# Experience Report and Challenges for Systems-of-Systems Engineering: A Real Case in the Brazilian Defense Domain

**Carlos Eduardo de B. Paes**[1,2], **Valdemar Vicente Graciano Neto**[2,3,4], **Flávio Oquendo**[3], **Elisa Yumi Nakagawa**[2]

[1] Pontifical Catholic University of São Paulo
São Paulo – SP – Brazil

[2]ICMC – University of São Paulo
São Carlos – SP – Brazil

[3]IRISA-UMR CNRS – Université Bretagne Sud, Vannes, France.

[4]Instituto de Informática – Universidade Federal de Goiás (UFG)
Goiânia – GO – Brazil

carlosp@pucsp.br, valdemarneto@inf.ufg.br

flavio.oquendo@irisa.fr, elisa@icmc.usp.br

***Abstract.*** *Defense domain is a crucial branch of the government of any country since it ensures the national security and supremacy. Hence, it is imperative that the technologies adopted that support their operations must be aligned with the cutting-edge scientific research results and technologies. In particular, software systems play a crucial role in defense domain, since they are usually connected among themselves, forming alliances known as System-of-Systems (SoS). In the last decades, SoS conception has evolved and Brazil has also adopted SoS' development strategies. However, we identified a lack of reports communicating challenges faced and strategies adopted to carry out SoS Engineering (SoSE) in Brazil. In this direction, this paper reports experience, results, and challenges of a real project carried out in Brazilian Department of Defense, particularly in the Navy's context. We report the conception of an SoS named Blue Amazon Management System (SisGAAz), a SoS for Brazil's defense and maritime surveillance. We claim that this report contributes to SoSE since it offers a panorama of SoS conception in Brazil, representing, as a matter of fact, the state of the art in SoSE for Defense in our country, a paramount artifact for the advance of SoS research.*

## 1. Introduction

Software-intensive Systems-of-Systems (SoS)[1] are strategic. They have been developed and adopted as a first class citizen in a range of domains, particularly in defense domain. Many systems have been combined, supporting reliable and trustworthy operations performed by army, navy, and aeronautics, delivering complex functionalities by means of defense SoS, and exhibiting crucial emergent behaviors, such as, the defense of a country itself as a result of the interoperability among all those individual systems. In this sense,

[1]For sake of simplicity, SoS will be used interchangeably to express singular and plural.

SoS Engineering (SoSE) has become strategic, guiding how to develop constituent systems, and offering insights on how to interoperate them to accomplish complex missions.

For a long time, interoperable systems in the defense area were largely used as instances of SoS. However, a small focus has been given on reports of challenges, problems, and limitations identified during the SoSE development life cycle. Those lessons learned are important since they represent expertise knowledge and experience that could be shared to aid the conduction of other SoSE development projects, in military domain or even other civil domains. Such lessons can foster the development of SoSE research, exposing the gaps, challenges and difficulties, to point out directions to which the investigation efforts should be invested. Unfortunately, this knowledge is often restricted to project documents in defense departments, and, particularly, there is a lack of reports of SoSE projects, specially in Brazilian defense area.

In this direction, the main goal of this paper is presenting an overview on the current practice on SoSE in Brazil, externalizing the challenges faced during the conduction of a real project, and arising challenges that still must be overcome. In particular, we perform an analysis of the challenges identified in the early stages of the SoSE life cycle (Concept of Operations, Requirements Engineering, and Architecture Design), carried out in a real project of the defense area. In this paper, we analyze SisGAAz, a Brazilian Navy Management SoS project composed of a set of interoperable systems to collect, share, analyze, display operational information, and provide decision support regarding the Blue Amazon. Many of these systems are already in use by the Brazilian Navy and others will be developed and integrated during the systems development [Chaves 2013]. The main contribution of the paper is exposing those challenges, opening research opportunities for SoSE in Brazil. The remainder of this paper is organized as follows. Section 2 presents a background. Section 3 presents an overview about SisGAAz project. Section 4 outlines challenges, limitations and problems identified during the first phase of the project. Section 5 points out for advancements we should conduct in order to overcome the highlighted challenges. Finally, Section 6 presents final remarks.

