Model Based System Engineering Process for ESS AB

R. Linander, M. Gohran, R. Duperrier, F. Plewinski, P. Radahl

European Spallation Source ESS AB, P.O. Box 176, SE-221 00 Lund, Sweden

Rikard.Linander@esss.se

Abstract. The design of European Spallation Source (ESS AB) is a major challenge for the community of intense neutron sources due to the inherent complexity of such facilities, with a large number of different systems that are essential for the operation. The project is managed through a worldwide distribution of design activities, which are based on a combination of in-kind and cash contributions. Furthermore, it will be essential to have transverse approaches for functions, constraints and interfaces such as nuclear safety, security, alignment, etc. Due to the large variety of design and work contributions in combination with a complex set of functions, an essential condition for the success of this project is a systematic approach to be adopted by all contributors to ensure consistency between requirements, design and expected performance.

This centralized approach will allow a holistic view that enhances the quality of the system of interest and the configuration management during the implementation when change requests are unavoidable. This paper describes the model-based systems engineering processes being implemented for ESS AB to support the construction of the facility.

1. Introduction

Model Based Systems Engineering - MBSE - will be a part of the activities for the ESS facility construction. For a green field facility like the ESS, MBSE will enhance the overall project management and quality assurance. Therefore, the ESS Systems Engineering Management Plan - SEMP - is created with the intention to describe the processes, products, roles and tools that will be used by the Systems Engineering team to support the construction of the ESS facility. The SEMP is applicable to all Systems Engineering tasks to be performed in support of the ESS programme.

The main objective by incorporating MBSE to the ESS project is to build a virtual model that will encompass a behavioural and physical description of the facility. This model will establish the requirements and functionality of systems, components and items in a graphical model based point of view. This setup provides the means to define and describe interfaces in a precise and traceable manner. It will enhance the possibilities to discover inconsistencies and to avoid integration problems at an early stage of the design work. If the MBSE models are coherent and reliable, the overall reliability and robustness of the facility design will be improved. The MBSE models will also be used as a tool for traceability of design change impacts of interfacing items.

2. Methodology

Traditionally, projects have been carried out using a document centric System Engineering - SE - approach to manage interfaces, requirements and definition of operational modes. In contrast to the document centric approach the MBSE has been developed into a useful tool for engineers managing design and construction of large intricate devices such as airplanes and other large complex machines.

2.1. Modelling Language

A prerequisite for a successful implementation of a SE platform and models is the choice of an appropriate and standardized modelling language. The System Modelling Language - SysML – is a general-purpose graphical modelling language, see e.g. [1] and [3], that supports the concept and practice of model based system engineering. SysML is derived from the Unified Modelling Language - UML -, where parts of UML are reused with some extensions to build the SysML semantics and structure. SysML uses nine different diagrams in order to describe and specify system requirements, system structure, functional behaviours and specifications during the facility lifecycle, see Figure 1.
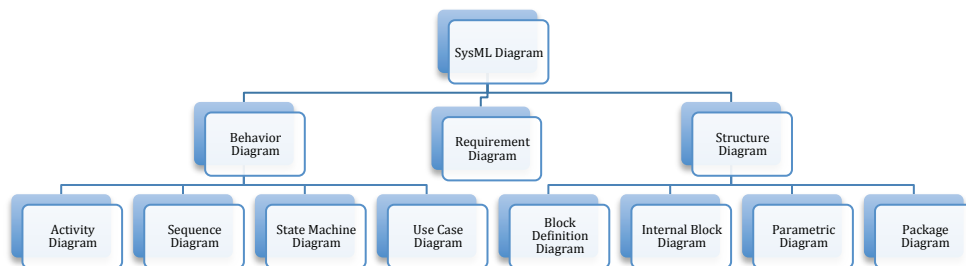

Figure 1. SysML standard diagrams

The diagrams comprise data used for the overall definition of a system. In order to achieve a comprehensive understanding of a system, diagrams from all main branches (behaviour, requirement, and structure) have to be completed.

2.2. System Engineering Platform

The MBSE activities at ESS will rely on efficient communication with stakeholders of the product descriptions (interfaces and internal structure), management of traceability links with their logic and the behavioural descriptions of the product. Utilization of the SysML language will allow development of parametric models of the ESS facility that will enable trade studies and simulation of change propagation in support of the configuration management.

