An Approach for System of Systems Requirements Management

Renata M. de Lima, Lisandra M. Fontoura
Programa de Pós-Graduação em Ciência da Computação, UFSM. Santa Maria, RS, Brazil
{rlima, lisandra}@inf.ufsm.br

***Abstract*—Systems of Systems (SoS) are becoming more common in our society due to the diversity of stakeholders, heterogeneous technology, ubiquity, or large-scale networks involved in the development of software intensive systems. Consequently, there is an increasing interest in closing the gaps still existing in the System of Systems Engineering field, which is in constant growth due to its complexity. Most of these gaps are related to the SoS requirements management, the lack of well-defined and standardized methods and techniques for Requirements Engineering for SoS makes difficult the development and the evolution of the overall SoS. In this paper we propose an approach to requirements management for SoS through a process capable of organizing the work among stakeholders and managing changes in requirements, thus contributing to a more adaptable evolutionary development of the SoS. The results of a conformity assessment show that our approach is in conformity with the most of international standards related to Systems Engineering and Requirements Engineering.**

Keywords-System of Systems; SoS; Capability Objectives; Requirements Management; Requirements Engineering.

I. Introduction

In different application domains, it is possible to find different systems working together to satisfy a specific goal. This set of Constituent Systems (CS) collaborating with each other establish a System of Systems (SoS). SoS is "a set of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities" [4]. The structural differences and the diversity of stakeholders in a SoS development lead to complexities and challenges unexplored by the SE [7].

Several studies indicate that there is a real need for a requirements management that can facilitate the collaborative work among the stakeholders and that can support the evolution of SoS in a flexible and adaptable way [7]. A well-defined management process for SoS requirements would be capable to clarify the management relationships among CS and SoS stakeholders, facilitating the collaboration among them in the life cycle of the SoS development. Thus, more consistent management options would also facilitate the Requirements Engineering for SoS activities, ensuring a more adaptable SoS development. In this paper, we present an approach for SoS requirements management, which aims to organize the work among the stakeholders and the activities of the RESoS.

II. Process Description

The proposed approach consists of a Requirements Management Process for SoS (RMP-SoS) organized in three activities: Manage Stakeholders, Elicit SoS Requirements and Manage Changes & Validation. The process can be implemented through the production of the documentation using templates and following the workflow (illustrated in Figure 1). The RMP-SoS is more adequate to be applied in Directed SoS, because this type is built with a specific purpose, so even though the CS can operate independently, they are subordinated to a central managed purpose [8].

A. Activity 1: Manage Stakeholders

The first group of tasks aims to identify who are involved in the RE activities for SoS, as well as their abilities, knowledge and assign roles and responsibilities to the stakeholders, develop a work plan for the next activities, and facilitate the agreements among stakeholders.

1) Identify Stakeholders: This task has the main goal to identify the main people who are involved in the requirements activities related to the SoS and to the CS and describe their abilities and knowledge. The responsible for this task is the SoS Requirements Manager and the document generated is the Stakeholders Register.

2) Assign Roles and Responsibilities: The purpose of this task is to distribute roles and responsibilities among the registered stakeholders. Roles and responsibilities may vary over the SoS evolutionary development. The responsible for this task is the SoS Requirements Manager, and the artifact created is the Stakeholders Roles and Responsibilities.

3) Establish Collaboration Agreements: This task aims to generate documents that express agreements and the level of collaboration among stakeholders from CS and SoS involved in the RE activities. In some ER tasks, it is essential that the SoS requirements engineer get involved with the CS requirements engineer to understand the nature of the CS changes and evaluate impacts on SoS [4]. To meet this need, it is necessary that agreements about sharing data and information be clearly established. The SoS Requirements Manager is responsible for this task and produce the Collaboration Agreements.

4) Prepare RE Work Plan: In order to prepare and authorize the beginning of the RE activities, it is necessary to provide a summary of the work that will be done. It is needed to define the SoS type, specify the project (SoS) scope and context, the work purpose for this project phase, the constraints, the requirements traceability policies, as well as the list of stakeholders and their responsibilities [1]. The SoS Requirements Engineer is responsible for the RE Work Plan.