## 2. Foundations

SoS result from other operationally independent systems (so-called constituents) working together to reach common goals. Each constituent performs its individual mission contributing to the success of the global missions [Silva et al. 2014]. SoS have been consensually distinguished by a set of inherent characteristics [Maier 1998]: (i) operational independence of the constituent systems, since constituents have their own operation even if they work within the scope of the SoS; (ii) managerial independence of the constituent systems, which can be independently managed by distinct organizations and stakeholders; (iii) evolutionary development of the SoS, since the SoS evolve as a consequence of the individual evolution of its constituents, their functions and purposes; (iv) distribution, since constituents' interoperability relies on some communication technology, and (v) emergent behavior, which enables the SoS to provide new functionalities from interactions among constituent systems and that are not localized in a single constituent. In particular, missions are an important SoS concept, specially in military SoS.

SoS are categorized according to its degree of authority that it has over the constituent systems that cooperate to the accomplishment of the missions

[Department of Defense 2008]. In this perspective, four basic categories of SoS can be defined: (i) Directed – The SoS is controlled by a central entity and it is designed and operated for fulfilling specific purposes. Constituent systems can have their operational and managerial independence, but their behavior is subordinated to the central control and its purposes; (ii) Acknowledged – Goals, resources, and central control of the SoS are all recognized, but the constituent systems retain their independent management and their behavior is not subordinated to the central managed purpose; (iii) Collaborative – Constituent systems voluntarily collaborate in a greater or lesser degree in order to address shared or common interests. In this case, a central control has little coactivity over the behavior of the constituent systems and it typically offers standards to allow the collaboration among those systems; and (iv) Virtual – There is no central control and universal purposes, and such purposes are neither designed or expected in many cases, so that constituent systems operate in a distributed and uncoordinated environment where the mechanisms to maintain the SoS are not evident.

For a long time, SoS modeling has been carried out under a traditional document-centric perspective, suffering of (i) replication of information, (ii) lack of traceability between documents, (iii) inconsistencies of information and business rules, and (iv) difficulties to handle and search information in such documents. However, over the past decade, SoS engineers have significantly increased the adoption of Model-Based Systems Engineering (MBSE), a SoSE approach that shifts from a document-centric perspective for a model-based reality, emphasizing the development and adoption of models in SoSE [Ramos et al. 2012]. MBSE promises (i) a more effective knowledge management that can enhance the ability of stakeholders to understand the system and its behavior and performance, (ii) enhanced team communications, (iii) explicit processes for reasoning about system issues, (iv) early detection of errors and omissions, (v) improved systems architecture, (vi) detailed design integrity, and (vii) effective design traceability [Kalawsky et al. 2013, Do et al. 2014]. SysML[2] (System Engineering Modeling Language), an extension of UML[3] (Unified Modeling Language), is considered a central standard notation in MBSE [Industries Alliance 2003, Delligatti 2013].

## 3. SisGAAz Overview

The Brazilian Navy has developed a system called SisGAAz to meet the guidelines of the National Defense Strategy of Brazil for managing an area known as Blue Amazon. The Blue Amazon is associated with an oceanic area and corresponds to Brazilian Jurisdictional Waters (BJW), international areas of responsibility for the Search and Rescue operations (SAR - Search and Rescue) and areas of specific interest that go beyond the BJW and the SAR area [Chaves 2013].

SisGAAz supports Brazilian Navy's activities in its constitutional allocations and subsidiaries assignments, such as protecting the national frontiers and ensuring the brazilian sovereignty. The project involves a diverse team with several professional roles such as Telecommunication Engineers, Systems Engineering, Software Engineers, Domain Experts, Project Managers, Quality Managers, Information Managers, and Infrastructure Technicians, totalizing around 30 people involved.

[2]http://sysml.org/
[3]http://www.uml.org/

SisGAAz's mission is *monitoring and controlling, in an interoperable way, the national maritime area (waters under brazilian jurisdiction), international areas of responsibility for the search and rescue operations, and areas of specific interest that go beyond the previous ones, in order to contribute to the strategic mobility, represented by the ability to respond promptly to any threat, emergency, aggression or illegality*. SisGAAz's main purposes are: (i) information sharing and interoperability among the institutions with interest in the sea (national and international, public and private); and (ii) supporting network centric operations and the providing decision support by a shared virtual environment. Figure 1 depicts the government entities that maintain relations and operations with the Brazil's Navy that benefit from the SisGAAz. Entities are illustrated as actors and include several ministries, Army, and Justice.