The SE models for ESS will embed several of the diagram types described above, mainly block definition diagrams, internal block diagrams, use cases, requirements, activity diagrams and parametric diagrams. These diagrams will describe the ESS facility with its systems and sub-systems including interfaces and will create a structural and behavioural model. This model or part of this model will be instantiated to define a baseline and alternative baselines to support the trade studies and change impact assessment. Complete traceability between items will be developed through the SysML model including links to documents such as requirement rationale. Instantiated baselines will rely on a set of data stored in csv tables. A software package will support the development of the ESS SysML model. This software package will be capable of:

- Generating traceability matrices
- Extracting any part of the model into reports
- Exporting the model in an html format that will include a navigator feature

Changing the stored values in the csv tables will be possible through the SysML model. The Figure 2 depicts the architecture of the SE platform for ESS.
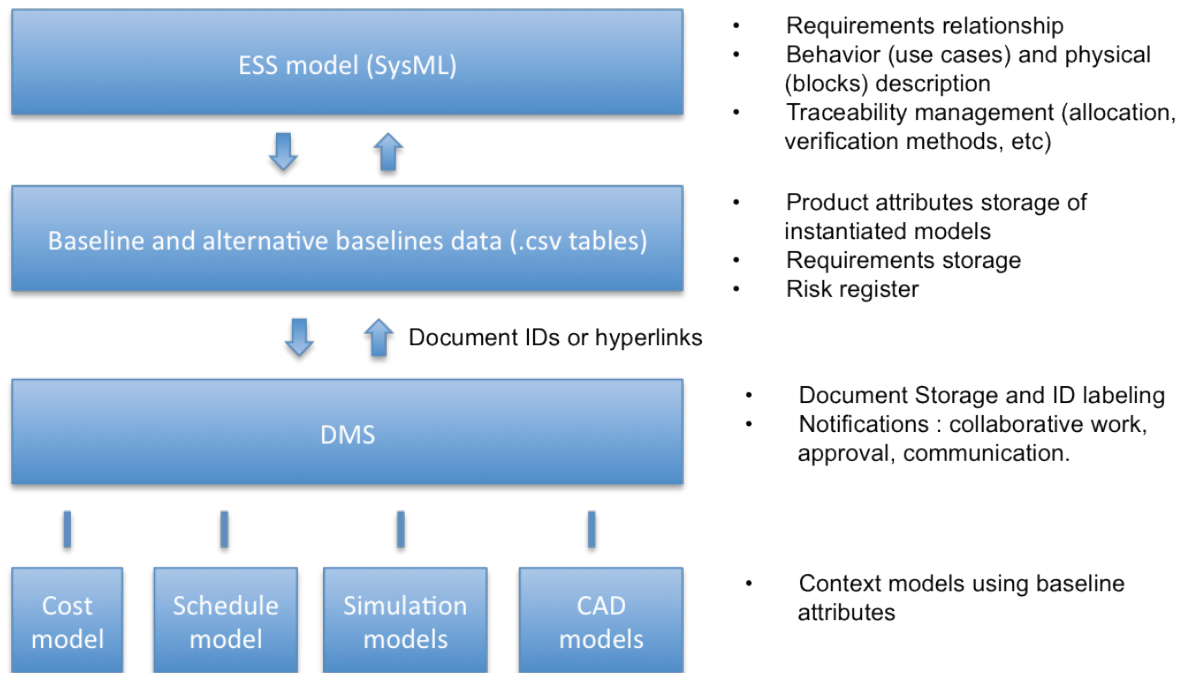

Figure 2: Architecture of the SE platform for ESS

3. Processes
Since MBSE is identified as an important contribution to the pre-construction and construction phase of the ESS project, key SE functions will be developed in order to set the framework and support to the global project. All activities associated to MBSE are developed in parallel in order for each activity to refine the others. The construction phase will require monitoring of the technical tasks and the maintenance of the traceability for an efficient management of non-compliance and change requests.

