An iterative and recursive Model-based System of Systems Engineering (MBSoSE) approach for Product Development in the medical device domain

Ciancia, Pierfelice
Systems Engineering consultant
at FRIKART Engineering GmbH
Bern, Switzerland
pierfelice@frikart.ch
www.frikart.ch

***Abstract*—*In this paper, an iterative and recursive method for the application of a System of Systems (SoS) view to medical Product Development will be presented. Using a top-down approach, the method starts with the definition of business processes and leads to the definition of single systems architectures which satisfy requirements on a System of Systems level. A modeling technique based on the use of Business Process Model and Notation (BPMN) and System Modeling Language (SysML) will support the requirements engineer and the system architect during their design activities and all stakeholders to share information along the entire lifecycle. In conclusion, an example of application in the medical device domain will be presented.*

Keywords—Product Development; System of Systems (SoS); Systems Engineering; MBSE; iteration; recursion; top-down; business processes; BPMN; SysML; medical devices; healthcare

I. Introduction

The added value of using Model-based Systems Engineering (MBSE) techniques along the entire lifecycle of complex systems has been already discussed in many works ([1] [2] [3] [4] [5]). Some MBSE methodologies have been evaluated in an interesting work where differences between different approaches have been clearly shown [6]. A top-down analysis is common in some of those methodologies, but not all of them place emphasis on development of business processes or link those processes in a clear and defined way to further development steps. IBM Telelogic Harmony-SE [7] is a good V-model based approach where the model/requirements repository directly support the different SE processes. It uses a "service request-driven" modeling approach. INCOSE OOSEM [8] is a valid top-down approach where stakeholder needs are analyzed and the main SE processes are covered following an iterative approach. Vitech's MBSE methodology [9] is based on a "Onion model" which leads the development of a system on different level of abstractions with different level of details. The approach presented in this paper has been defined while setting up the Product Development activities for a System of Systems in the medical device domain. Inputs coming from different MBSE methodologies have been considered during the definition.

According to the International Council on Systems Engineering (INCOSE) definition, a System of Systems (SoS) is itself a System whose elements are further systems [10]. Applying MBSE to a SoS means to build up a model which covers multiple levels of abstraction. The top level is the SoS level, where the scope and the main business processes are defined. On a system level, single systems inherit SoS interfaces requirements and high level functionalities allocation. On a system element level, system requirements are allocated to single elements. The SoS hierarchy used in this paper, according to the INCOSE definition, is shown in Figure 1.

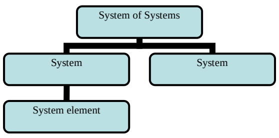

Figure 1 – Hierarchy within a System of Systems

The level of detail for information at each level of abstraction can be adjusted to the organization(s) needs. A SoS is characterized by systems that are managerially and/or operationally independent. Usually, different development teams work on their definition.

It is possible to describe a complex environment made of systems interconnected between them, in the same organization portfolio, as a SoS. Potentially, once that a framework has been defined, systems (made of one or more products) can be added or removed from the SoS. Risk analysis (especially in the medical device domain) and change management play a fundamental role every time a system is added or removed. Different system versions can be considered as different systems, if they coexist in the same SoS view. The boundary of a system or a SoS is fixed by the organization itself with the help of a systems engineer.

The approach presented in this paper is based on recursion applied to a “single system” top-down approach which starts with the definition of high level business requirements and goes on with the system functional analysis to the definition of a system architecture. The main steps of this single system approach are shown in Figure 2.

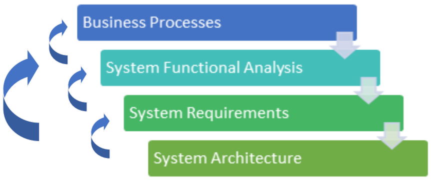

Figure 2 - Main steps of an iterative top-down approach for single systems

This approach can be applied to different domains and represents a systematic way to define requirements and to evaluate the effect of their changes on different levels of abstraction. The philosophy behind aims to use one model as a central source of information for all stakeholders. One of the scope of this paper is to show how business analysts and systems developers can work together to deal with challenges coming from an always more complex world.

In Chapter 2, a detailed description of those steps is given. In Chapter 3, the same approach is extended to a System of Systems. In Chapter 4, an example of application in the medical device domain is shown.

