OBEO

THALES

CapellaDays
ONLINE 2021

POLITECNICO
MILANO 1863

DIPARTIMENTO DI
SCIENZE E TECNOLOGIE
AEROSPAZIALI

Advanced Space Technologies 4
Robotics & Astrodynamics

Enhancing CubeSat design through ARCADIA and Capella: a concrete application

Presenter:
Mr. Paolo Minacapilli

Authors:
Mr. Paolo Minacapilli
Prof. Michèle Lavagna

November 17, 2021

Overview of the presentation

Objectives

Introduction to MBSE and benefits

ARCADIA and Capella for a CubeSat design

Conclusions

ARCADIA

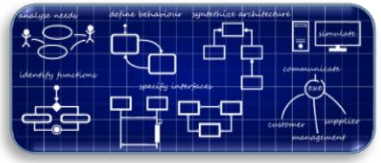
Capella
Open Source MBSE Solution

Need of an MBSE approach for space industry

AeroCube-10

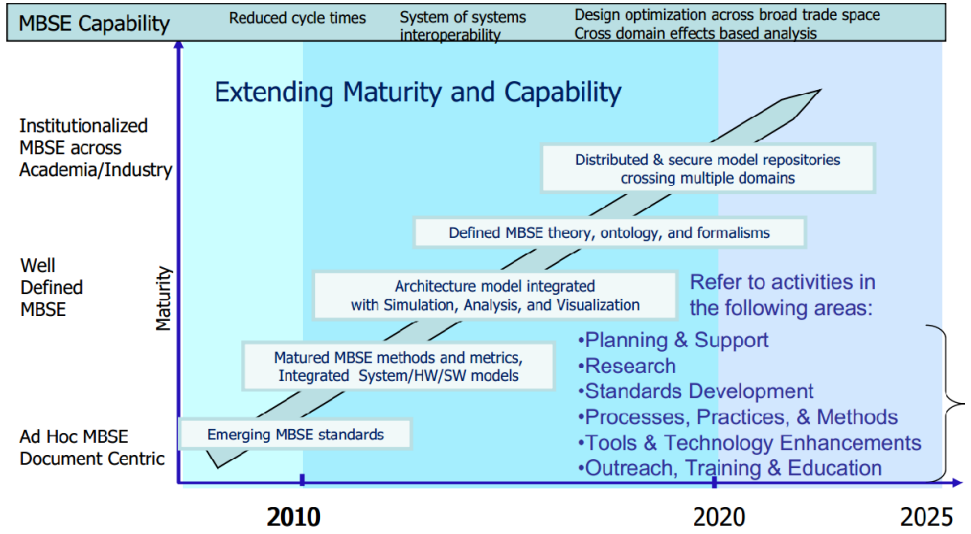
Assessed benefits of MBSE
Repulsion by engineers who feel comfortable with text-based procedures

NEED OF **IMPROVING MODEL-BASED SYSTEMS ENGINEERING (MBSE) MATURITY** THROUGH PRACTICAL APPLICATIONS, TO BETTER RIDE THE **NEW SPACE ECONOMY WAVE**



Low maturity of MBSE in the context of small satellites design

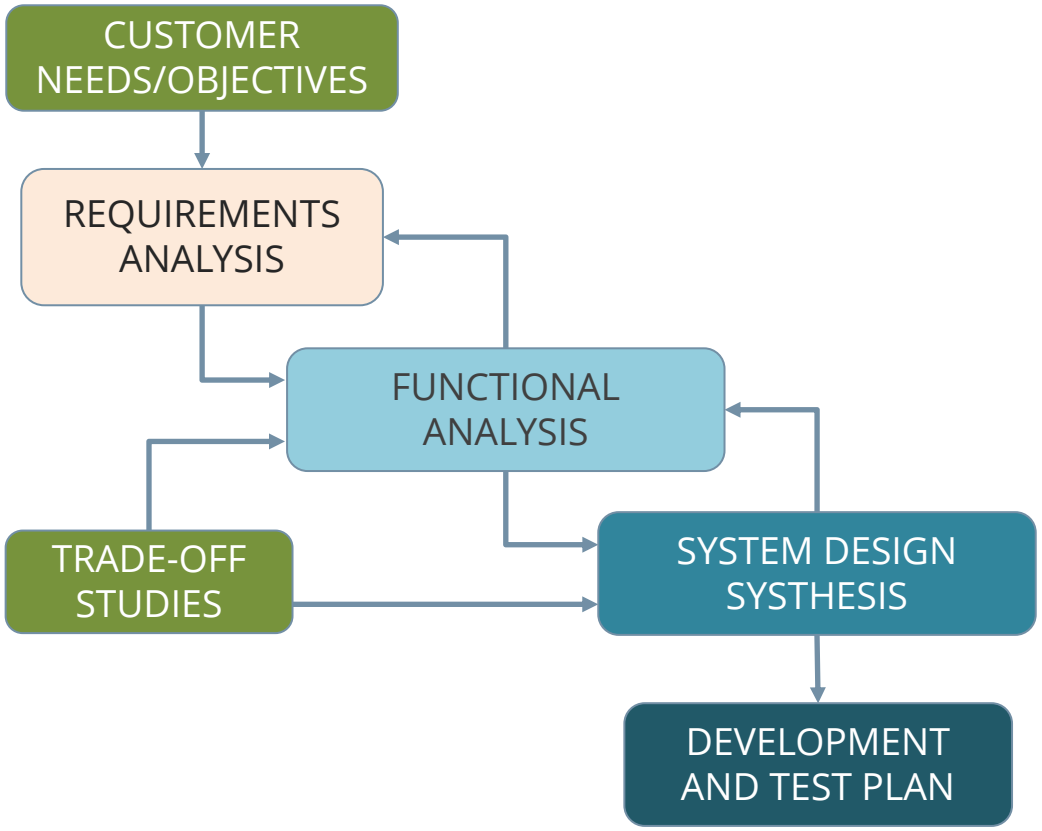
INCOSE MBSE Roadmap[1]



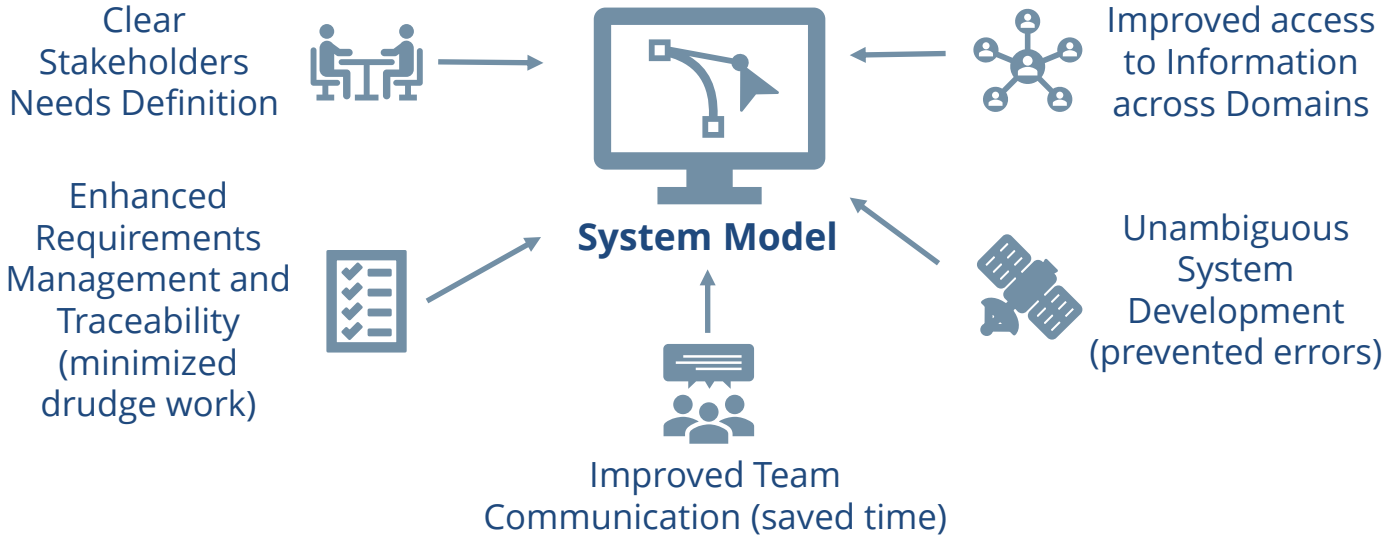
[1] International Council on Systems Engineering. *Systems Engineering Vision 2020*. INCOSE Technical Operations, Seattle, WA, 2007

Introduction: Space Systems Engineering practices

SPACE SYSTEMS ENGINEERING → MBSE



Increased awareness of MBSE potential in the last years



METHODOLOGY
Logical sequence of tasks defining "what" is to be done and "how".

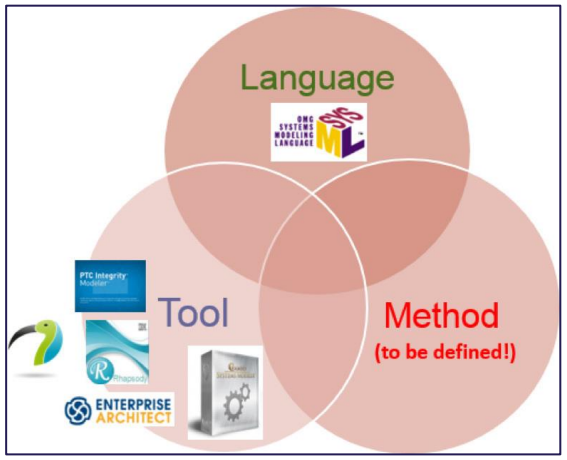
Object Oriented Systems Engineering (OOSEM) · OPM · ARCADIA

TOOL
Instrument that facilitates the accomplishment of tasks.

