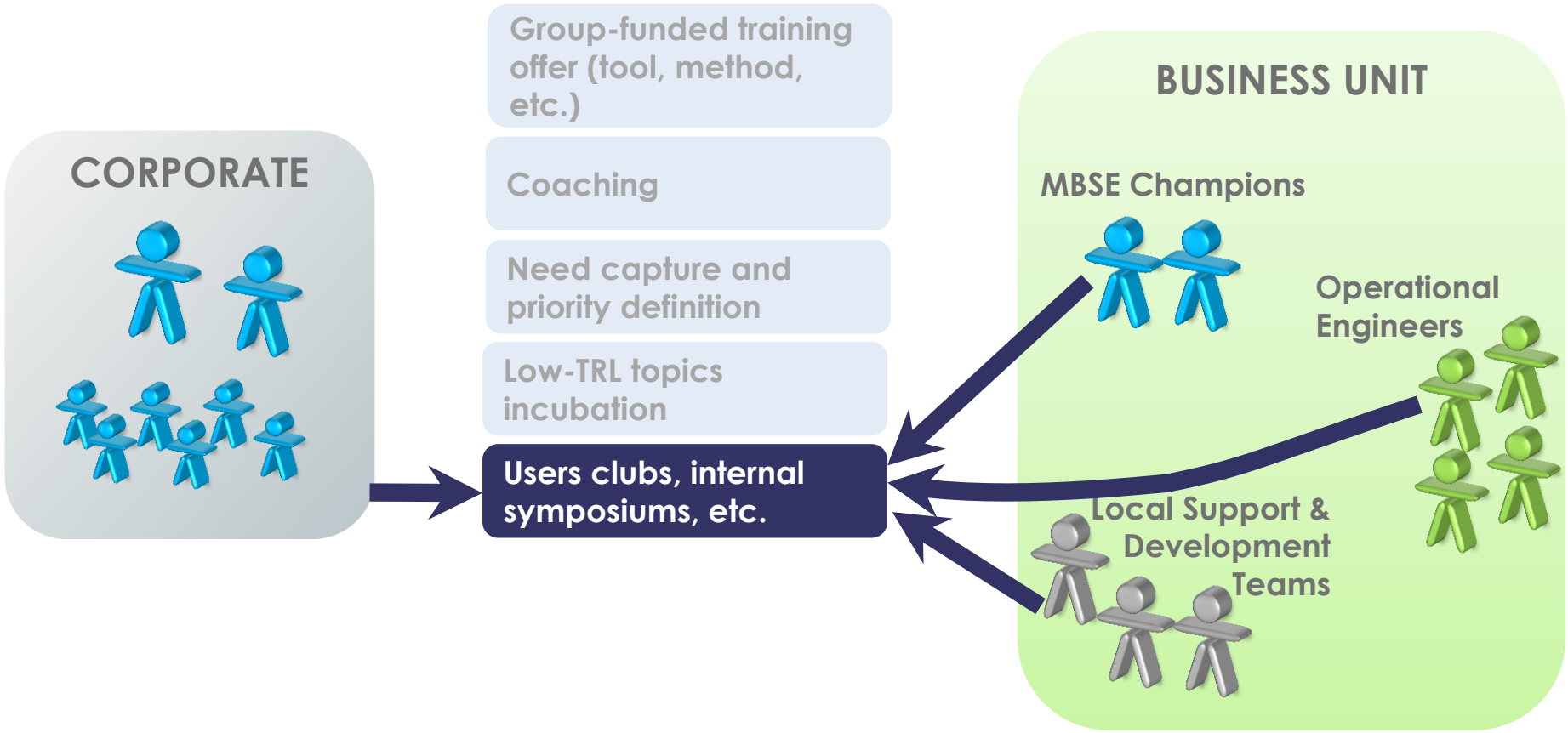


Identify potential threat sources, **threats** and attacked assets, **feed risk analysis**