The SE models for ESS will be organized and developed top-down. In Table 1 a levels of a break down structure is exemplified. Iterative processes will for each system and sub-system provide the mechanism for specifying the lowest level seen appropriate. This will be repeated and retained for each project phase or state of the facility and for each system lifecycle.

Table 1. Break down structure

Engineering Levels	Area or System of Interest
Level 1	ESS Programme
Level 2	ESS part (example: Target Station)
Level 3	System of a Part
Level 4	Subsystem of a System
Level n	(n)-subsystem of a ($n-1$)-subsystem

The progressive work with the different diagrams of the SE platform should be synchronized meaning that the top-level requirements, operation and architecture should be elaborated in parallel. Nevertheless the foundation should always be the system requirements. The development process is copied for the next level in the break down tree and then further detailed in a hierarchal order. The procedures and methodology include a possibility to iterate with higher levels in order to refine the model and make the platform more robust.

The SE elements identified for the construction of ESS are listed and explained in the following sub-chapters. All three main diagram types, see Figure 1, have to be incorporated in a system description in order to get a coherent and redundant view of the system.

3.1. Requirement Identification and Management

The working group for requirements will be responsible for assigning the requirement owner for each separately identified system. The requirement owner will be responsible for specifying the method of verification for fulfilment of the specified requirement. All ESS requirements will be divided in three different categories that have a strong relationship to each other. The requirement tree has to be properly built in order to provide necessary traceability and change request management. The requirements could be categorized using three different stereotypes, these are:

- Functional Requirement – What is performed by the system
- Constraint Requirement – Limitation of the range of acceptable system function solutions
- Performance Requirement – Is a statement of need, something that the stakeholders want.

The performance requirement shall quantify the satisfaction level of a function or constraint requirement, for example the fulfilment of a temperature range between 300 and 350 K or fulfil the compliance of an ISO standard.

The management of the requirements is handled with a database that is a part of the ESS platform hosting the ESS virtual model. In order to manage the inputs and interfaces between different database instances, each requirement should be described and specified in accordance with the following inputs:

- Owner
- Parent Requirement
- Child Requirement
- Reference or Rationale
- Verification Method
- Risk
- Verification Results
- Flexibility or Status

In order to validate the consistency of a described requirement, the architecture and operations concept has to satisfy the requirement. The information flow from a parent requirement to a child requirement in the requirement tree also has to fulfil the top-level requirement in order not to loose information or to have information without constraints.

3.2. Behaviour and Operation Concepts

The operations concept or the behaviour of the systems will be defined using different behavioural diagrams as for example use cases and state machine diagrams. The behavioural diagrams will define how a system is used and by whom. Requirements are often related to the behavioural diagrams in order to describe the functionality of a system. The behavioural diagrams are refined and updated throughout the requirements and architectural design processes and will be a part of the ESS operation plan.

Within the operation concepts, the state machine diagram will describe the system lifecycle. Depending on the function of the system these diagrams will look different but nominally the lifecycle of the facility should be seen as in Figure 3.
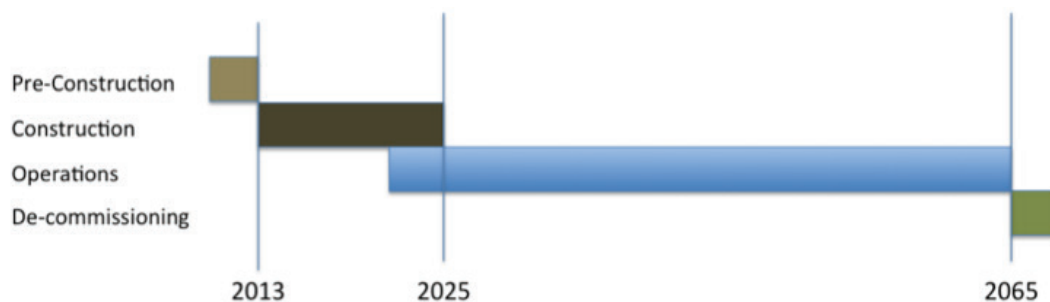

Figure 3. ESS Life Cycle

3.3. Architecture and Design Synthesis

The architecture and design is developed in parallel to the operation concepts and requirement flow-downs. The activity will decompose the ESS facility in systems and the systems in sub-systems. Interfaces between the elements will be addressed as flows consisting of the following types:

- Matter
- Energy
- Data
- Force

Parametric models will describe the dependencies of the flows through the element hierarchy. This parameterization is a prerequisite for the subsequent configuration control. The architecture will be built using block definition diagrams and internal block diagrams with a large number of different attributes to the above described flow types. The flow from one block to another will be depictured as an interface between structures. As an example of a block definition diagram, the top level ESS facility is described with in-flow and out-flow ports in Figure 4. The internal block diagram in Figure 5 describes the flow of knowledge, protons and neutrons within the ESS facility. Here it can also be seen that the main parts of the facility are the conventional facility, control systems, accelerator, target including beam dump and scientific instruments. The block definition diagrams and internal block diagrams are evolved in accordance to Table 1.
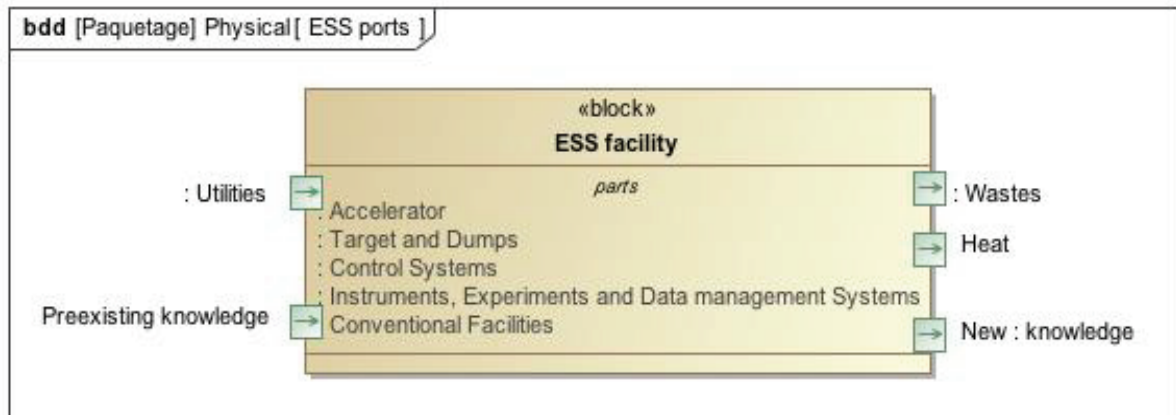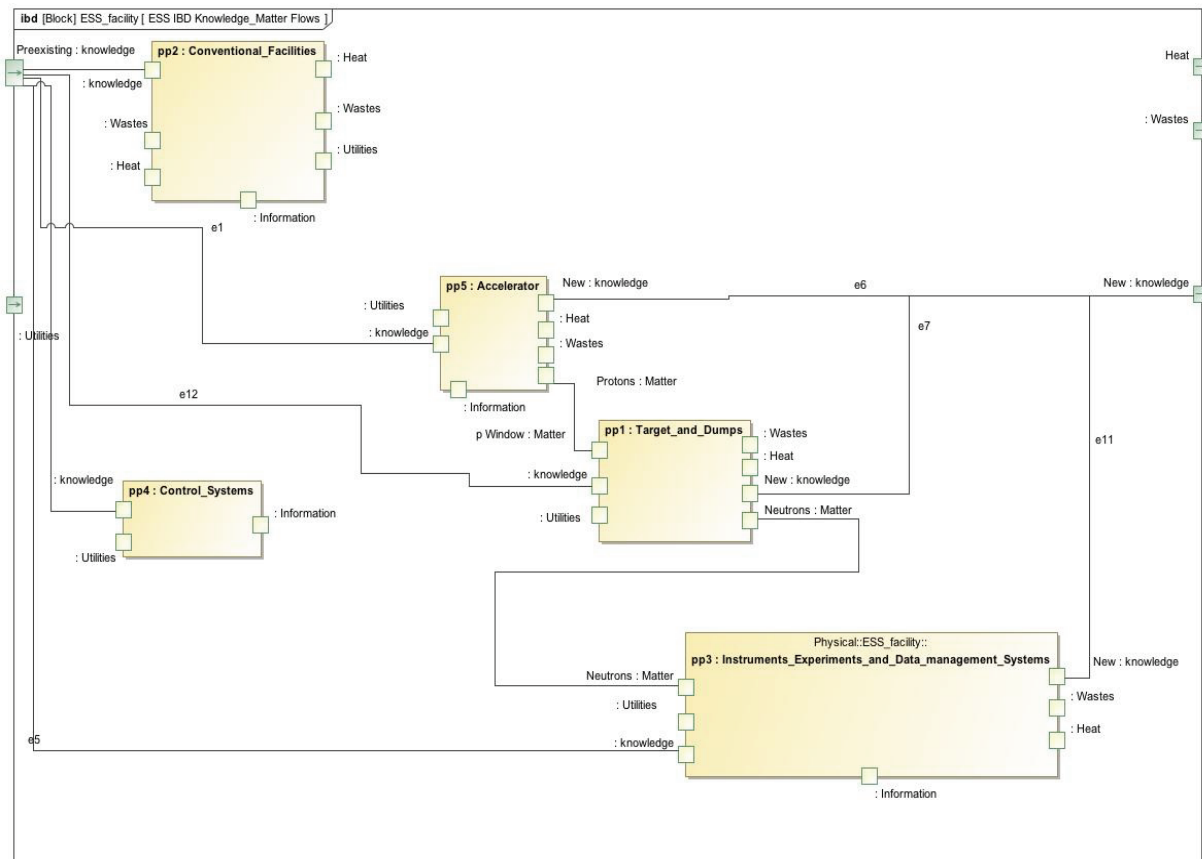
Figure 4. ESS Top Level Block Definition Diagram