Cameo Systems Modeler · OPCAT · Capella Open Source MBSE Solution

LANGUAGE
Communication rules with precise syntax and semantics.

OMG SysML · OPL

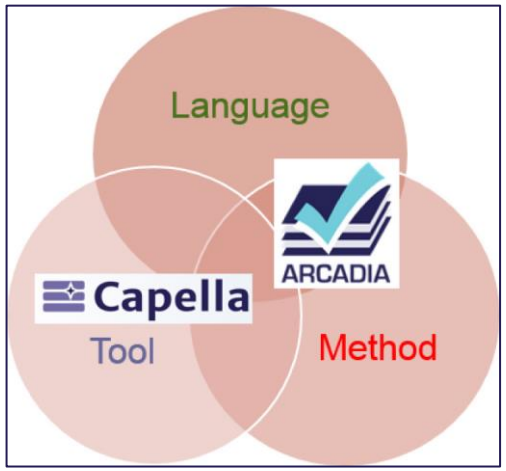


"Classic" MBSE with SysML

SysML is just a language → Needs a tool and a methodology that implement it

Object-oriented nature → Difficult to understand by non-software background engineers

No distinction between functions and components → Semantically confusing



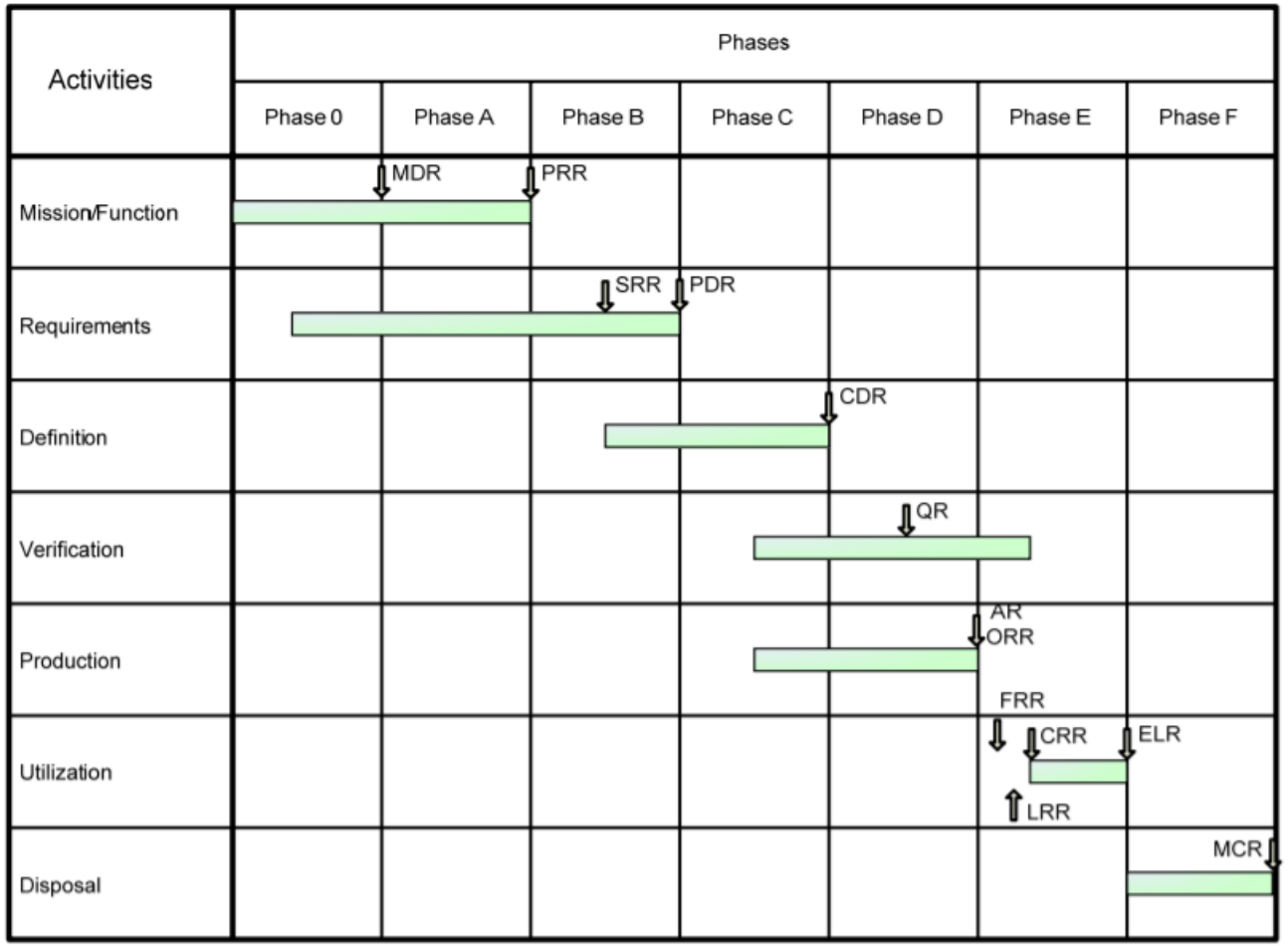
ARCADIA (ARChitecture Analysis & Design Integrated Approach)

- Embeds methodology and language
- DomainSpecific Modeling Language (DSML)
- Does not requires modelling experts
- Less steep learning curve

ARCADIA → Perfectly integrated by the tool → Capella Open Source MBSE Solution

- Open-source
- Intuitive
- Customizable

Space projects lifecycle: ESA standards ECSS-E-ST-10C Rev.1



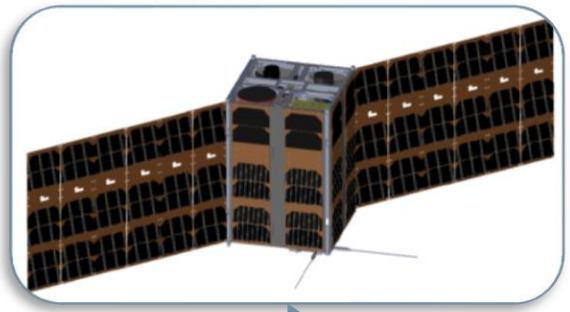
- Phase 0 - **Mission analysis/needs identification**: understand customer needs, propose mission/system concepts
- Phase A - **Feasibility**: propose system solutions to meet the customer expectations
- Phase B - **Preliminary Definition**: preliminary define the system solution
- Phase C - **Detailed Definition**: establish the system detailed definition
- Phase D - **Qualification and Production**: finalizes the development of the system, prepare for operations
- Phase E - **Utilization**: operate the system, support to anomaly investigations and resolutions
- Phase F - **Disposal**: safely dispose all products launched into space as well as ground segment

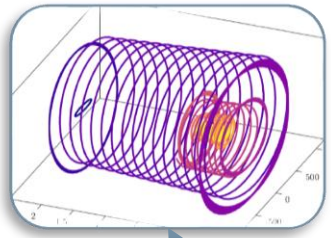
Getting started: CubeSat mission framework

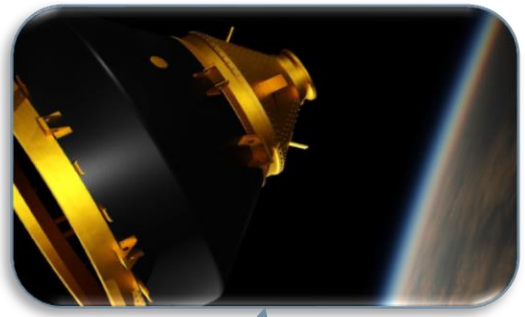
12U CubeSat

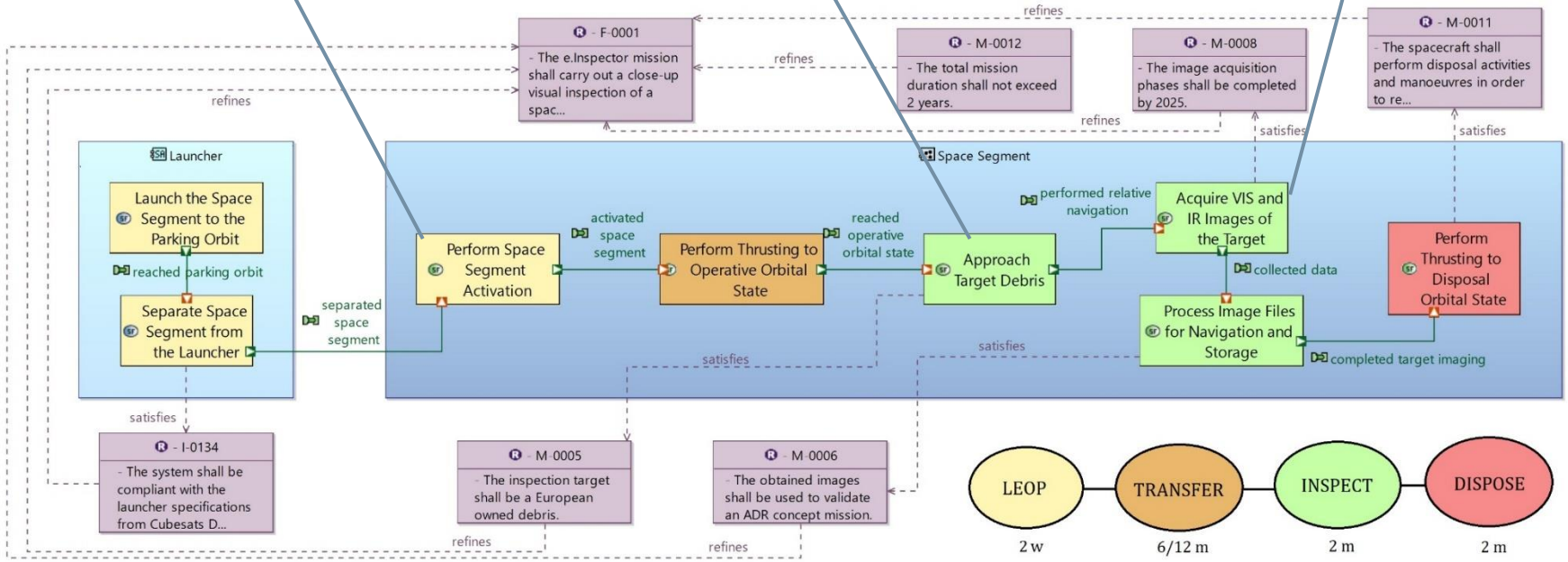
High Level Mission Goal:
Carry out a close-up visual inspection of a European space debris.

