Modelling Avionics Interfaces and Generating ICDs for the Propulsion Subsystem of the MPCV-ESM

Délia Cellarier

28/09/2020

ORION

Taking Humans to the Moon and Beyond

Multi-Purpose Crew Vehicle

European Service Module

For the first time, NASA will use a European-built system as a critical element to power and propel an American spacecraft.

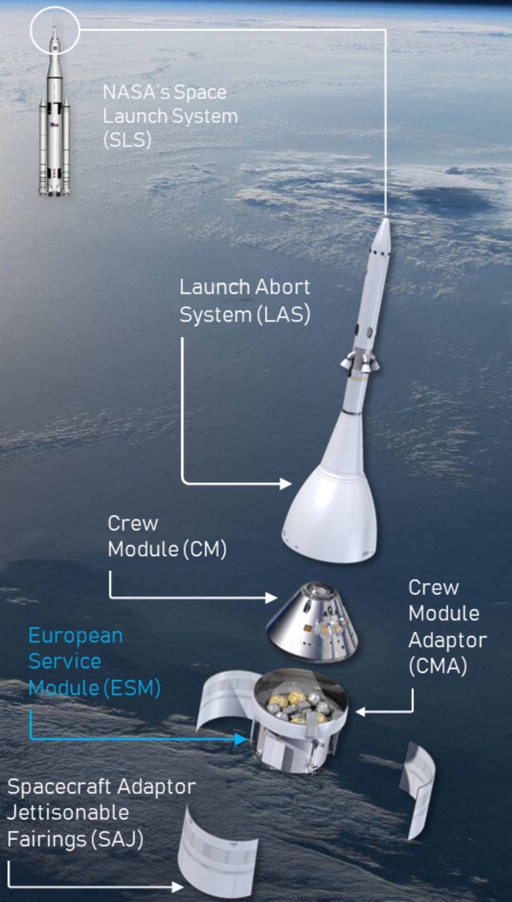
Provided by ESA and its prime contractor Airbus Defence and Space, ESM is Europe's contribution to NASA's Orion spacecraft, supplying critical functions for Artemis missions:

- Spacecraft **propulsion**
- **Electrical power** supply
- **Thermal** control
- **Consumable storage** (air and water)

20,000

It is the number of parts and components in ESM that must fit together perfectly and perform reliably.

Orion FM-1 is fit for flight



ESM Avionics architecture

ESM is controlled by a set of electronic units:

- **Propulsion Drive Electronics (PDE)**

Controls the propulsion HW

- **Pressure Regulation Unit (PRU)**

Controls the pressure in ESM propulsion tanks

- **Power Control & Distribution Unit (PCDU)**

Provides 28V and 120V power, and controls the power provided to / supplied by Crew Module batteries

- **Solar Array Driving Electronics (SADE)**

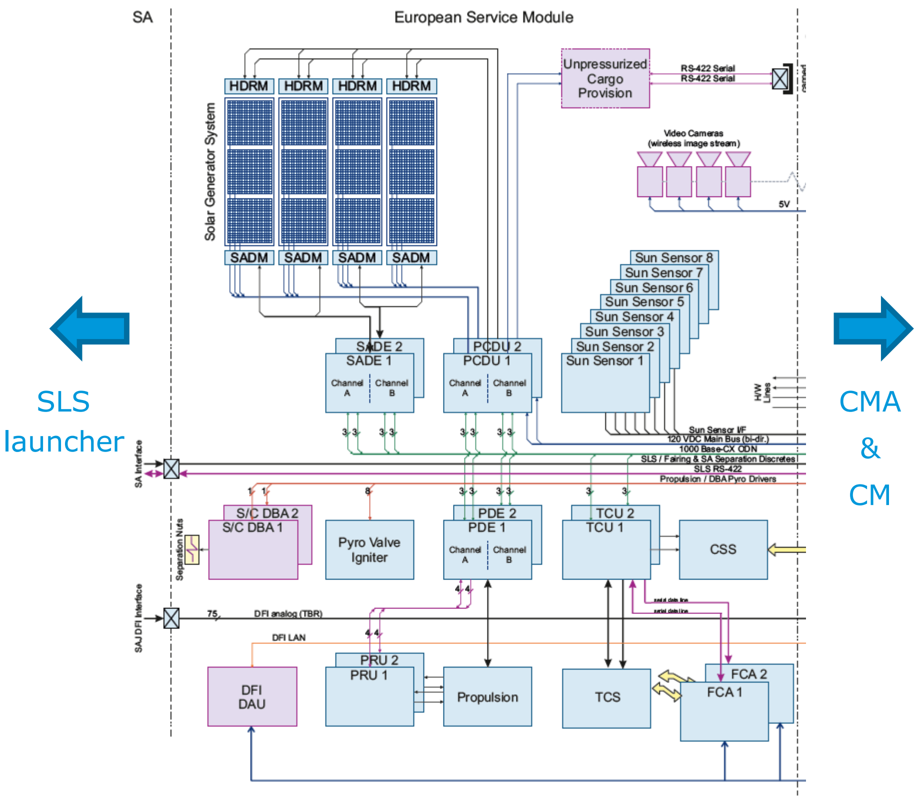
Controls the rotation of ESM solar arrays via 4 SADM

