

A Model-Based Systems Engineering Metamodel – Moving an Organisation to Common System Modelling

Jan P. Weber^{a*}, Fernando Perez Lopez^b, Alicia Alvarez Pascual^c, Fran Luque^d, Silvia Carosi^e, Ricardo del Pino^f, Mauro Pantaleoni^g, Alan Baker^h, Antonio Accardoⁱ, Ignacio Oliva^j, Alexander Rudolph^k, Dennis Landmann^l

^a EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, janpeter.weber@eumetsat.int

^b EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, fernando.perezlopez@eumetsat.int

^c EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, alicia.alvarezpascual@eumetsat.int

^d EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, fran.luque@eumetsat.int

^e EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, silvia.carosi@eumetsat.int

^f EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, ricardo.delpino@eumetsat.int

^g EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, mauro.pantaleoni@eumetsat.int

^h EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, alan.baker@eumetsat.int

ⁱ EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, antonio.accardo@eumetsat.int

^j GMV GmbH, Europaplatz 2, 64293 Darmstadt, Germany, ignacio.oliva@external.eumetsat.int

^k EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, alexander.rudolph@eumetsat.int

^l EUMETSAT, Eumetsat Allee 1, 64295 Darmstadt Germany, dennis.landmann@eumetsat.int

* Corresponding Author

Abstract

The European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT) is currently operating a total of 10 satellites across a wide range of programmes, with more to come in the next few years. The operation and utilisation of long-standing meteorological missions, alongside the introduction and development of new missions, poses a significant challenge in terms of systems engineering and the teams involved. Ensuring commonality between mission design approaches while catering to mission-specific needs in designs becomes essential.

Traditionally, EUMETSAT managed all missions using a document-based systems engineering approach or via a mixed document and Model-Based Systems Engineering (MBSE) approach, relying heavily on the expertise and organisational knowledge of each systems engineer. Several attempts at establishing MBSE methodologies were made, with the use of Enterprise Architect (EA) and SysML becoming the most widely adopted. This effort started with significant contributions from projects such as the Sentinel-3 mission. The Sentinel-3 Re-Engineering project, initiated in 2018, made extensive use of this approach and contributed to its further development, marking a significant step in the organisation's shift towards a standardised MBSE methodology, sharing the same underlying modelling approach and establishing a model as the accepted reference for system design.

The introduction and development of a common MBSE approach and the creation of a model as the accepted reference for system design quickly started to become a recommended practice for other, already in-flight missions as well as future missions such as Meteosat Second and Third Generation, Sentinel-6, CO2M, EPS-Aeolus, and EPS-Sterna.

Throughout this transition, it was necessary to integrate long-standing interface and system design approaches with new features such as model-driven information item generation, automated diagram re-drawing, organisation-specific structural and functional decomposition modelling, and the addition of essential SysML element attributes and new system modelling features. Specifically, this required changes in how data is documented in the system model to cater to both current and future mission designs. Each mission has/had different documentation approaches, yet many pieces of information are similar in content. Identifying a common ground was essential, leading to the establishment of essential attributes for system modelling elements. Furthermore, new system design features were added to the modelling toolbox, including elements for the modelling of functional flows, product dependencies, and respective document generation and export.

This paper presents a retrospective analysis of EUMETSAT's transition to a common MBSE approach, providing insights into the evolution of the EUMETSAT MBSE framework. It discusses the challenges, benefits, and practical lessons learned from integrating stereotyping via an MBSE meta-model, model-driven information item generation, functional decomposition modelling, and the migration of system-level documentation. In addition, the paper outlines future directions to further enhance both modelling and system analysis capabilities.

Keywords: Copernicus, MBSE, Metamodel, EUMETSAT, SysML

Acronyms/Abbreviations

ADD	=	Architecture Design Document
ADU	=	Application Data Unit
EA	=	Enterprise Architect (Sparx Systems)
EPS	=	EUMETSAT Polar System
EPS-SG	=	EUMETSAT Polar System – Second Generation
EUMETSAT	=	European Organisation for the Exploitation of Meteorological Satellites
ICD	=	Interface Control Document
IVV	=	Integration Verification and Validation
MBSE	=	Model-Based Systems Engineering
MDG	=	Model Driven Generation
MFG	=	Meteosat First Generation
MME	=	Multi-Mission Elements
MSG	=	Meteosat Second Generation
MTG	=	Meteosat Third Generation
SDD	=	System Design Document

1. Introduction

EUMETSAT's roots trace back to the early operational Meteosat satellites in the 1980s, including the initial Meteosat First Generation (MFP) program. An era where documentation was predominantly paper-based and tailored to single, self-contained missions or programs. As Meteosat systems grew more complex, Meteosat Second Generation (MSG) followed, adding layers of technical and organisational requirements that challenged existing processes. Subsequently, EPS/Metop introduced a new dimension by adding a low earth orbiting polar component to the so far strictly geostationary set of satellites operated by EUMETSAT, thereby multiplying the need to handle distinct mission data flows and operational timelines. By the 2010s, EUMETSAT's responsibilities further diversified with the Copernicus Program by the European Commission, adding in 2016 Sentinel-3 and in 2020 Sentinel-6 to the list of operated missions by EUMETSAT and expanding the scope and technical complexity of ever more missions to be managed. The 2010s were also filled with work in preparation for the third generation of Meteosat satellites (MTG) as well as the second generation of EPS/Metop (EPS-SG) seeing launches both in 2022 and more scheduled for 2025 and 2026. Additionally, EUMETSAT is taking responsibility in developing, preparing and operating the future missions of CO2M (part of the Copernicus Program), EPS-Aeolus and EPS-Sterna further underscoring the necessity of handling multiple programs with a shared set of consistent engineering standards. With 10 satellites currently operated by EUMETSAT and adding additional 7 satellites to the list in the next 24 months, alongside extensive data exchanges with international partners such as NOAA and global data service responsibilities (e.g., serving as a data node for missions like Jason-3), the complexity in ground segment systems engineering and coordination among involved teams significantly increases. Ensuring commonality between mission design approaches while catering to mission-specific requirements is essential - not to enforce a rigid set of nomenclatures or practices, but to establish a unified framework that standardizes system structure, interface modelling, and documentation, yet remains flexible enough to incorporate necessary adaptations.

