



[www.thalesgroup.com](http://www.thalesgroup.com)

# Arcadia and Capella on the Field: Real-World MBSE Use Cases

MBSE Symposium, Canberra  
October 27<sup>th</sup>, 2014

Stéphane Bonnet, Fabrice Lestideau, Jean-Luc Voirin

OPEN

THALES

## ENGINEERING KEY QUESTIONS

- How is the customer need received? How are its **consistence and feasibility checked**?
- Which are the **engineering phases in the solution elaboration**, how are they related?
- **How is complexity managed?**
- How are different alternatives evaluated, **how do the specialists collaborate?**
- **How is the solution justified** w.r.t. the need and the different constraints?

## FACTS

- Needs and solutions are more complex, more stakeholders, more constraints, less time
- **The approach Doors / Word / Visio / Excel reaches limits**
- Manual processes are not compatible with agility and short loops



Model-Based Engineering Method for Architectural Design

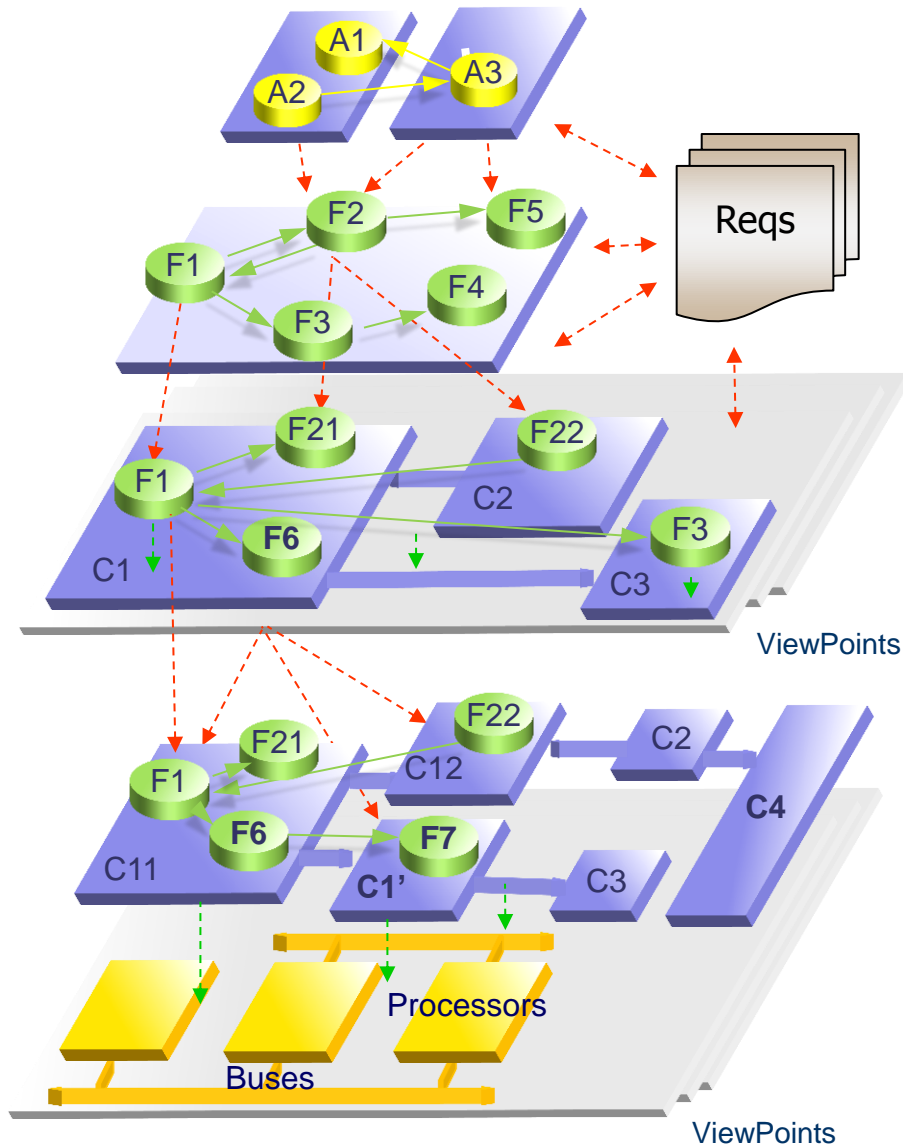


Graphical Modelling Workbench supporting Arcadia



NEED UNDERSTANDING

SOLUTION ARCHITECTURAL DESIGN

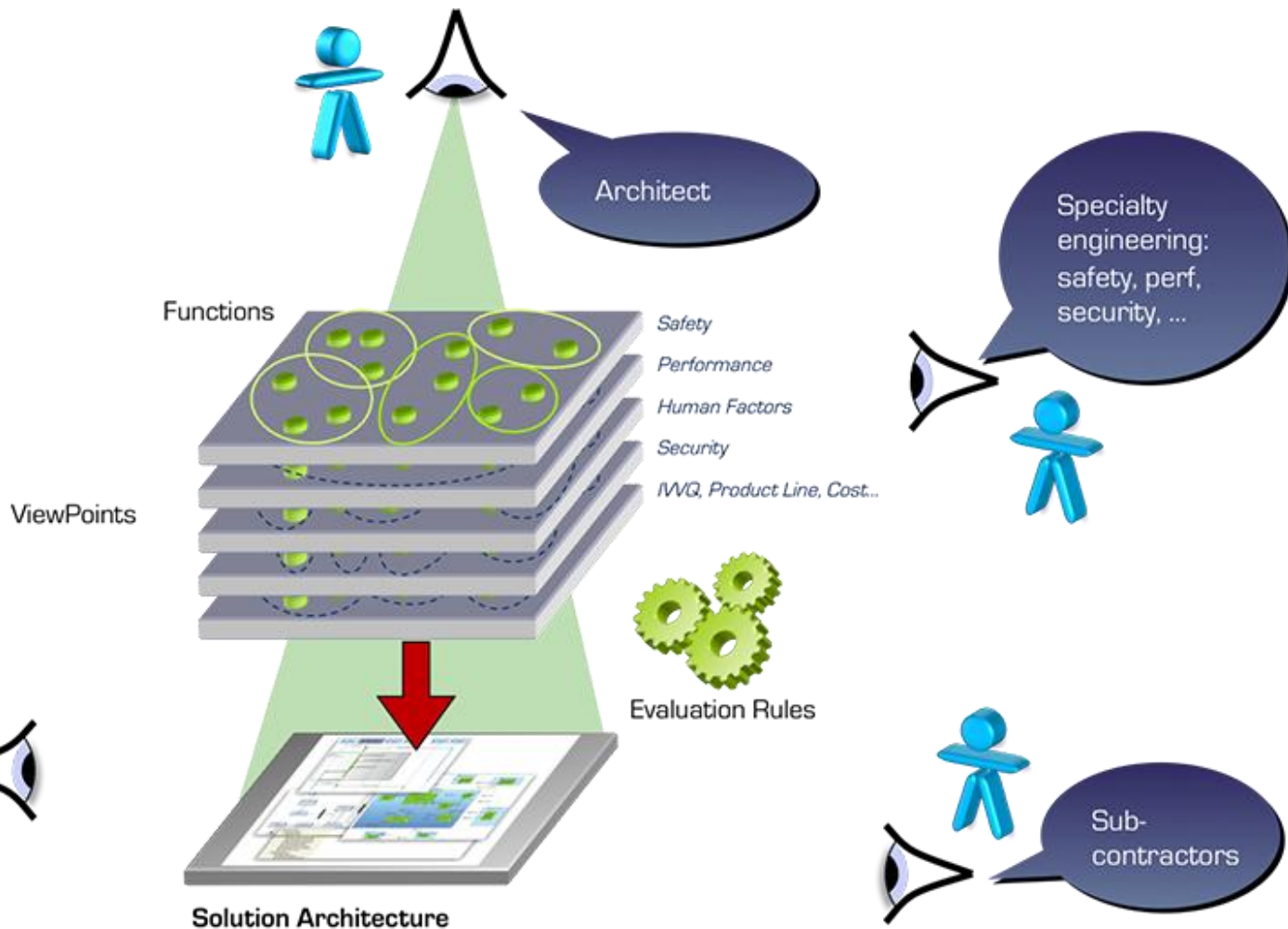


What the users of the system need to accomplish

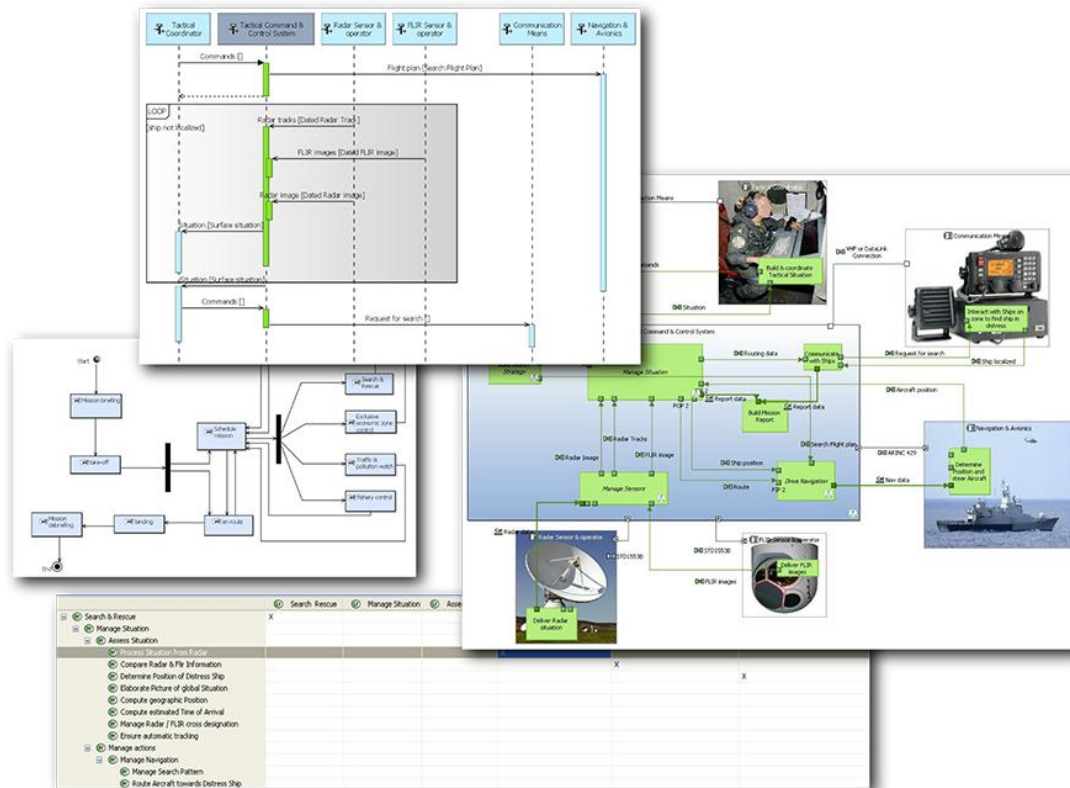
What the system has to accomplish for the users