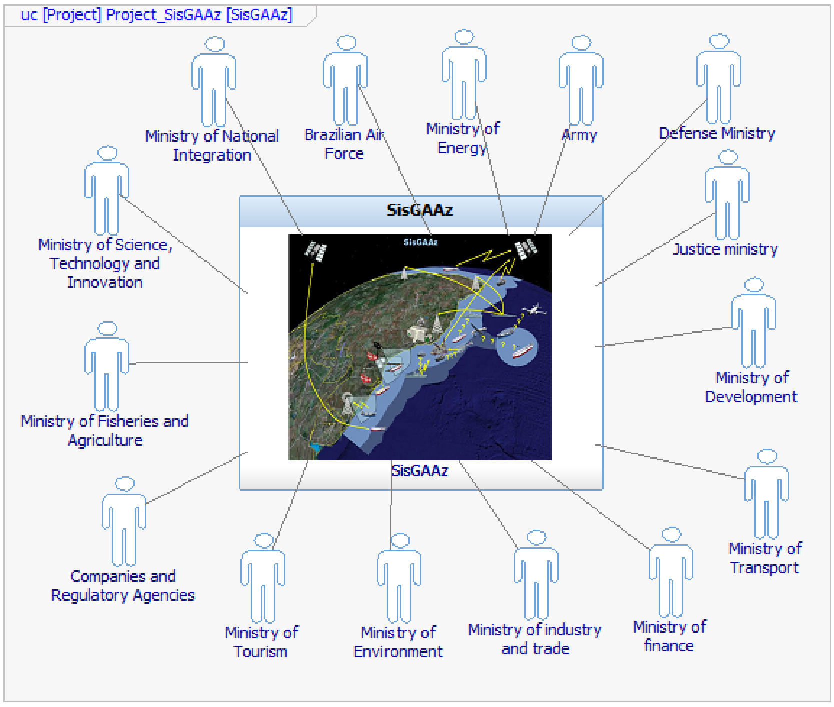


**Figure 1. SisGAAz external entities and stakeholders.**

SisGAAz development was divided in two phases: conceptualization and development. The conceptualization phase involved the system specification and design (operational concept and requirements), elaboration of project management plans, and architecture design. This phase was completed in two years and it was developed in the following four tasks: (i) project management planning; (ii) elaboration of Concepts of Operation (ConOps); (iii) elicitation, analysis, and documentation of system requirements; and (iv) conception and design of SoS architecture. Figure 2 illustrates the project life cycle.

The development of SisGAAz follows an evolutionary approach (iterative and incremental). Initially, a small core of functionalities is already implemented and deployed. Subsequently, the implementation activities are performed in a way that new functionalities are added to constituents according to the local demands of an operational area. For example, the Navy district of Rio de Janeiro is an operational area, as it is São Paulo. Rio de Janeiro district hosts the first constituents implemented according to the local de-