Throughout this evolution, EUMETSAT had to reconcile legacy methods with emergent technologies, while also addressing the challenges inherent in managing shared resources. This dual focus has driven the development of innovative engineering solutions that bridge legacy practices and modern technological demands.

This paper specifically focuses on the systems engineering challenges and solutions relating to the ground segment design; it explicitly excludes the spacecraft design aspects.

2. Overview of EUMETSAT Missions and Systems Engineering Challenges

Historically, many satellites were developed as largely standalone projects, with certain functions – like data archiving, real-time dissemination, and interfacing – being handled on a mission-specific basis and sometimes partially designed externally of EUMETSAT. As the number of operational satellites grew, EUMETSAT recognized that not only was duplicating core components unsustainable, but that the continual mutation and evolution of these components necessitated a transition to a shared resource concept. The adoption of Multi-Mission Elements (MMEs) [1] was driven by organizational needs for resource optimization and standardization, independent of MBSE. However, the introduction of MBSE provided substantial improvements in standardizing and efficiently documenting MMEs,

greatly enhancing their effectiveness. This shift required extensive coordination across historically separate projects so that existing missions could be upgraded to make use of the MME backbone. At the same time, brand-new missions like Sentinel-3, MTG, and Sentinel-6 were designed or setup with the usage of MME shared functions either from start or by making use of them as much as possible when deployed at EUMETSAT, reducing duplication of efforts.

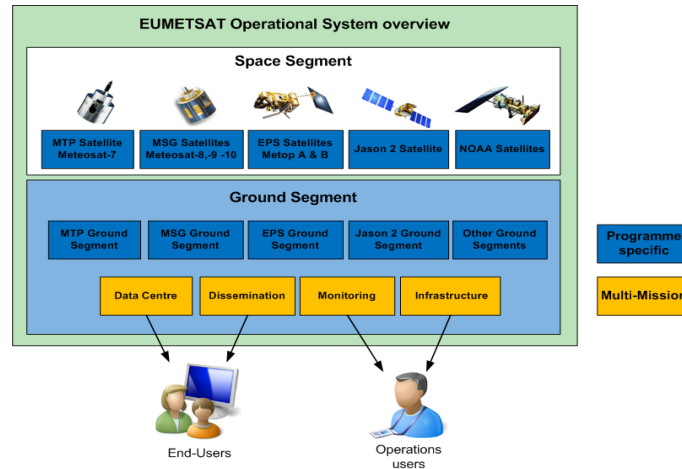


Fig. 1: Early 2015 High-Level MME Overview – Illustration of Shared Resource Utilization

The organisational approach to unify common functions under the MME umbrella also drove a need for consistent technical specification of interface definitions and document structures. Where older missions or externally provided designed missions maintained Interface Control Documents (ICDs), System Design Documents (SDDs) and Architecture Design Documents (ADDs) in varied, yet often similar formats, Word and/or PDF documents, EUMETSAT saw a need for an integrated framework centred on engineering shared resources, encompassing both shared and federated ownership models that support consistency across missions. Moreover, the introduction of more new missions demanded for a forward-compatible approach – no more re-inventing the wheel for the ground segments supporting each spacecraft and a coherent set of system design tools. This synergy – finding common ground among legacy documentation styles and ensuring new designs fit a shared standard – represented a complex but necessary endeavour to sustain a growing fleet of satellites.

3. From Document-Based to MBSE – Early Efforts

EUMETSAT's early missions, including Meteosat, MSG, and the EPS/Metop program, operated predominantly under a document-intensive paradigm for the system engineering, where each engineering document – like an ICD or SDD – was manually authored and updated by experts from the organisation. Over time, it became clear that as missions multiplied, this approach was vulnerable to version control headaches, cross-referencing errors, and reliance on the knowledge and know-how of engineers maintaining the documentation and linking everything together.

With the introduction of more advanced mission ground segments in the early to mid-2010s, internal teams began with the introduction of Model-Based Systems Engineering (MBSE) approach. Early on, EPS-SG had introduced a system model using Vitech CORE [2] in parallel with standard documentation, but the tool's limited flexibility and eventual end-of-life meant EUMETSAT needed an alternative.

Significant groundwork was done by the Sentinel-3 Re-Engineering initiative [3], where entire sets of system-level documentation were planned to be driven by a central system model. For this the team evaluated alternatives and selected EA (by Sparx Systems [4]).

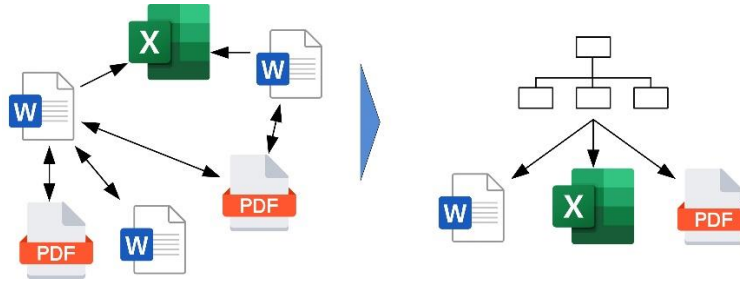


Fig. 2: Document Based SE vs MBSE

While the initial learning curve was steep as the projects needed to not only familiarise with the new tool chosen by the organisation but also a new modelling approach had to be established that combined the needs, lessons learned and advantages of many missions while streamlining and simplifying the modelling and documentation approach. Early on the Sentinel-3 Re-Engineering was able to showcase to the organisation modelling and information management successes using the developed MBSE model in EA including the synchronization with IBM DOORS [5](requirements allocation to components and functional traceability, see section 5.6).