How the system will work to fulfill expectations

How the system will be developed and built



- **Guidance**  
[Embedded methodological browser]
- **Complexity management**  
[Abstraction via computed information]
- **Productivity tools**  
[Automated transitions and diagram creation accelerators]
- **Model Analysis & Navigation**  
[Model validation, semantic browser]
- **Multi-criteria analysis**  
[Viewpoints and management framework]




# Capella

First operational deployments in 2009

Used on most major engineering projects

Currently being open sourced

OPEN

THALES



Melody Advance - platform:/resource/SAR/SAR.air.d/[PAB] Implementation Vs Behavioral - Processing(Physical Component Physical System) - Melody Advance

File Edit Diagram Navigate Search Project Run Window Help

SAR - Overview

## Overview of SAR

**Operational Analysis**

**Define Stakeholder Needs and Environment**

Capture and consolidate operational needs from stakeholders  
Define what the users of the system have to accomplish  
Identify entities, actors, roles, capabilities, activities, concepts

**System Analysis**

**Formalize System Requirements**

Identify the boundary of the system, consolidate requirements  
Define what the system has to accomplish for the users  
Model functional dataflows and dynamic behaviour

**Logical Architecture**

**Develop System Logical Architecture**

See the system as a white box: define how the system will work  
so as to fulfill expectations  
Perform a first trade-off analysis

**Physical Architecture**

**Develop System Physical Architecture**

How the system will be developed and built  
Software vs. hardware allocation, specification of interfaces,  
deployment configurations, trade-off analysis

**EPBS**

**Formalize Component Requirements**

Manage industrial criteria and integration strategy: what is  
expected from each designer / sub-contractor  
Specify requirements and interfaces of all configuration items

Introduction | Operational Analysis | System Analysis | Logical Architecture | **Physical Architecture** | EPBS

## Physical Architecture

Logical Architecture   **Physical Architecture**   EPBS

Develop System Architectural Design

Transition from Logical Functions

- Perform an automated transition of Logical Functions
- Create Traceability Matrix

Refine Physical Functions, describe Functional Exchanges

- [PFBD] Create a new Functional Breakdown diagram
- [PDFB] Create a new Functional Dataflow Blank diagram
- [FS] Create a new Functional Scenario

Define Physical Components and Actors, Manage deployments

- Perform an automated transition of Logical Actors
- Perform an automated transition of Logical Components
- [PCBD] Create a new Physical Component Breakdown diagram
- [PAB] Create a new Physical Architecture diagram
- Create a new Physical Component / Logical Component Matrix

Allocate Physical Functions to Physical Components

Delegate Logical Interfaces and create Physical Interfaces

Enrich Physical Scenarios

Introduction | Operational Analysis | System Analysis | Logical Architecture | **Physical Architecture** | EPBS

[OIS] Doors closing Scenario(Scenario Scenario)

75%

Cabin Crew Cabin Crew   Doors Doors   Flight Crew Flight Crew   Cabin Cabin

Lock command

Lock status

Doors closed & Slides armed

[PAB] Implementation Vs Behavioral - Processing(Physical Component Physical System)

75%

Determine Position and steer Aircraft

Communicate with Ships

Aircraft position

Manage Situation

Route Aircraft towards Distress Ship

Drop time top

Manage Rescue Means

Ship position

Consolidated tracks

Commands

Assess Situation

Search Zone

Acquire radar tracks

Acquire radar image

Acquire FLIR image

128M of 762M



## Use Case 1:

# Managing System Design Complexity

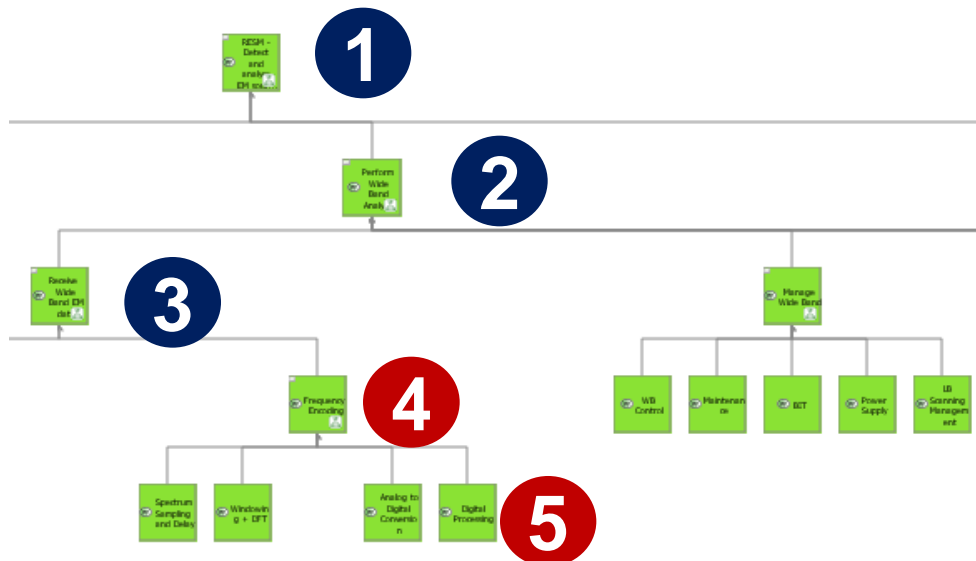
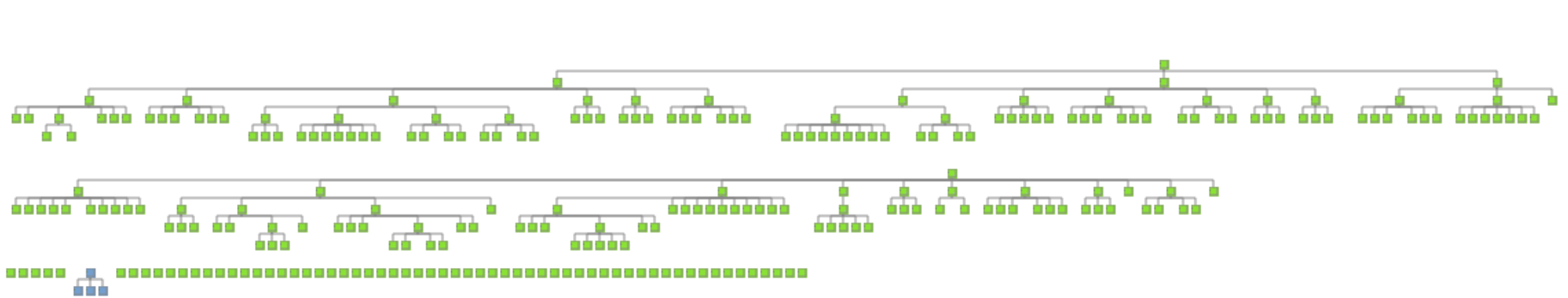


## Context



## MBSE usage

- Issues in the latest phases of operational validation
  - Very good design documents, but in silos
- 
- 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