DOI reference number: 10.18293/SEKE2018-178

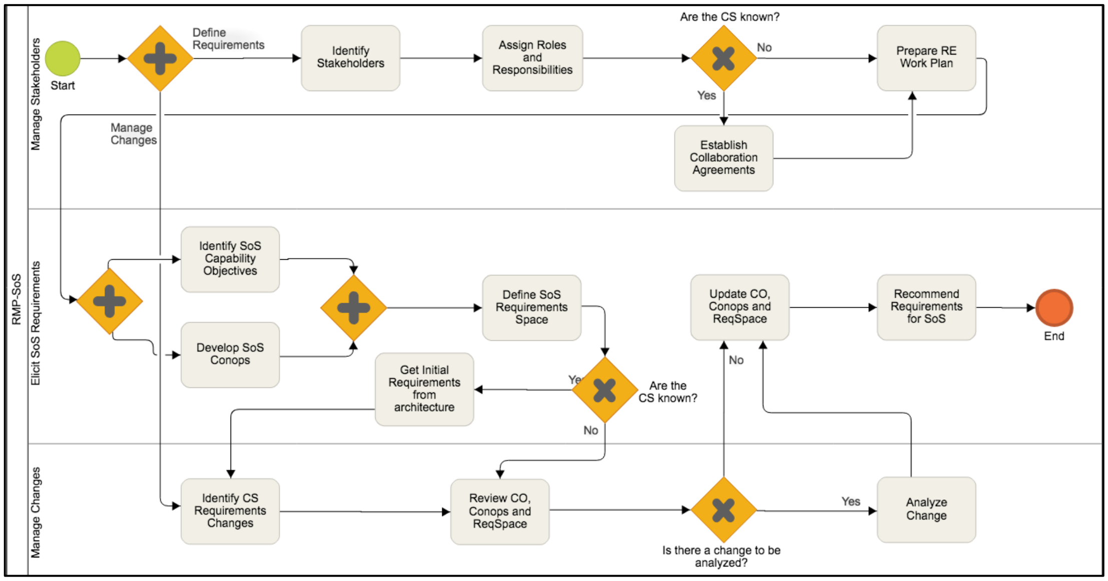
RMP-SoS
Manage Stakeholders
Start
Define Requirements
Identify Stakeholders
Assign Roles and Responsibilities
Are the CS known?
No
Prepare RE Work Plan
Yes
Manage Changes
Establish Collaboration Agreements
Elicit SoS Requirements
Identify SoS Capability Objectives
Develop SoS Conops
Define SoS Requirements Space
Update CO, Conops and ReqSpace
Recommend Requirements for SoS
End
Get Initial Requirements from architecture
Ye
Are the CS known?
No
No
Manage Changes
Identify CS Requirements Changes
Review CO, Conops and ReqSpace
Yes
Analyze Change
Is there a change to be analyzed?

Figure 1 - BPMN Diagram - Workflow of the RMP-SoS

B. *Activity 2: Elict SoS Requirements:*

This second activity has the purpose to organize all the tasks for RESoS in order to generate a set of requirements ready for incorporation to a future functional baseline for SoS.

1) Identify SoS Capability Objectives (SoS CO): It aims to identify the capability objectives for SoS, describing in the users language primary and secondary missions, the operational constraints, and a general vision of the operational environment [1]. The responsible is the SoS Requirements Engineer, and the artifact is the SoS Capability Objectives.

2) Develop SoS Concept Of Operation (SoS Conops): This task has the goal to describe the way the SoS functions works by the user's point of view [3]. SoS Conops document is developed by the SoS Requirements Engineering with actively participation of the SoS Users, who will describe the way they are planning to use the SoS to meet their objectives.

3) Define SoS Requirements Space (SoS ReqSpace): In order to limit the scope of SoS users primary needs and define functionalities to promote the capabilities. And considering the users environment variability, which impacts the way those functionalities will be executed; the SoS Engineering team develop the SoS Requirements Space document [2]. The responsible for this task is the SoS Requirements Engineer.