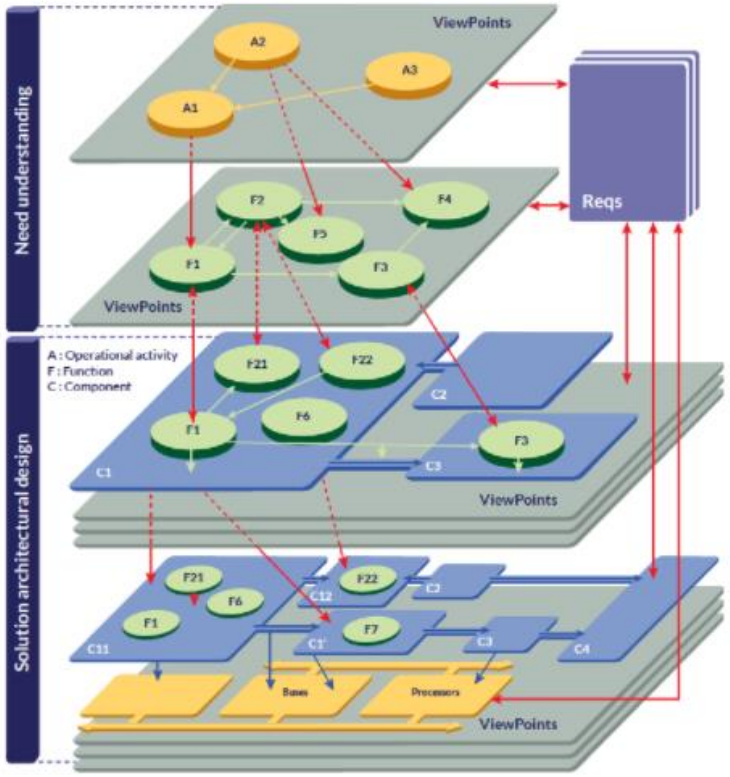
1) Understand the **debris status** at the time of flight
2) Validate **GNC sensors** to be used for a next capture of the debris
3) Reduce risks of future **Active Debris Removal (ADR)** missions











METHOD-EMBEDDED DEVELOPED TASKS

Operational Analysis
What the users of the system need to accomplish

- ✓ High Level Objectives
- ✓ Involved Stakeholders (Entities) and Responsibilities (Capabilities)

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

- ✓ System Capabilities
- ✓ External Functional Analysis

Logical Architecture
How the system will work to fulfill expectations

- ✓ Subsystems Modeling
- ✓ Internal Functional Analysis
- ✓ Functional Interfaces

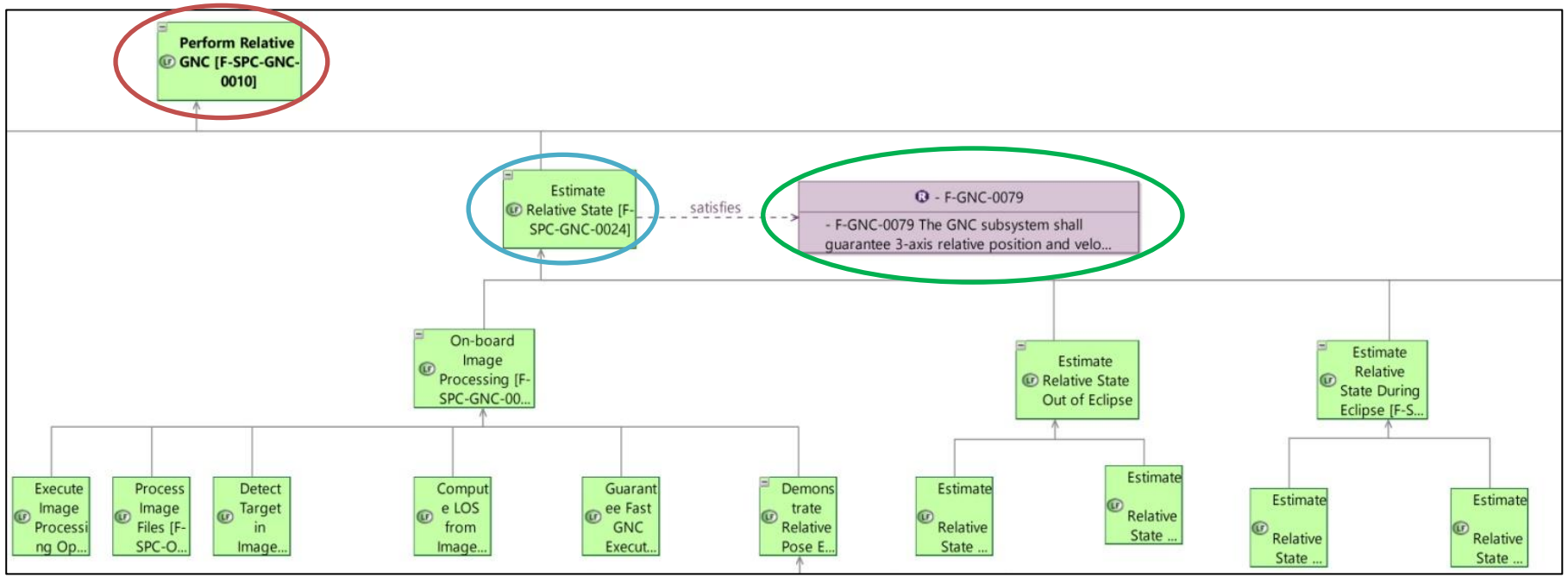
Physical Architecture
How the system will be developed and built

- ✓ Physical Components
- ✓ Physical Interfaces
- ✓ Mass and Cost Budgets
- ✓ Product Tree

- ✓ Requirements
- ✓ Phases and Modes
- ✓ Concept of Operations

+ dedicated AIV/AIT plan development diagrams

From Functional Analysis to Requirements



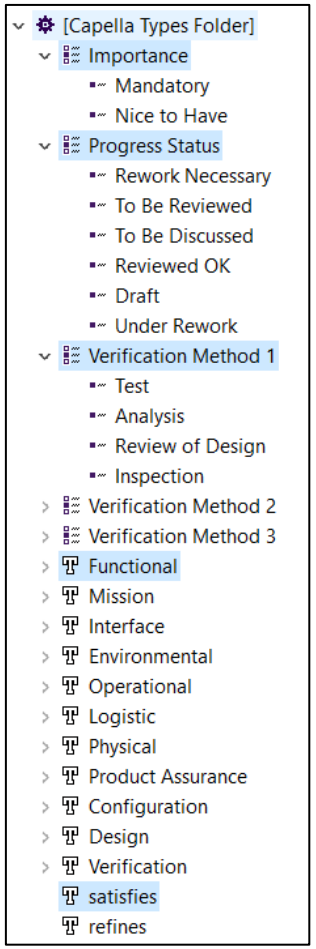
Perform Relative GNC
- Estimate Relative State
- Execute Relative Maneuvers
- Perform Attitude Target Tracking

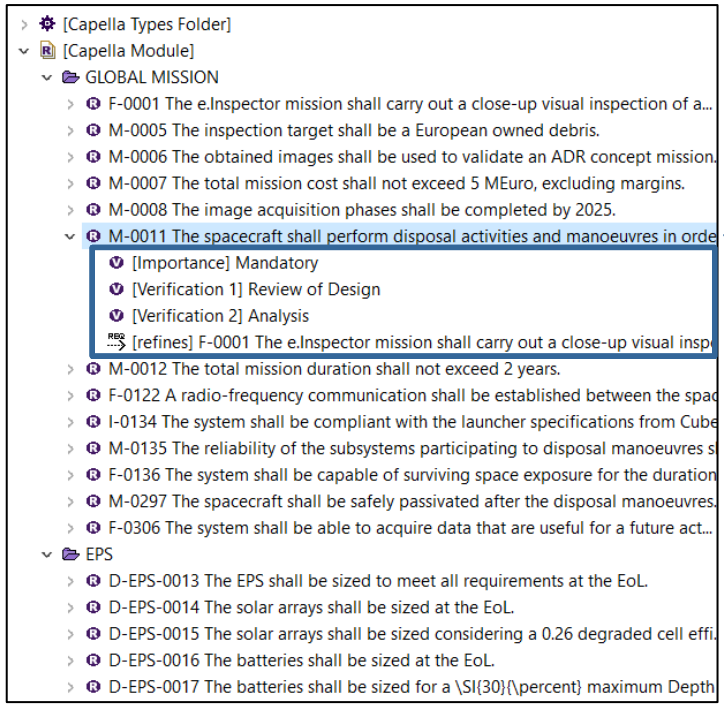
Estimate Relative State
- On Board Image Processing
- Estimate Relative State Out of Eclipse
- Estimate Relative State During Eclipse
- Sensor Fusion for Enhanced Estimation