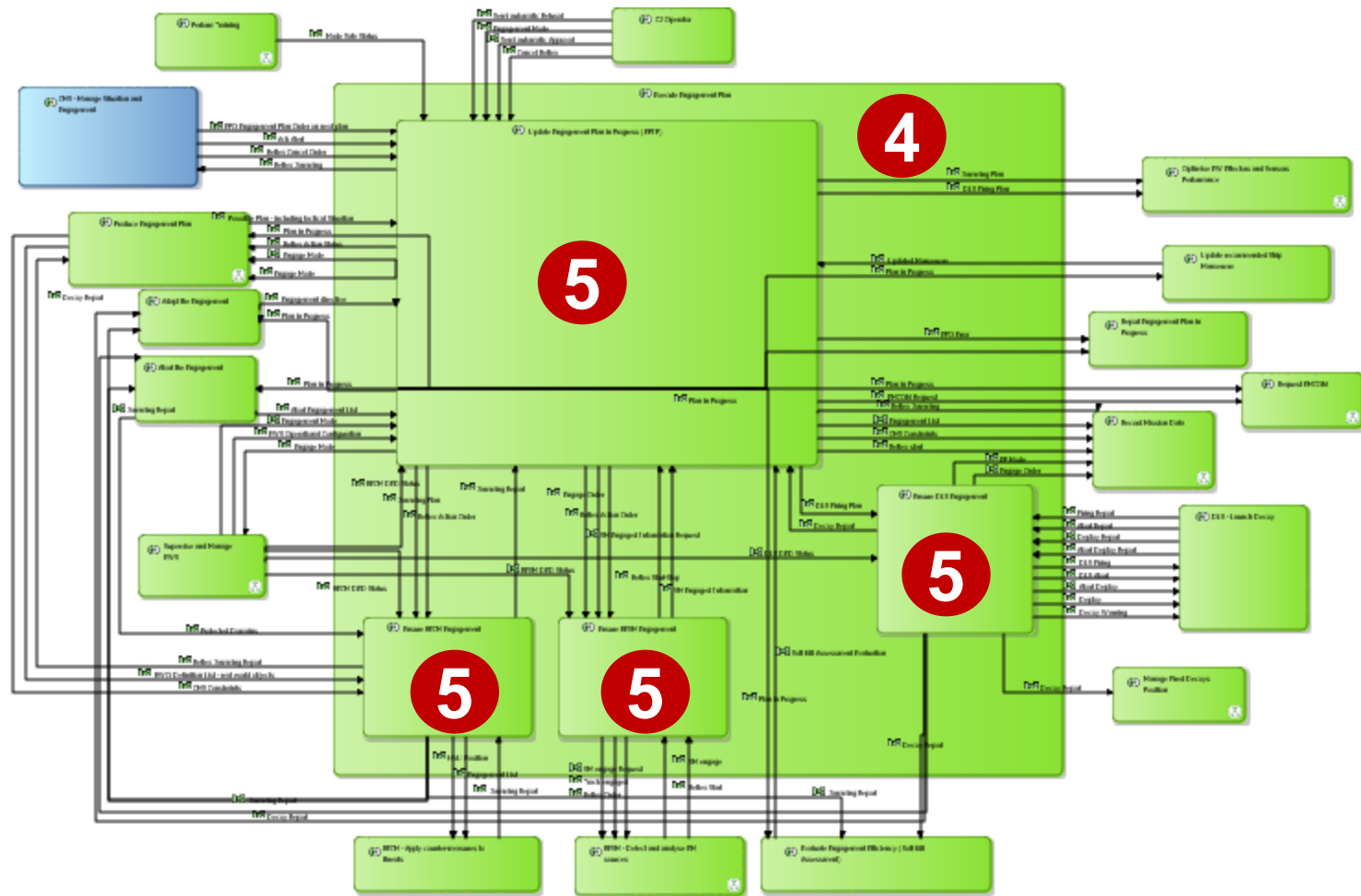


**275** Functions  
(230 Leaves)

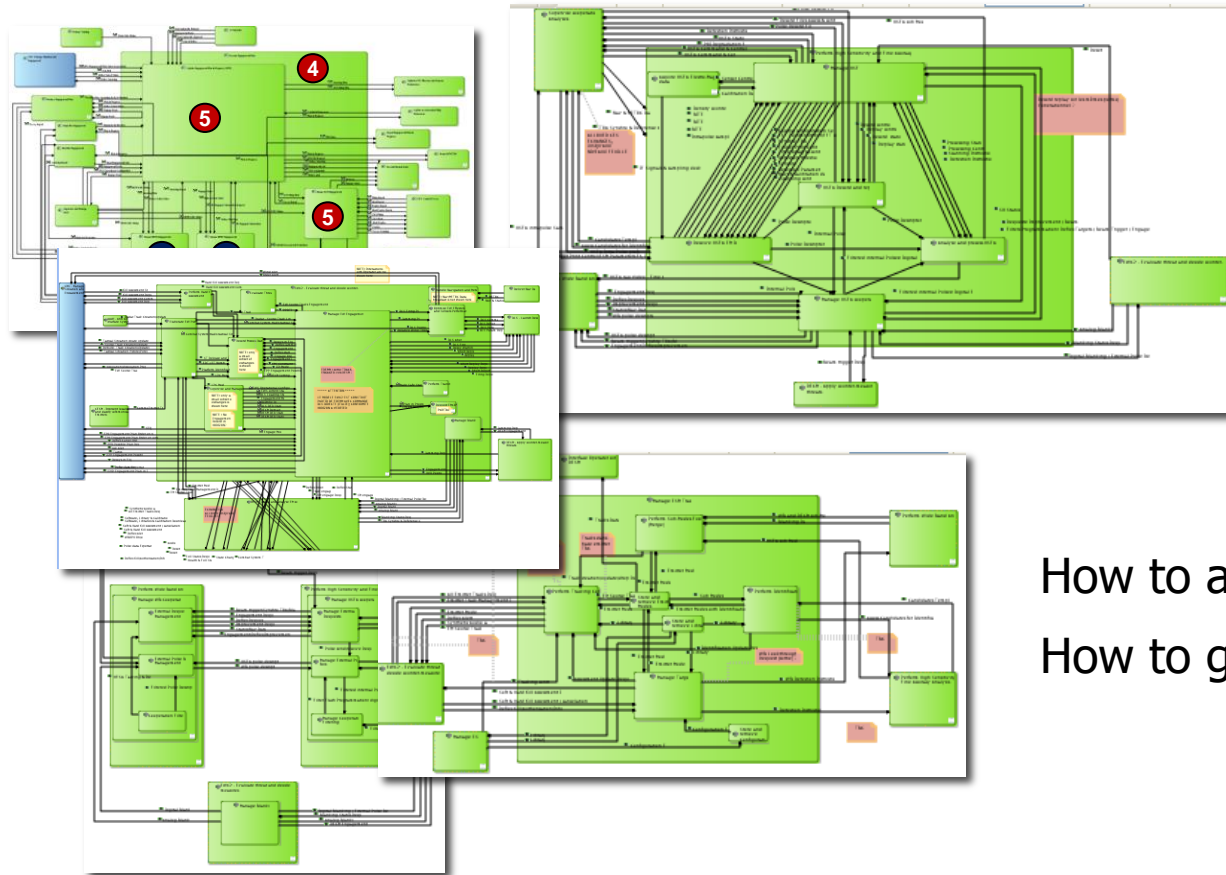
**578** Functional  
Exchanges between leaf  
functions

**5** levels of decomposition

## Contextual Diagrams: Low-level internals



This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2014 – All rights reserved.

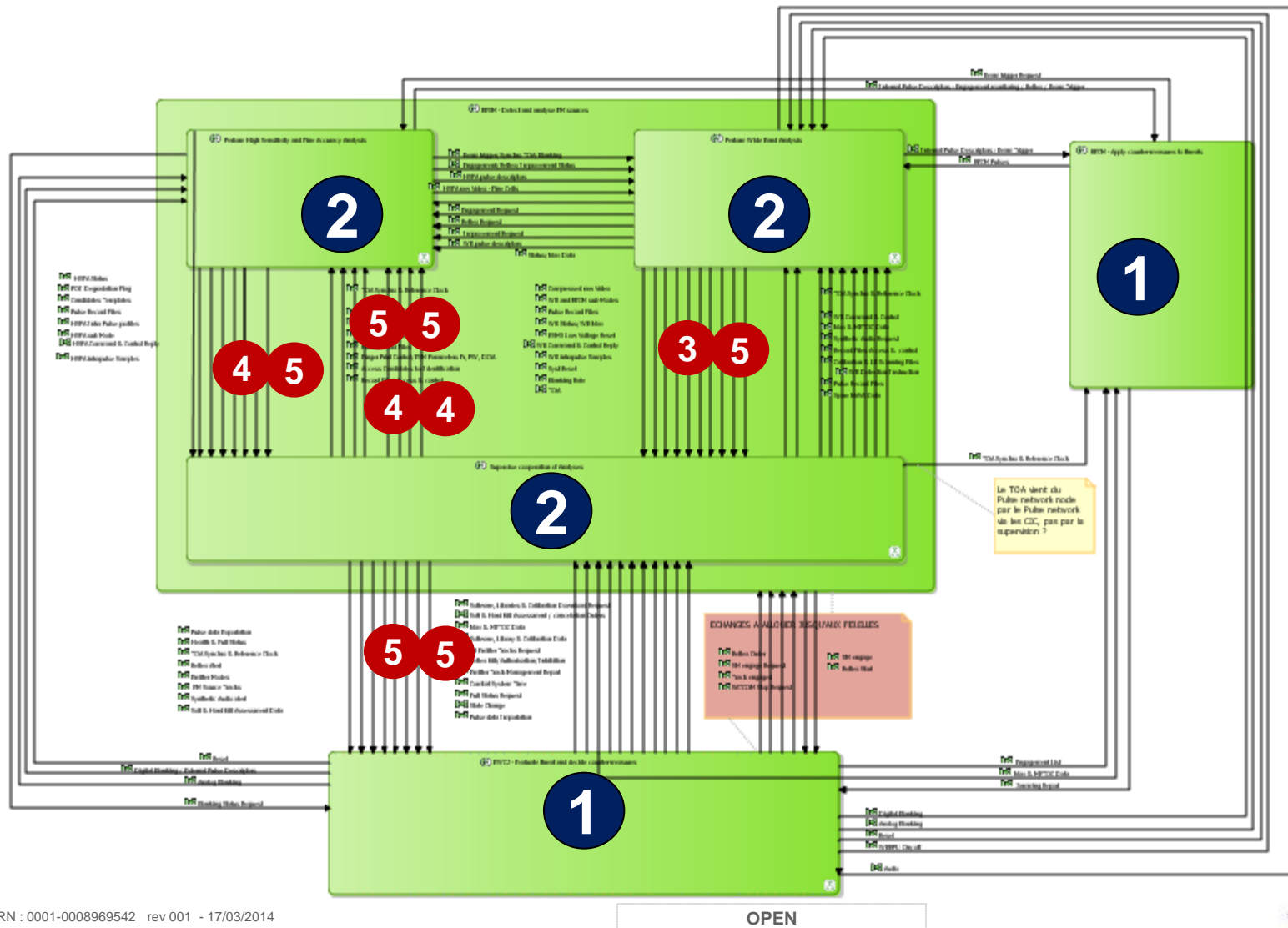


# X 40

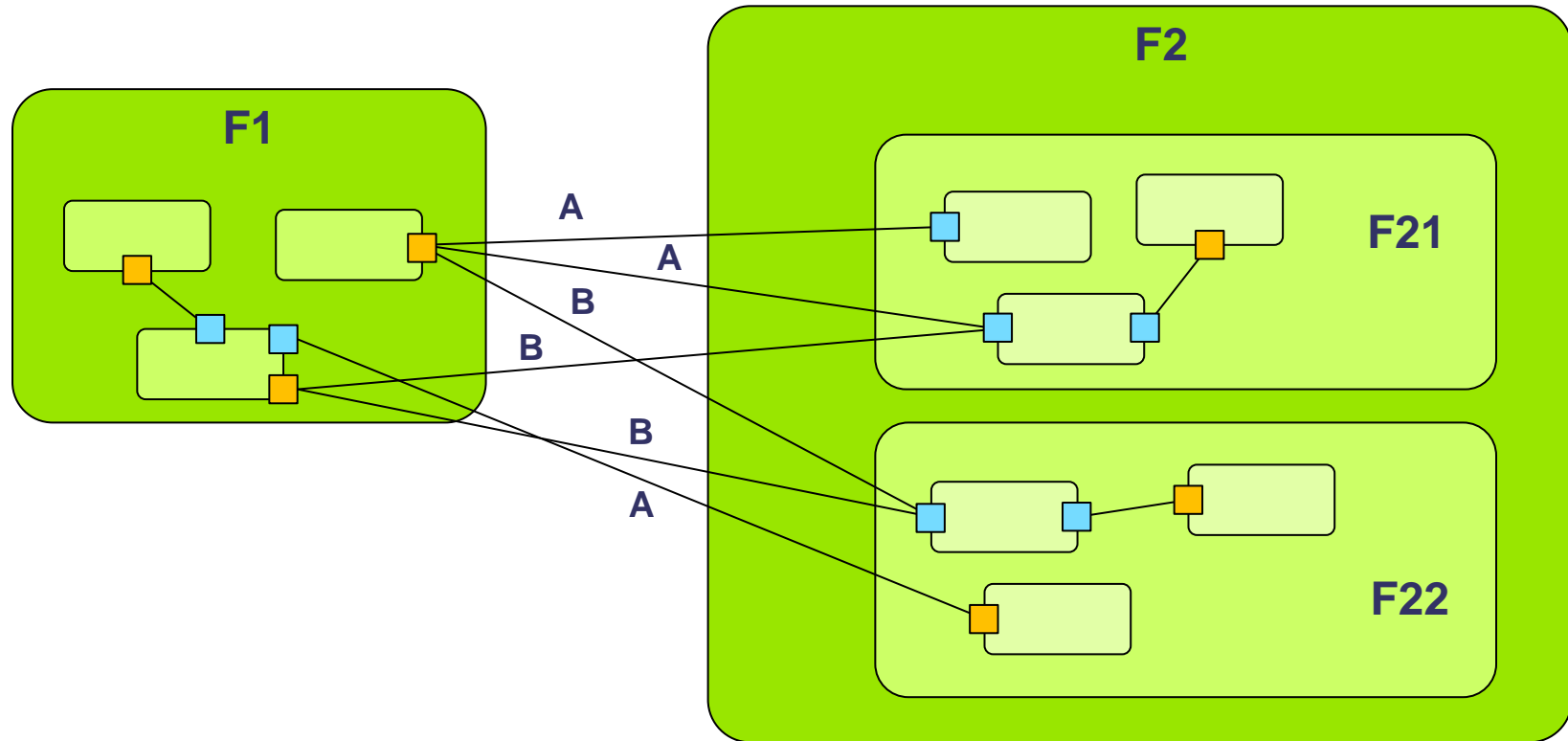
How to analyze transverse topics?  
How to get transverse overviews?