II. “Single system” approach

The top-down approach presented in this paper is characterized by iteration. Iteration steps and duration are tightly organization depending. People from the organization should define together the level of iteration. At this purpose, methodologies like agile or lean could be integrated to the approach presented in this paper.

One of the main scopes of this work is to highlight how close business analysts and system developers should work while defining scope and requirements of complex systems. Clarity of information shared between system stakeholders is directly dependent on the complexity of a system. The use of a systematic approach and a structured model is a good starting point to the development of successful systems.

A. Business Processes

Business processes describe interactions and shared information between actors on a business level. At this level, stakeholders for the System of Interest (SOI) are defined and requirements are described as activities that need to be carried out to satisfy business requirements. The scope of describing business processes is to focus on what are the real needs and which of those needs shall be satisfied using the SOI. Additional requirements (functional and non-functional) can be directly defined in the business processes and they are part of the model. Usually, information concerning business processes are gathered by interviews to the stakeholders and/or by meetings and/or workshops. Examples of business processes in the healthcare industry are interactions between patients and healthcare professionals (e.g. MDs). Other processes existing in many different industries are maintenance, logistics, manufacturing, customer care, sales, etc. In Figure 3, an example of model-based business process description is shown.

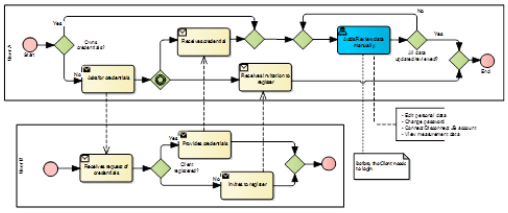

Figure 3 - Example of model-based businesses process description

While describing business processes, the involved stakeholders only focus on “what” needs to be done and not on “how”. A good technique to model business processes is to define activities performed using the SOI in a way that single use cases can be directly derived from them.

B. System functional analysis

Once that some business processes are already defined, a list of use cases can be traced towards them (example on Figure 4). Using the defined business process as input, a systematic approach for the definition of a complete list of use cases is guaranteed. A good functional analysis can be performed by the description of single use cases in a graphical way. Using behavioral diagrams is possible to define main functionalities that need to be developed by a black-box description. A lower level functionalities definition can be performed by a white-box description of the use cases, where the main system parts can be defined.

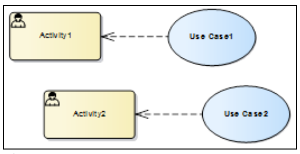

Figure 4 - Example of traceability between use cases and business activities

While performing a white-box analysis, some functionalities are allocated to system elements. That is a first step for definition of a system architecture.

C. System requirements

In a model-based SE perspective, the main concept is to have a shared model between stakeholders to collect all requirements. Different stakeholders have different needs and provide different feedback. The system modeler is usually the one responsible for keeping the model up-to-date. Ideally, everyone should access the model to get the needed information and all changes must be evaluated in terms of impact to other parts of the system.

In many industries, document-based requirements management is still an important issue. For instance, in the medical devices domain documents still play a fundamental role in regulatory matter. Using a model, it is possible to collect all needed information and to export them as documents in the preferred format. Time saving and enhanced change management are only two of the benefits coming from the application of MBSE.

D. *System architecture*

During system functional analysis, defined functionalities and, then, non-functional requirements are already allocated to system elements. Some additional requirements come from the development team, where specialists provide their feedback. A functional architecture provides a full description of the system in terms of main functionalities. Some of these functionalities need a further breakdown. Then, a physical architecture is essential for the definition of the interfaces between system elements and to the external world and for the development of the system.

III. System of Systems approach

The iterative top-down approach described in Chapter 2 can be extended and used on different level of abstractions. A SoS consists of different level of abstractions. At the top level, the "big picture" is defined and it leads the development of single systems which are located on a lower level of abstractions.

While considering to apply the approach to a SoS, the described steps can be used in a recursive way to define a general SoS architecture and single system architectures. This concept is expressed in a graphical way in Figure 5.