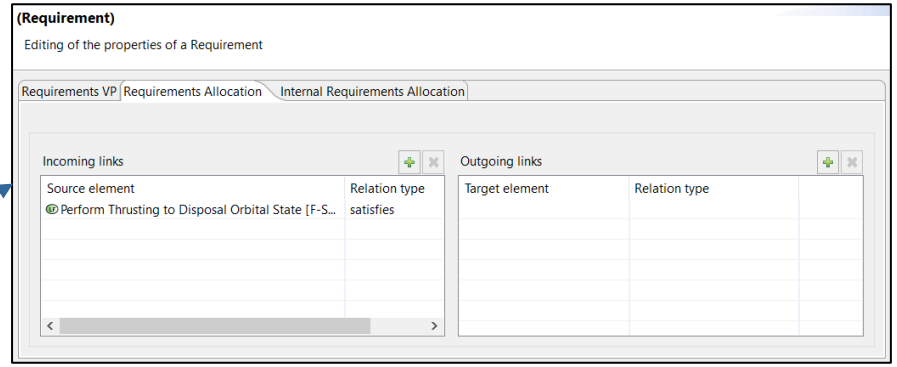
F-GNC-0079:
The GNC subsystem shall guarantee 3-axis relative position and velocity states estimation

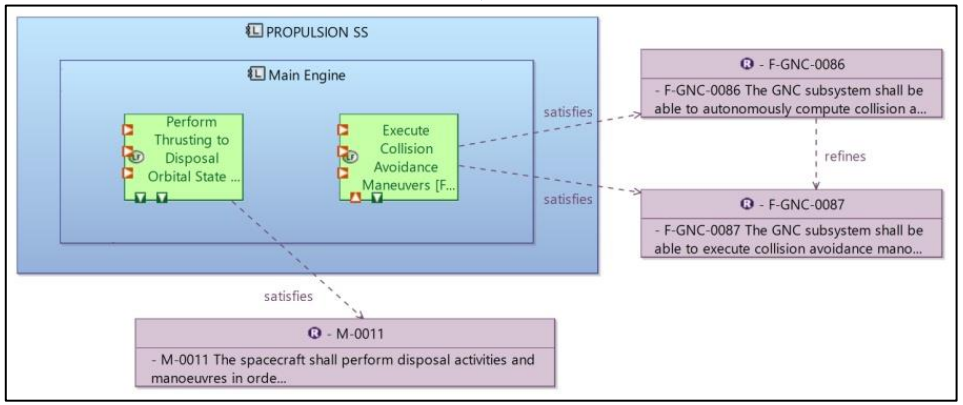
Requirements traceability

Traceability between requirements and model elements is managed through the **Capella Requirements Viewpoint** which provides a graphical output too



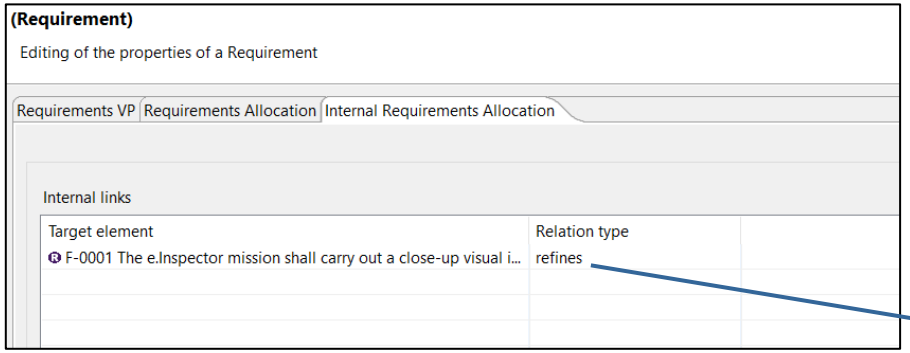




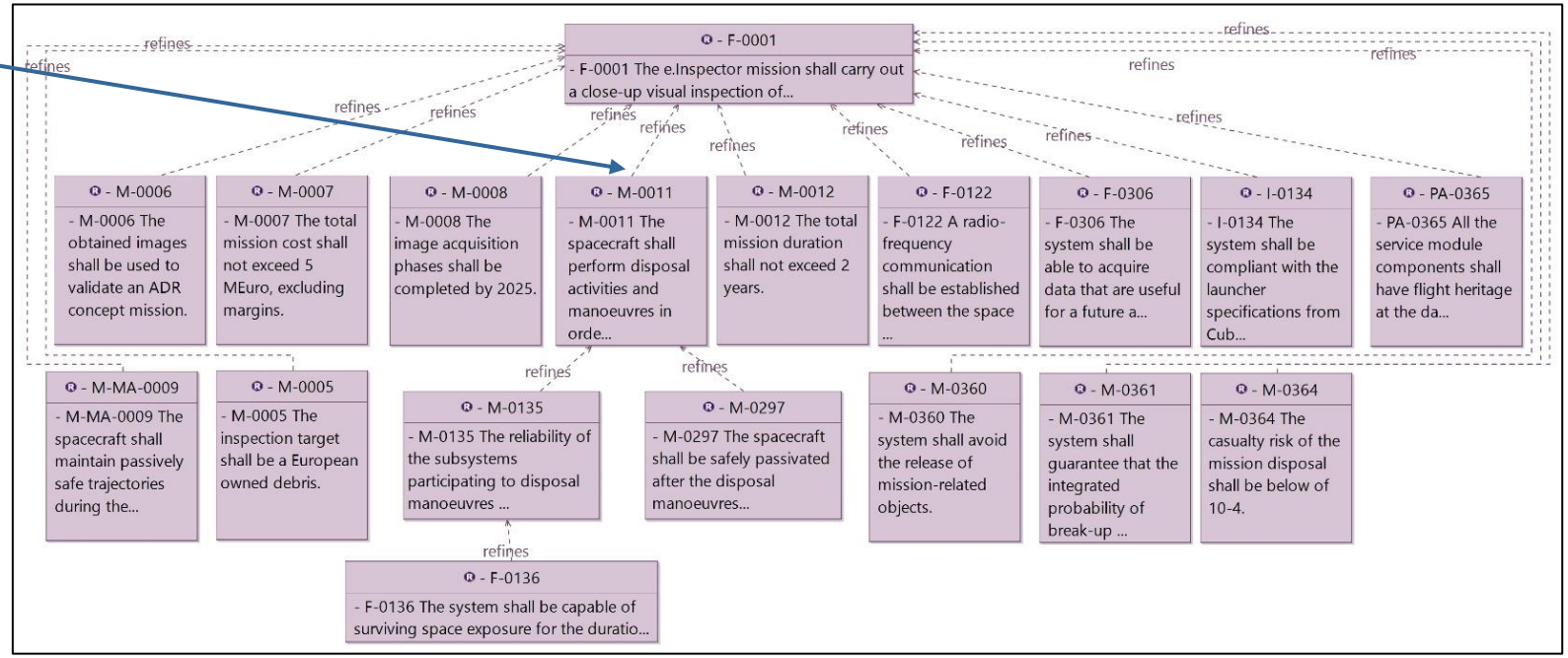


Internal Requirements allocation

Requirements trees are generated once internal relations are defined



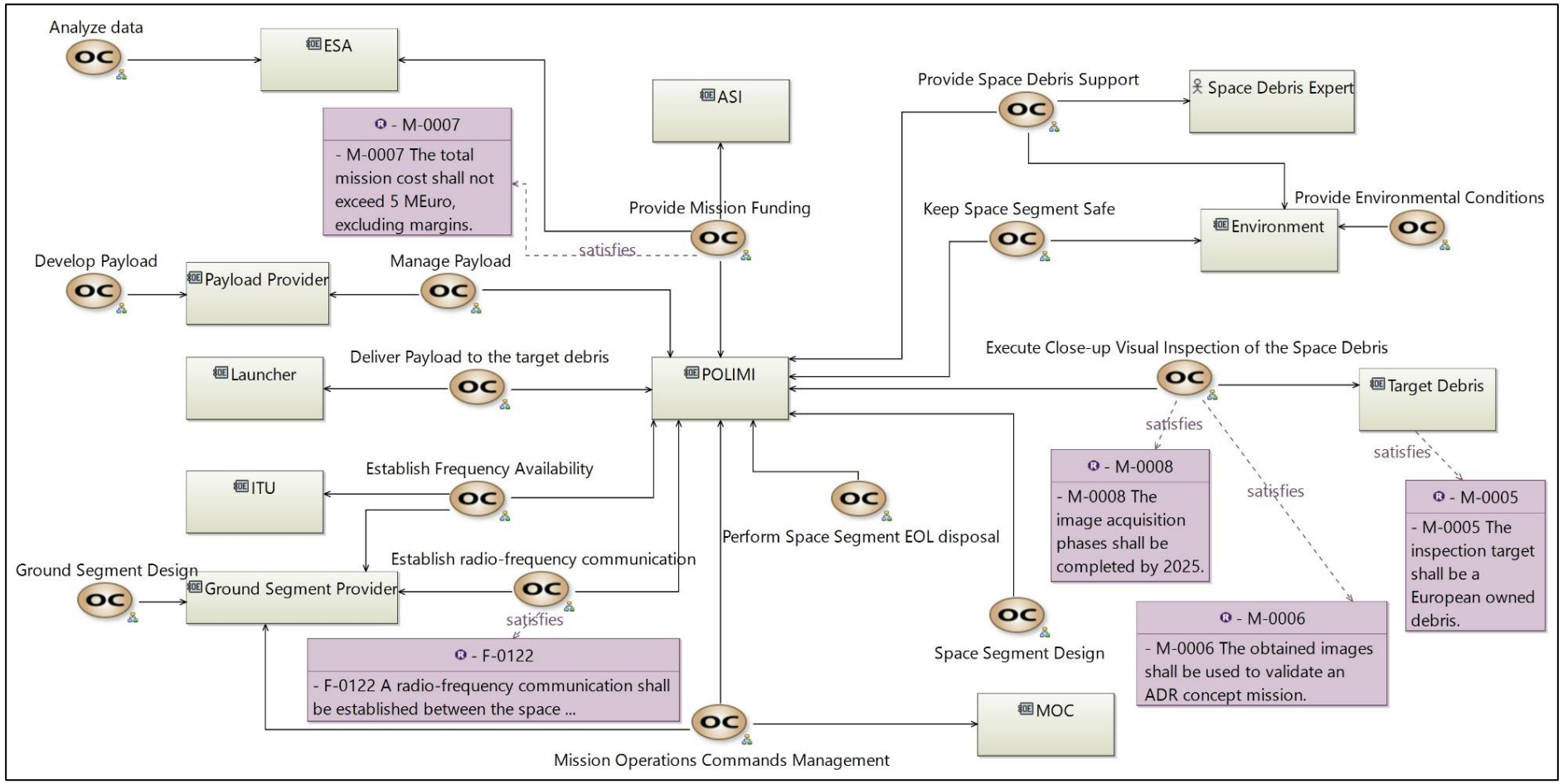
***Example*: Requirement M-0011 Internal Link**