**Challenge: Build and maintain simplified views**

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

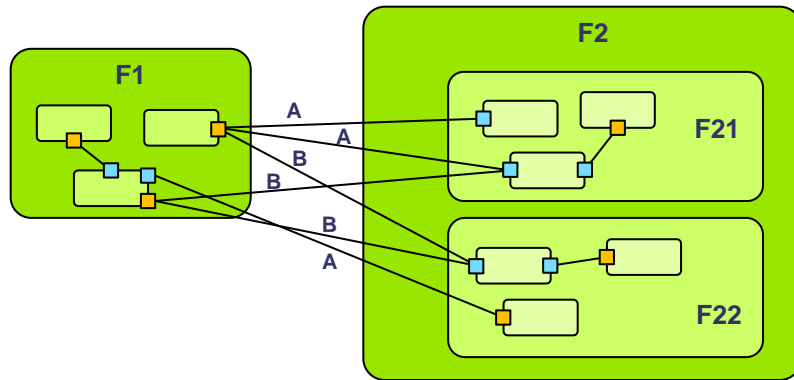


## MODEL

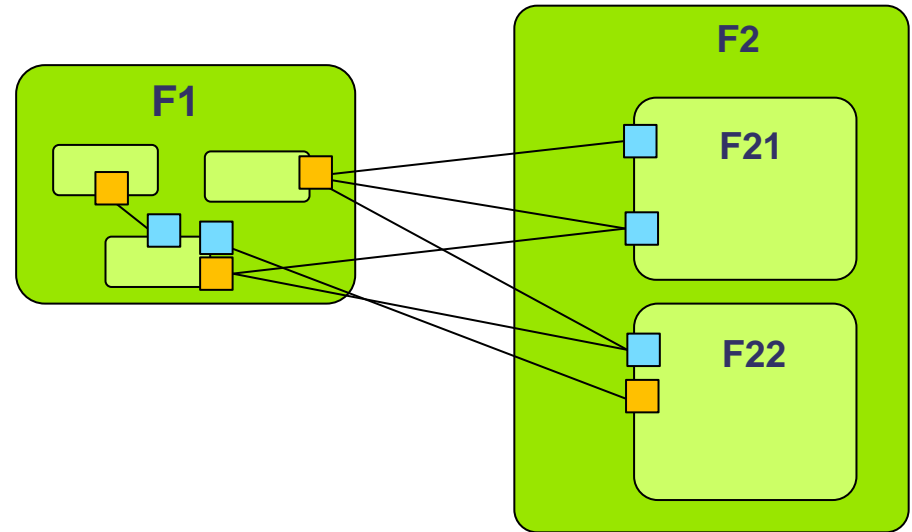




## MODEL



## VIEW

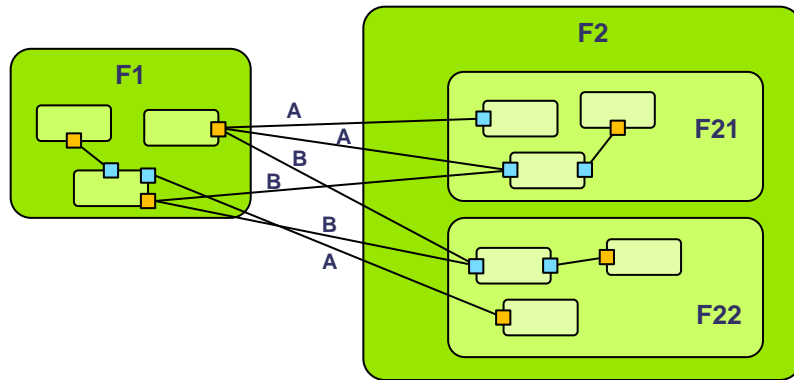


**THEORY:  
DELEGATION**

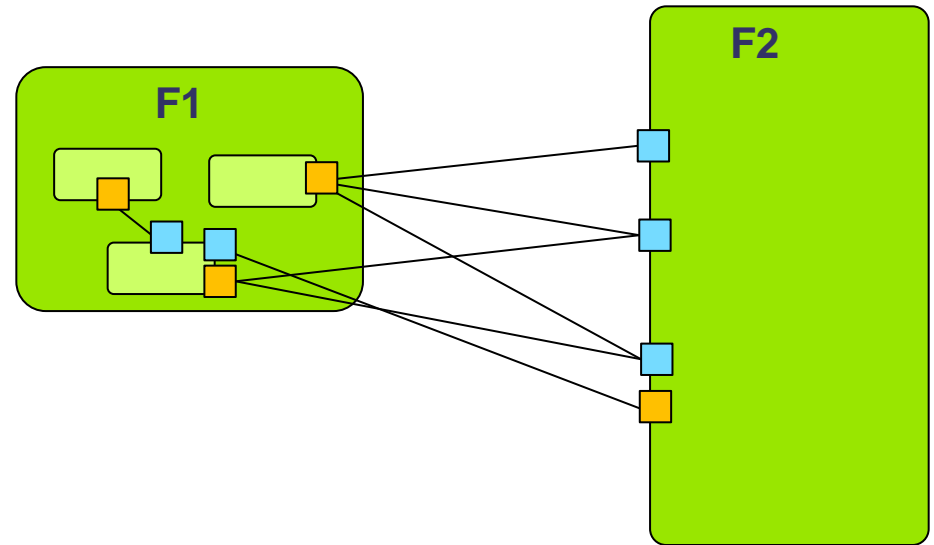
Children of F21 and F22 not displayed

Ports on F21 and F22 are graphically  
computed

## MODEL



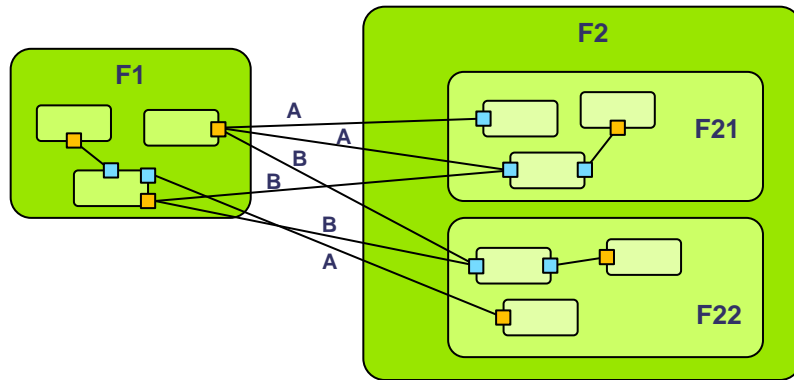
## VIEW



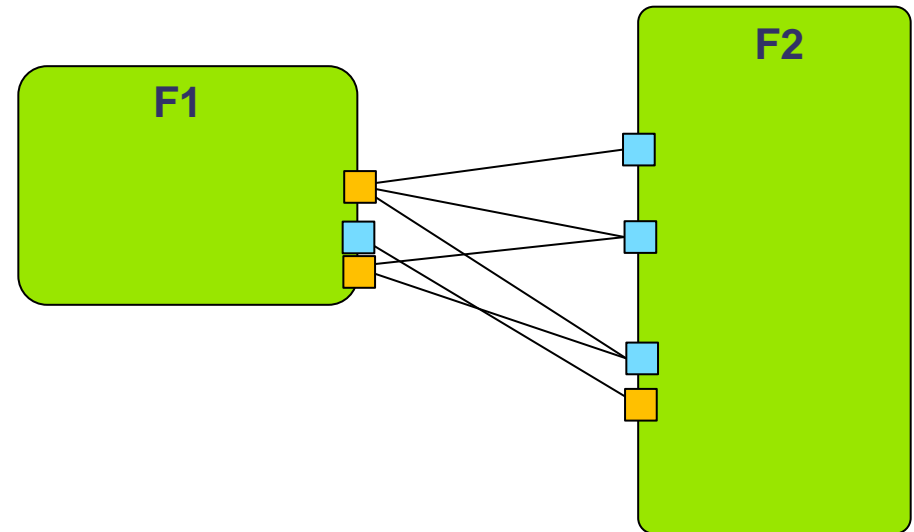
Children of F2 not displayed

Ports on F2 are graphically computed

## MODEL



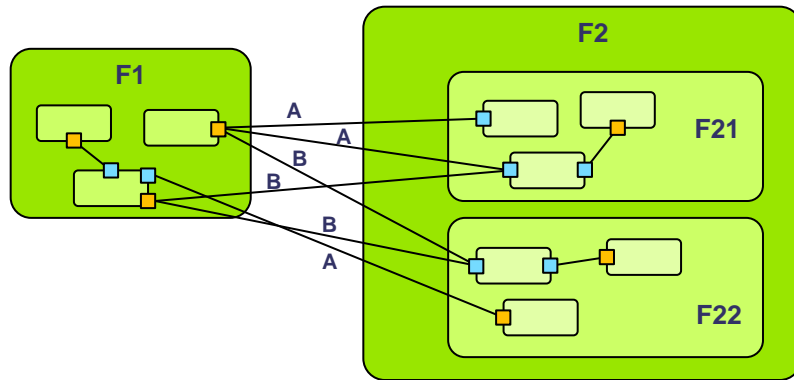
## VIEW



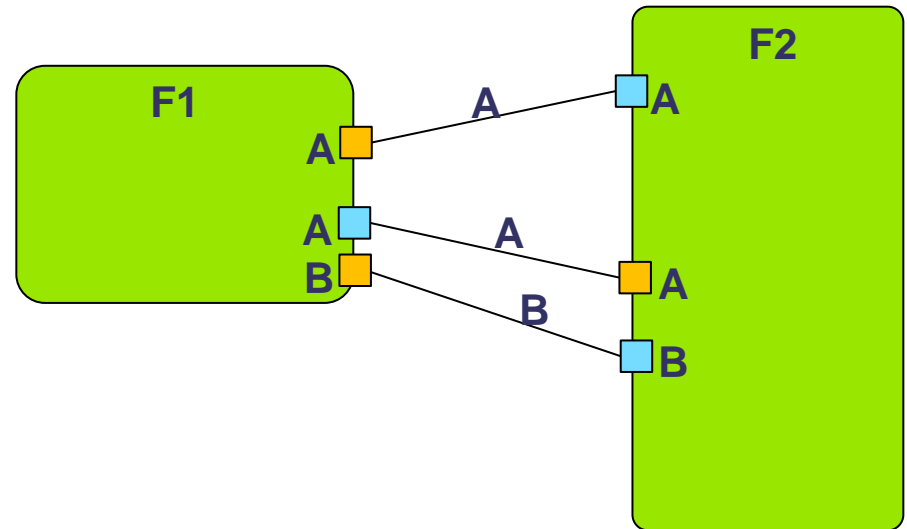
Children of F1 and F2 not displayed

Ports on F1 and F2 are graphically computed

## MODEL

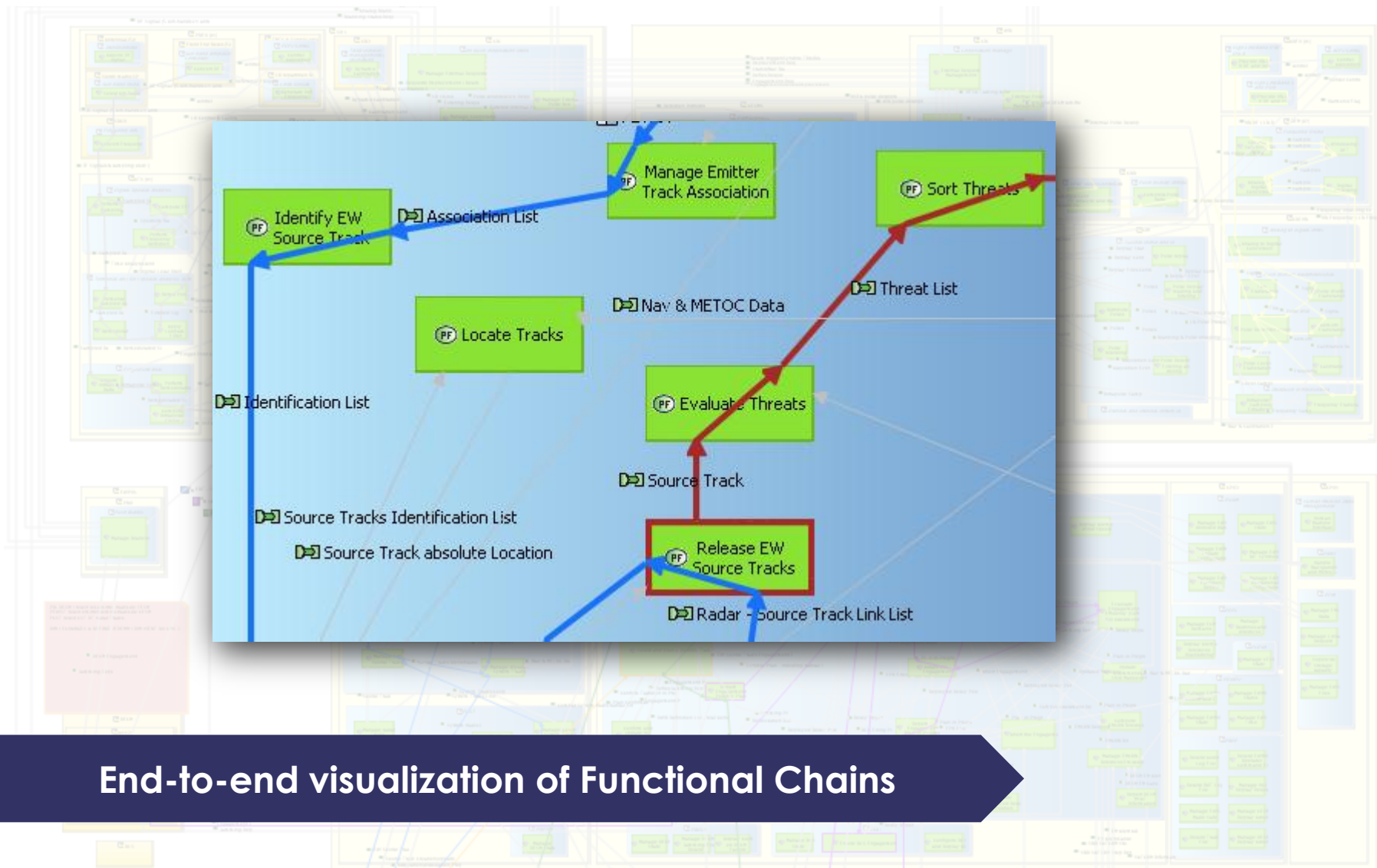


## VIEW



Tag-based simplification mechanism

Computed graphical simplifications free engineers from tedious and error-prone maintenance of abstraction levels



## End-to-end visualization of Functional Chains



## Use Case 2:

# MBSE-based Change Management



## Context

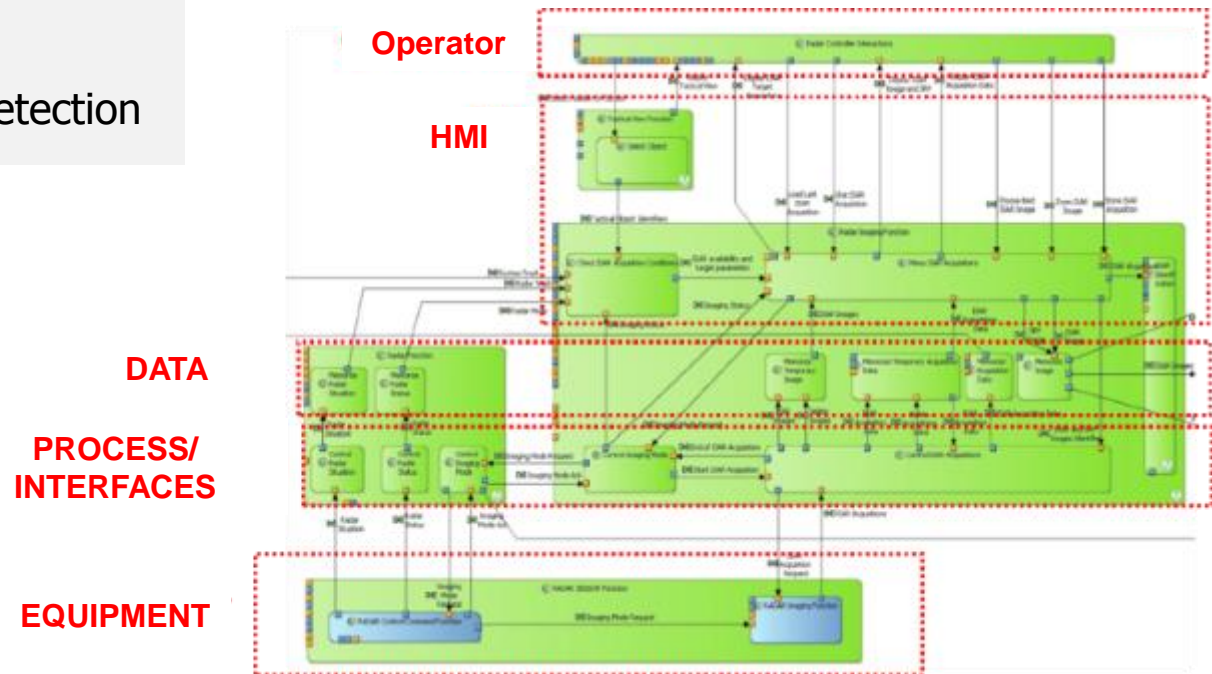


## MBSE usage

- Maritime Patrol Program delivered to the Customer
  - New functionalities asked by the Customer
- 
- Up-to-date model of the delivered System available
  - Modification of the model in order to:
    - **Estimate feasibility, cost and risks**
    - Drive developments and IVVQ
  - **Product line** management

## A regular reading pattern

- Multiple contributors modelling the same way
- Facilitates first access to diagrams
- Eases diagram review
- Allows quick inconsistency detection

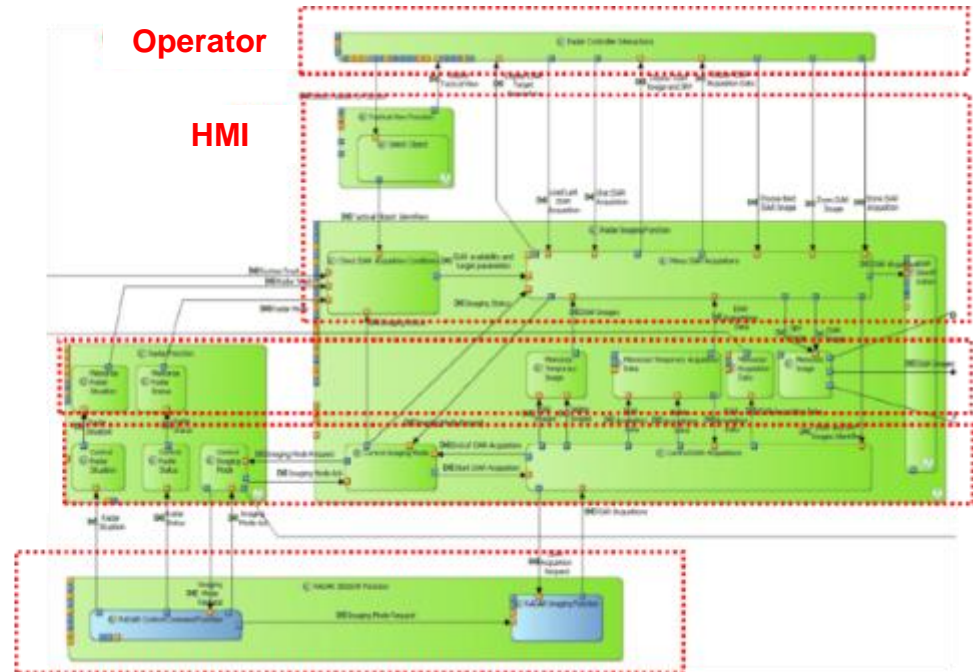



SSS: Need



PIDS: Reverse  