- **Thermal Control Unit (TCU)**

Manages active & passive thermal control systems, and storage and delivery of consumables

- **Fluid Control Assembly (FCA)**

Control of pumps and valves for the active thermal control system



Problematic

Challenge:

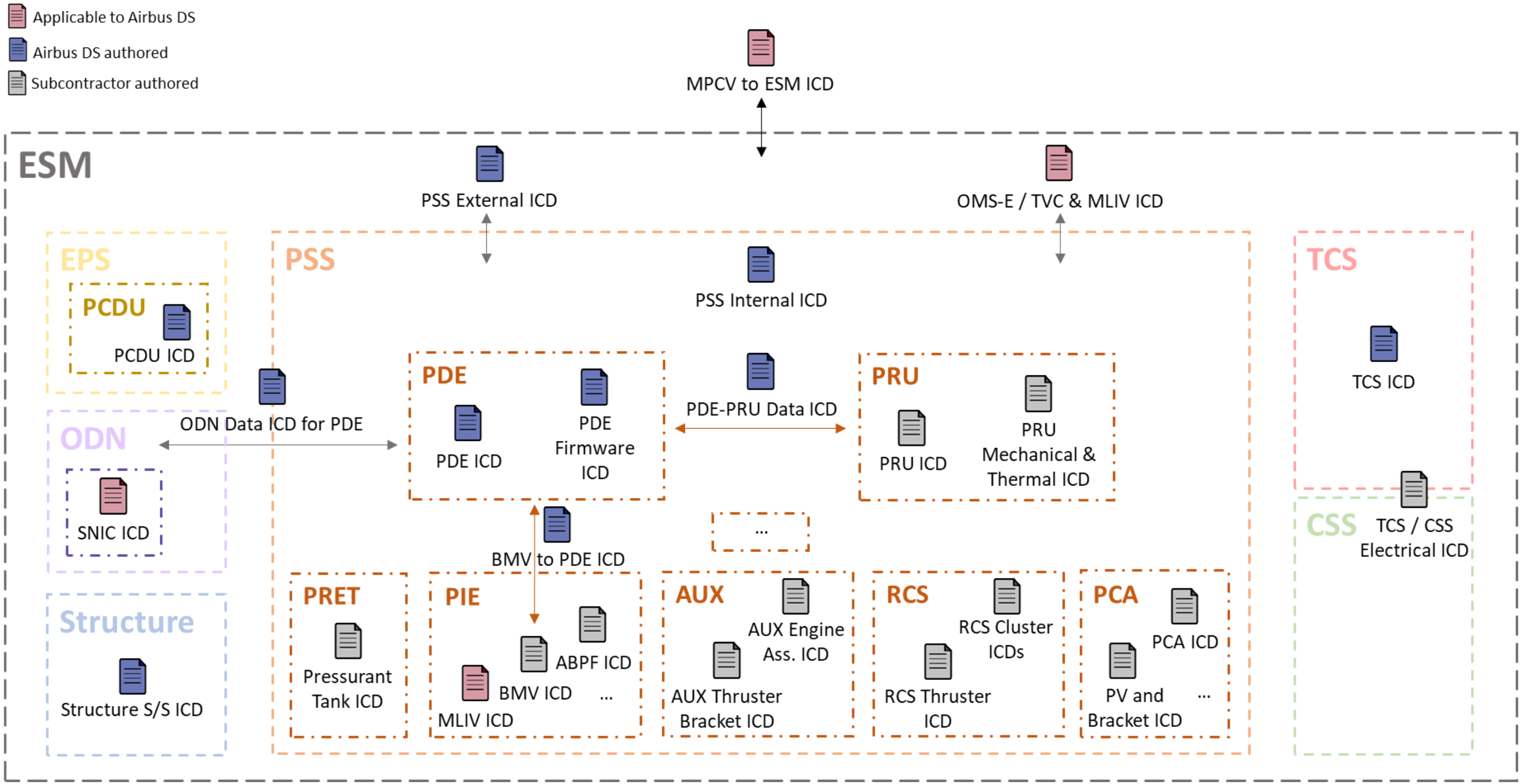
Management of

Interface Control Documents (ICDs)

- Technical and contractual aspect (⇒ standardized change process)
- Redundant information between ICDs of different levels or separated ICDs for one equipment
- Largely manual maintenance and verification

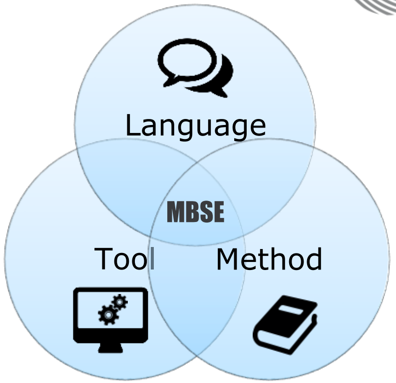
→ Inconsistencies between documents

ICD map (focus on Propulsion S/S)