4) Get Initial Requirements from Architecture: This task has the goal to obtain the needed requirements to provide a structure that allows changes in the SoS functionalities over its evolution. SoS architecture is itself a requirements generator, because when SoS engineers develop an architecture, they need to overlap the CS in a structured way to make them work together and share data. However, sometimes this structure can be different from the original design of the CS.

Hence, some changes in the CS may be necessary to support the SoS architecture [4]. In addition, after an architectural construction or simulation, if the designer has applied a good strategy, it is possible to identify and predict emergent behaviors, which will be interesting for the definition of requirements. SoS Requirements Engineer is the responsible, with intensive collaboration of the CS Requirement Engineers. The artifact is known as SoS Initial Architecture Requirements.

5) Update SoS CO, SoS Conops and SoS ReqSpace: This task has the main goal to update the documents based on the corrections recommended by the Review Record, or include/adapt an alteration approved by the Change Record from the task Review SoS CO, SoS Conops and SoS ReqSpace. The responsible is the SoS Requirements Engineer, and the artifacts are SoS Capability Objectives, SoS Conops and SoS Requirements Space documents updated.

6) Recommend Requirements for SoS: This task aims to define the requirements that will be addressed in the next SoS increment. The CS Requirements engineers work together with the SoS Requirements Manager and Engineer to prioritize and recommend relevant requirements for each iteration and produce the SoS Recommended Requirements. Then, the selected requirements are moved to the functional baseline to be treated and addressed by the activities associated to the Core "Addressing Requirements & Solution Options".

C. *Activity 3: Manage Changes*

The tasks of this activity aim to review the documents for validation and analyze problems or change requests.

1) Identify CS Requirements Changes:

To identify changes in the CS requirements, SoS RE team needs to connect to the process that manage the CS requirements, and identify the changes that are capable of affecting its capability objectives. The evolution of SoS is directly related to the evolution of its CS [4]. Therefore, it is important to keep a record of the changes and evolution occurring in the CS and their impact on SoS. The SoS Requirements Engineer and SC Requirements Engineers perform this task. The document generated is CS Information which impacts SoS Requirements.

2) Review SoS CO, SoS Conops and SoS ReqSpace:

Meetings among key stakeholders are performed to review and validate the SoS Capability Objectives, the SoS Conops, and SoS Requirements Space. All the key stakeholders may participate, but the responsible is the SoS Requirements Engineer. This task can generate a Change Request that is a problem identified or a specific change purpose, which will be analyzed in the sequence. Also, it can generate a Review Record that points out necessary corrections to the documents.

3) Analyze Change Request

The task begins with a critical analysis of a problem identified by the Review SoS CO, SoS Conops and SoS ReqSpace, or of a specific proposal of change. The Change Control Board performs an analysis of the problem or proposal in order to verify its validity. If it is valid, the effect and impact in terms of cost for the project is evaluated. Once the analysis is done, the decision about to implement the change is taken. The Change Control Board is responsible for generating a Change Record or a new Change Request.

III. Conformity Assessment

A conformity assessment is a demonstration that specific requirements related to a product, process, system, person or organizations are attended. This section summarizes the assessment conducted to identify where the RMP-SoS is in conformity with the international standards ISO|IEC|IEEE 15288 [8] and ISO|IEC|IEEE 29148 [5].

ISO|IEC|IEEE 15288 establishes a common framework of process descriptions for describing the life cycle of systems created by humans, defining a set of processes and associated terminology. These processes can be applied at any level of hierarchy of a system's structure. Its Annex G brings an application of these system life cycle processes to a SoS.

The Agreement Processes (G.3.2) says "they are crucial for SoS because they establish the modes of developmental and operational control among the organizations responsible for the SoS and the independent CS". The RMP-SoS has an Establish Collaboration Agreements task, which aims generate collaboration agreements to express how stakeholders will collaborate with each other in a way that the operational management independence of the CS is maintained.

Organizational project-enabling processes (G.3.3), "in a SoS, the owners of a CS usually retain responsibility for engineering their systems as they each have their own processes. The SoS organization implement these processes for the SoS for those considerations that apply to the overall SoS". For example, RMP-SoS is concerned with this topic knowing all the stakeholders in the SoS requirements activities and assigning them with roles and responsibilities adequately through the tasks Identify Stakeholders and Assign Roles and Responsibilities respectively.