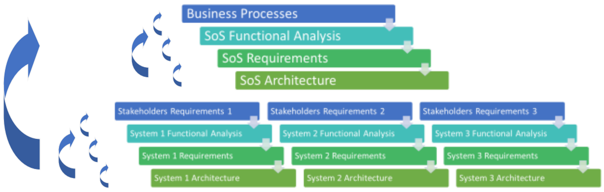

Figure 5 - Main steps of an iterative and recursive top-down approach for a System of Systems (SoS)

On a SoS level, the focus is on interactions between systems in the SoS, interfaces definition (including communication protocols) and allocation of high level functionalities. Further concepts related to the communication (e.g. privacy, security, etc.) can be developed at this level. On a system level, requirements coming from the upper level are inherited during the single systems analysis, to define system architectures which shall satisfy those requirements.

A. *System of Systems level*

As a first step, the involved stakeholders define their own business processes. The business analyst builds up a business model. During this phase, business processes can be adapted and perfected in agreement with the stakeholder needs. Business processes could also include "journeys" or "stories" while describing operational processes. For instance, "personas" definitions for therapy processes are common in the medical devices industry. Ideally, the analyst models solution-independent processes, focusing on the information shared between the actors and avoiding to give too detailed information concerning used technologies. Otherwise, this last practice would add constraints to the definition of a system(s) solution.

Some of the use cases derived from the business processes involve the use of more than one system in the SoS. These use cases, which could be called "SoS use cases", should be described to define the interfaces between the systems and communication protocols.

Additionally, an allocation of functionalities can be performed at this level too. In an early phase, when the SoS is not yet completely defined, different alternatives of same use cases can be described and evaluated and a tradeoff analysis can be performed. The use cases are traced towards the activities in the business processes. At this level, a white-box description would be represented by several messages shared between the systems in the SoS. Further descriptions should be postponed and completed on a system level.

The SoS architecture simply includes the systems and their interfaces. A tradeoff analysis can help to choose the right architecture between different candidates.

B. *System level*

On a system level, the first requirements which are inherited from the upper level are the interface requirements. Each system in the SoS needs to satisfy them, to be able to communicate with the other systems and to carry on those activities which are requested by the business processes.

Each system owns its additional stakeholder requirements and, in some cases, it could even own specific business requirements. Those requirements can be added to the model on a system level and shall be satisfied by related system requirements as well. Some of these stakeholder requirements could lead to the definition of new use cases which do not affect the other systems. A system functional analysis leads to the definition of functionalities which are allocated to single system elements and to the creation of a functional architecture. A physical architecture including interfaces between system elements is the last step at this level.

IV. Example of Application

An added value to the described approach comes from the use of a good organized model, which allows to share information between the stakeholders in a clear and systematic way.

A model can be adapted to the presented methodology by creating the right links between its objects. The example in this chapter concerns a project in the medical device domain. The involved organization aims to develop independent systems and to connect them to other existing systems to satisfy new business requirements. Single independent products of the company portfolio have been classified as "single systems" and the overall picture looking at the interactions between them has been defined as the SoS view. The importance of considering single products as separate systems in the medical device domain comes mainly because of regulatory reasons. Each system needs to be developed as a single entity following the right product classification and regulations. At the same time, while talking about interconnected and interdependent devices, a complete technical overview from a high-level perspective is indispensable.

In this case study, all systems follow business and marketing requirements which are specific for each of the products. Different project managers and development teams work on their definition in an independent way. Each of the systems need to be classified and to be compliant with different regulatory standards. In some cases, specific system elements are treated from marketing and regulatory perspectives as single products. In this context, the model plays a fundamental role in requirements allocation and management and it gives a shared view between the systems during the development process. Furthermore, the model structure needs to be compliant with the systems engineering methodology and the internal people organization. Some of the documents needed for regulation, risk analysis and other topics are printed out directly from the model. Figure 6 shows the concept behind the structure of the model used in this case study.

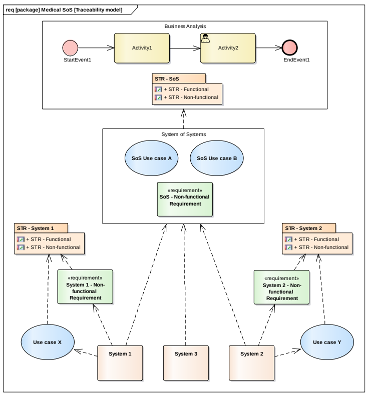

Figure 6 - Example of dependencies between objects in a model

All links between the objects in the picture are generic dependencies and aim to show the traceability paths. A complete meta-model has been defined as well, but it is not shown in this paper.