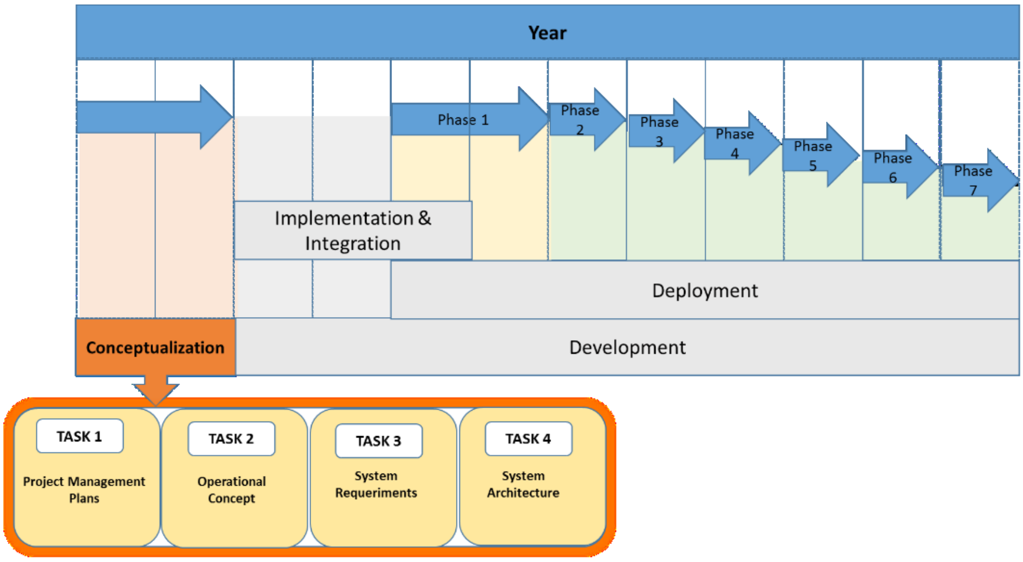


**Figure 2. SisGAAz Project's Life cycle**

mands. After that, the implementation will take place São Paulo. The new constituents developed there matches their specific demands, and after developed, they are joined to the first ones to form the preliminary body of the SoS (with constituents from São Paulo and Rio de Janeiro). Next, this work is iteratively carried out in other regions of Brazil, creating new constituents, connecting them to the others, realizing the SoS, and expanding the SiSGAAz SoS progressively.

The proposed SoS has five types of constituents, each one responsible for one of the following assignments: logistics support, cybersecurity, intelligence, command and control (C2), communications, and remote sensing. Logistics constituent aims to provide support to logistics operations regarding Navy's equipment, such as acquisition and transport of material to national oceans supervision. Cybernetics constituent provides capabilities related to SisGAAz safety and security aspects, supporting (i) availability and assurance of confidentiality of information and services, (ii) defense and protection of networks (internal and external), (iii) monitoring the full spectrum of cyber warfare operations, and (iv) safety and protection of services and constituents that forms the SisGAAz. The command and control (C2) constituents are military systems that provide features to support planning of command operations and control of constituents in the accomplishment of the mission, playing a type of central authority role. The communications constituents provide the technological infrastructure to integrate the various military organizations and SisGAAz constituents. The intelligence constituent aims to support planning and execution of operations, according to the ConOps. Finally, the remote sensing constituent provides functionality for collecting, processing and distributing data from a variety of sensors, new or legacy, which make up the SisGAAz and data received through cooperation with other organizations. Each brazilian geographic area can host a subset of those types of constituents that are linked together to form the national scale SoS known as SisGAAZ.

## 4. A Panorama of SoSE in Brazil

After our participation in the first phase of SisGAAz project, we glimpsed a panorama about how SoS is currently developed in Brazil. Next, we distill our view on SoSE in Brazil under some remarkable perspectives.

### 4.1. SoS characterization

Most of current military systems are part of a SoS, even if not explicitly recognized as such. Although SisGAAz project has focused on the development and acquisition of independent systems, many of those systems have been and will be created and integrated into the SisGAAz without explicit consideration of SoS type. In a first perspectice, SisGAAz could be considered as a directed SoS, specially for the presence of a C2 constituent with some degree of central authority. However, SisGAAz's constituents share objectives, management, and independent resources. They are, in practice, a collaborative SoS, but (i) they have not been thought as such, (ii) other constituents have been arbitrarily added to such SoS, and the level of control is not clear, which reflects on how the SoS is engineered [Dahmann and Baldwin 2008]. In SisGAAz project, those issues and SoS characterization were not considered. This issues impact directly in the dynamics of project management, systems engineering and architectural decisions in the system.

### 4.2. System Engineering Process

During the conceptualization phase of SisGAAz, a traditional document-centric system engineering process was adopted. Such process is based on American Department of Defense (DoD) MIL-STD-498, a process to guide how to perform activities for defining the SoS system architecture [Department of Defense 1994] and standardize software development and its documentation. It proposes different types of documents (known as Data Item Descriptions (DIDs)) to elaborate the project plans, concept of operations, requirements, design, test, software, support manual etc. Such approach exhibits important limitations that make the SoS development process costly, bureaucratic, and heavy, sharing all of the aforementioned drawbacks brought by document-centered approaches. Specific issues problems due to (i) information spread across several documents, (ii) lack of correspondences between requirements, analysis, and system design, becoming them hard to assess, (iii) difficulties of comprehension of the whole SoS and its architecture, (iv) difficulties to maintain such disjoint set of artifacts, what significantly impacts the total cost and effort, and (v) obsolescence of the SoS architecture [Industries Alliance 2003]. Indeed, even with the high cost, the artifacts become inconsistent and obsolete [Friedenthal et al. 2008, Delligatti 2013].

