Astro MBSE: overview on requirement management approaches for astronomical instrumentation

Marco Riva, Andrea Balestra, Marco Xompero, Alessio Zanutta, Matteo Genoni, et al.

Marco Riva, Andrea Balestra, Marco Xompero, Alessio Zanutta, Matteo Genoni, Marcello Agostino Scalera, Runa Antonio Briguglio, Giacomo Dinuzzi, Davide Fierro, "Astro MBSE: overview on requirement management approaches for astronomical instrumentation," Proc. SPIE 12187, Modeling, Systems Engineering, and Project Management for Astronomy X, 121871Q (25 August 2022); doi: 10.1117/12.2630404

Event: SPIE Astronomical Telescopes + Instrumentation, 2022, Montréal, Québec, Canada

ASTRO MBSE: OVERVIEW ON REQUIREMENT MANAGEMENT APPROACHES FOR ASTRONOMICAL INSTRUMENTATION

Marco Riva[a], Andrea Balestra[b], Marco Xompero[c], Alessio Zanutta[a], Matteo Genoni[a], Marcello Antonio Scalera[a], Runa Briguglio[c], Giacomo Dinuzzi[d], Davide Fierro[e].

[a] Osservatorio Astronomico di Brera – INAF; [b] Osservatorio Astronomico di Padova – INAF, [c] Osservatorio Astronomico di Arcetri – INAF, [d] Istituto di Astrofisica e Planetologia Spaziali – INAF, [e]INAF - Istituto Nazionale di Astrofisica

ABSTRACT

Systems Engineering requires the involvement of different engineering disciplines: Software, Electronics, Mechanics (often nowadays together as Mechatronics), Optics etc. Astronomical Instrumentation is no exception to this. A critical point is the handling of the requirements, their tracing, flow down and the interaction with stakeholders (flow up) and subsystems (flow down) in order to have traceable and methodical evolution and management.

In the Italian Astronomical Community, we are developing methodologies and tools to share the expertise in this field among the different projects. In this paper we will focus on the requirement management approach among different projects (ground and space based, …). The target and synthesis of tis work will be a support framework for the Requirement management of the Italian Astronomical Community (INAF) projects.

Keywords: system engineering, requirement management, templates for astronomy, ASTROMBSE,

1. INTRODUCTION

One of the target of ASTROMBSE project [2] is to provide to the astronomical community proper tools for Engineering activity and a key element is the requirement management. The aim of this paper is to describe within the *AstroMBSE* initiative the proposed approach to model the requirement management.

Requirements Management is the process of gathering, analyzing, verifying, and validating the needs and requirements for the given product or system being developed. Successful requirements management ensures that completed deliverables meet the expectations of the stakeholders.

In order to tailor the needs of the astronomical community, it has been developed a customized stereotype in SysML, starting from the classical extended requirement. We will describe in detail here the approach and its implementation using Maory (aka MORFEO) as case study. It is then optimized for ground based projects.

2. ASTRONOMY ORIENTED REQUIREMENT STEREOTYPE

A dedicated astronomy oriented stereotype has been created in order to match all the needs of astronomical instrumentation projects.

Several properties have been added (or modified):

- ***Verification method*** (including all the type specified by different customers)
- ***Project phases,*** linked to verification method, to enable different verification for the different project milestones
- ***Compliance status*** (to enable compliance and verification matrixes automatic generation)
- ***Flow down source*** to enable some generic queries on the data packs that collect the parent reqs,
- ***Priority***, to enable prioritization of user needs (very useful during phase A and definition of science needs
- ***Requirement Kind*** to enable the characterization of requirements or infos

Modeling, Systems Engineering, and Project Management for Astronomy X, edited by
George Z. Angeli, Philippe Dierickx, Proc. of SPIE Vol. 12187, 121871Q
© 2022 SPIE · 0277-786X · doi: 10.1117/12.2630404