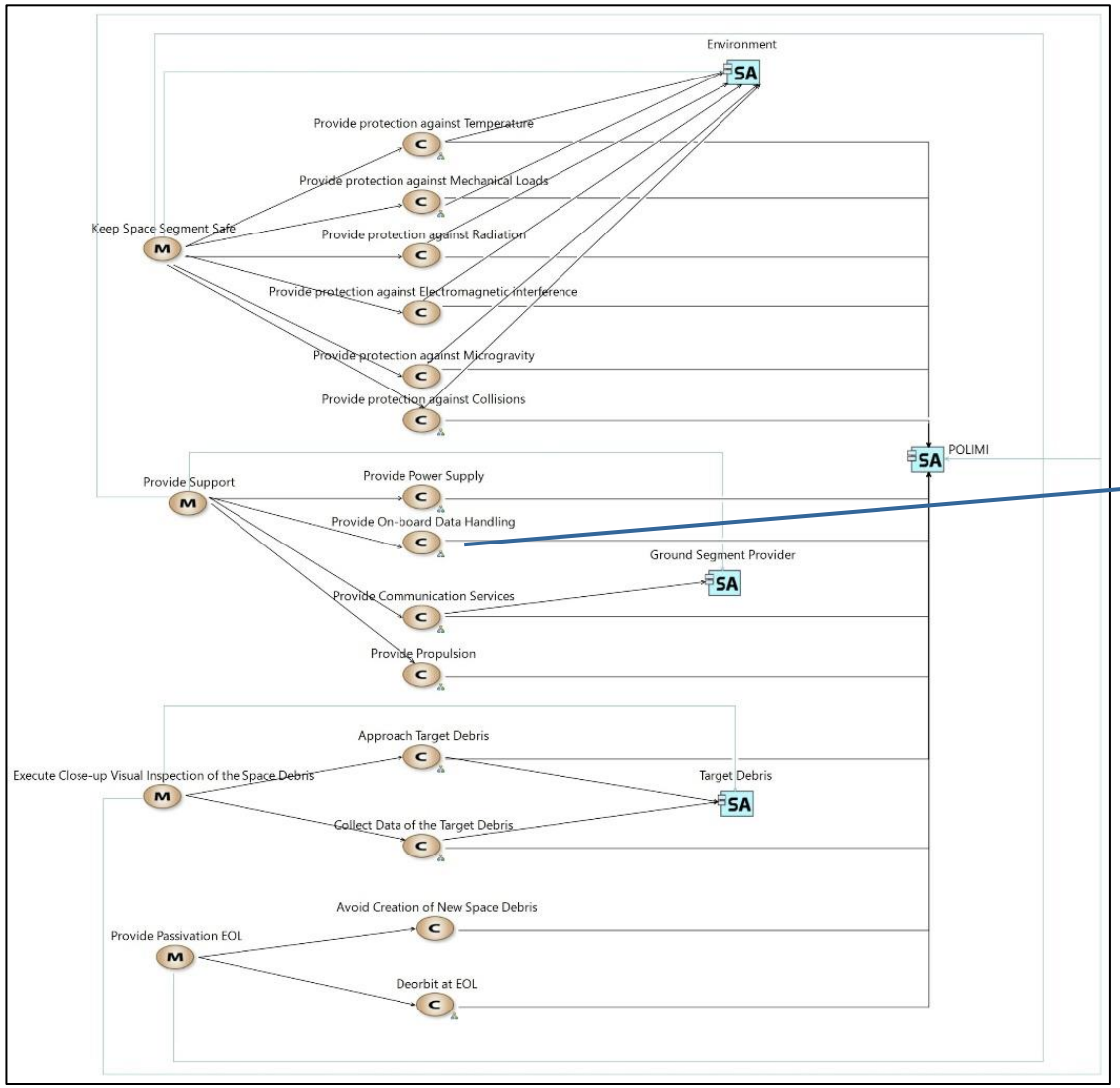


Customization of Operational Architecture Blank diagrams

Outputs: High Level Objectives, Involved Stakeholders and Responsibilities

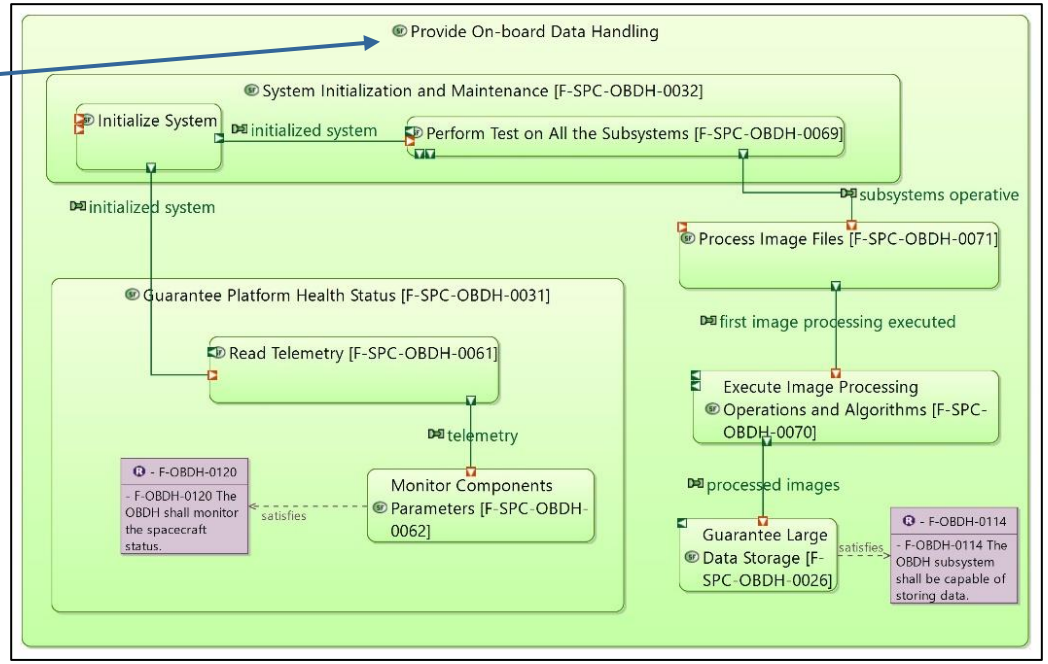
***Example*: Operational Architecture Blank diagram @Operational Analysis**