Together, Sentinel-3 Re-Engineering and Sentinel-6 mission teams experiences rapidly highlighted common needs - such as standard ways to model data streams, interfaces, functional flows, and system decompositions. EUMETSAT recognized that not only could documentation be produced more consistently, but new synergies emerged: cross-team referencing was much simpler, and external engineering services could be sourced more readily since EA usage is widespread in the aerospace domain. Thus, MBSE was no longer just a concept; it was actively delivering benefits, such as more robust traceability, fewer documentation discrepancies, and improved readiness for major reviews. Nonetheless, challenges remained around balancing new MBSE approaches against established, heavily regulated review processes requiring PDF or Word deliverables. The solution was a hybrid: a robust, central MBSE repository used day-to-day, with automated document exports generated for major milestones and official sign-offs.

Early MBSE initiatives demonstrated concrete improvements in standardizing design artifacts, reducing rework, and enhancing data integrity. The development of the internal “EUMETSAT meta-model” and subsequent MDG technology objectively validated that the primary benefit of MBSE lies in its ability to deliver consistent, traceable, and efficient design processes rather than in its status as a novel methodology.

4. Establishing a Common MBSE Approach – History, Challenges and Benefits

The Sentinel-3 Re-Engineering project joined later by the Sentinel-6 mission project were showcasing MBSE advantages in EA. These missions developed by different teams shared the same fundamental architecture of the early Copernicus mission ground segment designs. The difference was that at the time of 2018-2020 when the foundational work of the MDG was made, the Sentinel-3 mission had one, respectively two satellites flying and could build upon an operationally used set of system design of a live mission, while Sentinel-6 team had to wait till 2020 to see their satellite flying.

One of the first steps was to establish a common modelling approach for interfaces, data streams, data objects, protocol definitions, dependencies, and overall model structure. Although EUMETSAT does not formally designate this as an “EUMETSAT ontology”, its principles are implicitly defined by the developed MDGs and established working practices. This became crucial as these core functions and approaches would be at the very heart of every other mission adopting the same modelling technologies within EUMETSAT. Initially, the Sentinel-3 project played a key role in contributing to the MBSE approach within EUMETSAT. To coordinate these efforts better across multiple missions, the MBSE working group was established - bringing together parallel projects whose combined contributions have driven a unified and scalable modelling strategy.

This approach resulted in the already mentioned MDG or Model Driven Generation Technology (see section 5) that encapsulates EUMETSAT’s unique stereotypes, tagged values, and document generation templates.

The MDG Technology allows engineer to pull from a library of standardized elements – such as EUM_Interfaces and EUM_Data-Streams with predefined sets of EUMETSAT common attributes and a background meta model linking the information during document exports – when modelling system components and interactions.

To allow for a broad adoption the Systems Engineering team developed two sets of documentation, an EA Modelling Guide explaining the principles and usages of the new established system modelling approach and an EA Administration Guide providing processes for user and group management, ensuring that only authorized engineers could modify packages, while read-only groups had easy transparent access for review. This was important as with the growing experience of using the MDG and EA models by the Systems Engineering teams a growing interest in non-Systems Engineering teams emerged. Teams in operations responsible for the daily operations of the EUMETSAT satellites as well as data processing and flight operations ground segments as well as the beforementioned MMEs became more and more interested in accessing the models directly to receiving access to the most up-to-date information impacted by the system engineers.

To allow for better readability by all parties involved and common system architecture management a standard folder structure (or “View” structure) within the system model in EA was introduced, covering aspects like System Views, Interface Views, Data Views, so that each mission project had a consistent baseline (see Fig. 3).

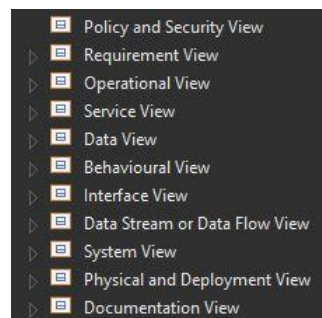


Fig. 3: EUMETSAT System Model Structure

To handle parallel development and sign-off across multiple missions, EUMETSAT adopted a combination of Git file based model version and branch handling and LemonTree (a LieberLieber Software GmbH tool [6]) (see section 5.3) to handle model compare and merge checks, which also facilitated the collaborative editing.

By tying model elements directly to official documentation (e.g. ICDs or ADDs), EUMETSAT minimized the risk of discrepancies between the “single source of truth” (the central model of a mission) and external PDF/Word deliverables. This approach significantly reduced duplication of information and errors, an outcome especially important during formal reviews. On the other hand, the procedural generation of documentation meant that now documentation was generated standardized with no information to be left out, resulting in often larger documentation due to the often enlarged pool of information that could be handled in the model due to collaborative contributions by the engineers involved in contrast to individual document updates prior. Moreover, this approach requires extra effort when external partners (e.g., ESA, NOAA, or JPL) need different document structures or levels of detail. In these cases, teams often spend time adapting or “massaging” the outputs, reducing some of the efficiency gained through automation.

