IAC-18,D1,4A,4,x45081

The relationship between the Model Based Systems Engineering models and Information Systems to support space products lifecycle processes

Silva, A.[a]*, Loureiro, G[b]

[a] *Brazilian Institute for Space Research (INPE), Av. dos Astronautas, 1758, São José dos Campos, Brazil,12227-010*, ana.claudia@inpe.br

[b] *Brazilian Institute for Space Research (INPE), Av. dos Astronautas, 1758, São José dos Campos, Brazil,12227-010*, geilson@lit.inpe.br

* Corresponding Author

Abstract

Motivated by the needs of the Integration and Testing Laboratory (LIT) of the Brazilian Institute for Space Research (INPE), this paper aims to analyze how the information of space products life cycle processes is determined from the models used for their development, considering a model-based approach. With a focus on Space Systems Engineering, exemplified by the satellite Assembly, Integration and Testing (AIT) process, we analyzed the relationship between Systems Engineering models with a future Information System to support the product lifecycle process. There are various business processes to perform the satellite life cycle phases. Integrated with each other, they deliver the final result. A business process is a set of activities or related tasks that are performed to deliver an expected result. Information Systems are a good way to improve the business processes performance. AIT is one of the satellite life cycle processes. Doing AIT is so complex and a lot of equipment, people, tasks and information are involved in it. So far, at LIT/INPE there is not an Information System that supports the whole AIT process practiced there, and the storage and exchange of AIT information are document based. The use of documents as a primary source of data makes it difficult to retrieve information during the process, requires a lot of effort from the team involved in it, and sometimes causes delays in activities. There is a worldwide trend towards a shift from the document-centric to the model-centric approach for engineering complex systems. Model Based Systems Engineering (MBSE) improves Systems Engineering practices using models to represent the system in various aspects. Models are built to represent requirements, structure and behaviour of systems. These models, among other advantages, become consistent and reliable sources of information to support the various phases of the product lifecycle. Concerning LIT's case, the MBSE benefits as well as the adoption of Information Systems can improve the AIT process. We must consider that in order to build good models that faithfully represent a system it is necessary effort and skilled workforce. Equivalent effort and workforce are needed to build Information Systems that faithfully meet the needs of a business process. There is a relationship between these two activities and collaboration between them can be helpful to reduce development effort.

Keywords: MBSE, Information Systems, Space Products, SysML, UML

Acronyms/Abbreviations

AIT: Assembly, Integration and Testing
BDD: Block Definition Diagram
CBERS: China-Brazil Earth Resources Satellite
IBD: Internal Block Diagram
INPE: Brazilian Institute for Space Research
IS: Information System
LIT: Integration and Testing Laboratory
MBSE: Model Based Systems Engineering
SysML: System Modelling Language
UML: Unified Modelling Language

1. Introduction

There are various business processes to perform the satellite life cycle phases. Assembly, Integration and Testing (AIT) is one of these phases. Doing AIT involves a lot of equipment, people, tasks and information, what makes it a complex process.

Currently the Integration and Testing Laboratory (LIT) of the Brazilian Institute for Space Research (INPE) is working on AIT of two satellites, Amazonia-1 and CBERS 4A.

The AIT team at LIT/INPE uses a series of tools to help them to manage the many tasks they have to perform and all information that flows through the whole AIT process. So far, at LIT/INPE the storage and exchange of AIT information are document based. There are too many documents involved in the AIT process. The use of documents as a primary source of data makes it difficult to retrieve information during the process, requires a lot of effort from the team involved in it, and sometimes causes delays in activities.

Motivated by the need to improve the way the information is stored, manipulated and exchanged during the AIT process and taking advantage of the AIT activities that are taking place there, LIT/INPE started the project of an Information System (IS) to support the AIT process. This project is led by the Information Systems Development team at LIT, which has a close collaboration of the AIT team.

As well as adopting an IS to support the business process of the AIT phase, using a model-based approach to engineer the satellites that arrives at LIT for AIT would also bring great benefits to the LIT activities and this is worth considering as a future project.

In this context, being aware that building good models that faithfully represent a system demands effort and skilled workforce and that equivalent effort and workforce are needed to build Information Systems that faithfully meet the needs of a business process, this work begins a study on the relationship between these two activities, using the satellites AIT process as example. This study aims to analyze how the information of space products life cycle processes is determined from the models used for their development, considering a model-based approach.