Proposed approach

Implementing a **model-based approach**, using existing MBSE technologies, to effectively **manage information** and **generate ICDs** from a model

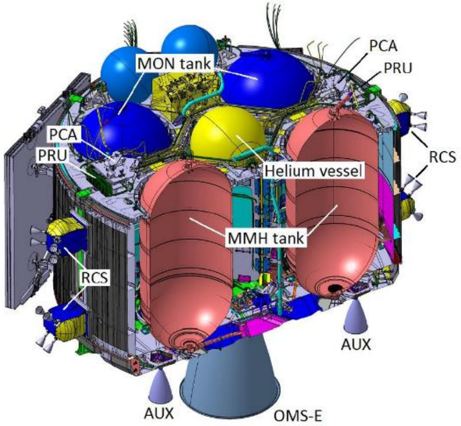


Case study:

Avionics interfaces of ESM3 Propulsion Subsystem (PSS)

→ Shadow engineering

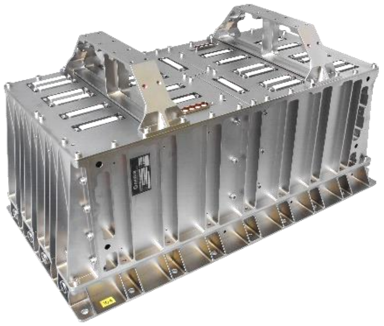
Propulsion Subsystem (PSS)