Despite these challenges, the MBSE approach delivers well-known benefits of a single repository: better consistency, collaborative access, and version control. It also forces a more systematic approach, prompting engineers to address potential gaps that might have been ignored in traditional document-based methods. On the other hand, customizing or improving the presentation of information - something often done easily in Word or PDF - can require specialized effort in SysML [7].

There is also a learning curve: both the tools and working practices demand significant training and support from experienced users. Updates to the metamodel often require broader discussion and agreement, sometimes slowing progress. Yet EUMETSAT has achieved notable successes, such as by the MDG technology by aligning with the organization’s common “ontology” for design and interfaces, and a standardized set of views that covers most system engineering activities—from initial mission development to routine operations. This standardization makes it easier for engineers to locate the information they need, whether they are joining a new mission or moving between projects.

Overall, this common MBSE approach laid the groundwork for scaling EUMETSAT's systems engineering to support both current and emerging satellite missions under a single, unified framework.

With the experiences coming from the Sentinel-3 Re-Engineering project and the Sentinel-6 project as well as multiple other MDG exploiting projects, it was possible to go through multiple iterations with further missions adopting the modelling approach including the MTG, EPS-SG, CO2M, EPS-Aeolus and EPS-Sterna programs.

Key Benefits of a Common MBSE Approach:

1. **Reduced Onboarding Effort:** With an established meta-model and well-documented best practices, a new engineer or external contractor can more quickly understand how systems are structured, where to find mission data, and how to generate standardized documents.
2. **Cross-Mission Synergies:** Missions can replicate existing data streams or interface definitions from the central MME model as well as other mission models, ensuring that each new or legacy satellite doesn't re-invent fundamental structures and already defined interfaces.
3. **Automated Documentation:** ICDs, ADDs, or SDDs can be reliably output from the model, preserving consistency, accelerating reviews, and reducing manual editing.
4. **Flexible Tool Support:** EA's widespread use in aerospace means external companies can more easily provide specialized services, from advanced scripting to automation of tasks like auto-redrawing ibds or exporting data for operational configuration (e.g. data dissemination configuration). One of the core advantages of EA is that it supports multiple modelling notations and methodologies, allowing for flexible and customised use of said features, views, and definition of stereotypes.
5. **Forward Compatibility:** By initiating MBSE from day one, missions such as CO2M, EPS-Aeolus, and EPS-Sterna can leverage pre-established models - for instance, for MME components - instead of recreating them independently. Using MBSE with pre-established resources not only eliminates the need for complex tool migrations mid-program but also ensures continuity, consistency between missions and reduces legacy gaps.

A key challenge was to migrate mission data from legacy documents managed by EUMETSAT into EA models without disrupting ongoing operations and mission enhancements while at the same time confirming the information in the documents was accurate and not impacted by creeping in of inconsistencies due to past manual documentation updates and all this while developing the MDG further.

Early missions, such as MSG and EPS/Metop had decades of manually maintained artifacts, each with distinct naming conventions and sometimes incomplete cross-references due to non-aligned documentation updates (a fact overcome by the MDG procedural documentation generation from a single central model).

It became important to break down the EUMETSAT system modelling into mission specific and common models. While missions like MSG or EPS/Metop would have their specific model repository in Git, both and all other missions would leverage common system models such as the one of the MMEs by importing "Shared Assets" relevant to all EUMETSAT EA projects. As a result, older missions could gradually adopt new common building blocks (like shared data archiving or EUMETCast dissemination modelling) by referencing the same instance from the base repository instead of duplicating it. Meanwhile, future missions such as CO2M, EPS-Aeolus and EPS-Sterna directly started with the MBSE approach, inheriting the structures, stereotypes, and documentation templates from the base model provided by the MDG.

In practice, bridging the old and new required consistent training, documentation, and pilot conversions, but gradually, each mission's design and interface definitions lined up with EUMETSAT's overarching MBSE guidelines. The result was a significant reduction in custom solutions, more streamlined updates to multi-mission functionality, and a clear path to incorporate new missions with the benefit of solution reuse.

5. Model-Driven Enhancements and Tooling – The MDG

The *MDG Technologies* allow users to extend EA's modelling capabilities to specific domains and notations. *MDG Technologies* seamlessly plug into the chosen MBSE tool to provide additional toolboxes, UML profiles, patterns, templates and other modelling resources.

For EUMETSAT, the following MDG technologies have been and are continuously further developed, namely:

- **EUMETSAT Stereotypes / EUM Systems Modelling** (section 5.1 and 5.2)
- **Document Templates** (section 5.4)

- **Document Creation Wizard**
- **Project Creation Wizard**

While the first two MDG technologies will be discussed more in detail in the following sections, the two wizards allow to make it easier for a given Mission/System to set-up a new document containing the model information respectively a new model view structure (cf. Fig. 3) much easier. The Wizards allow for a pre-defined automatic generation of initial structures needed for either the system model (Project Creation Wizard) or for the documentation export (Document Creation Wizard), therefore reducing time efforts needed for frequently repeated activities.

5.1 EUMETSAT Model Structure

To simplify and organize the information management within the central model, a high-level view structure shown in Fig. 3 was created. Each View is containing a set of sub structures to further break down the information and structure of the inner view information management. The Data View and Data-Stream View shall be described exemplary in more detail:

The *Data View* provides information of the Application Data Units or ADUs and its respective hierarchy (see Fig. 4). It is used as Data Dictionary and specifies the domain specific items used in the model. In the ADU Dictionary all the ADUs relevant to the mission architecture are reported. Organisation can occur as preferred by the mission but is generally recommended to be organized by grouping of similarity or by facilities of origin.