At the top level in the model, the business analysis leads the development of the SoS. The described processes include therapy processes (interactions between patients and healthcare professionals), which are described starting from the definition of different “personas”. The organization prioritized the description of some processes. Use cases are defined directly from business activities analysis. The ones involving more than one system are classified as “SoS use cases” and lead the team to the definition of interface requirements and communication protocols between some systems. But some high-level functionalities are allocated to the systems at this level as well. Figure 7 shows the SoS architecture of the case study.

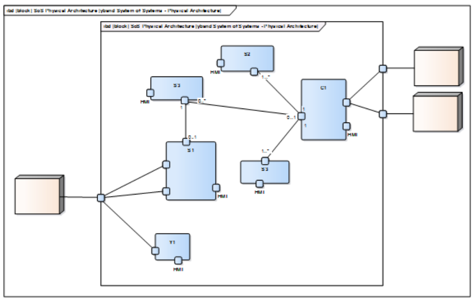
Figure 7 - System of Systems Architecture

The systems include insulin pumps, software systems, clouds and other medical devices. The interfaces between the systems are described as communication protocols and are also included in the model. Complete traceability between interfaces and protocol messages is guaranteed. The protocols are defined on a SoS level. When new systems are incorporated into the SoS, communication requirements need to be satisfied.

For each of the systems, a project leader and a development team are involved. Each team independently carries out a stakeholder requirements analysis for its own project. Specific use cases are defined for the single systems and non-functional requirements as well. An example of use case description is shown in Figure 8. On a lower level, for each of the systems all functionalities coming from the SoS use case and from the single stakeholder analyses are allocated to the system elements.

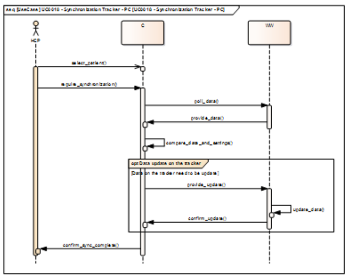
Figure 8 - Dynamic description of a use case

All the involved systems are described by a system model which is linked to the main high-level SoS model. Each team works on its own model having access directly to the shared database. Management and business analyst can also have access to the model to work on business requirements and to access to lower level requirements as well.

Risk analysis and safety concepts are applicable to different level of abstractions in the model. On a system level, medical devices are registered and compliant with applicable standards. On a SoS level, the overall privacy, security and safety concepts are evaluated as well.

V. Conclusions

An iterative and recursive top-down approach for the application of a System of Systems view to medical Product Development has been described. The definition of this approach has been carried out together with managers and developers within an existing Product Development environment and its use has been extended to other organizations. To define the approach, different MBSE methodologies have been evaluated. Inputs coming from those methodologies have influenced the definition of a new methodology which better fits the need of the involved organization. This systematic approach involves business and systems analysis and leads to the definition of systems architectures that satisfy given business requirements. The business analysis is on a SoS level and leads the development of new services and systems. High-level functionalities and interface requirements between single systems are defined. Furthermore, the model is used as input for risk analysis on both SoS and system levels.

REFERENCES

[1] INCOSE Systems Engineering Vision 2020, Sep 2007

[2] Z9: What is Model Based Systems Engineering, Hazel Woodcock, Jan 2012

[3] A primer for MBSE – 2nd edition, D. Long, Z.Scott, Oct 2011

[4] Model Based Systems Engineering: Fundamentals and Methods, P. Micouin, Oct 2014

[5] INCOSE Systems Engineering Vision 2025, Oct 2014

[6] Survey of Model-Based Systems Engineering (MBSE) Methodologies, J. A. Estefan, JPL California, May 2008

[7] Systems Engineering Best Practices with the Rational Solution for Systems and Software Engineering, H. Hoffman, IBM, Feb 2011

[8] A Practical Guide to SysML: The Systems Modeling Language, Friedenthal, Moore, Steiner, 2008

[9] Systems Engineering (SE) 101, Long, James, Vitech Corp, 2000

[10] INCOSE Systems Engineering Handbook – 4th edition