- ***Flowdownkind*** to immediately identify the kind of flow down (with or without analysis)
- ***Assigned person*** to identify the responsible that has to answer to the requirement
- ***Redmine (or JIIRA)*** to have a direct link to eventual ticket opened that contains discussion on the related requirement
- ***Derivation*** to group in one property either derivation from requirement or from use cases
- ***Requirement*** status to monitor if it is just a proposal of requirement or has been approved
- ***Documentation*** to be filled with the section of the document that describe how the requirement is fulfilled (and complete the verification matrix)

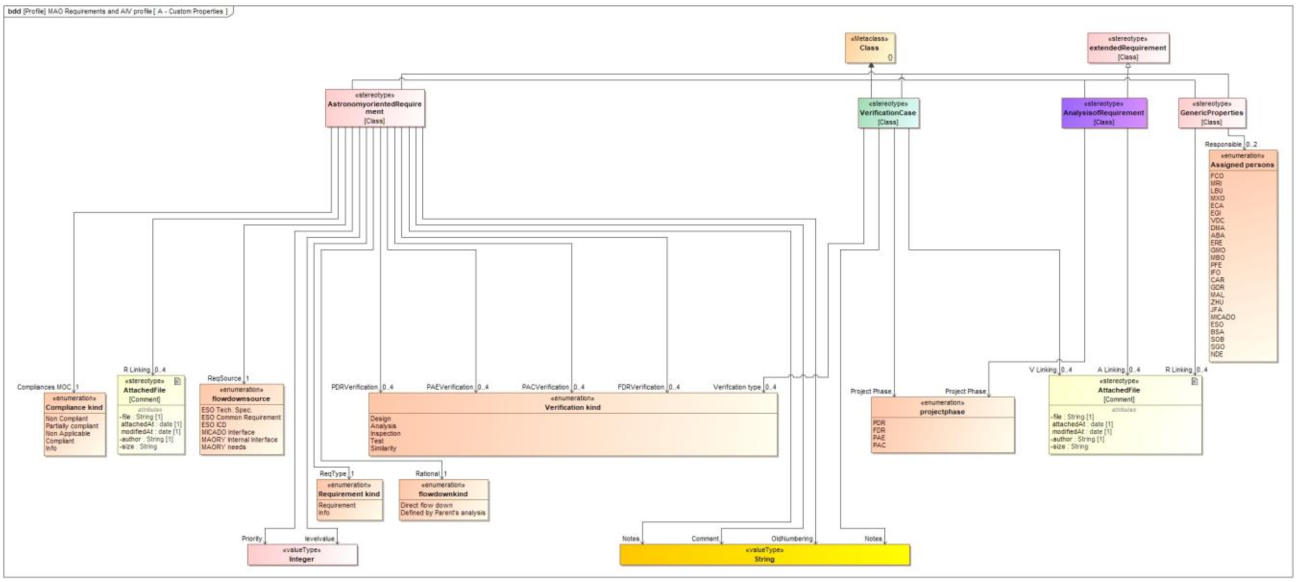

Figure 1 View diagram of the custom properties of the requirements

3. REQUIREMENT MANAGEMENT

Different approaches can be adopted for requirement management:

- Dedicated software is used to manage the requirement (i.e. DOORS) and it is coupled with a sysml model
- The Requirements are directly modelled and managed into the sysml model

The first method allows a more robust control of the requirement, allowing easier modification and updating of the flowdown. It helps also the user with a simple interface that can be used also to communicate with different partners. The link to the model is quite complex and may risk to fail if not accurately handled.

The second method has been adopted for Astrombse. It leads to a more complicated modelling and management of the requirement, in particular regarding the modification and update of children requirement, but it make much more easy the link of the interface to the rest of the model, like interfaces, use cases and so on…

By implementing this method, we are able to:

- Keep Verification Cross-Reference Matrices, and Verification Activities under control.
- maintain a list of Applicable and Reference Documents including standards and linking these with requirements. This allows visibility into where and how modifications to documents will affect the requirements.
- Adoption of a graphic derivation tree representation gives excellent visibility to the requirements and links between requirements across the MAORY project.