An important information feature originating from the Data View is the relationship building between ADUs and the data-streams they are conveyed by using the custom EUMETSAT stereotyped connection *EUM_Conveyed ADU* and the relationship building between ADUs and the originating (producer) and target (consumer) facilities using the *EUM_Produced by* and *EUM_Consumed by* custom EUMETSAT stereotyped connections (see Fig. 5)

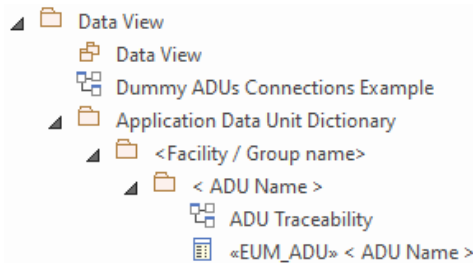


Fig. 4: Application Data Unity ADU Package Content

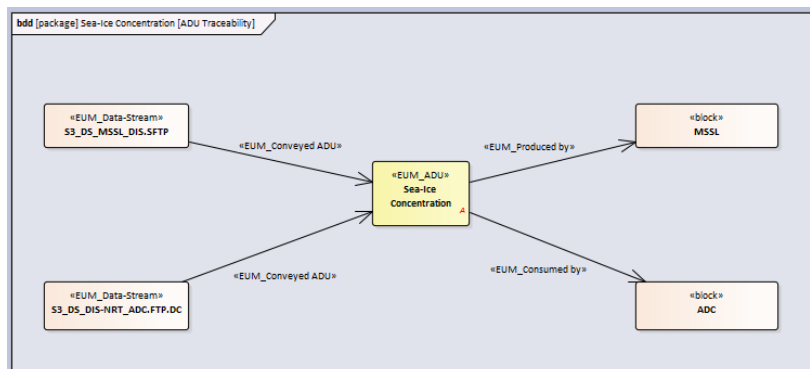


Fig. 5: ADU Traceability

As a second example, the *Data-Stream View* provides information on the transferred data, the direction and the type of protocol involved in a data/information exchange. The data-stream itself must be comprised within interfaces and are described via document export in formal ICDs.

The Data-Stream View can be broken down into lower-level packages that contain a defined set of data-streams linked to a model structural management level. The usage of level breakdown was introduced to allow the separation of external and internal interfaces/data-stream to/from and within EUMETSAT as well as to further break down according to the hierarchy of the structural design at system and component level.

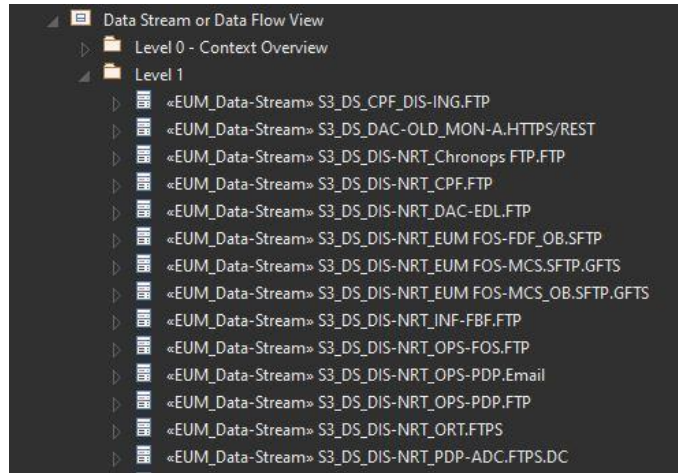


Fig. 6: Data-Stream Package Content

Data-Streams must be related to ADUs with the beforementioned *EUM_Conveyed ADU* stereotype relationship and in return need to be also related via the *EUM_Used Interface* stereotype to an interface the data-stream is belonging to.

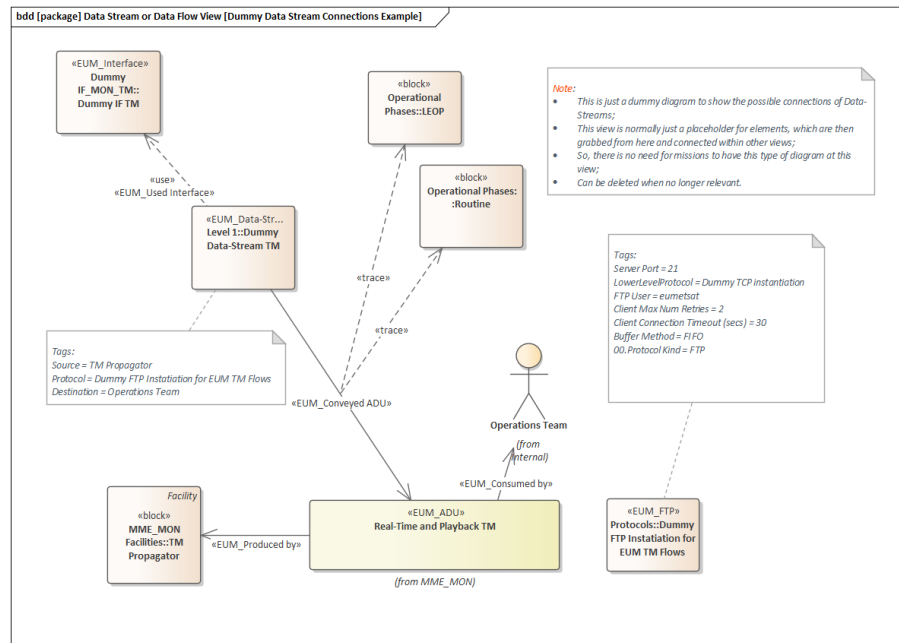


Fig. 7: Data-Stream Traceability Diagram

The data-stream shall further be associated to the applicable protocol used to implement it (see Fig. 7). Furthermore, as demanded by phase mission design, the data-stream can detail an *Operational Phase* trace on top of the *EUM_Conveyed ADU* association to further detail which specific project phases an ADU is conveyed on by the data-stream and in which it isn't conveyed.