Many of the current works in the area of Space Systems Engineering are looking to improve the practice of Systems Engineering by exploring Model Based Systems Engineering (MBSE). In the space community, we find works that focus on the construction of methods, languages and tools to model complex systems such as [1] and [2]; we also find works applying some language, tool or methodology to model the structure and/or behaviour of some specific space product and its subsystems, such as [3] and [4]. This paper has another focus. This paper focuses on the identification of the relationship between the modelling made from the perspective of the systems engineer, who is concerned with building the space product, and the modelling made from the perspective of the software engineer who is concerned with building an IS to support the product lifecycle business processes.

Section 2 presents important concepts for understanding the article. Section 3 presents, at a high level, the business process to perform the AIT activities at LIT. Section 4 presents the relationship between an IS system model for LIT's AIT business process case, and a simplified engineering model of a satellite assembled, integrated and tested at LIT. Section 5 discusses the results of this research. Section 6 presents the next steps to continue this research, and section 7 concludes the article.

2. Important concepts

2.1 MBSE

MBSE is a world trend to engineer complex system. It improves Systems Engineering practices using models to represent the system in various aspects.

In the MBSE approach, domain models are the main source of information and the main means of exchanging information among those involved in the engineering process, rather than exchanging information based on documents. Models are built to represent requirements, structure and behaviour of systems [5]. These models become consistent and reliable sources of information to support the various phases of the products lifecycle. MBSE brings benefits like enhanced communications, reduced development risk, improved quality, increased productivity and enhanced knowledge transfer [6].

Models can be created in several languages at the direction of the team. One of the most widely used languages nowadays is SysML.

2.2 Business Processes and Information Systems

A business process is a set of activities or related tasks that are performed to deliver an expected result, as a product or a service [7]. A good way to improve business processes performance is adopting Information Systems to support them. An IS is a system that collect, process, store, and disseminate data and information and that provide a feedback mechanism to monitor and control its operation [8]. An adequate IS facilitates access to information and eliminates possible inconsistencies and the need to redo tasks related to information management.

The main components of Information Systems are custom built software. The first thing one needs to build this kind of software is to understand the business process that will be supported by it and the information that flows in such process. In some points of the process certain information will be required by the IS for that, in other points, it can be provided to be used, for example, to assist decision-making or to perform other operations.

2.3 UML

UML is a visual language, adopted internationally, much used to model software systems based on the object orientation paradigm [9]. UML in its version 2.5 has 14 diagrams, some to model the structure of the systems and some to model the behaviour of the systems [10]. The Class Diagram and the Object Diagram from UML are used in this paper. They are structural UML diagrams.

A Class specifies a classification of Objects and the features that characterize the structure (attributes) and behaviour (operations) of those Objects. Objects of a Class contain values for each attribute of the Class, according to the characteristics of the attribute, for example its type and multiplicity [10].

A Class Diagram presents Classes of the system and the relationship between them. An Object Diagram shows Objects and data values; it corresponds to an instance of the Class Diagram, showing the state of the system at a given point in time [11].

2.4 SysML

SysML is a graphical modelling language, an extension of UML, used to model complex systems. It is a general-purpose modelling language that can support many different MBSE methods [5]. SysML supports the specification, analysis, design, verification and validation of systems [12].

SysML defines three categories of diagrams: structural, that describes the static structure of a system, behavioural, that shows the dynamic view of the systems, and Cross-Cutting, that enables the modelling of relationships across both structural and behavioural elements [12].

The Block Definition Diagram (BDD) and the Internal Block Diagram (IBD), that are used in this paper, are used to represent the structure of the system.

A Block is the basic architectural unit in SysML and, as UML Classes, it may include both structural and behavioural features. Blocks have structural properties named Parts. While Block is the definition of an element, Part is a usage or instance of that element. [12].

SysML BDD focus on definition and captures the system hierarchy and the definition of a Block in terms of its properties, operations and references. It shows high-level structure but also external interfaces and flow [12].

SysML IBD focus on usage and describes the internal structure of the system in terms of its Parts, Ports, and Connectors [5].