4. SYSTEM LEVEL REQUIREMENTS

The system level requirements are captured from the stakeholders needs. They typically consist in

- a set of technical specification that are flow down from the scientific requirements.
- A set of environmental requirement that define the context (observatory or space mission)
- A set of interface requirement that define the relationship of the system with the other parties (telescope, other instruments, …)

It is important to properly analyse the requirement at system level, as there could be the need of splitting some of them or to directly asses their compliances without further flow down.

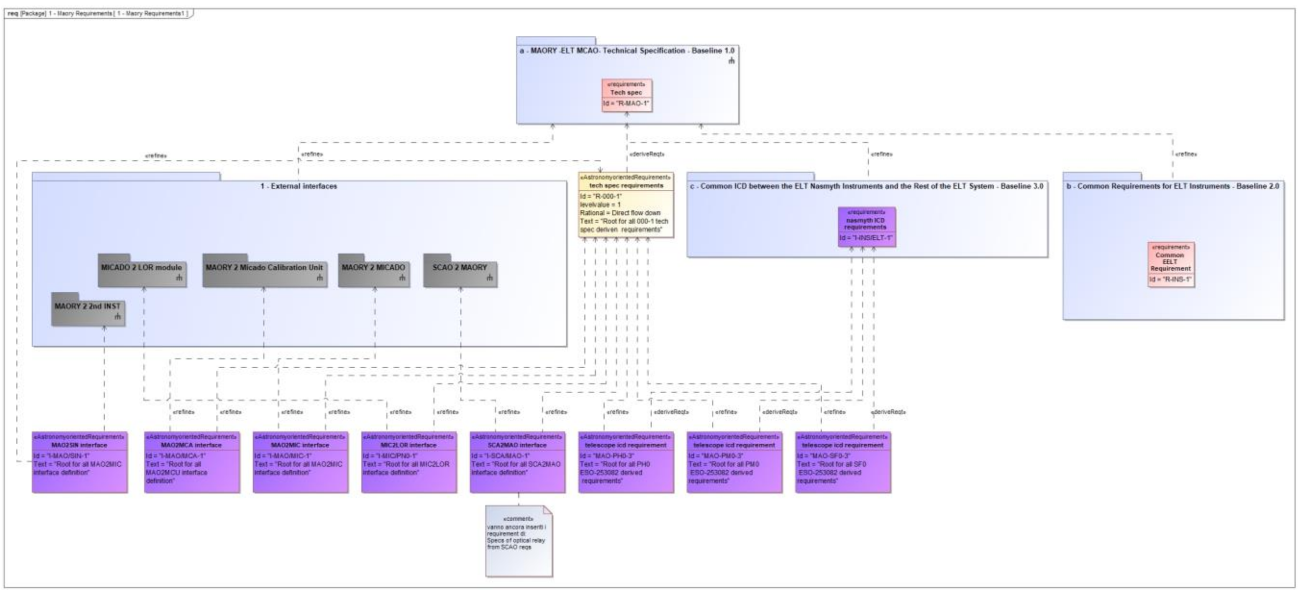

Figure 2 System level requirement

5. FLOWDOWN TO SUBSYSTEM

From the system level requirements then subsystem level ones are derived.

All requirements are analyzed in term of applicability to the subsystems and then have been flown down to the subsystems. In case the requirement was not directly applicable to the subsystem proper analysis have been conducted to obtain the right subsystem requirement, or the related budget, or the engaged interface.