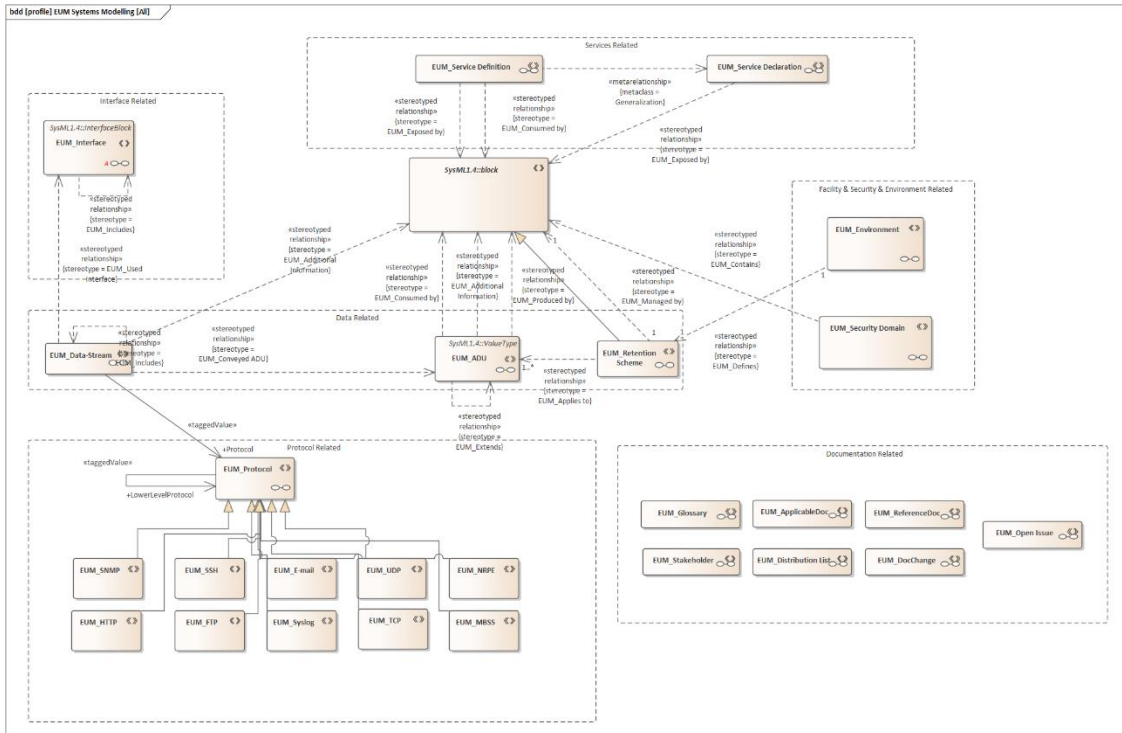


Fig. 8: Overview of all Stereotypes and Their Relationships

5.2 EUMETSAT Meta Model

The Meta Model developed by EUMETSAT is at the core of each modelling activity Domain using the custom stereotypes touched already. A custom stereotype applies in different characteristics to a basic SysML model component or feature. Thereby it allows for the extension of the Unified Modelling Language UML by developing a domain-specific toolset. EUMETSAT’s toolset has grown over time and extends further than two items of EUM_ADU and EUM_Data-Stream highlighted in section 5.1. Each stereotype is set into a specified relationship within the meta model to other stereotypes created to allow for both systematic model building and likewise systematic model information extraction (e.g. for document generation).

Fig. 8 shows a full overview of all stereotypes and relationships in use.

5.3 Git & LemonTree

When the Sentinel-3 Re-Engineering project started to use EA, a classical approach using a database server was chosen. While this proved reliably at first, frequent modifications by multiple stakeholders accessing the system model and needing to work on the same object or objects belong to the same structure (lower level decomposition) posed challenges as concurrent engineering on the same model object was not possible and posed – especially in the home office intensive time of 2020-2022 (COVID Pandemic time) – significant offline coordination needs. Furthermore, managing database servers where change management was less controllable further drove the decision to move to a Git file-based approach rather than database approached for EA.

By using Git, it is possible to return to a previous version/state of the model without much effort due to the commit management options within Git. However, undetected mistakes can likewise propagate between commits without being detected and can result in time demanding efforts to deal with them when they affect shared branches between multiple teams.

For this reason, GitLab offers different roles to control the permission of the User using the repository. Furthermore, the usage of protected branches defining who can push and merger changes becomes important.

In this context the important work of merger management becomes a central activity to assess if changes are supposed to be merged and what impact on the model the merger of a branch has and if necessary to resolve merge conflicts occurring. EUMETSAT decided to use LemonTree [6] for this as it provides a Plug & Play model version management of differ and merger.

5.4 Automated Document Generation

The MBSE tool allows the use of report templates to generate documents based on models. A template defines the type of content of the report and present the information in a specific, pre-defined format and style. A range of system templates were created by EUMETSAT to cater generic as well as mission specific needs within the Document Generator. Furthermore, when needed it is possible to design custom templates which provide the additional advantages of being further tailored to single mission needs without the need to re-import them into the overall MDG. This MDG Technology includes styles and templates for generation of Interface Control Documents (ICDs) and System Design Documents (SDDs). These templates provide the necessary resources to make use of the model package structure, diagram, «model documents», templates and elements to generate a EUMETSAT document. The user only must update the notes on the elements and select elements from the model to include those in the document generation. To generate an ICD a user simply needs to drag-drop include the interfaces that shall be exported via the document generation; the correspondent *Data-Streams*, *ADUs* and *Facilities* will be automatically added (using a set of searches) to the document if their relationships are correctly modelled in the EA System/Mission model. Thus, further simplifying the complexity of document export.