### 4.3. System Engineering Lifecycle Management

SisGAAz involves several professionals from different knowledge areas distributed in two states (São Paulo and Rio de Janeiro). The communication between the team members was realized by e-mails, phone and files exchanged through a repository of version control. As a consequence, we have identified some problems: (i) excessive number of documents and files stored in a huge repository; (ii) fragmented records of project information scattered in numerous emails; (iii) problems to share the information between the engineers during the project; and (iv) loss of important information during the meetings by telephone. This poor and ineffective communication brought a negative impact on the progress of the project and mainly for the SoS architectural specification, particularly in this real case, in which the project is large, complex and multidisciplinary. The system engineering life cycle was realized without an integrated and collaborative system-engineering environment. Consequently, the team did not have a clear view of the system engineering process used in the project and it was very difficult to realize the tasks and activities proposed to the project.

### 4.4. Models

Some system engineering models were developed during the architectural system design step. Business Process Model and Notation (BPMN)[4] and Data Flow Diagram (DFD) were used for the representation of operational processes. A combination of UML and SysML models were adopted for materializing the System and Subsystem Design Description (SSDD). Many models (BPMN, SysML and UML) created during the project used incompatible elements of language to represent the SiSGAAz architectural description. This combination produced incomplete and inconsistent models according to the syntax and semantics proposed by the language. This is a result of the lack of an appropriate MBSE method to support the correct use of SysML. Distinct modeling approaches can cause a lack of standardization during the development phase. In addition, models in SysML and UML are largely static and descriptive and do not capture the dynamics and executable aspects of emergent behaviors of SoS.

## 5. Challenges and Advancements

As reported, there are types of limitations in the way SoS are currently engineered in Brazil. As a matter of fact, the SoSE practice in Brazil has been divergent from the world-wide state of the art and practice that we yielded previously. Thus, it is paramount to take those observations as motivations to research and advance in our practice to produce SoS. In this direction, we argue that each of the aforementioned key viewpoints configure specific challenges raised. In this direction we raise the following challenges (C) for the current practice of SoSE in our country:

**C1. Conception of SoSE processes that are aligned with Maiers' taxonomy**: Current SoSE practice in Brazil does not take into account the aforementioned taxonomy [Maier 1998]. Hence, it is important to adopt and conceive SoSE processes that aides the stakeholders involved to discern the nature of the SoS being produced, and in which the activities support the realization of the SoS architecture, avoiding architectural degradation [Gurgel et al. 2014]. Hybrid cases, i.e., those ones in which multiple types of central authority can co-exist, as in SiSGAAz, must also be investigated;
**C2. Substitution of Ancient Processes**: Current practice in Brazil still adopts document-centric approaches. In alignment with international trends, SoSE processes must be migrated to modern proposals;
**C3. Absence of Process Standardization and Automation**: Even with the adoption of some industrial standards, there is a lack of standardization in the processes. The process recommendations are not strictly followed. Hence, it is necessary to propose some methods to facilitate the adherence of the stakeholders to the processes elaborated and/or adopted and their institutionalization. Software Process Improvement practice can

[4]http://www.omg.org/spec/BPMN/2.0/

give some guidance on this direction and Governance politics must also be proposed for this scenario;
**C4. Models and Automation**: It is necessary to shift the SoSE practice in Brazil from the outdated document-centric approaches to the cutting-edge technologies and processes for SoSE, in particular MBSE. There is a lack of models in the current approaches, a low level of abstraction in the documents, and a low possibility to automate SoSE production activities tasks. All this scenario increases the complexity in the development, reduces the maintainability and traceability of the final product, and causes a lack of sync between the final product and the correspondent documents. Solutions must be adopted, extended, and proposed to tame those drawbacks.