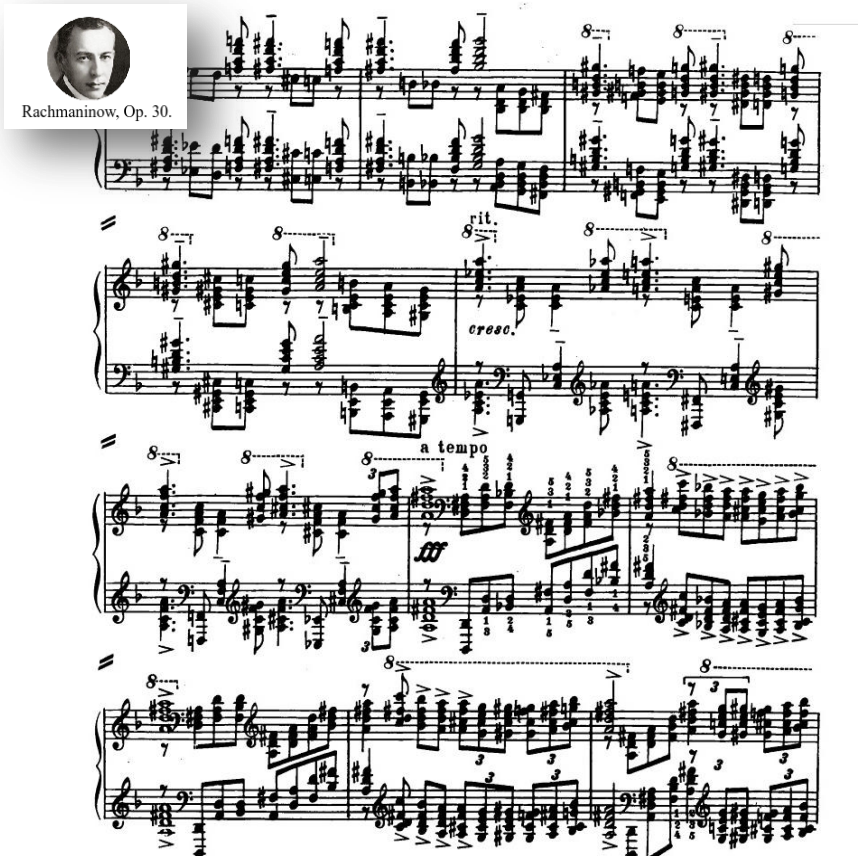
Identify high level **primary assets** and characterize their **security needs**, identify external interfaces and **trust boundaries**

MBSE deployment actions



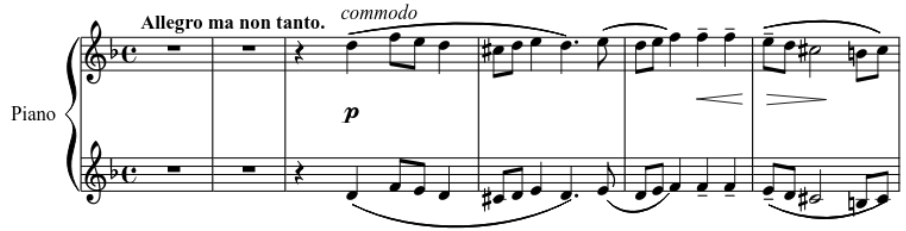
Capella Thales Users Day
Yearly event, > 100 people

Adapt the deployment actions to your organization



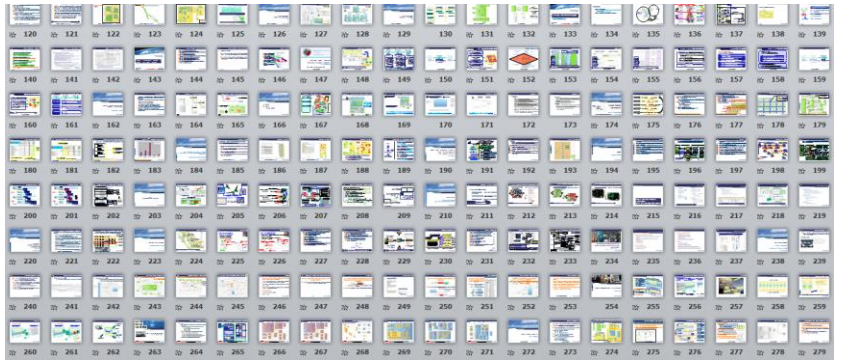
DON'T SEEK THE BIG BANG!