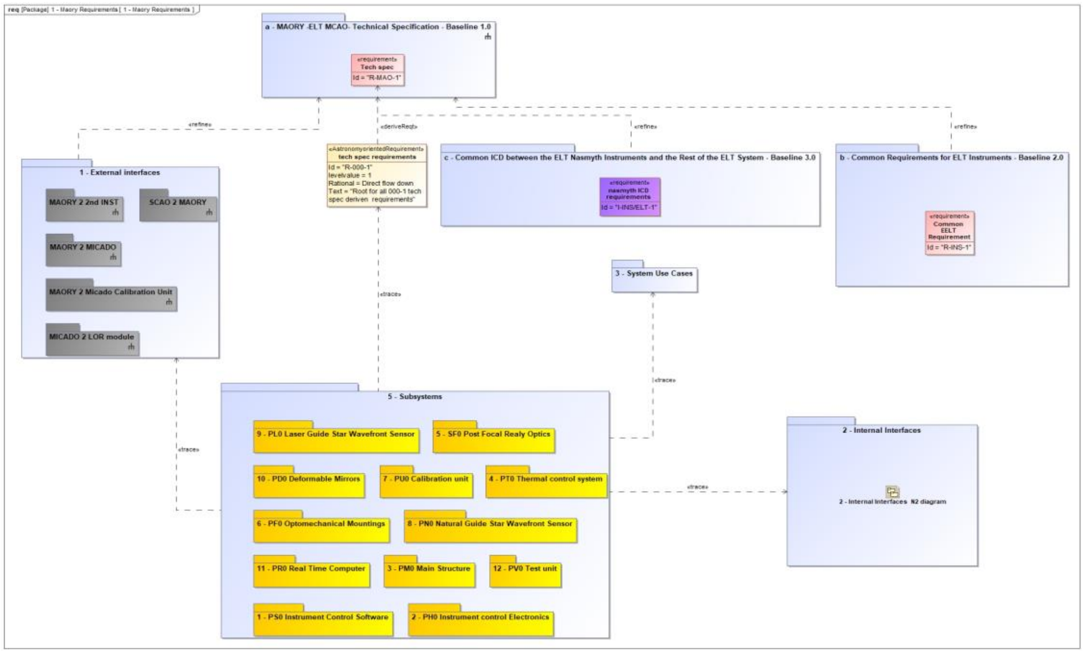

Figure 3 Requirement flow down diagram

5.1 Requirement analysis

In Figure 4 is shown a typical example of flow down through analysis. The stake holder requirement (dark pink block), is directly flow down to system level requirement (yellow block). At this stage it is refined by the system analysis (purple block). In this case two subsystem requirements (Mao-PS0-1.2.2.6 and MAO-PS0-1.3.5.1 light pink block) are traced to this analysis and logically derived from the parent requirement. In this particular case the MAO-PS0-1.3.5.1 open an interface requirement package which specify the interface which is "created" by this requirement.

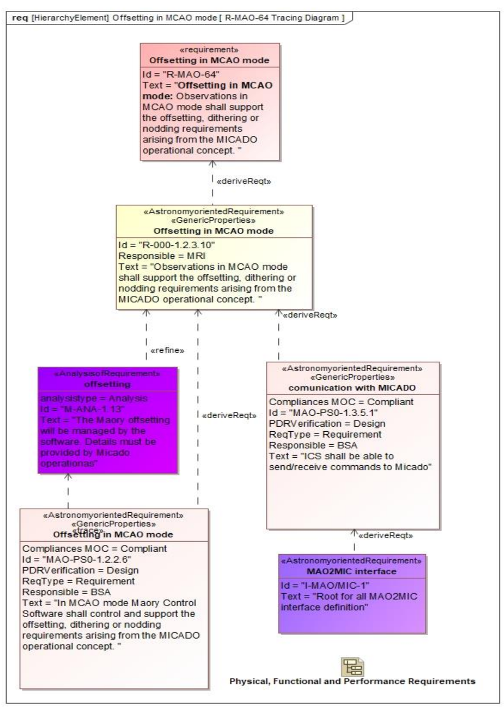

Figure 4 Typical requirement analysis and related flow down

Similar to this process is the splitting of system requirement in subsystem budget requirement, where from a total allocated value, the subsystem partition is split and allocated to the proper item. In Figure 5 is shown as an example, the cooling budget with arrow diagram