Commands

Measurements

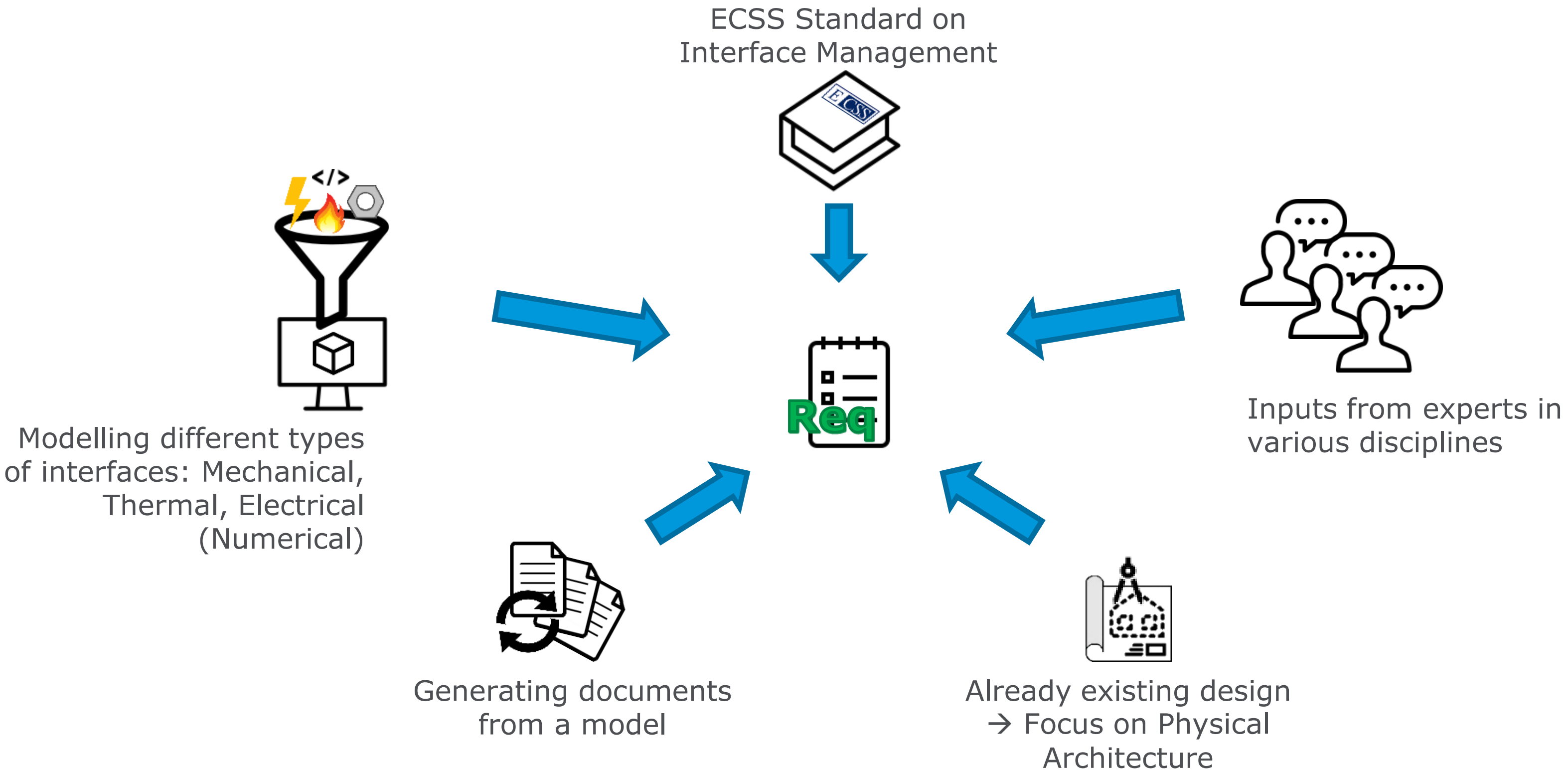
PDE Box



- Propulsion Subsystem (PSS)
  - Electronic Pressure Regulation (EPR)
    - Pressure Regulator Unit (PRU)
    - Pressure Control Assembly (PCA)
  - Pressurant Tanks
  - Propellant Tanks
  - Propellant Isolation Equipment (PIE)
  - Propulsion Drive Electronics (PDE)
  - Main Engine Assembly (MEA)
    - Thrust Vector Control Assembly (TVC)
    - OMS-Engine
  - Equipped Auxiliary Thrusters (AUX)
  - RCS Cluster Assembly (RCS)
  - Supports, Brackets and Tubing (SBT)
  - PSS Internal Harness
  - Test Ports (TP)

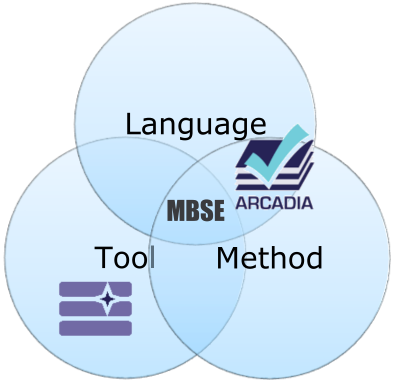
Avionics

Activity’s inputs and drivers

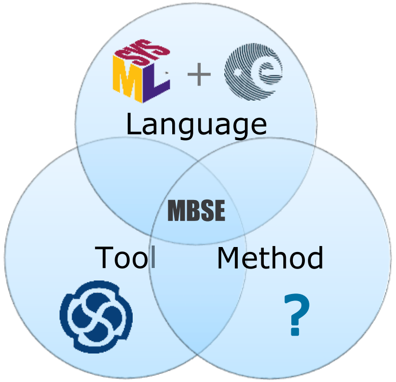


Trade-off for an MBSE solution

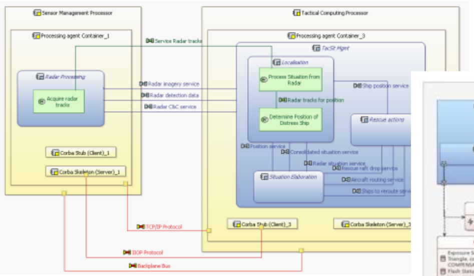
Capella



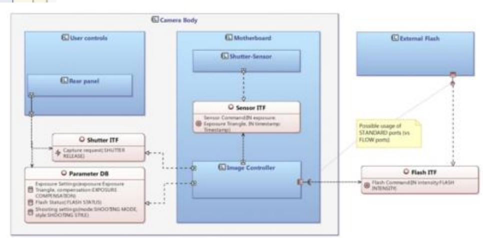
ENTERPRISE ARCHITECT



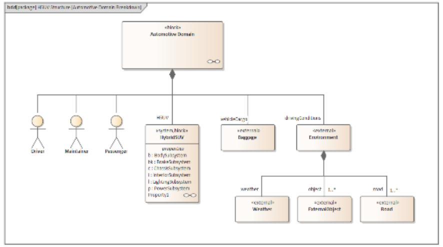
Physical Architecture Blank



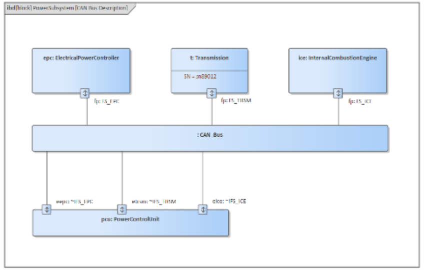
Interface Diagram Blank



Block Definition Diagram



Internal Block Diagram



Trade-off for an MBSE solution – cont’d

Capella

- ✓ Open-source
- ✓ Method to guide the user
- ✓ User-friendly
- ✓ Flexible redundancy management (REC/RPL)
- ✓ Open-source add-on to generate fully custom Word documents
- ✓ Version control with EGit
- ✓ Assisted extensibility with Viewpoint technology

- ✗ No integrated features to model documents

“Entry point” to MBSE, and a SW engineer can get the maximum from the tool

Enterprise Architect

- ✓ SysML community (Nasa,…)
- ✓ ESA SysML Toolbox
- ✓ Typed ports and connectors
- ✓ Integrated and complex features for document generation
- ✓ Modelling of document artefacts
- ✓ Baseline concept and version control (CVS)