Figure 5. Knowledge, proton and neutron flow

4. Integration and Verification

In order to verify that the produced model is correct, coherent and reliable, the system design and implementation is verified versus the requirement. ESS will use four different tools to verify the compliance of a system [2]:

- Inspection – Examination of the system together with its eventual compliance documentation (certification). A similarity is expected in this category.
- Analysis – Analytical data or simulations that show the theoretical compliance.
- Demonstration – A qualitative exhibition of the expected functions. For example, the commissioning of the accelerator is a demonstration.
- Test – Control via a simulation and/or dedicated test equipment. In the last case, the test may add the performance verification to the function verification satisfied by the demonstration.

Different verification methods are used during different phases of the project. Analysis and mock-up or test stand tests will be done mainly during the pre-construction phase. Inspections, demonstration and verification tests will be considered mostly during the construction phase.

In addition to the verification methods, an important part of the integration is to rank the requirements for risk and flexibility. These entities will have an impact on lower level requirements and thus focus and development efforts can be concentrated to specific identified systems.

The integration process should confirm that all boundaries between system elements have been correctly identified and described. This includes physical, logical and man-machine interfaces and interactions. The verification of the integration process should also confirm that all functional, performance and design requirements/constraints are satisfied.

5. Conclusions

To the knowledge of the authors, this is the first time MBSE is used as one of the tools to design, build and use a large research facility as ESS. MBSE is chosen in favour of the classical document-based approach because it provides a more rigorous means for capturing and integrating system requirements, design, analyses and verification information. The information is also covered for the complete life cycle of the system. The main benefits of using MBSE according to [1] are:

- Enhanced Communication
- Reduced Development Risk
- Improved Quality
- Increased Productivity
- Enhanced Knowledge Transfer

ESS will use SysML as the language for implementing the MBSE. SysML does not in itself provide any guidance or systematic approach of how to implement the working approach. Hence a well-defined methodology and guideline of how MBSE should be incorporated within the ESS project is currently developed, i.e. the ESS System Engineering Management Plan. The fixed top-level requirements for the ESS facility will be incorporated in the ESS MBSE platform during the design update phase of the ESS project.

References
[1]   S. Fridenthal, A. Moore, R. Steiner, *A Practical Guide to SysML*, Morgan Kaufmann Publishers, 2008
[2]   R. Duperrier, *Systems Engineering Management Plan*, ESS AB; to be published.
[3]   http://omgsysml.org/
[4]   R. Duperrier, *ESS System Architecture and Requirements,* ESS AB; to be published.