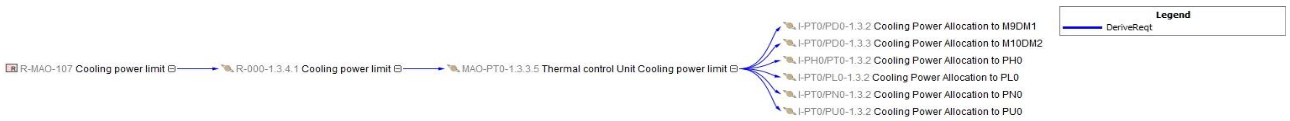

Figure 5 Budget flow down

5.2 Interface *requirements*

The internal interfaces have been modeled considering all the internal relationship and have been organized per topics. Five types of interfaces have been considered:

- Optical
- Mechanical Thermal
- Fluid
- Electrical Software

Whenever there is a relation between two subsystems, an Interface is identified. Each Interface is handled with proper documentations, according to the architecture shown below.

All the 29 relations between subsystems are summarized in the following N-squared diagram:

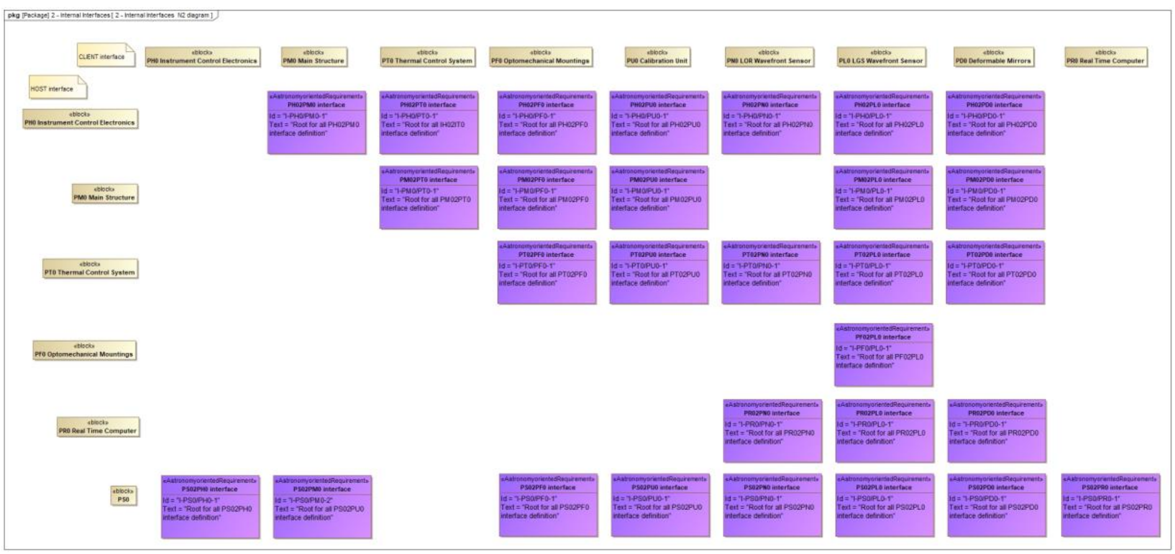

Figure 6 N 2 diagram of Morfeo internal interfaces.

Any of the violet blocks of Figure 6 represent an interface between two subsystems. Depending on the case, the interface can be of a single or multiple types. The carrier of interface is on the rows, while the passenger is on the columns.

The control of each interface is handled at different level as shown in Figure 7 At system level, it is created one Interface requirement document for Each of the 29 interfaces (brown block of Figure 7). In this document are listed all the agreed Interface related requirements. It will define then all the numbers and the duties of each side of the interface. It is handled by The SE and require the approval of both Interface sides. For completeness there will be also a short description of the interface.