Since we identified those challenges, we also propose some intervention proposals that can aide the advancement of current practice of SoSE in Brazil. Among these practices, we can highlight the following:
**SoS Processes (C1, C2, C3)**: we noticed a lack of guidance to support the SoS characterization and processes that reinforce the adherence of team members to their tasks. New processes specially tailored for SoS have been proposed [Goncalves et al. 2015]. They must be adopted and/or extended to overcome such difficulties and to support such processes institutionalization and a strict adherence to SoS taxonomy;
**Conception of Collaborative and Distributed Software Development Environments for the conception of SoS (C1, C2, C3)**: Engineering SoS involves a range of people that inherently collaborates to deliver a final product. Specially in SoS, there is a high degree of diversity of the stakeholders and institutions involved in the production life cycle, many times geographically spread. Hence, it is paramount to develop software environments to support a suitable management of the SoS development with all sort of complexity involved. Those tools should support collaborative work with an additional dimension of global distributed development, supporting a current reality of Distributed Software Development [Marczak et al. 2014, Santos et al. 2015], a reality in SoS conception. Hence we propose that collaborative web-based environments must be suitably conceived in order to match the modern demands imposed by SoS production;
**Endorsement of models, metamodels and MBSE approaches in SoSE (C4)**: Models and metamodels foster an institutionalized practice of documenting software, even in SoS context, since they become first-class citizen in the production chain [Graciano Neto et al. 2015], becoming a golden feature in software and systems development. MBSE approaches and processes for SoS are imperative, since models contribute to automation, traceability, abstraction. They must become the *de-facto* standard to drive SoS conception and specification, overcoming the drawbacks deeply related to document-centric approaches, and adopting the cutting-edge technologies recently proposed to skip the ancient frontiers of SoS production. Remarkable examples include the use of architectural description languages (ADL) for documenting SoS software architectures, such as the recently proposed SoSADL [Oquendo 2016];
**Simulation technologies (C4)**: Simulations are a well-recognized technique in SoSE [Graciano Neto et al. 2014]. Since it offers a model that facilitates the visualization of the SoS operation before properly being deployed, it anticipates the detection of deffects and problems promoting an early correction of them, provides a dynamic view for SoS architectural description models to aide SoS architects in their activities in the software architecture lifecycle, facilitates validation and verification tasks [Michael et al. 2011],

specially tackling dynamic properties inherent to SoS, such as emergent behaviors [Graciano Neto and Nakagawa 2015]. A remarkable example is DEVS (Discrete Event System Specification), a modeling formalism for SoS that relies on sequence diagrams, state diagrams, ports specification, and input/output events [Zeigler et al. 2012].

These are important directions for the progress of SoS engineering practice and improvement of systems engineering industry in Brazil. We assume that each of these factors creates limitation and problems for SoS engineering. Overcoming them is imperative, and the brazilian SoS community can contribute in order to extend international instances of SoS production, proposing solutions and advancing the state of the art and state of the practice in Brazil to ground a better future of technology to our country, specially regarding the national security and supremacy.

## 6. Final Remarks

SoS have emerged and SoS engineering (SoSE) has become the heart of the development of such systems. This paper presented a big picture of the practice in SoSE in Brazil. Our main contribution is presenting an experience report of a real project conducted in Brazilian's defense context, externalizing the state-of-the-practice currently followed in Brazil. We believe that the SoS projects for defense areas need to incorporate new practices and approaches to system engineering and architecture design. Nowadays the systems engineering has been undergoing a major transformation that consists in the adoption of consolidated practices from modern software engineering to SoSE. We presented the challenges that we will face in SoSE in Brazil in the next years, and we propose some directions to support a sustainable path to realize such improvements that are in alignment with other recently proposed research directions [Graciano Neto et al. 2016]. We are aware of other international initiatives on reporting experience in SoSE [Do et al. 2014]. However, from the brazilian perspective, this is an innovative artifact. Future work include working on some of the branches we raised, and extending our reports in other directions, such as externalizing the software architectural description of SiSGAAz, and adopting such SoS as case study scenarios for solutions we have worked on in the research groups we are enrolled (START/ICMC and ArchWARE/IRISA).

**Acknowledgments** We thank PIPEq/PUC-SP, CNPq (grant:201230/2015-1/SWE) and FAPESP (grant: 2014/02244-7).

## References

Chaves, S. F. A. (2013). Blue amazon system management (sisgaaz): the first step towards effective control of the brazilian maritime area. Technical report, Brazilian Department of War Studies School.