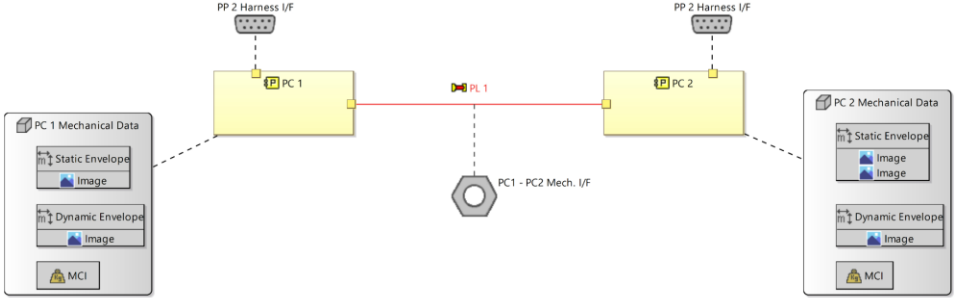
- ✗ Licensed
- ✗ Not intuitive
- ✗ Basic SysML validation rules not included
- ✗ Template customization not straightforward

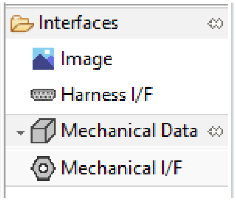
Steep learning curve, but can be really powerful

Modelling avionics interfaces – Capella

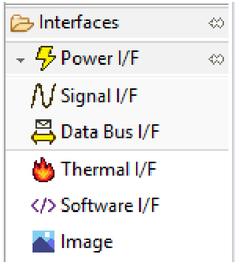
1. **Methodology**: choice of a mapping between Capella's model elements and types of I/Fs
2. **Interfaces Viewpoint**: development of a Capella extension which introduces new model elements to specify I/Fs' data, extends diagrams and includes custom validation rules

Physical Architecture Blank

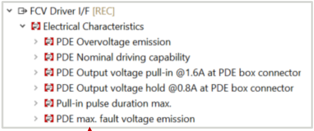




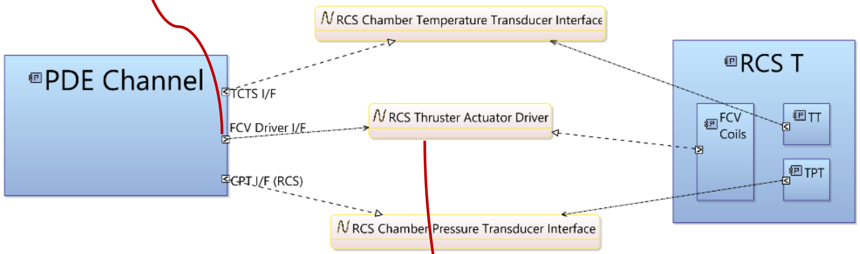
Interface Diagram Blank



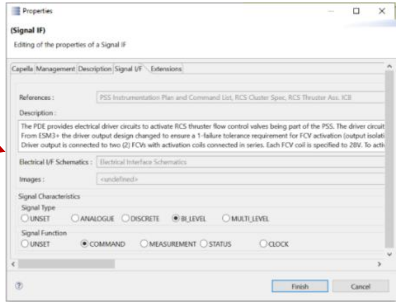
Modelling avionics interfaces – PDE-RCS example