Be incremental



Some Use Cases

A wide variety of MBSE deployments in Thales and other companies



Use Case 1 – Thales

Managing System Design Complexity

Context

- Issues in the latest phases of operational validation
- Very good design documents, but in silos
- 5 levels of functional decomposition, 275 functions

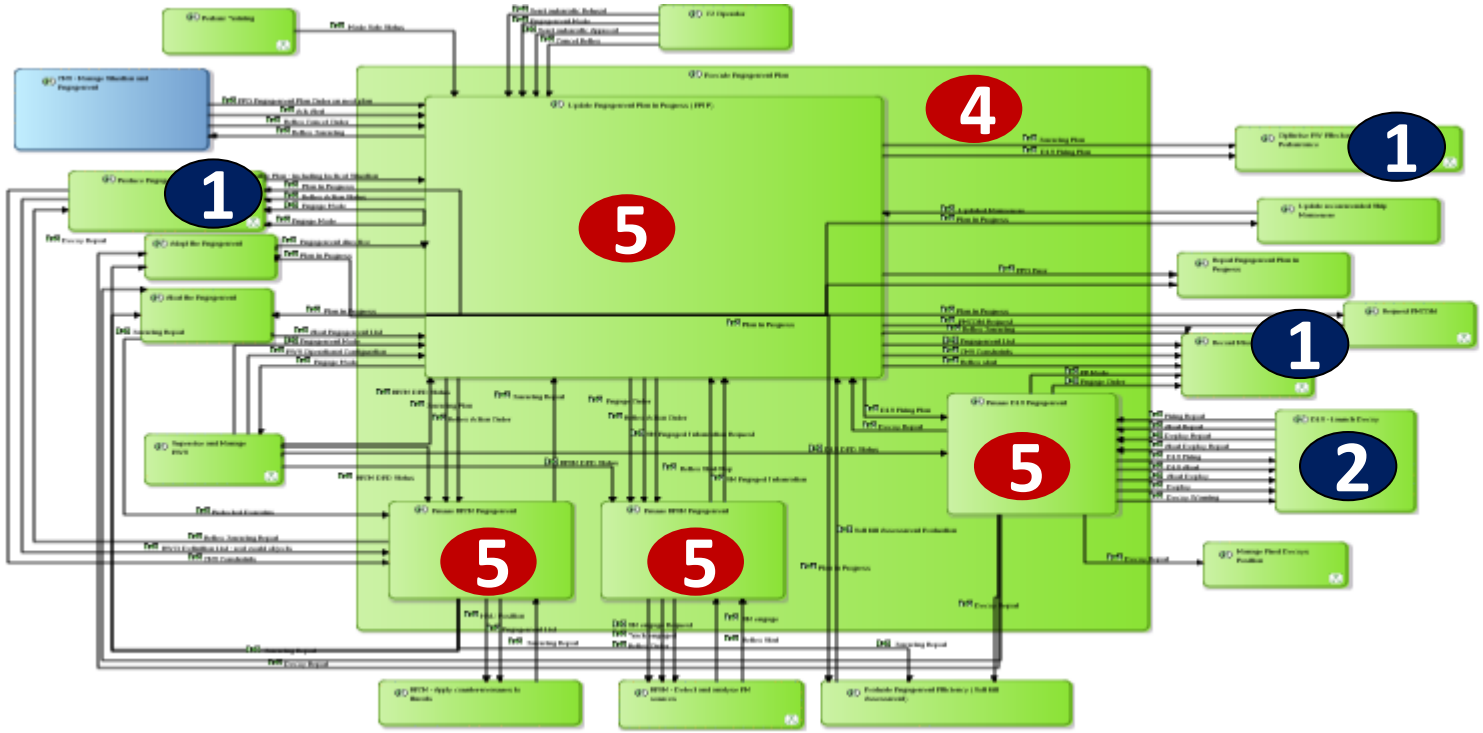
MBSE usage

- 1 man month to **reverse a first level of detail** in a model, based on existing documents
- **First time overall views have been available**
  - Good support for discussion
  - Visualization of transverse functional chains