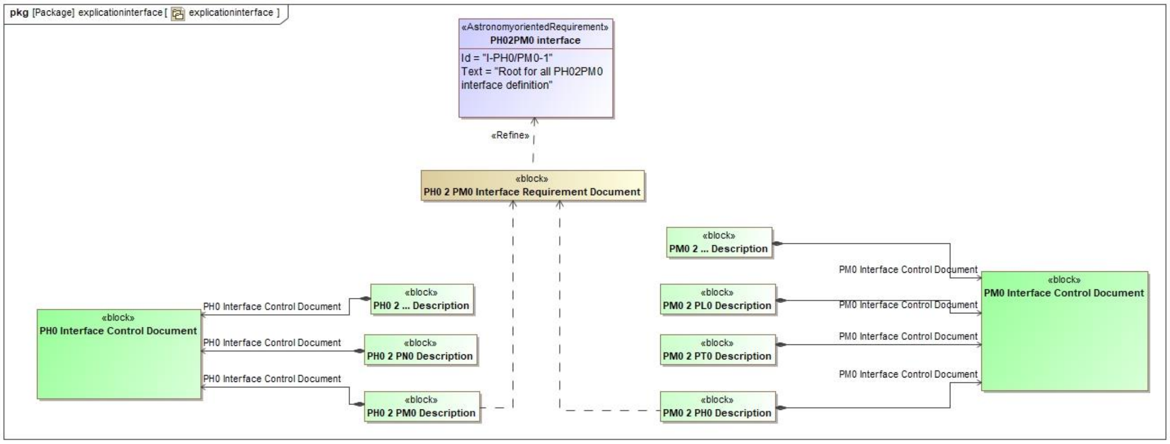

Figure 7 Scheme of the Interface Management (PM0 2 PH0) as an example.

Each Subsystem will have then a dedicated Interface Control Document (green Blocks) which will collect all the interfaces of this subsystem with any of the other subsystems. Each chapter of those documents (light green blocks) will describe, the subsystem's side of the interface (Carrier or passenger) and will show the compliances w.r.t. the related IRDs. In summary there will be 29 Interface Requirement Documents (brown blocks), 10 Interface control documents (green blocks) and in each Interface Control Document *n*-chapters, where *n* is the number of violet blocks of Figure 6 that belongs to the related work packages This document is the Interface Requirement Document (brown block) between Instrument Control Hardware and LOR unit.

6. SUBSYSTEMS DATAPACK

The final results is the subsystem requirement tree (shown in Figure 8). The main root is made by the subsystem tech-specs. It branches out to subsystem requirements of different type:

- Physical
- Functional
- performance
- operational
- interfaces

This root branches also to set of requirements derived by environmental requirement (dark pink block).

To enable an easy process, requirement management, discussion and definition has been conducted thanks to the exporting excel sheets in order to easy the reader and then the communication and updating of any requirement.