Engineering from  
Software Specification

**DATA**  
**PROCESS/  
INTERFACES**  
**EQUIPMENT**





## Elementary work decomposition

- Panels
- External / Internal Interfaces
- Data Memorisation
- Processing Complexity

### Computed Data: Estimated Cost

## Capella Outputs (model export)

DE	F	G	H	I	J	K	N	O	P	Q	R	T	AI	AJ	AK	AL	AM	AP	AQ	AR
Fct 1	Fct 2	Fct 3	Fct 4	Fct 5	Fct 6	Fct 9	Panneaux OM ou	Interfaces interfaces	Interfaces extemes	Memorisation: 1 st	Algo: Niveau 1	Consommateurs	ire	Fonctions chiffrees		Complexité Calculée	Implem TU retenue sans		à développe r l'Exemple 40% stimul	Implem et TU avec Reuse et
							528	1485	82	45	697		238			13650			100%	
							800	1300	70	50	400					14843		New	77%	
							7	2	5	2	10		1	####				Reuse	15%	0
							Complexité													
							Totaux	528	1485	82	45	697	234	238		16009	13912	COTS	8,4%	9419
								1	3	0	0	5	1	1		69	69	New	100%	69
								0	3	0	0	1	1	1		20	20	New	100%	20
								0	4	0	1	0	1	1		18	18	New	100%	18
											0	0	0							
											0	0								
											0	0								
								3	15	0	0	2	2	1		101	101	New	100%	101
								0	11	0	1	0	1	1		32	32	New	100%	32
								2	8	0	0	4	2	1		96	96	New	100%	96
								2	8	0	0	2	2	1		70	70	New	100%	70
								0	2	0	0	2	1	1		30	30	New	100%	30
											0	0	0							
								4	9	0	0	0	1	1		57	57	New	100%	57
								0	5	0	0	0	0	1		10	10	New	100%	10
											0	0	0							
								0	4	0	0	0	0	1		14	14	New	100%	14
								0	9	0	1	0	1	1		31	31	New	100%	31
								2	3	0	0	0	0	1		20	20	New	100%	20
											0	0	0							
								0	6	0	1	0	1	1		17	17	New	100%	17
								0	6	0	1	0	1	1		17	17	New	100%	17
								0	8	0	1	0	1	1		23	23	Reuse	75%	17
								4	8	0	0	5	1	1		103	103	COTS	50%	52
								0	6	0	0	0	0	1		10	10	New	100%	10