Use Case 1 – Thales

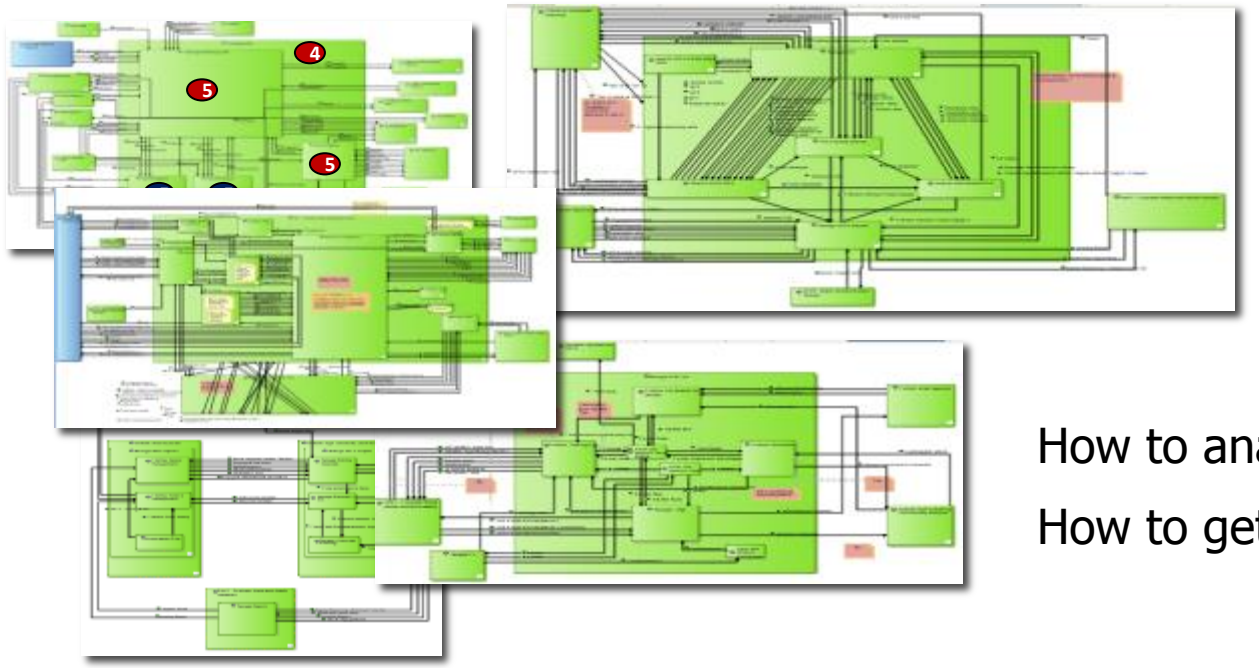
Managing System Design Complexity

Contextual Diagrams: Low-level internals, high-level neighboorhood



Use Case 1 – Thales

Managing System Design Complexity



X 40

How to analyze transverse topics?