Technical Management Process (G.3.4) "is applied to the particular considerations of SoSE - planning, analyzing, organizing and integrating the capabilities of a mix of existing and new systems into a system of systems capability". Also, "the SoS organization must plan an integrated life cycle that recognizes the independent changes in the CS, in addition to the SoS-initiated changes in a life cycle that threats the SoS as the system-of-interest". In this context, we consider that RMP-SoS uses some artifacts produced across the life cycle and the evolution process of a SoS. For example, we consider the SoS Architecture as generator of requirements, even though its design is out of the RMP-SoS scope. Also, our proposed process can be integrated in a life cycle that considerate changes in the CS through the task Identify CS Requirements Changes.

Finally, Technical Processes (G.3.5) are concerned with technical actions throughout the life cycle", and some of them can be related to SoS. These technical processes are directly related to requirements and they are addressed in our tasks related to the RESoS in the Elicit SoS Requirements pool.

ISO|IEC|IEEE 29148 (2011) provides a unified treatment of the processes involved in RE throughout the life cycle of systems and software. For Stakeholder requirements definition process (6.2) "the purpose is to define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment". In this sense, RMP-SoS attends all the activities proposed, because we are able to Elicit Stakeholder Requirements (6.2.3.1) through the task Identify Stakeholders, where we can identify the individual stakeholders and their interest in the SoS throughout its life cycle. Also, through the task Identify SoS Capability Objectives we are able to elicit stakeholders requirements, that is the capability objectives, from the identified stakeholders and establish the traceability of the those requirements.

In order to Define stakeholder requirements (6.2.3.2) our process uses the task Develop SoS Conops, to "define a representative set of activity sequences to identify all required services that correspond to anticipated operational and supported scenarios and environments" and "identify the interaction between the users and the system", by means of describing the operational environment and the operational scenarios in the SoS operational users view.

The issues related to Analyze and maintain stakeholders requirements (6.2.3.3) are addressed in part by our task called Review SoS CO, SoS Conops and SoS ReqSpace, which reviews the documents generated by the tasks mentioned above and make sure that there is no conflicting, missing, incomplete, ambiguous, inconsistent or unverifiable requirements. If there is some problem, the task generates a Review Record or a Change Request in order to fix this.

In Requirements Analysis Processes (6.3) the purpose is "to transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services". RMP-SoS addresses this recommendation by the definition of the SoS Requirements Space. In order to Define system requirements (6.3.3.1), the task Define SoS Requirements Space aims to limit the primary needs of the stakeholders and define the functions to provide those needs achieving the traceability between capabilities and functionalities. Analyze and maintain system requirements (6.3.3.2) is also addressed in part by the task Review SoS CO, SoS Conops and SoS ReqSpace to ensure that the specified system requirements adequately reflect the capability objectives to address the stakeholders needs and expectations.

The standard also emphasizes that there are RE activities in other technical processes. In Requirements in architectural design (6.4.1), the Definition of the architecture (6.4.1.1) is out of the scope of the RMP-SoS. However, we perform the Analyze and evaluate the architecture (6.4.1.2) in a way that is possible to identify which integration requirements are missing by the tasks Get Requirements from Architecture and in part by the Identify CS Requirements Changes.

About Requirements in verification (6.4.2), the purpose is to confirm that the specified design requirements are fulfilled by the system. For SoS, these issues related to the verification should be done in execution time, and the activities are the responsibility of the processes related to the Core Elements "Assessing Performance to Capability Objectives" and "Addressing Requirements and Solution Options". The RMP-SoS only provides input for defining and conducting a verification plan based on requirements. The same assumption is valid for Requirements in validation (6.4.3).

Finally, the clause related to Requirements management (6.5) talks about the importance of "establish procedures for defining, controlling, and publishing the baseline requirements for all levels of the system-of-interest". Our RMP-SoS is totally about that, all the tasks pursue to identify, define, maintain and prioritize consistent requirements during the increments along the SoS life cycle.