Fig. 9: Sentinel-3 Mission Master ICD Export Configuration of Interfaces

5.5 HTML Exports

A huge advantage within EA is the ability to generate nightly HTML exports of the model to make the model accessible to a wider public within the organisation without the need to provide EA licenses. Exposed via a HTML server, each link provided allows the user to navigate through each mission's or each MME model.

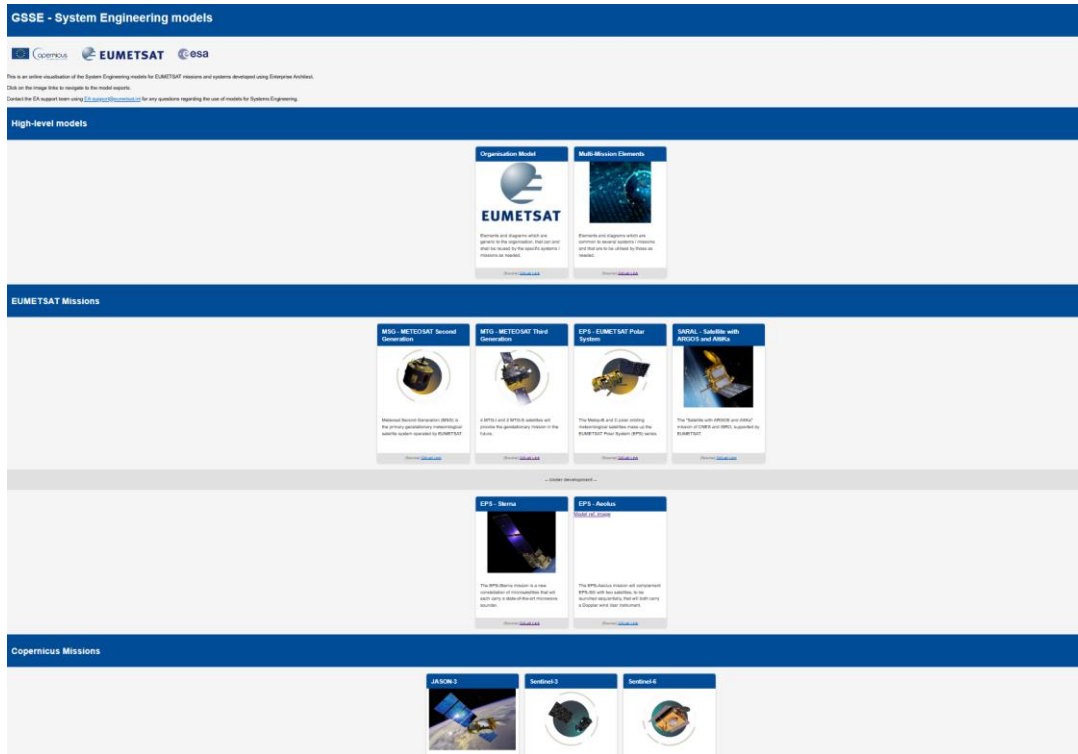


Fig. 10: EUMETSAT HTML – Systems Engineering Model Server

5.6 DOORS Synchronization

While a system model is great on its own, the integration with outside tools is an added value. EA allows for the synchronization and pulling of elements from IBM DOORS for the manual synchronization of a requirement package in EA with the Requirements managed in a DOORS Module.

By selecting inside the Requirements View (cf. Fig. 3) a package designated to contain the requirements of a specific DOORS module it is possible to import the DOORS module requirements into EA. For this it is essential to map the EA properties and DOORS Requirement Elements.

As the DOORS module might change due to requirement changes or updates, it is possible to re-synchronize the EA Package requirements.

Once imported it is up to the engineers to link the requirements to the designated EA objects such as facilities, behaviours/activities and more.

6. Roadmap and Ideas Towards 2030

While the current MDG has enabled EUMETSAT to unify much of its system modelling and documentation - particularly through automated exports, standardized structures, and cross-mission referencing - there remains substantial potential to further refine and expand the MBSE framework. In interviews with various stakeholders, several overarching objectives emerged:

1. Enhance Operational Integration and Automation

- **Redrawing of Interface Diagrams:** Automate and streamline diagram generation to reduce the manual overhead on engineers and ensure consistently updated views.
- **Export of Dissemination System Configurations:** Building on the CO2M project’s pilot, leverage the model to generate configuration files (e.g., for operational tools currently under evaluation) directly, closing the loop between design documentation and live mission implementation.
- **System Technical Budget (STB) Analysis:** Use the model as a data source to drive simulations or system technical budget analysis, enabling early detection of performance bottlenecks and facilitating iterative, model-driven refinements.

Additionally, this integrated approach supports model-informed Integration, Verification and Validation (IVV), aligning system verification processes with design activities to enhance overall project assurance.

2. Strengthen Collaboration Across the Full Mission Lifecycle

- **Integrated Collaboration Among System Engineering, IVV, and Ops Prep:** Provide distinct but interconnected model views (or stereotypes) for each discipline. This ensures that requirements, design constraints, verification items, and operations-relevant data can all coexist in the same repository without overwhelming any single user group.
- **Early Co-Development:** Encourage further the collaborative development during the early concept phase of missions so that IVV and Ops Prep teams can shape relevant stereotypes or data fields from the start. This avoids late rework and fosters a sense of shared ownership in the model.