## Use Case 3:

# Multi-Level Engineering

OPEN

THALES

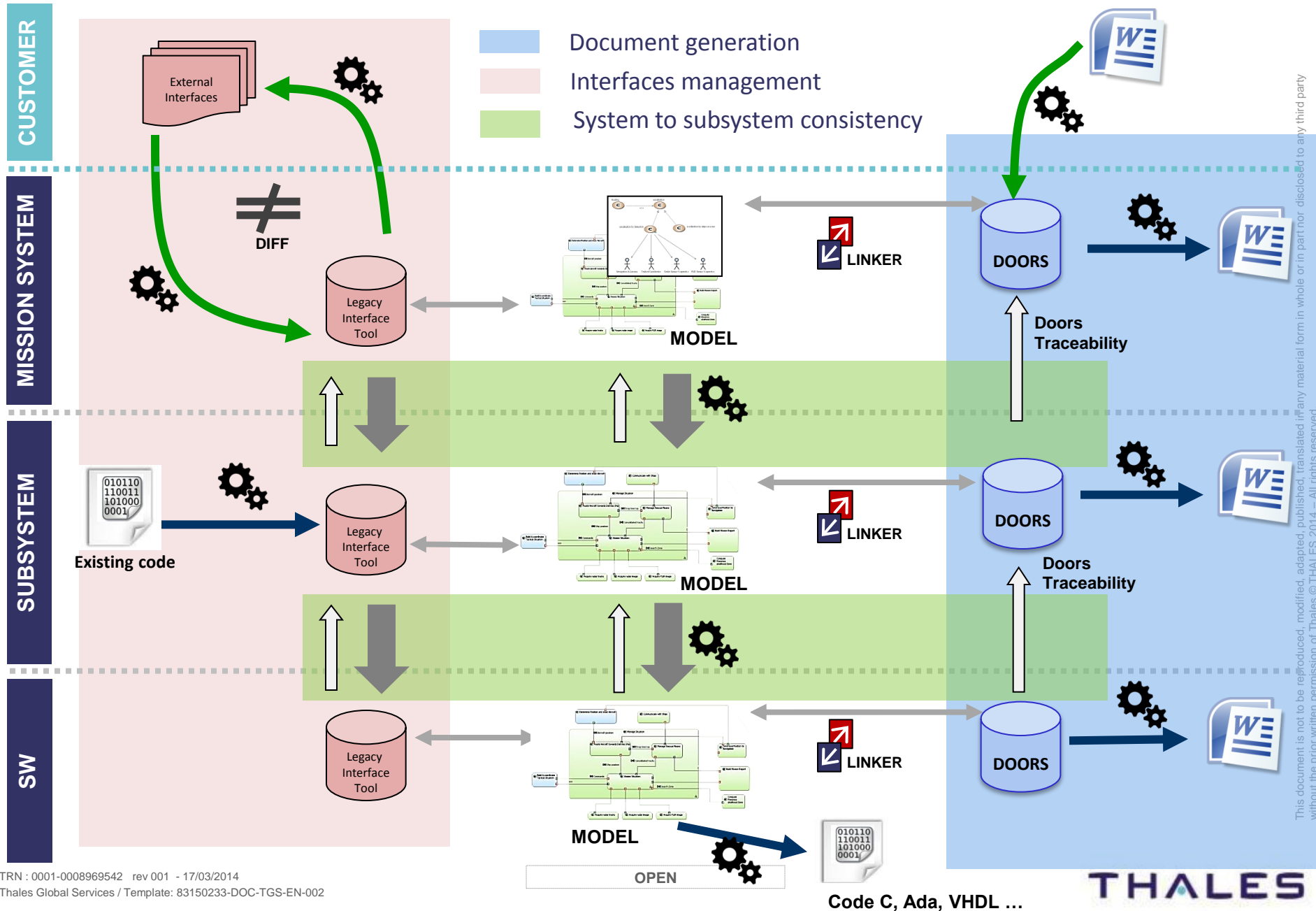


## Context




## MBSE usage

- Complex systems with full Thales responsibility (from Mission System to SW Component)
- Setup a global, **multi-level engineering approach**
- Joint effort with Thales Airborne Systems / Thales Corporate to **specify and develop an automated, iterative transition**
- Incubation on two projects
- Now integrated in the product and used in other contexts

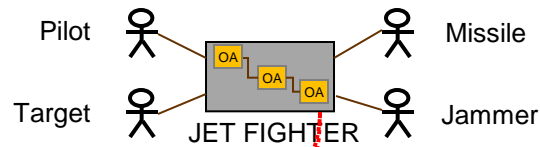


## Radar Engineering

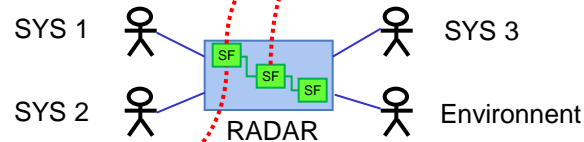
## Processing Unit Engineering

 Traceability link

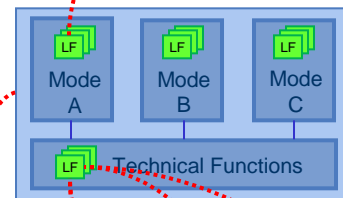
## Operational Analysis



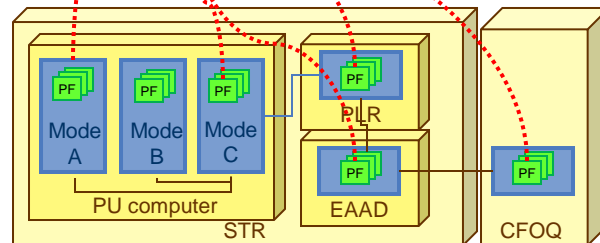
## System Analysis



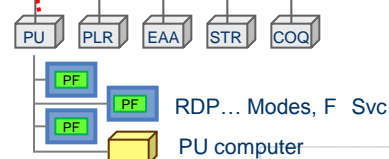
## Logical Architecture



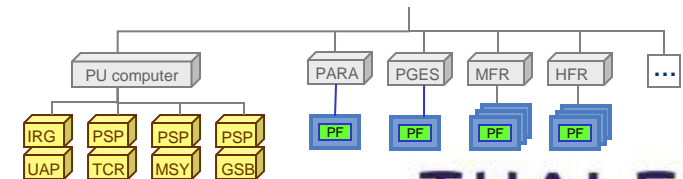
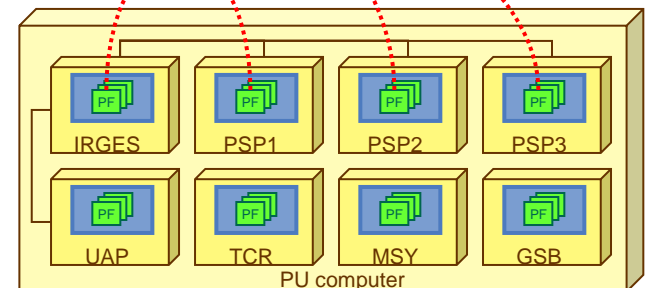
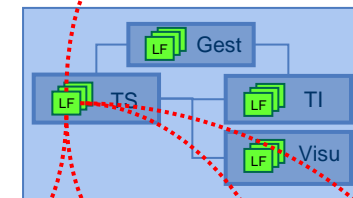
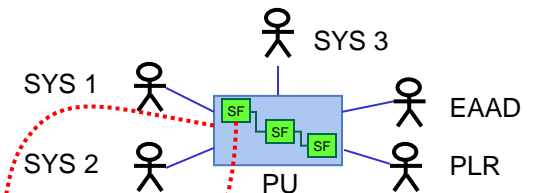
## Physical Architecture