Outputs: System Capabilities, External Functional Analysis

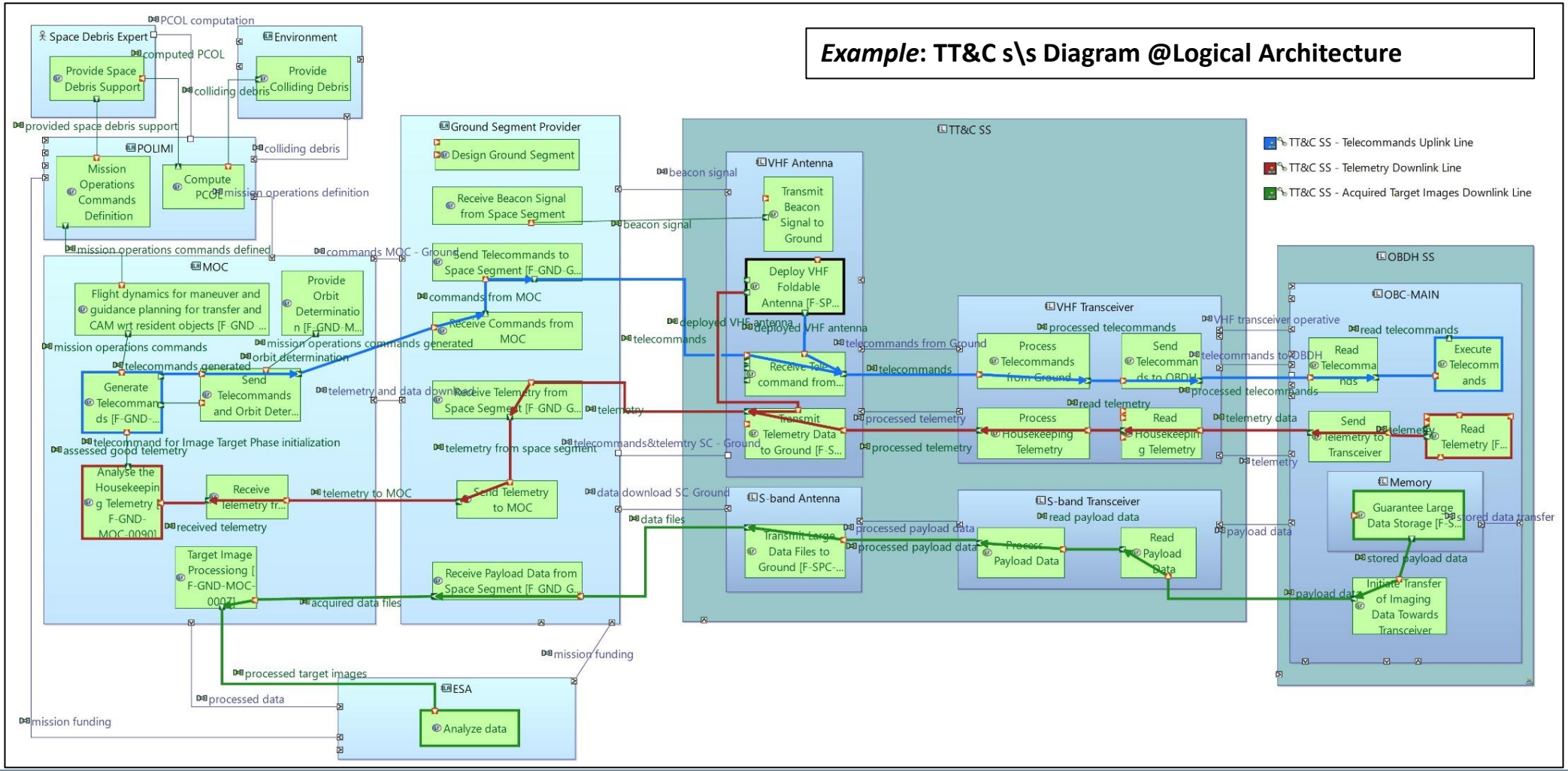
Example:
L: Mission Capabilities Blank diagram
R: OBDH Data Flow Diagram @System Analysis



Logical Architecture

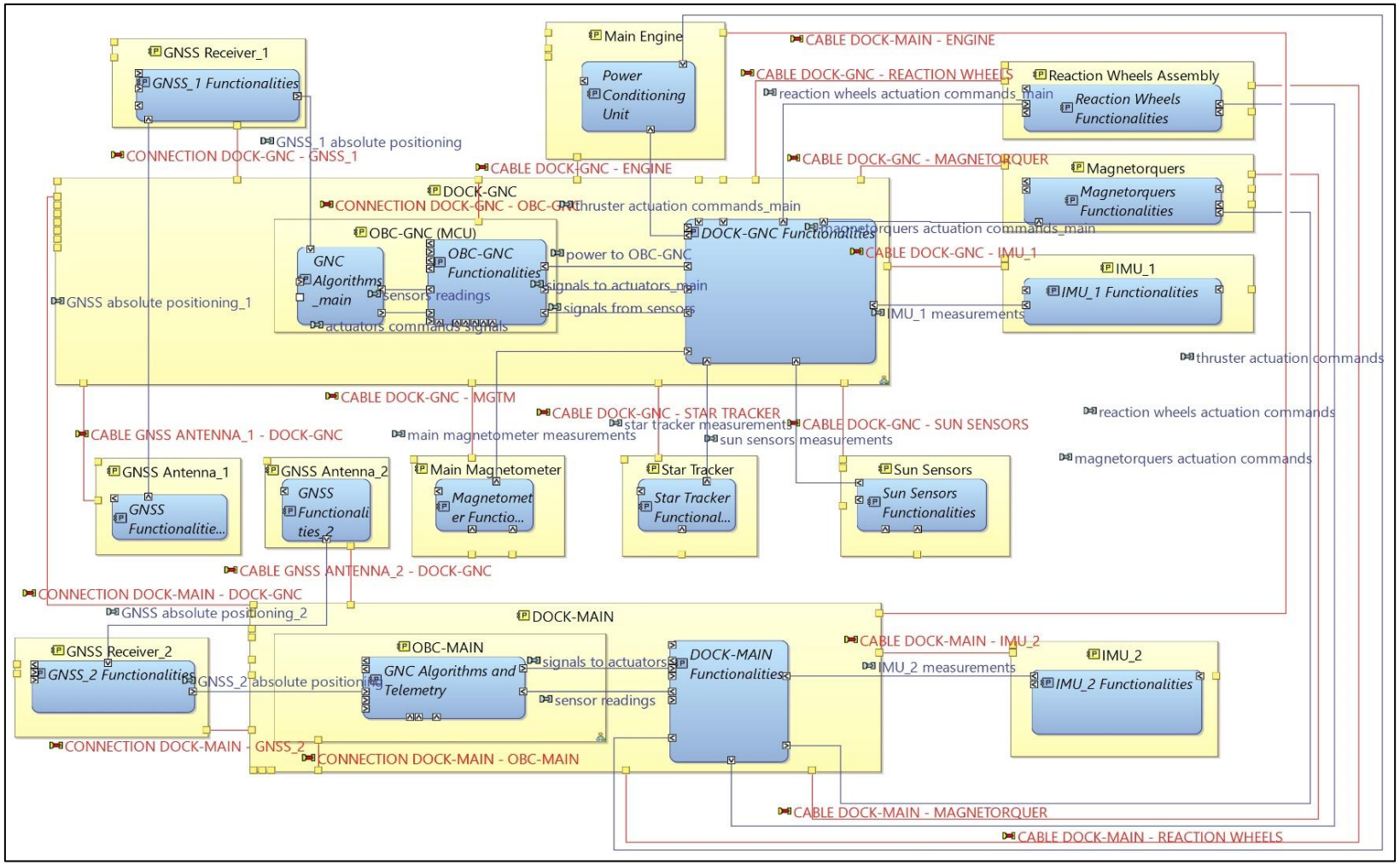
Outputs: Subsystems Modeling, Internal Functional Analysis, Functional Interfaces

***Example*: TT&C s\s Diagram @Logical Architecture**



Physical Architecture

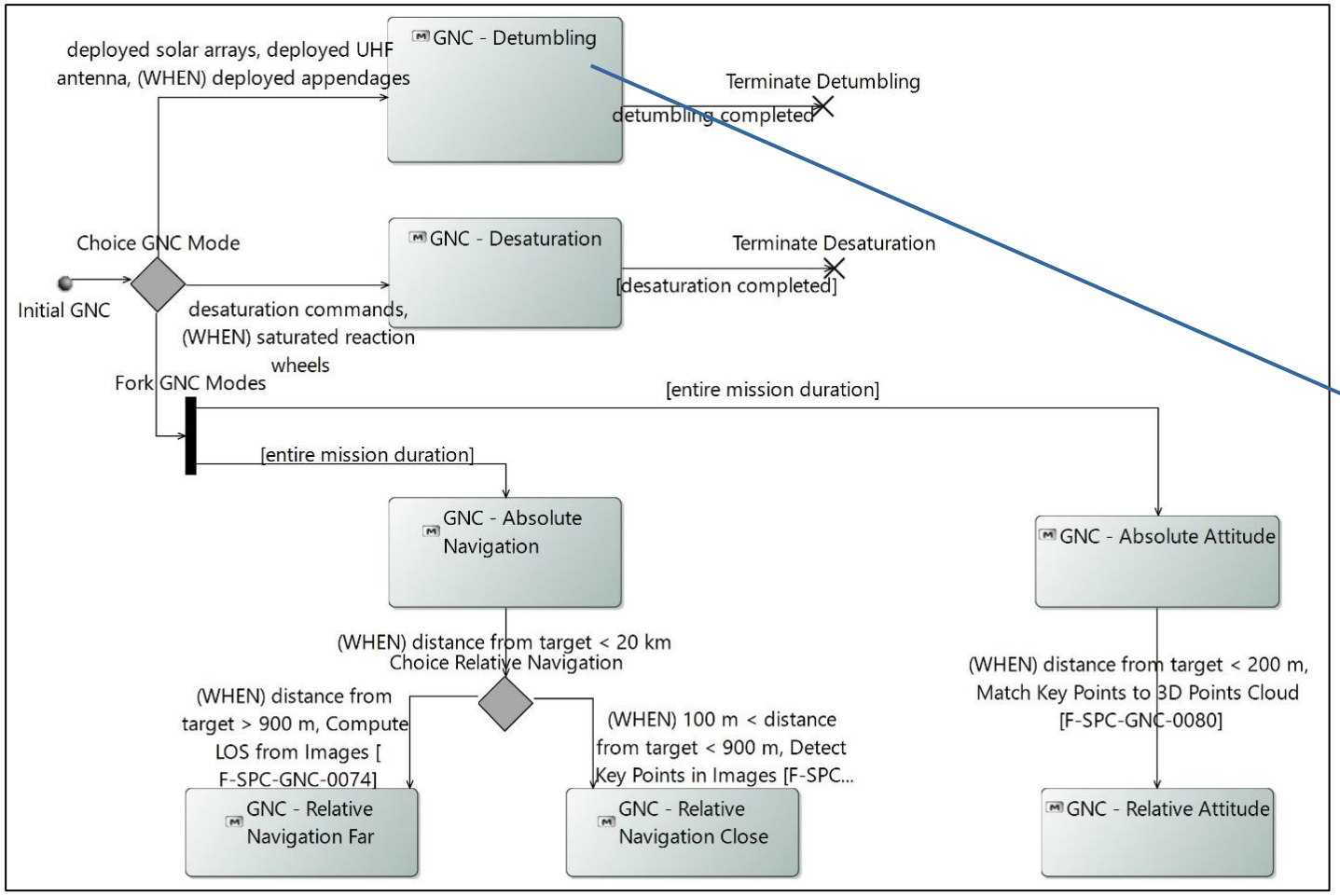
Outputs: Physical Components, Physical Interfaces, Mass and Cost Budgets, Product Tree



***Example*: OBDH - GNC interfaces**

Modes modeling

Modes are characterized by several **functions** already modeled in the previous analysis.