3. LIT's AIT Business Process and its information

The satellite life cycle processes are proposed by NASA and ESA. According to ESA [13], the satellite life cycle phases are:

- Phase 0: Mission Analysis/Needs Identification;
- Phase A: Feasibility;
- Phase B: Preliminary Definition (Project and Product);
- Phase C: Detailed Definition (Product);
- Phase D: Production / Ground Qualification Testing;
- Phase E: Utilization;
- Phase F: Disposal.

According to the nomenclature adopted by ESA, the AIT activities of a satellite take place during Phase D of the product life cycle.

There is a business process to perform each satellite life cycle phase, and the execution of these various business processes lead to the expected result.

The AIT process is complex, and many activities are carried out so that its purpose is fulfilled. Aiming to build an IS to support LIT's AIT business process, a initial model for the LIT's AIT business process was built, according to the high-level activities pointed out by the LIT's AIT team. This model starts with 5 macro processes: *AIT Definition*, *AIT Preparing*, *AIT Management*, *AIT Execution* and *Resources Control* [14]. Each of these macro processes broken down into lower level processes or activities. A brief description of each of these macro processes follows.

- AIT Definition: Here one defines the activities that will be executed (from the macro activities, through all the levels of unfolding, to the level of procedure) and the sequence in which these activities are to be executed.
- AIT Preparing: Here one defines the information that the operator needs to perform each activity, such as setup, required items, and step-by-step operations.
- AIT Management: Here the AIT manager organizes the execution of tasks, prepares the schedule, releases tasks for execution and deals with the issues and problems that occur during the process.
- AIT Execution: Here the tasks are effectively carried out and records are made about the activity itself (such as executor and date of execution), about the laboratory conditions (such as temperature, humidity and contamination) and about intercurrences (as non-conformities and unexpected events). During the AIT Execution, lessons learned should be recorded.
- Resources Control: Here, the resources needed for the AIT process are controlled (such as materials, tools, GSEs, staff).

From these macro processes, the high-level information that is generated and used by each of them was identified, arriving at a set of information that is important for the AIT process. Fig. 1 summarizes it. Like the macro processes, certainly this high-level information can also be broken down in more detail.

4. The relationship between a IS model and models of MBSE

As a basis for the comparison between the models, part of an IS model for the LIT's AIT business processes, and two simplified engineering models of the Amazonia-1 satellite are used.

Amazonia-1 is the first Earth observation satellite entirely designed, integrated, tested and operated by Brazil. It is a polar orbiting satellite that will generate images of the planet every 5 days. Its objective is to provide data for environmental monitoring, especially in the Amazon region [15].

4.1 Modelling an IS for LIT's AIT business process

Currently, the information arriving from Phase C to LIT, shown in Fig. 1, comes in documents. LIT's AIT process also happens based on documents and there is no IS to support the whole process. However, considering an IS for this purpose, the information that arrives at LIT, as well as the information generated at LIT itself, would have to be inserted in this IS so that it could be provided by the IS throughout the process. This IS would have to be structured so that this information could be entered and retrieved.

To exemplify how part of a model of an IS for meeting the needs of the LIT's AIT process would be, consider the *satellite log* information. As shown in Fig. 1, this useful information flows from the macro process *AIT Execution* towards the macro process *AIT Management*.

Satellite log is high-level information. Included in its decomposition, among many other items of information, are those about assembly and disassembly of parts of the satellite, that will be used as an example.