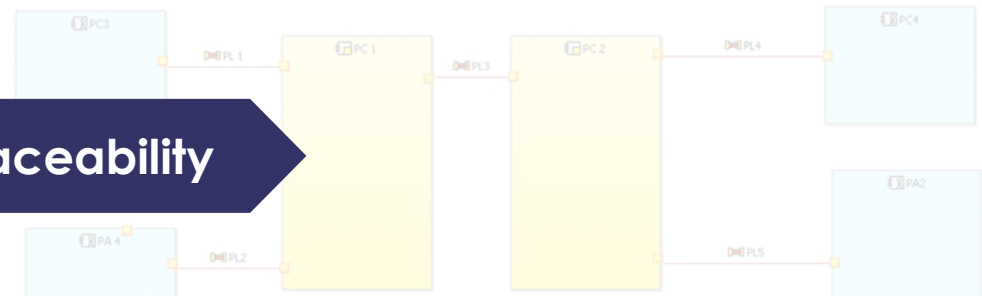
## Product Breakdown Structure



OPEN



THALES



## Computed system - subsystem traceability

OPEN

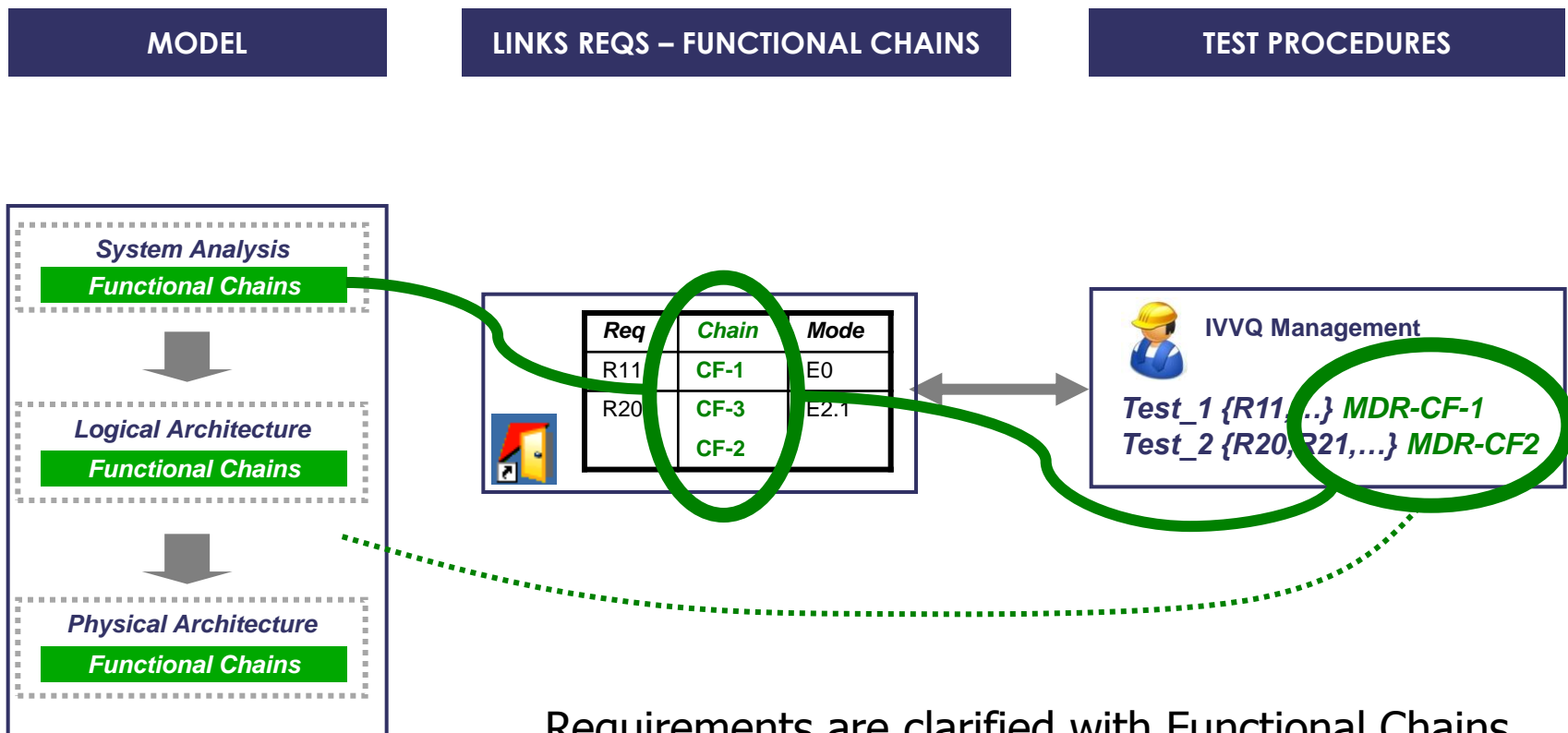
THALES



## Use Case 4: Model-driven IVV

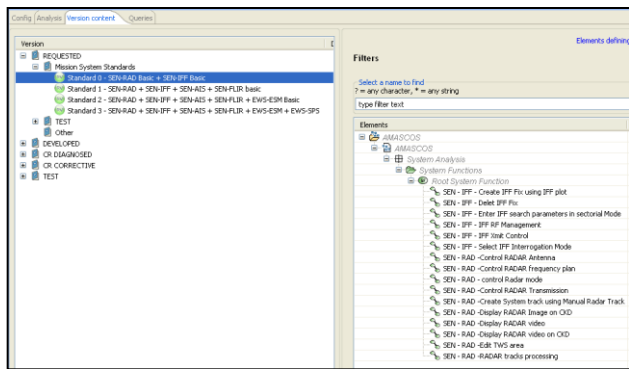
OPEN

THALES



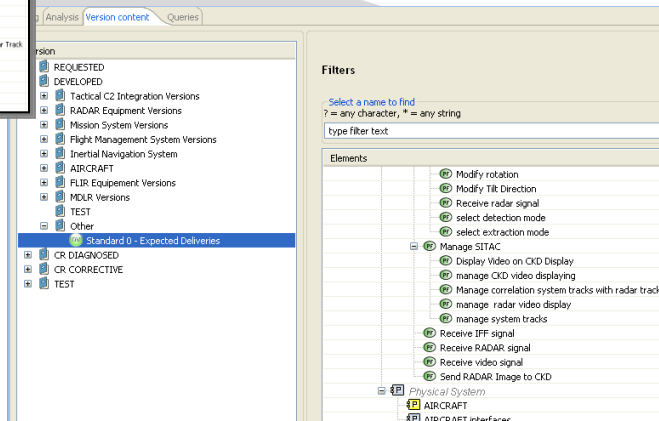
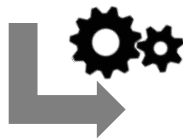
Requirements are clarified with Functional Chains

Test Procedures are linked to Functional Chains

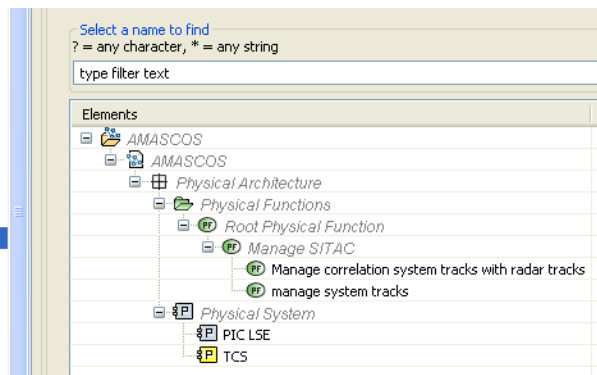


Define operational content expected for each project milestone

Deduce functional content and components to be delivered



- [-] Tactical C2 Integration Versions
  - [+] C2 Version 5.1
  - [+] C2 Version 5.2
  - [+] C2 Version 6.1
  - [+] C2 Version 6.2
- [-] Components
  - [-] NAV Component Versions
    - [+] PIC NAV 15.1
  - [-] Tactical Situation Versions
    - [+] PIC LSE 17.1
    - [+] PIC LSE 17.2
  - [-] RADAR Component Versions
    - [+] PIC RAD 16.1
    - [+] PIC RAD 16.2
  - [-] Video Component Versions
    - [+] PIC VID 19.1
    - [+] PIC VID 19.2