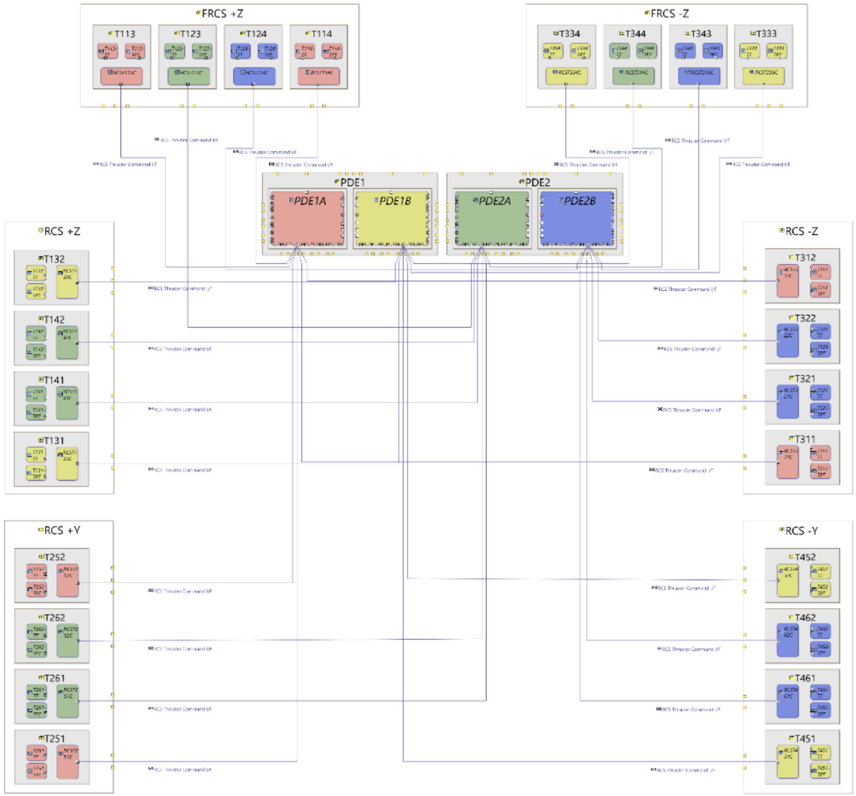


→ Electrical Characteristics added as Property Values at Component Port level



→ Dedicated tab in the Property Sheet for I/F data

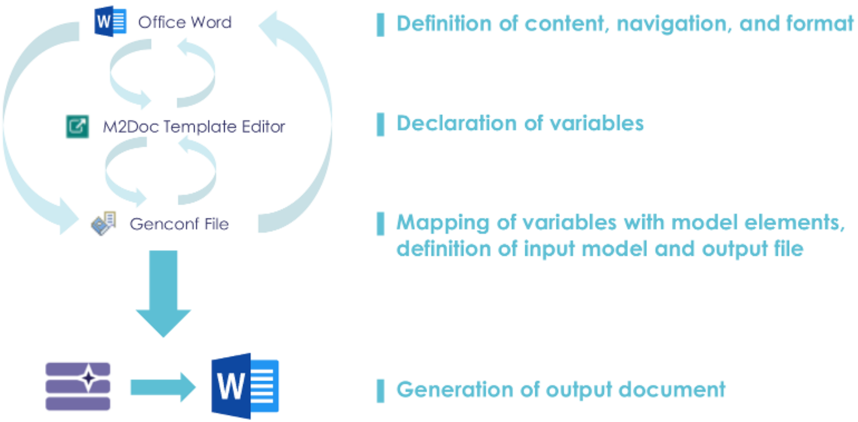




Generating ICDs

M2Doc

→ Open-source add-on to generate MS-Word documents from Capella models



Generating ICDs – cont’d

Templates

- → Edited using **Microsoft Word** editor
- → Uses Word **fields**
- → M2Doc language built on top of Acceleo Query Language (**AQL**) for querying the model
- → Can be extended with **custom services** (**Java**) ↓

Extract of a template provided in the example ‘In-Flight Entertainment System With M2Doc’:

{ m:if lc.ownedFeatures->filter (fa::ComponentPort).componentExchanges->size() <> 0 }

2.1.4 Component Exchanges

{ m:for ce | lc.ownedFeatures->filter (fa::ComponentPort).componentExchanges }

Name	{ m:ce.name }
Direction	{ m:ce.getCeDirection(lc) }
Destination Component	{ m:ce.getDestinationComponent(lc) }

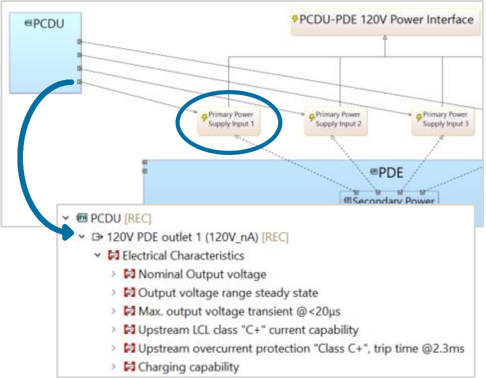
{ m:if ce.description.trim().size() <> 0 }

{ m:ce.description.trim().fromHTMLBodyString().replaceLink(ce) }

{ m:else }

No description

{ m:endif }




```
/**
 * Construct an MTable with the Electrical Characteristics of the
 * Component Ports involved in the given electrical Interface
 */
public MTable getElecCharacMTable(ElectricalIF elecIF) {
```

{ m: ... }

Template

Electrical characteristics	Unit	PDE interface	PCDU Interface
Nominal Output voltage	V	-	[redacted]
Nominal Input voltage	V	[redacted]	-
Min. Output voltage steady state	V	-	[redacted]
Min. Input voltage steady state	V	[redacted]	-
Max. output voltage steady state	V	n/a	[redacted]
Max. input voltage steady state	V	[redacted]	-
Max. output voltage transient @<20µs	V	-	[redacted]
Max. input voltage transient @<20µs	V	[redacted]	-
Input isolation to ground	MΩ	[redacted]	-
Upstream LCL class "C+" current capability	A	-	[redacted]
Upstream overcurrent protection "Class C+", trip time @2.3ms *)	A	-	[redacted]
Charging capability	mAs	n/a	[redacted]
Input filter charging	mAs	[redacted]	-

Table 7-1 Primary Power Supply 1 Interface Description

Generated ICD

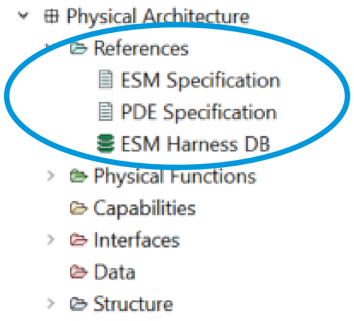
Traceability

With existing documents or resources:

In the **Interfaces Viewpoint**

- → Via References of documents
- → Link to external Harness DB

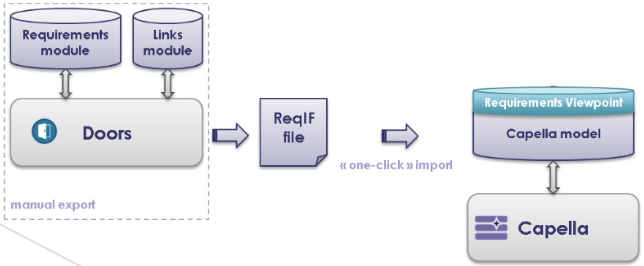
Custom validation rules included
Used in document generation



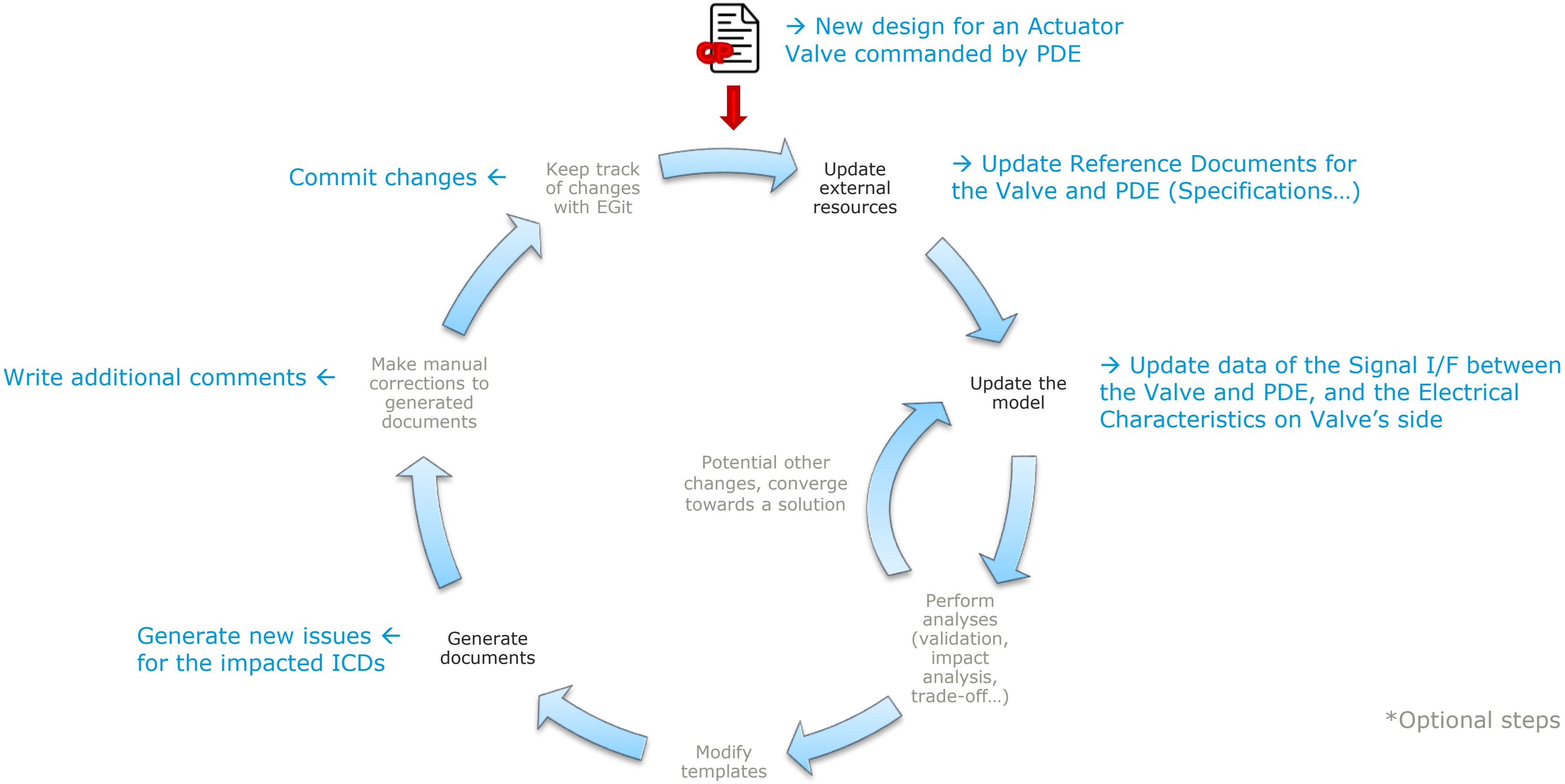
With elements across the lifecycle:

Other **existing Viewpoints** identified

- → Requirement Viewpoint
- → V&V / Test Means Viewpoint



Use Case



*Optional steps

Lessons learned

Choice of an MBSE solution

→ Has to be made according to one's needs

→ Several solutions exist and can be extended

→ Capella is a good option for a quick setup

ROI

→ There is a learning curve

→ An 'expert' can make things easier for end users (methodology, Viewpoint, custom validation rules, M2Doc custom services, templates…)

→ A model can help reducing the risk of errors caused by a wrong ICD and the cost of maintaining ICDs

Conclusion

→ MBSE can be introduced even in a document-oriented project

→ **MBSE technologies are ready** for contractual document generation. What need to be updated are engineering processes and standards to take them into account.