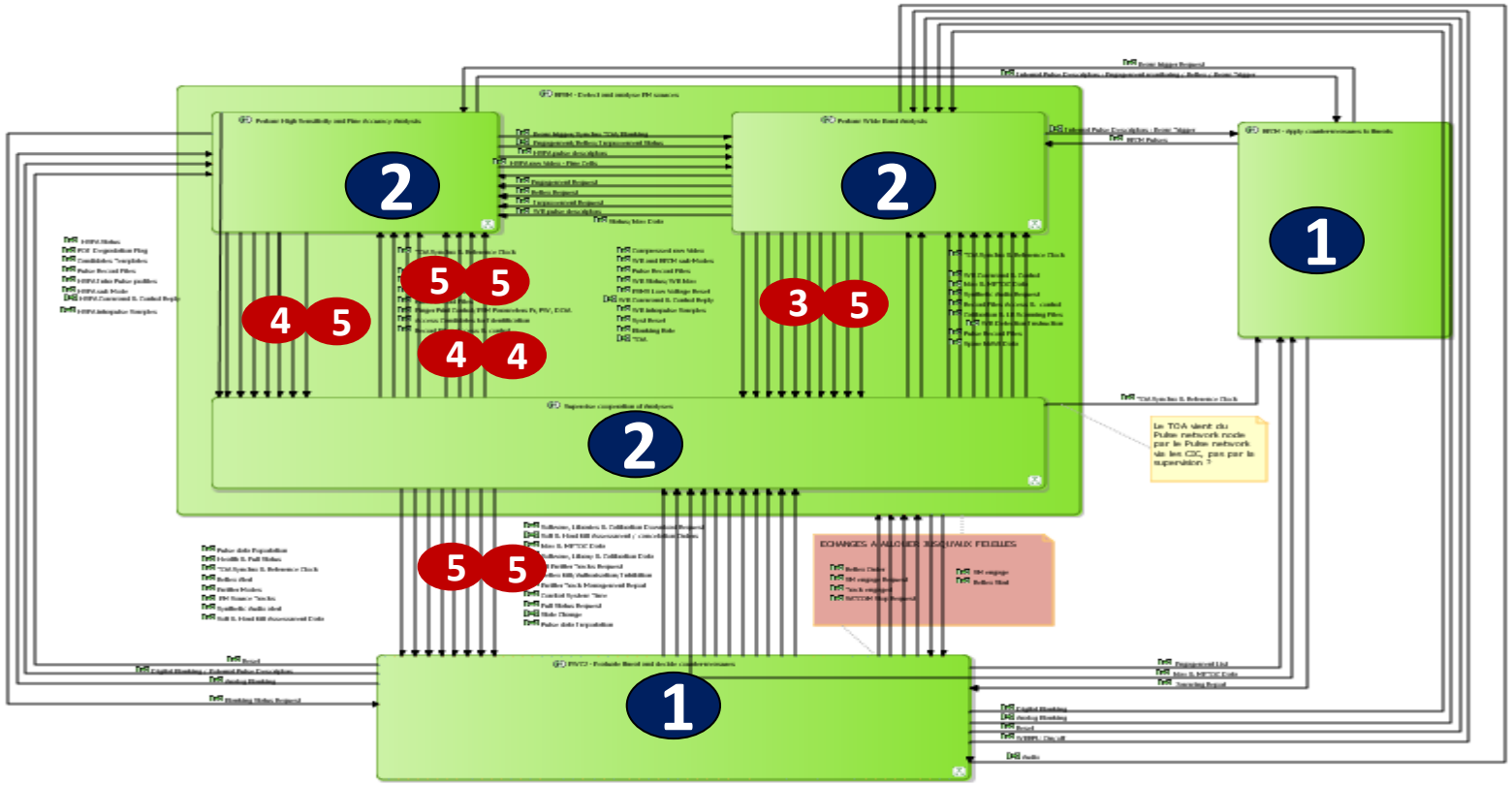
How to get transverse overviews?