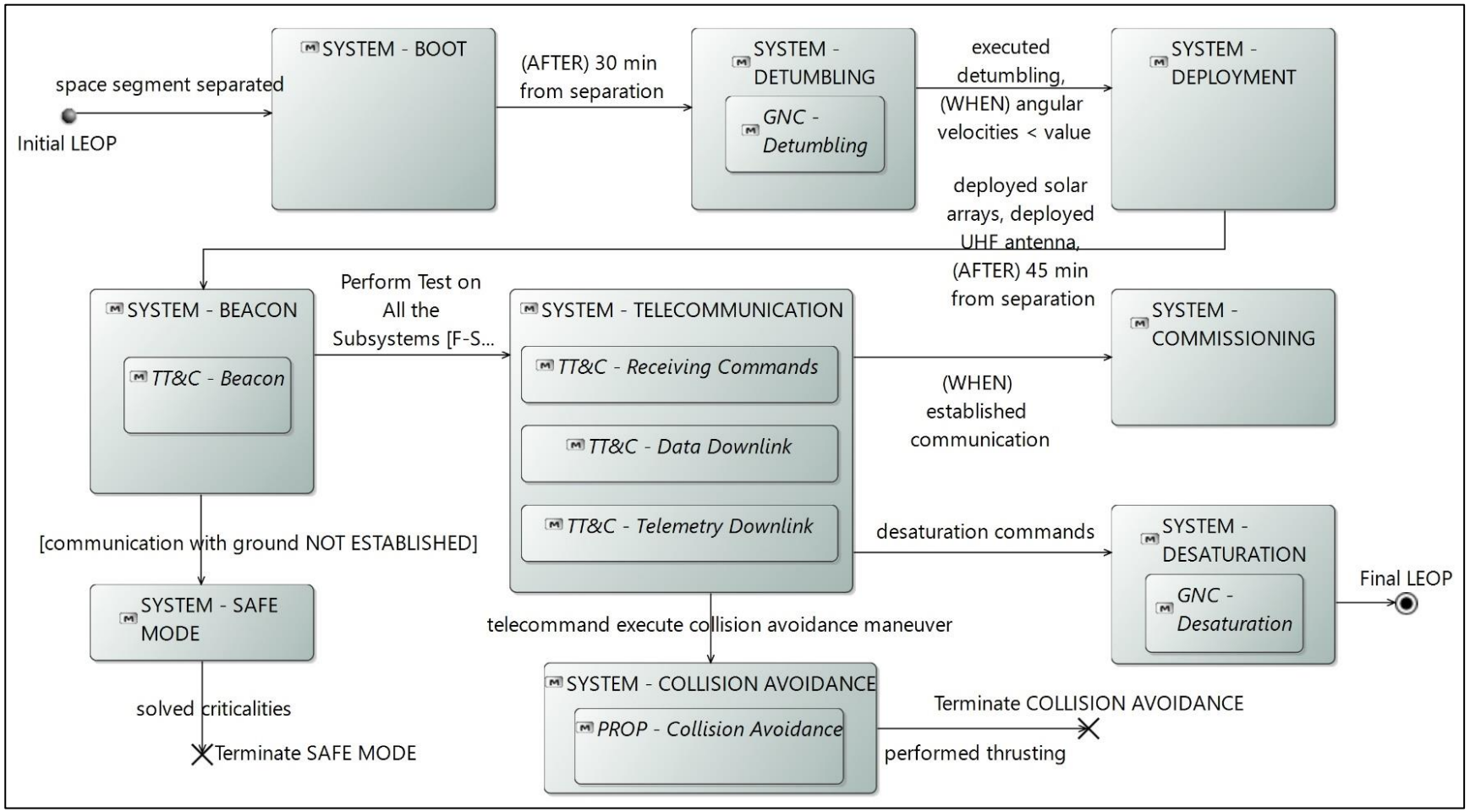
***Example*: GNC Subsystem Modes**

GNC - Detumbling

do / Measure Angular Velocity Related Quantity IMU_1 [F-SPC-GNC-0067]
do / Measure Angular Velocity Related Quantity IMU_2 [F-SPC-GNC-0067]_2
do / Gather Angular Velocities Readings
do / Dissipate Angular Velocity [F-SPC-GNC-0066]
do / Execute 3-Axis Attitude Control [F-SPC-GNC-0048]
do / Monitor GNC Components Parameters [F-SPC-OBDH-0036]
do / Magnetometers Data Acquisition
do / select magnetorquers functionality
do / Route Sensor Readings to OBC-GNC
do / Route Sensor Readings to OBC-MAIN
do / Route Commands from OBC-GNC to Actuators
do / Route Commands from OBC-Main to Actuators
do / Split Actuators Commands
do / Activate Magnetometer
do / Activate Magnetorquers
do / Provide Magnetic Field Components_IMU_MGTM
do / Provide Magnetic Field Components_IMU_MGTM_2
do / Provide Magnetic Field Components_MGTM
do / Generate Magnetorquers Commands

[region]

Modes modeling

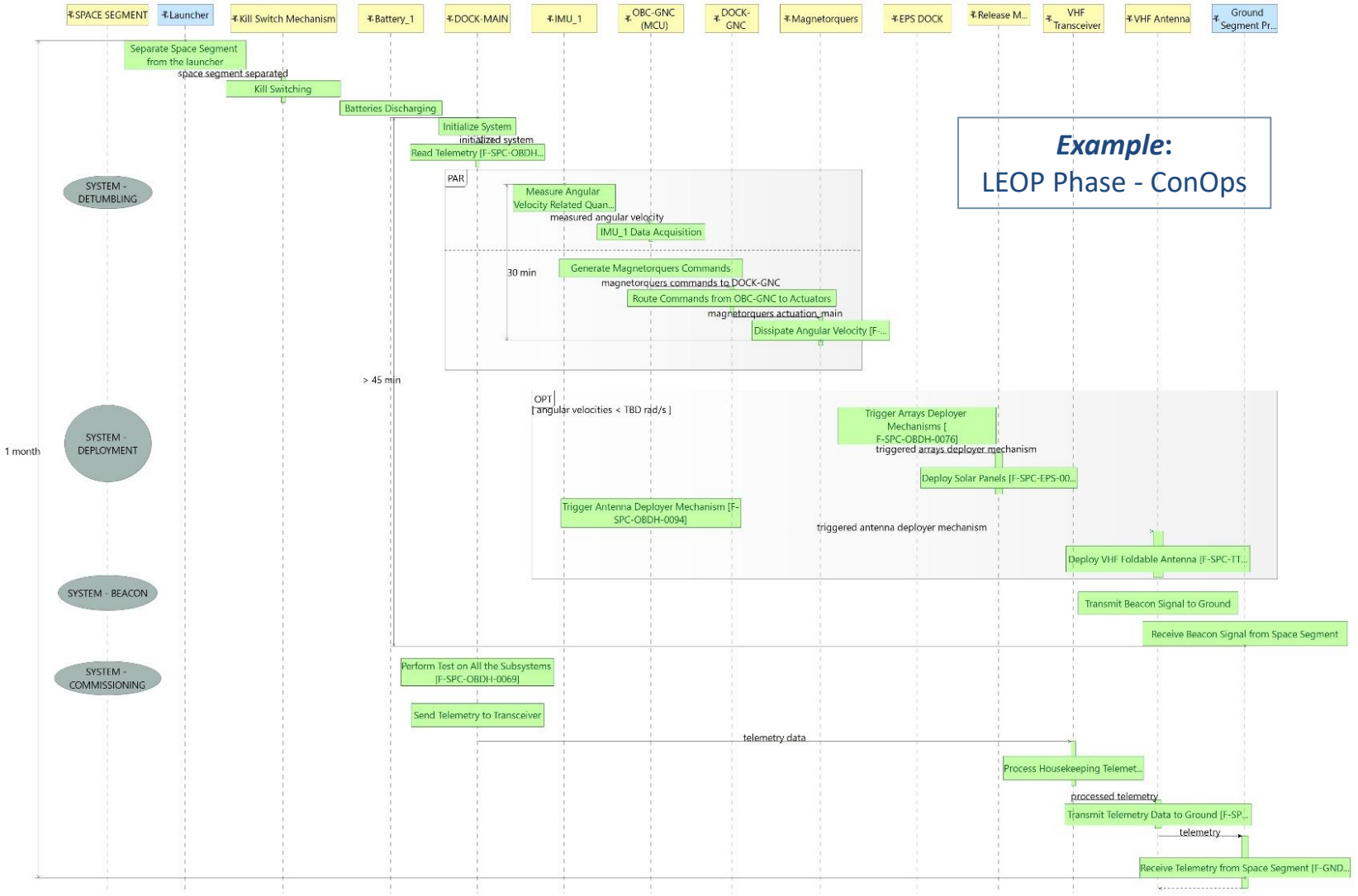


***Example*:**
LEOP Phase – System Modes

Subsystems Modes are exploited to easily define System ones

Saved time in developing **State Machine Diagrams**

Concept of Operations



Example:
LEOP Phase - ConOps

Satellites are operated relying on detailed **Concept of Operations** which are part of the system design.