About Change management (6.5.2) the standard says "change has to be managed by ensuring that proposed changes go through a defined impact assessment, review, and approval process, and by applying careful requirements tracing and version management". These issues, in the RMP-SoS are treated by the tasks Analyze Change and Update SoS CO, SoS Conops and SoS ReqSpace. The first one analyses and accesses the validity and the impact of the change in SoS requirements. And the second one implements the change if it was approved, or implement a correction pointed by the Review SoS CO, SoS Conops and SoS ReqSpace.

V. Discussion

Considering the conformity assessment presented, we assume that our proposed process for SoS requirements address the most of the recommendations suggested by the two International Standards related to its field. It is important to emphasize that there is no standard specific developed for SoSE, and that until now the unique way is the application of existing standards, such as ISO|IEC|IEEE 15288 [3].

In the case of ISO|IEC|IEEE 29148, we had to adapt some concepts in order to make them applicable for SoS. For instance, the stakeholder requirements mentioned can be considered our "capability objectives" in the SoS context. Some processes presented by this standard we understood that they are out of scope of this SoS requirements management. For example, Measurement for requirements (6.5.3) presents activities we understand that is part of the processes related to the Core Element Assessing Performance to Capability Objectives, even though RE could benefit from this process, it is not directly related to the tasks of the RMP-SoS.

In addition, the Information items (7), define three documents as part of the RE processes. However, in the SoS context these documents, even they are similar and have almost the same proposal, they have different names and different ways to be completed. For instance, the Stakeholder requirements specification document (StRS) would be our SoS Capability Objectives. In the same way, the System Requirements Specification document (SyRS) would be the union of our SoS Conops and SoS Requirements Space translated into the SoS Requirements document.

VI. Conclusion and Future Work

In this paper we presented a Requirements Management Process for System of Systems (RMP-SoS), which aims to organize the RE activities and the work among stakeholders, by integrating techniques and methods from project management and traditional requirements engineering, providing support for translating capability objectives and the definition of requirements and incorporating changes over the evolutionary development of SoS. As a result of a first conformity assessment, we understand that the process is in conformance with the international standards of systems life cycle processes and requirements engineering.

As future works we intend to improve templates to support the RMP-SoS documentation and develop an online tool to help the application of the process in real projects, which will facilitate the communication among stakeholders and ensure the geographical distribution of the CS is not a problem. After that, the process will be empirically evaluated in a real project.

VII. Acknowledgment

We gratefully acknowledge financial support from CAPES-Brazil, and the Brazilian Army through the SIS-ASTROS Project (813782/2014), developed in the context of the PrgEE-ASTROS 2020.

References

[1] ARNAUT, B. M., Ferrari, D. B., & e Souza, M. L. D. O. (2016, October). A requirements engineering and management process in concept phase of complex systems. In Systems Engineering (ISSE), 2016 IEEE International Symposium on (pp. 1-6). IEEE.

[2] DAHMANN, J., Rebovich, G., Lane, J. A., & Lowry, R. (2011). System engineering artifacts for SoS. IEEE Aerospace and Electronic Systems Magazine, 26(1), 22-28.

[3] DAHMANN, Judith; ROEDLER, Garry. Moving towards standardization for system of systems engineering. In: System of Systems Engineering Conference (SoSE), 2016 11th. IEEE, 2016. p. 1-6.

[4] DoD-USA (2008). Systems Engineering Guide for Systems of Systems. Washington, DC.

[5] International Organization for Standardization. 2011. ISO/IEC/IEEE 29148—Systems and Software Engineering—Life Cycle Processes – Requirements Engineering.

[6] International Organization for Standardization. 2015. ISO/IEC/IEEE 15288. Systems and Software Engineering. System Life Cycle Processes: Annex G Application of System Life Cycle Processes to SoS.

[7] LIMA, Renata M.; VARGAS, Daniel. FONTOURA, Lisandra M. System of System Requirements: A Systematic Literature Review using Snowballing. In: SEKE: Software Engineering and Knowledge Engineering. Pittsburgh, PA: KSI Research Inc. and Knowledge Systems Institute, 2017. v. 29. p. 97 – 100.

[8] MAIER, M. W. (1996, July). Architecting principles for systems-of-systems. In INCOSE International Symposium (Vol. 6, No. 1).