Dahmann, J. S. and Baldwin, K. J. (2008). Understanding the current state of us defense systems of systems and the implications for systems engineering. In *Systems Conference*, pages 1–7.

Delligatti, L. (2013). *SysML Distilled: A Brief Guide to the Systems Modeling Language*. Pearson.

Department of Defense (1994). MIL-STD-498, Software Development and Documentation. MIL 498, DOD.

Department of Defense (2008). Systems engineering guide for system of systems. Technical report, Department of Defense.

Do, Q., Cook, S., and Lay, M. (2014). An investigation of MBSE practices across the contractual boundary. *Procedia Computer Science*, 28:692 – 701. 2014 Conference on Systems Engineering Research.

Friedenthal, S., Moore, A., and Steiner, R. (2008). *A Practical Guide to SysML: Systems Modeling Language*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Goncalves, M. B., Oquendo, F., and Nakagawa, E. Y. (2015). Towards architectural synthesis of systems-of-systems. WDES, pages 25–32, Belo Horizonte, Brazil. SBC.

Graciano Neto, V. V., Garcés, L., Guessi, M., de Oliveira, L. B. R., and Oquendo, F. (2015). On the equivalence between reference architectures and metamodels. CobRA '15, pages 21–24, New York, NY, USA. ACM.

Graciano Neto, V. V., Guessi, M., Oliveira, L. B. R., Oquendo, F., and Nakagawa, E. Y. (2014). Investigating the model-driven development for systems-of-systems. ECSAW, pages 22:1–22:8, Vienna, Austria. ACM.

Graciano Neto, V. V. and Nakagawa, E. Y. (2015). A biological inspiration to support emergent behavior in systems-of-systems development. WDES' 15, pages 33–40, Belo Horizonte, Brazil. SBC.

Graciano Neto, V. V., Oquendo, F., and Nakagawa, E. Y. (2016). Systems-of-systems: Challenges for information systems research in the next 10 years. In *GRANDSI-BR*, pages 1–3, Florianópolis, Brazil. SBC.

Gurgel, A., Macia, I., Garcia, A., von Staa, A., Mezini, M., Eichberg, M., and Mitschke, R. (2014). Blending and reusing rules for architectural degradation prevention. MODULARITY '14, pages 61–72, New York, NY, USA. ACM.

Industries Alliance (2003). ANSI/EIA 632 - Processes for Engineering a System. ANSI 632, SAE International.

Kalawsky, R. S., O'Brien, J., Chong, S., Wong, C., Jia, H., Pan, H., and Moore, P. R. (2013). Bridging the gaps in a model-based system engineering workflow by encompassing hardware-in-the-loop simulation. *IEEE Systems Journal*, 7(4):593–605.

Maier, M. W. (1998). Architecting principles for systems-of-systems. *Systems Engineering*, 1(4):267–284.

Marczak, S., Perin, M., Prikladnicki, R., and Ayub, C. (2014). On the identification of factors that promote highperformance projects in distributed development. WDES' 14, pages 17–24, Maceió, Brazil. SBC.

Michael, J. B., Drusinsky, D., Otani, T. W., and Shing, M.-T. (2011). Verification and validation for trustworthy software systems. *IEEE Software*, 28(6):86–92.

Oquendo, F. (2016). Formally Describing the Software Architecture of Systems-of-Systems with SosADL. In *SOSE*, pages 1–6, Kongsberg, Norway. To appear.

Ramos, A. L., Ferreira, J. V., and Barceló, J. (2012). Model-based systems engineering: An emerging approach for modern systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1):101–111.

Santos, R., Junior, I. F., Lima, T., and Hernández, L. (2015). Análise histórica do wdds/wdes. WDES' 15, pages 1–8, Belo Horizonte, Brazil. SBC.

Silva, E., Cavalcante, E., Batista, T., Oquendo, F., Delicato, F. C., and Pires, P. F. (2014). On the characterization of missions of systems-of-systems. ECSAW, pages 26:1–26:8, Vienna, Austria. ACM.

Zeigler, B. P., Sarjoughian, H. S., Duboz, R., and Souli, J.-C. (2012). *Guide to Modeling and Simulation of Systems of Systems*. Springer.