3. Expand Customizability While Retaining Consistency

- **Flexible Framework, Common Core:** Continue refining the MDG so each mission can add specialized stereotypes where justified—yet all missions share a common backbone of data streams, interface blocks, and document templates.
- **Evolving Model Purposes:** Whether automating parts of the mission configuration (firewalls, data routes) or introducing simulation parameters (bandwidth, availability, timeliness), the MBSE environment must remain anchored in real user needs. As new features arise (e.g., functional flow modelling, advanced simulation), they should be introduced “cum grano salis,” ensuring the model continues to serve practical objectives.
- **Assurance and Tool Qualification:** As EUMETSAT’s MBSE outputs (configuration files, STBs, etc.) become increasingly critical, considerations around tool qualification and governance may gain importance—particularly where automatic outputs replace human-driven reviews.

4. Establish Functional Analysis as a Core MBSE Capability

- **New MDG from EPS-Aeolus:** A new MDG developed within the EPS-Aeolus project and recently presented in the MBSE working group and is planned to be integrated into the EUMETSAT toolbox and documented in the EUMETSAT Modelling Guidelines for later adoption in future developments of new missions.
- **Dedicated Stereotypes for Functional Analysis:** This MDG introduces specific stereotypes and connectors for functions, functional parameters, and functional chains, ensuring a structured and standardized representation of functional elements.
- **Decoupling Functional and Physical Architectural Domains:** By making the functional domain independent of the logical and physical architecture domains, this approach enables a more modular evolution of both, reducing dependencies and improving maintainability.
- **Laying the Foundation for Multi-Level Reuse:** The framework sets the stage for future reuse at multiple levels, allowing functional definitions to be leveraged across different missions, projects, or configurations, thereby enhancing efficiency and consistency across EUMETSAT’s MBSE ecosystem.

Looking towards 2030 and beyond, the vision for EUMETSAT’s MBSE ecosystem is one in which each mission model evolves seamlessly from initial concept through routine operations, continually adapting to new technologies, demands, and user needs. By incorporating lessons from initiatives like CO2M - where model-to-configuration exports are tested and incorporated - and further refining the MDG, the organization can keep strengthening the model as both an authoritative reference and a practical engine for engineering and operational tasks. Each enhancement - whether automatic diagram redrawing, advanced STB analysis, or domain-specific views for IVV and Ops Prep - must remain grounded in day-to-day usage, ensuring the model is never just another artifact but a vital, living repository. In this way, MBSE becomes the linchpin that unites diverse stakeholders, maintains institutional knowledge, and ultimately drives forward a more agile, efficient, and collaborative approach to satellite ground segment development and operations at EUMETSAT.

7. Conclusions

EUMETSAT’s move toward a standardized Model-Based Systems Engineering (MBSE) approach has shown that creating both mission-specific and shared models can effectively handle the challenge of running multiple missions simultaneously. This is even further enhanced when developing ground segments with common multi-mission core elements. By applying organization-wide guidelines and using adaptable tools - like Sparx Systems’ Enterprise Architect, customized MDG technologies, and Git-based version control - EUMETSAT has streamlined interface

definitions, system breakdowns, and document generation across its various satellite programs. While the transition initially posed difficulties, it led to clear improvements in design consistency, cross-mission reuse, and automation - benefits unattainable under older, document-based methods.

The Sentinel-3 Re-Engineering and Sentinel-6 projects demonstrate how MBSE can support both new and ongoing missions by centralizing information and enforcing consistent design practices. Over time, EUMETSAT refined its MBSE framework to accommodate specific project needs while keeping a shared core of stereotypes, attributes, and templates.

Looking to the future, EUMETSAT aims to shift MBSE from static documentation to a dynamic resource to better support development activities and, potentially, operational configuration management. Any broader operational usage is currently under internal discussion and evaluation. Planned enhancements - such as automated generation of dissemination configuration files, interface diagram redrawing, and integrated System Technical Budget (STB) analysis - reflect a growing ambition to ensure the model is a *living* resource at the heart of every mission lifecycle stage. As more missions and projects start out with MBSE, EUMETSAT plans to extend the framework further by linking system design with live operational data and advanced simulations. This approach cements MBSE as an essential strategic asset - one that promotes agility, collaboration, and long-term quality in Europe's meteorological satellite services.

Acknowledgements

The authors would like to thank the teams and colleagues involved in the development and evolution of the MDG technology at EUMETSAT together with all contracting companies who have contributed and are continuously contributing to this development.

References

- [1] K. Marston, Multi-Mission Elements: White Elephant or Essential Business Sense?, 14th International Conference on Space Operations, Daejeon, Korea, 16 - 20 May 2016.
- [2] Vitech CORE Software – Official Website: https://vitechcorp.com/core_software/
- [3] Fernando Pérez-López, Gianni Casonato, Jan P. Weber, Silvia Carosi, Eduardo Sanchez Suárez, Ricardo del Pino, Ignacio Oliva, Antonio Accardo, Andrew Holmes, Dennis Landmann, Hilary K. Wilson, Re-Engineering A Ground Segment – Preparing Sentinel-3 for the Future, 18th International Conference on Space Operations, Montreal, Canada, 26 - 30 May 2025.
- [4] Enterprise Architect by Sparx Systems - Official Website: <https://sparxsystems.com/products/ea/>
- [5] Overview of DOORS, <https://www.ibm.com/docs/en/engineering-lifecycle-management-suite/doors>
- [6] LemonTree by LieberLieber – Official Website: <https://www.lieberlieber.com/lemontree/>
- [7] SysML Open-Source Project, <https://sysml.org>