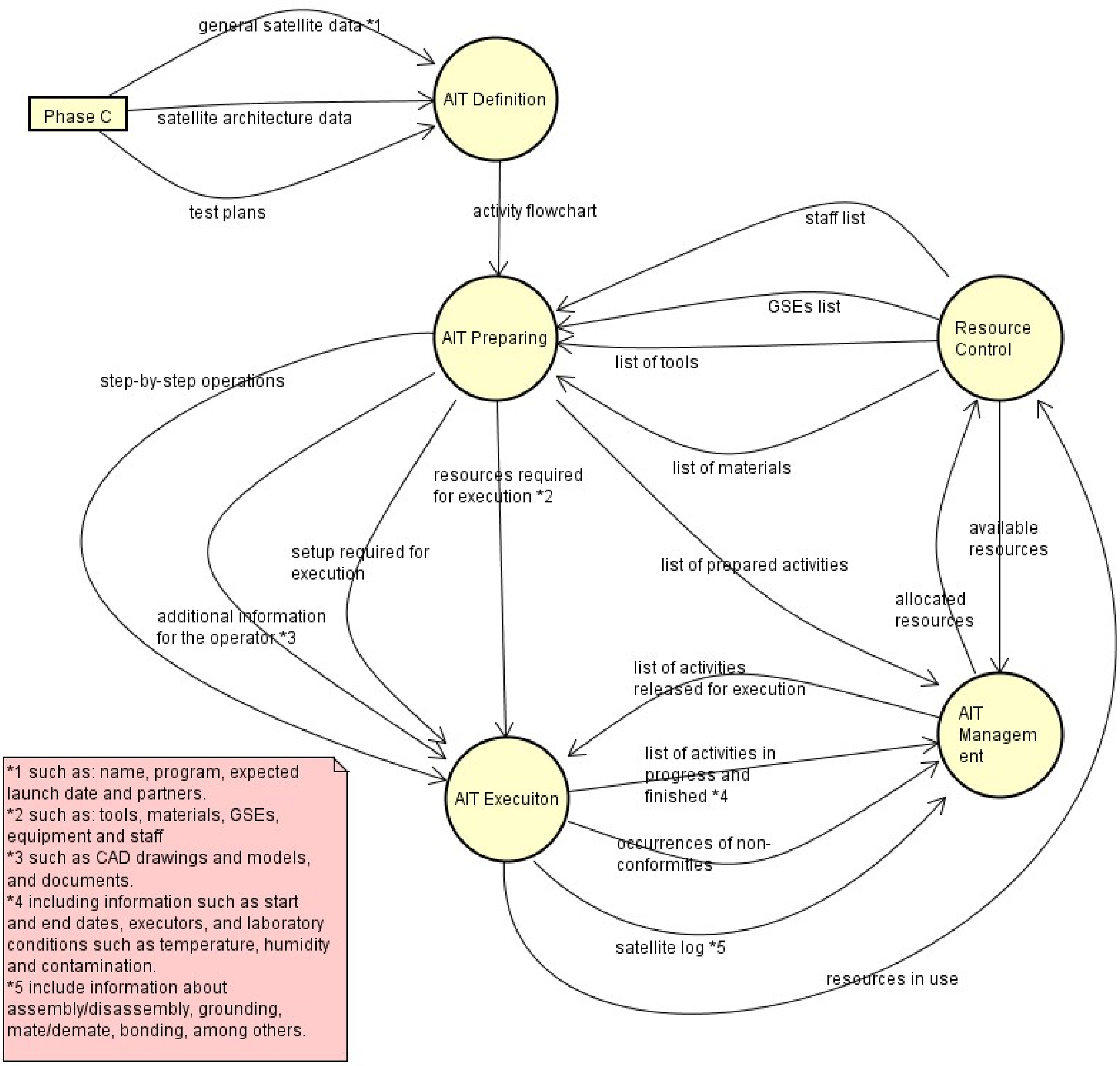

Fig. 1. LIT's AIT macro processes and their information

One of the questions that the IS should answer to the AIT manager, in the macro process *AIT Management*, is what is already assembled on a given subsystem or what was assembled on a certain date. This is important for the manager to let him know, for example, which subsystem can already be tested or so that he, or others, can analyse fault situations.

For the IS to provide this answer, three activities must be performing within the *AIT Execution* macro process, two of them are *register assembly* and *register disassembly*.

When some equipment is assembled or disassembled on the satellite, the operator shall record in the IS that this has occurred. This equipment must be one that was received at LIT for AIT and must be associated in the IS with the satellite part to which it corresponds. For this association to be possible, a third activity must happen first: *register entry of equipment into LIT*. In this activity the received equipment is registered in the IS with its proper identification (Part Number, Serial Number), among other information. Also, the satellite parts need to be previously registered in the IS, as well as the satellite subsystems, and the relationship between them. Fig. 2 illustrates this small part of the AIT business processes model.

For the IS to be able to support this process and to answer the question of the AIT manager, a structure such as that shown in Fig. 3 is required.

Fig. 3 shows a UML Class Diagram representing part of a software structure for an IS for the LIT's AIT business process. According to the UML notation, in Fig. 3 the boxes represent the Classes. Their first compartment is used to name the Class and the second one is used to show its attributes. The lines connecting the Classes represent the relationship between them. The number, and the "*" (which means several) at the ends of the lines indicate the number of possible instances in the relationship, for example, a satellite model may have multiple subsystems and a subsystem must belong to a satellite model.