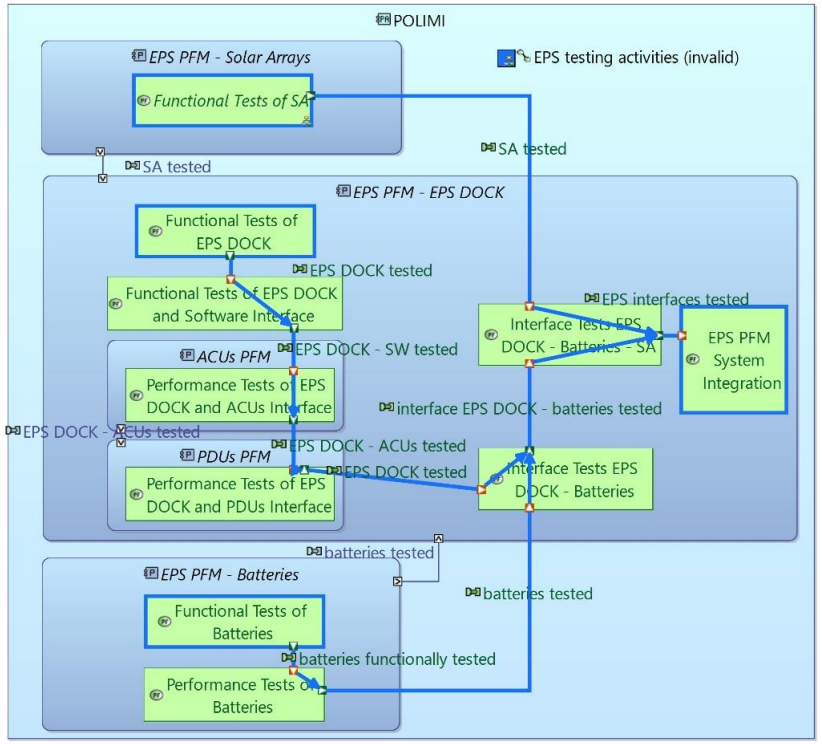
Scenario Diagrams are exploited for this purpose.

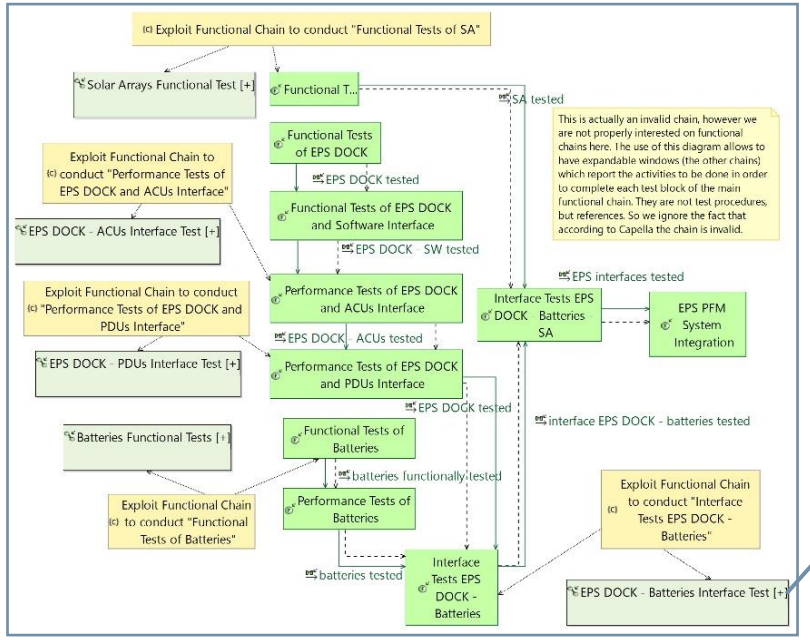
- Describe logic structures in a very **compact and concise manner**
- Rely on **already modeled elements**
- Force to think about **system utilization solutions**

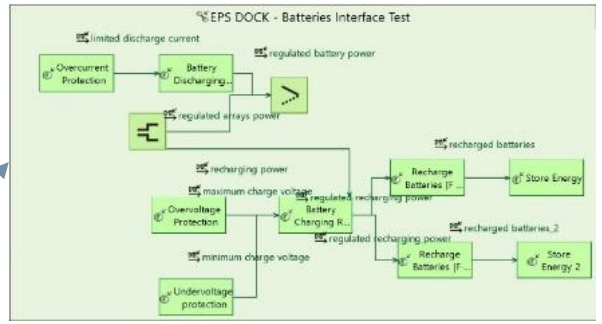
Assembly, Integration, Verification and Testing (AIV/AIT) plan

AIV/AIT plan definition is at a different level of modeling with respect to the system design. However, it is **inherently connected** with system functioning and architecture.

***Example*:** EPS testing activities





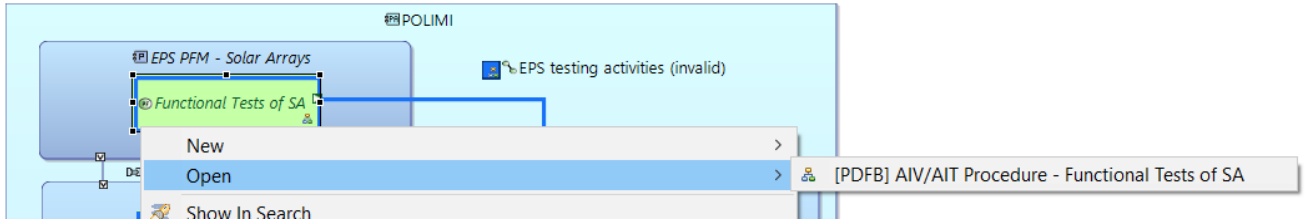


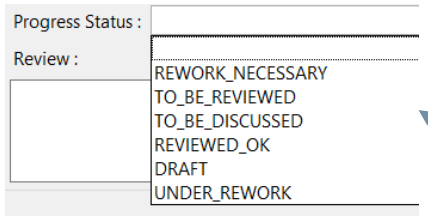
- Ad-hoc functionalities define **AIV/AIT activities**
- Bridge with model elements which provide **guidance** to the plan development

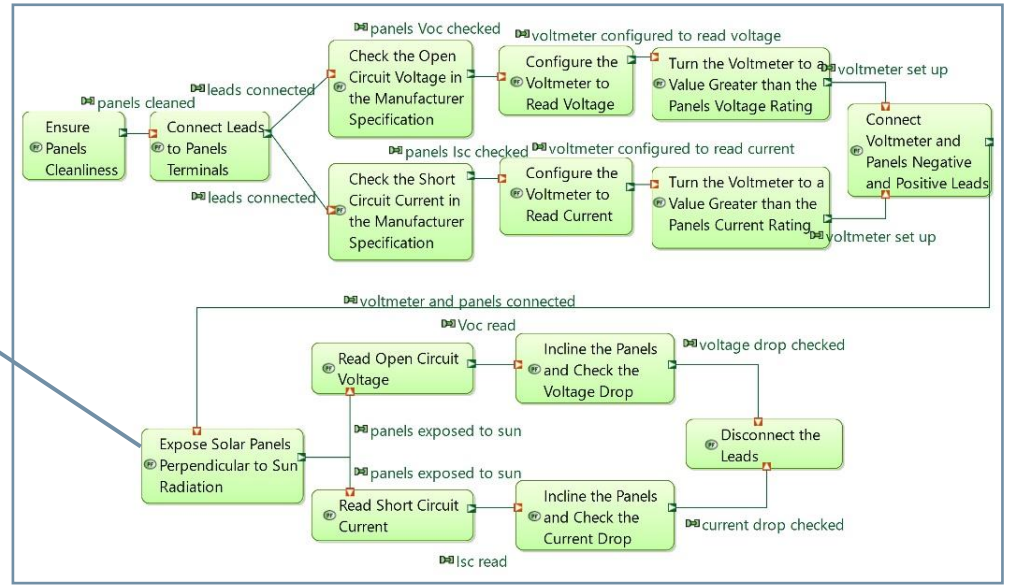
AIV/AIT procedures

Each activity is further described by a set of procedures which can be assigned to team members, monitoring the progress status.

***Example*:** Functional tests of Solar Arrays - Procedures







- Logical and temporal sequence of procedures can be used in the **operational context**
- Design changes are easily traced and related AIV/AIT **activities and procedures promptly updated**
- Improved **standardization** of AIV/AIT concepts
- Some ARCADIA rules have been **violated**, need of a formalization of the approach and its integration with Capella

Conclusions and Steps Forward

IS MBSE WORTH IT FOR SMALL SATELLITES?

MODELS EASE SYS. ENG. PRACTICES W.R.T. TEXT-BASED APPROACHES

- Solid, Effective and Efficient **approach** to manage satellites complexity
- System lifecycle further enhanced by **dedicated AIV/AIT plan modeling**

FUTURE STEPS

- Risk analysis
- Class diagrams
- Formalization of AIV/AIT syntax and semantics
- Overall model refinement toward Phase B design

ARCADIA

Capella
Open Source MBSE Solution

OBEO

THALES

CapellaDays ONLINE 2021

POLITECNICO MILANO 1863

DIPARTIMENTO DI SCIENZE E TECNOLOGIE AEROSPAZIALI

Advanced Space Technologies 4 Robotics & Astrodynamics

Thank you for the attention!

Questions?

linkedin.com/in/paolo-minacapilli/

paolo.minacapilli@yahoo.com

November 17, 2021