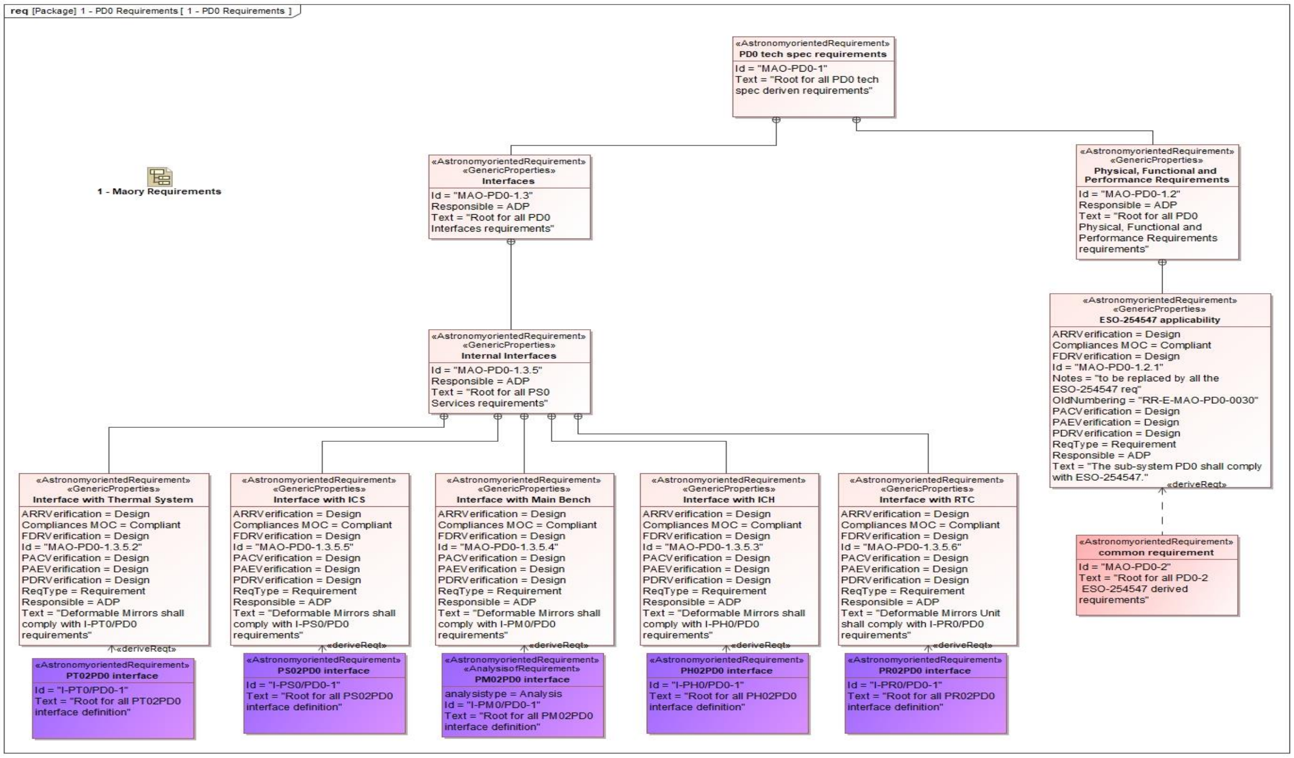

Figure 8 Subsystem requirement tree

7. CONCLUSIONS

The proposed architecture enables a deep modelling of the requirements, with the possibility to lock them to all the model stereotype, like Use cases, activity diagram, interfaces and block diagrams.

Having a common approach to the requirement management, would easy the interchangeability of knowledge and between different project within the community. Will also simplify the training of new resources.

The architecture has been developed in Cameo, but is almost tool independent, and a possible evolution is to translate in different tools.

Discussion are ongoing to evaluate the possibility of collaborate with ESA work on standardization also known as ESA SysML Solution [3].

REFERENCES

1. Riva, M., et al., “MORFEO@ELT: system engineering activity up to preliminary design review”, Paper 12187-61, SPIE “Modeling, Systems Engineering, and Project Management for Astronomy X”, Montreal (2022)
2. Riva, M., et al., “Astro MBSE: Model BAsed System Engineering synthesized for the Italian Astronomical Community”, Paper 12187-61, SPIE “Modeling, Systems Engineering, and Project Management for Astronomy X”, Montreal (2022)
Chinellato, S., et al., “MORFEO/MORFEO: the RAM analysis approach for the preliminary design”, Paper 12187-9, SPIE “Modeling, Systems Engineering, and Project Management for Astronomy X”, Montreal (2022)
3. Alberto González Fernández et al.,” ESA MBSE Evolution: From ESA SysML Toolbox to ESA MBSE Solution, September 2021, Model Based Space Systems and Software Engineering ~ MBSE2021