→ MBSE tools offer a wide range of features, and a model originally made for controlling interfaces **can easily be expanded for further use** (functional analysis, FDIR, Requirement management,…)

Thank you

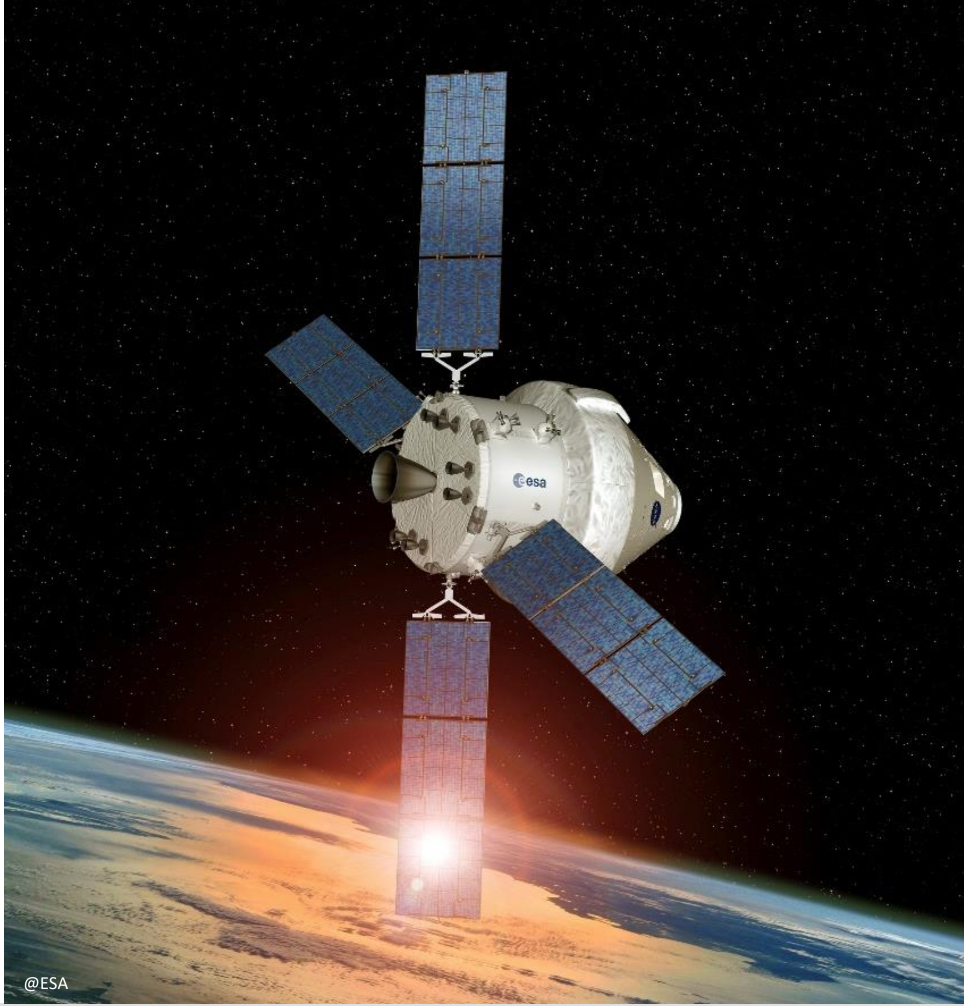
Any questions ?

Délia Cellarier

ESA/ESTEC – HRE-LS

Contact:

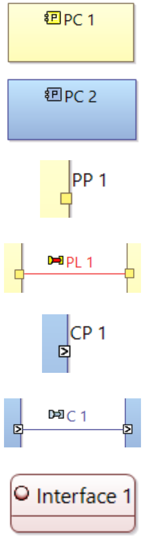
delia.cellarier@esa.int



BACK-UP SLIDES

Modelling avionics interfaces – Mapping

Chosen mapping to represent interfaces in Capella’s Physical Architecture:



Capella model element	Mapping
Physical Component (NODE)	Assembly, Avionics box
Physical Component (BEHAVIOR)	Avionics SW, sensor, actuator coil
Physical Port	Mechanical/Harness interface end
Physical Link	Mechanical interface plane
Component Port	Electrical/Thermal interface end
Component Exchange	Electrical/Thermal interface plane
Interface	Electrical/Thermal/SW interface

Next steps

Applying this approach to a real project

- → Application of MBSE for interface management can be included in the statement of work for new developments, supporting the system level definition of interfaces and generation of corresponding ICDs
- → The model should be delivered in complement to the model-generated ICDs, allowing the customer to perform analysis

Modelling for a multi-organisational project

- → As forward work, the interface modelling described in this paper is to be extended covering the multi-level collaboration and the integration of models provided by subcontractors, allowing to apply the concept over the complete development and production chain