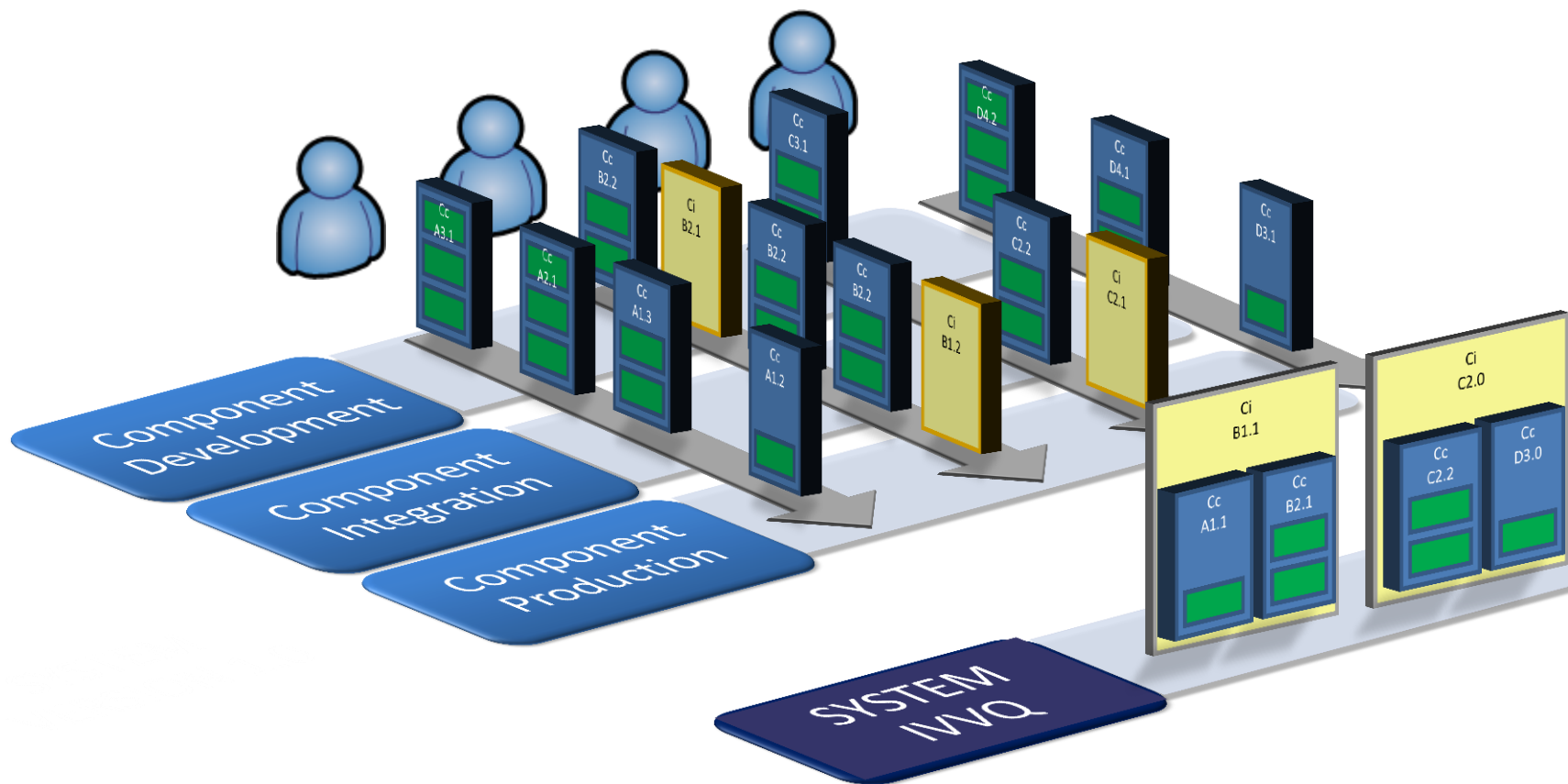


Define components versions and content

OPEN



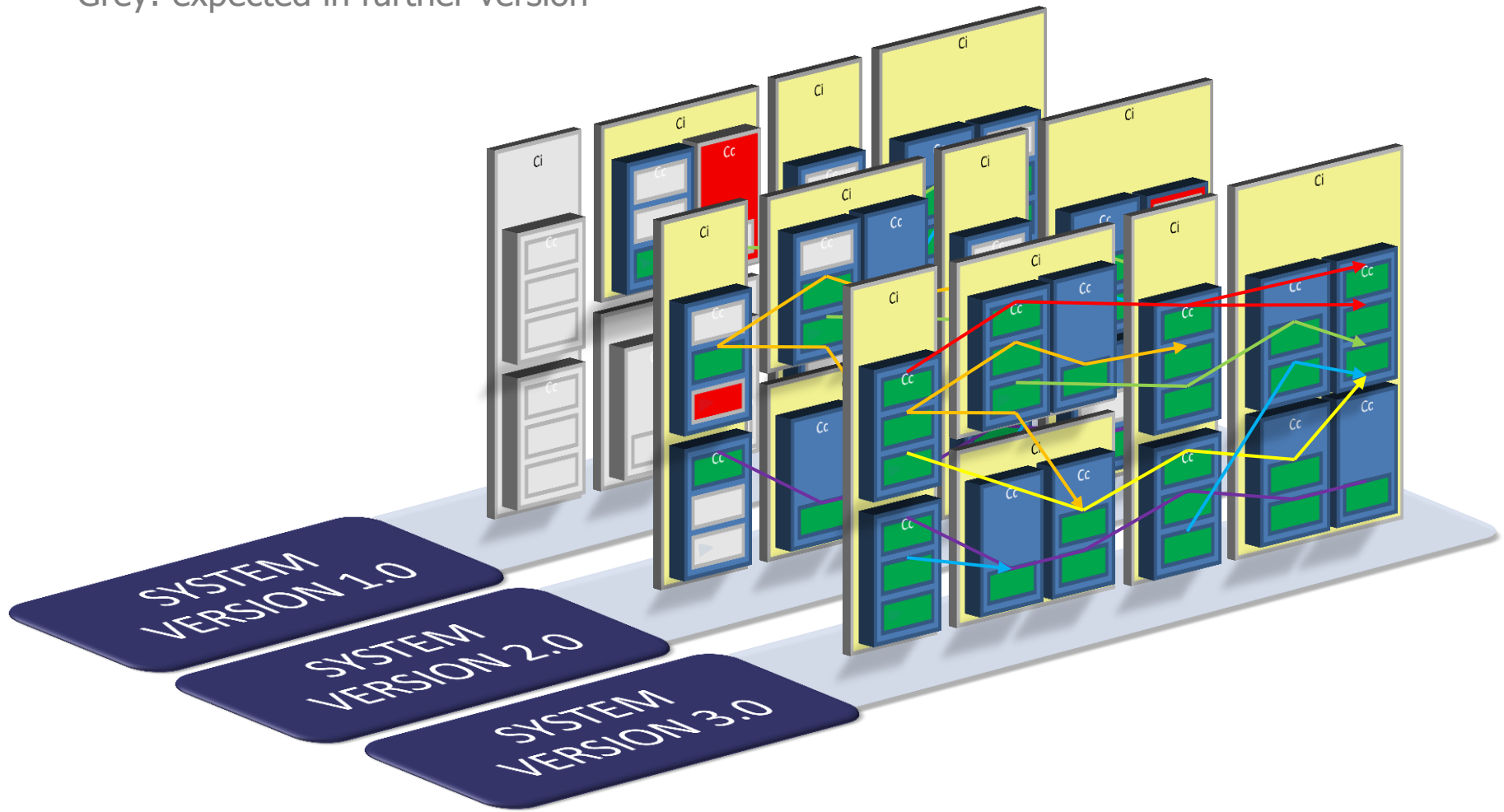
Blue: Software  
Yellow: hardware

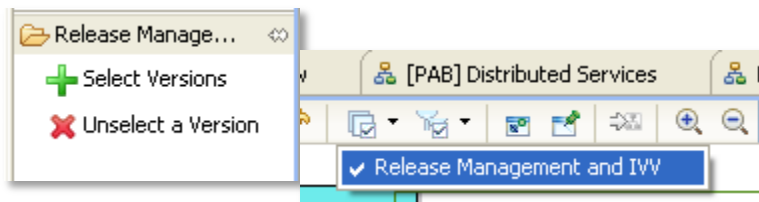




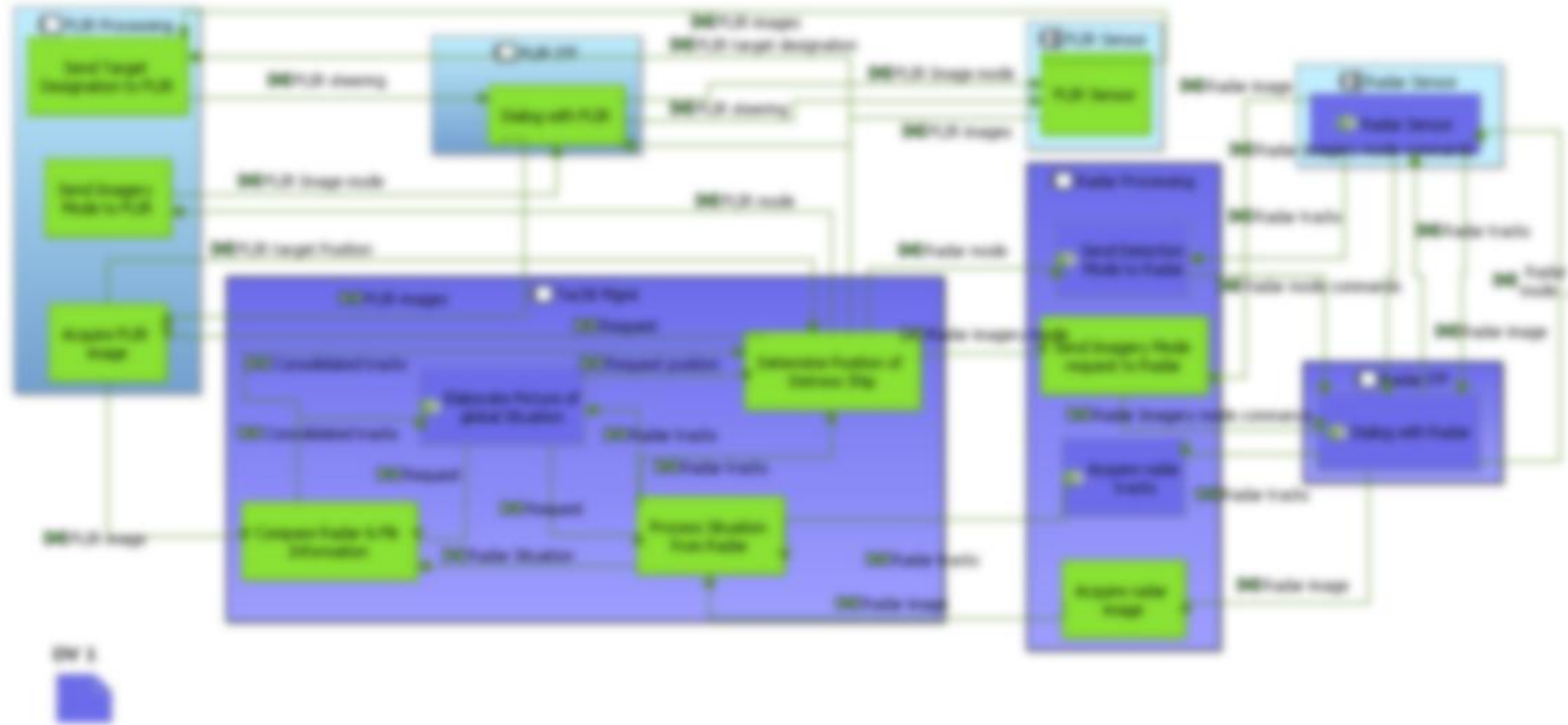
Red: Delayed, missing

Grey: expected in further version





Release management viewpoint:  
Automated visualization of versions



Developed Version 1  
Available elements in BLUE



This document  
without the pri



This document  
without the pri

## Compare Planned vs Developed versions

< Properties IVVQ [AMASCOS] REALIZATION [FREMM] DDV - Root Physical Function [FREMM] Console DAE Property View GLOBAL [AMASCOS] Semantic Browser

Config Analysis Version content **Queries**

Category
Queries

- List all DV provided elements with ES complement
- List all needed elements to integrate this DV
- Compare needed elements to integrate this DV and all elements provided by the Enabling System Real se

ES analysis

RV / RV comparison

DV / DV comparison

DV / RV comparison

DV, RV scheduling

Which expected elements are provided by this DV with ES complement ?

- Compare ES elements with the ES theoretic defined by the set of requiring DV
- Compare needed elements between two RV
- Compare requested elements between two RV
- Compare developed elements between two DV
- Compare provided elements between two DV
- Compare : DV developed elements versus RV needed elements
- Compare : DV developed elements with ES complement versus RV needed elements
- Compare : DV provided elements versus RV requested elements
- Compare : DV provided elements with ES complement versus RV requested elements
- Compare : DV provided elements with ES complement versus RV needed elements
- Export to graphml for scheduling representation

Diff C2 Version 5.1 vs Standard 0 - SEN-RAD Basic + SEN-IFF Basic

Version	Phase	Elements
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
C2 Version 5.1	Physical Architecture	
		SEN - IFF - Create IFF Fix using IFF plot
		SEN - RAD -Control RADAR Transmission
		SEN - RAD -Edit TWS area
		SEN - IFF - Select IFF Interrogation Mode
		SEN - RAD -RADAR tracks processing
		SEN - RAD -Display RADAR video on CKD
		SEN - IFF - Delet IFF Fix
		SEN - IFF - Enter IFF search parameters in sectorial Mode
		SEN - RAD -Control RADAR frequency plan

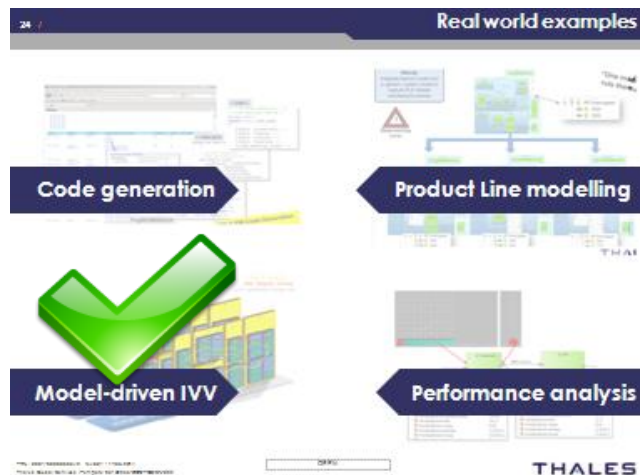
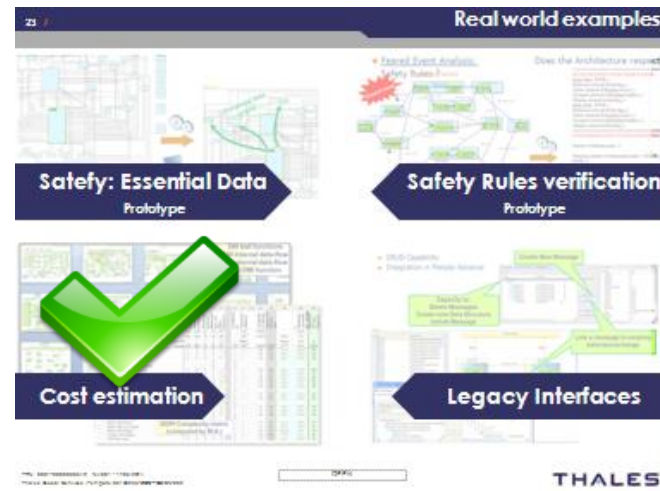
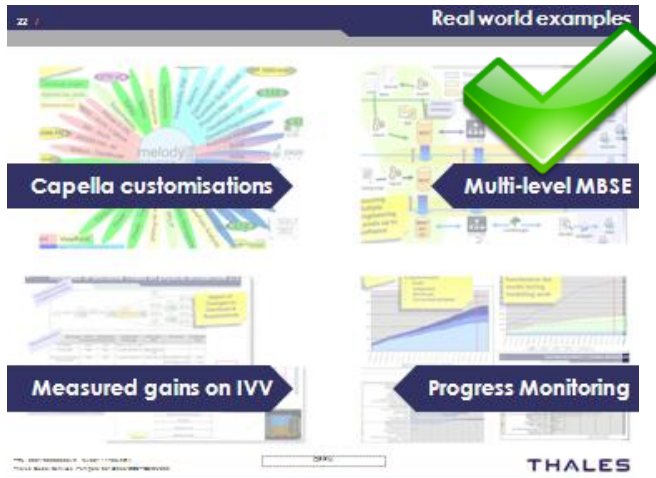
Send query

Clear All

Export

Display on Diagram





And more to come!





**Thank you for  
your attention!**

**Any Questions?**

## Capella Open Source Project

<https://www.polarsys.org/projects/polarsys.capella>

The screenshot shows the Polarsys website with the following elements:

- Logo:** POLARSYS Open Source Tools for Embedded Systems
- Navigation Bar:** Home, About, News, Projects, Downloads, Technologies, Community, Contact Us
- Icons:** A row of seven icons representing different domains: a truck, a heart, a crossed-out airplane, a car, a wind turbine, a radio tower, and a satellite.
- Breadcrumbs:** Home / Polarsys / Capella / Capella
- Project Navigation:** View, Downloads, Who's Involved, Documentation, Developer Resources, Contact Us
- Section Header:** Capella
- Description:** The *Arcadia* engineering method mainly focuses on functional analysis, complex architecture definition and early validation. It is highly extensible and customizable through viewpoints providing integrated specialty engineering support. Both the method (which is going to be published and standardized) and the Capella ecosystem are already operationally deployed within Thales in defense, aerospace, space, transportation and security business domains, across several countries, thanks to a large-scale rollout of model-based approaches, with hundreds of daily users worldwide, on critical operational projects.

## Contacts

Stephane.Bonnet@thalesgroup.com

Jean-Luc.Voirin@fr.thalesgroup.com

Daniel.Exertier@thalesgroup.com

## In Australia

Fabrice.Lestideau@thalesgroup.com.au

OPEN

**THALES**