Challenge: Build and maintain simplified views

Use Case 1 – Thales

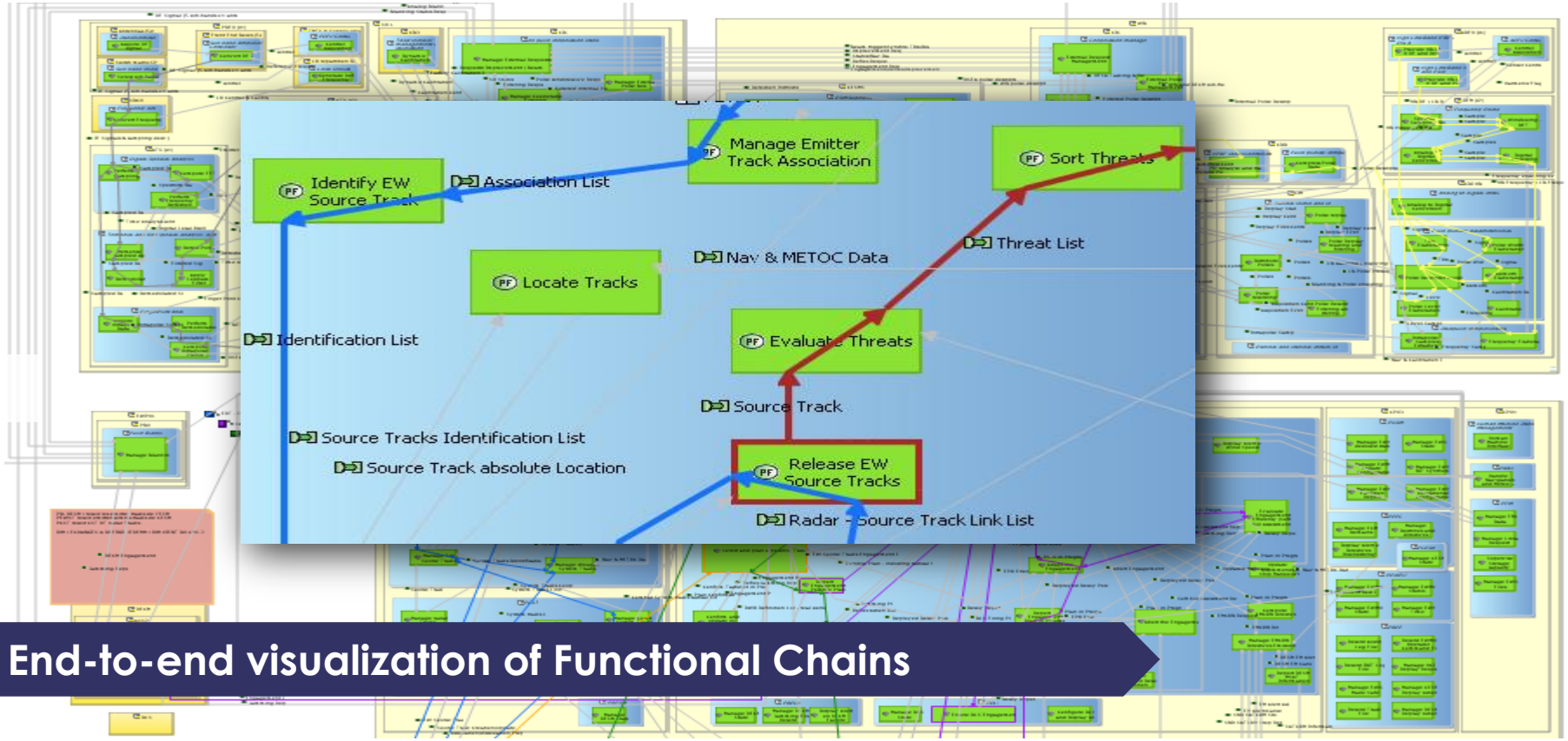
Managing System Design Complexity

Computed Diagrams: High-level Functions, Low-level Exchanges



Use Case 1 – Thales

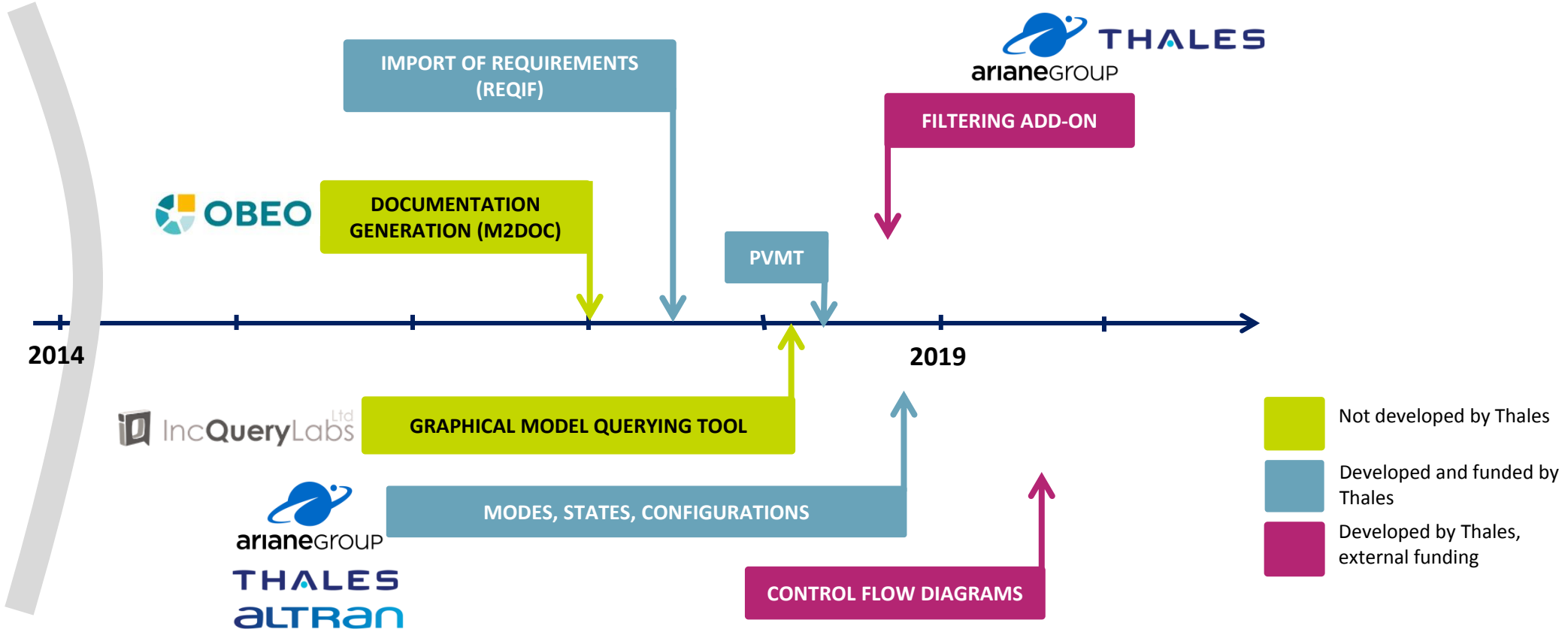
Managing System Design Complexity



End-to-end visualization of Functional Chains

What now, what next, where to

Capella still growing since going open source



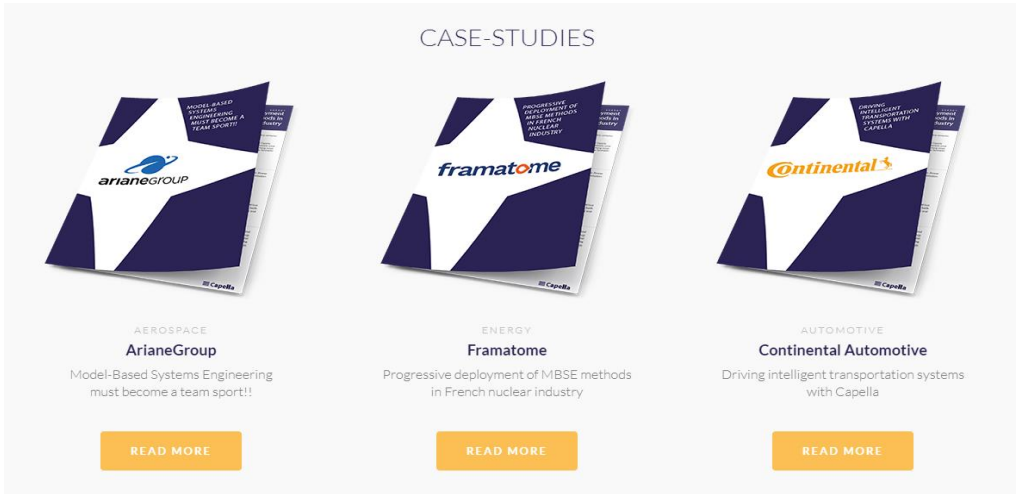
polarsys.org/capella

JEAN-LUC VOIRIN

Model-based System and Architecture Engineering with the Arcadia Method

PASCAL ROQUES

Systems Architecture Modeling with the Arcadia Method - A Practical Guide to Capella Modeling Tool



Stay tuned

Official Capella website

https://polarsys.org/capella/

Arcadia and Capella public forums

https://polarsys.org/forums/

Polarsys Capella YouTube channel

YouTube

INCOSE
Healthcare
Working Group

5th Annual Systems Engineering in Healthcare Conference

May 1-2, 2019
Minneapolis, MN

Thank you for attending!

Share your experiences at #HWGSEC