4.2 MBSE approach to engineering the satellite Amazonia-1

As mentioned earlier, information for the AIT process currently arrives at LIT by documents rather than by models. Taking MBSE as the way to engineering a system, models are built to represent system requirements, structure and behaviour. Considering a model-based approach rather than the document-based approach to engineering the space products that will be assembled, integrated and tested at LIT, information for AIT would arrive in models, for example in SysML models.

Taking the Amazonia-1 satellite as an example, it could be represented in a SysML BDD modelled as a composition of its subsystems, and its subsystem could be represented as a composition of its parts. A simplified example of a BDD for the electrical model of the Amazonia-1 satellite is showed in Fig. 4. Fig. 4 shows three of the subsystems of the Amazonia-1: *AOCS*, *OBDH* and *PSS*; and some of the parts of these subsystems. According to the SysML notation, in Fig. 4 the boxes represent the Blocks, and their first compartment is used to name them. The lines with a solid diamond at one end connecting Blocks represent a Part Association, that are commonly used to decompose a Block into its Parts. The numbers in the other end of the lines represent the number of instances or usages of the related Block. For example, the *PSS* subsystem are decomposed into 1 *PCDU*, 1 *SADE*, 2 *SADA* and 4 *Batteries*.

In a model-based approach, using SysML, Internal Block Diagrams could be used to represent the internal structure of each subsystem, describing how parts and external references are related in the context of the subsystem Block. For this purpose, instances of the subsystem parts would appear in the IBD, as exemplified in Fig. 5, which shows instances of the *RW* Block and *ACE* Block in an IBD of the *AOCS* subsystem. According to the SysML notation, in an IBD the boxes represent the Parts, and in their first compartment appear the name of the Part and the Block that the Part is a type of, in the format: *Part name: Type name*.

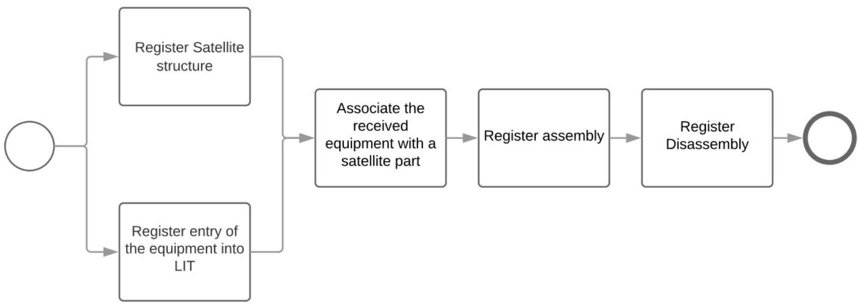

Fig. 2. Part of the LIT's AIT business process

4.3 A relationship between the two models

We have seen, as an example, that information on the assembly and disassembly of the satellite parts are important for the AIT manager and should be provided to him by the IS. We have also seen which tasks of the process are required for this information to be available in the IS: *register satellite structure*, *register receipt of equipment*, *register assembly* and *register disassembly*. Fig. 3 shows a model of the Software Engineering domain that allows the IS to support this process.

We can see that the satellite subsystems and the satellite parts modelled in the Class Diagram (Fig. 3) are also modelled in the satellite structural diagrams (Fig. 4 and Fig. 5) in the domain of Systems Engineering.

If we instantiate the model presented in the Class Diagram (Fig. 3) for the case of Amazonia-1, some examples of Objects are those shown in Fig. 6. Fig. 6 shows an Object Diagram from UML. Remembering that an Object Diagram shows instances of Classes. In practice, if we use, for example, a relational database to implement the IS, these Objects are data on tables and not database structure.

Comparing the BDD (Fig. 4) with the instances of the Class Diagram represented in the Object Diagram (Fig. 6), it can be noticed that the structural model of Fig. 4 is in part an instance of the Class Model (Fig. 3). The subsystem Blocks in the BDD are equivalent to instances of the *Subsystem* Class. In the case of the Blocks that appear in the decomposition of the subsystem Blocks, they do not correspond to instances of the *SatellitePart* Class, since they do not represent a specific satellite part. Instead, these Blocks are equivalent to instances of the *EquipmentType* Class. Instances of the *SatellitePart* Class do not appear in the BDD but appear in the IBD as Parts of the subsystems Blocks.

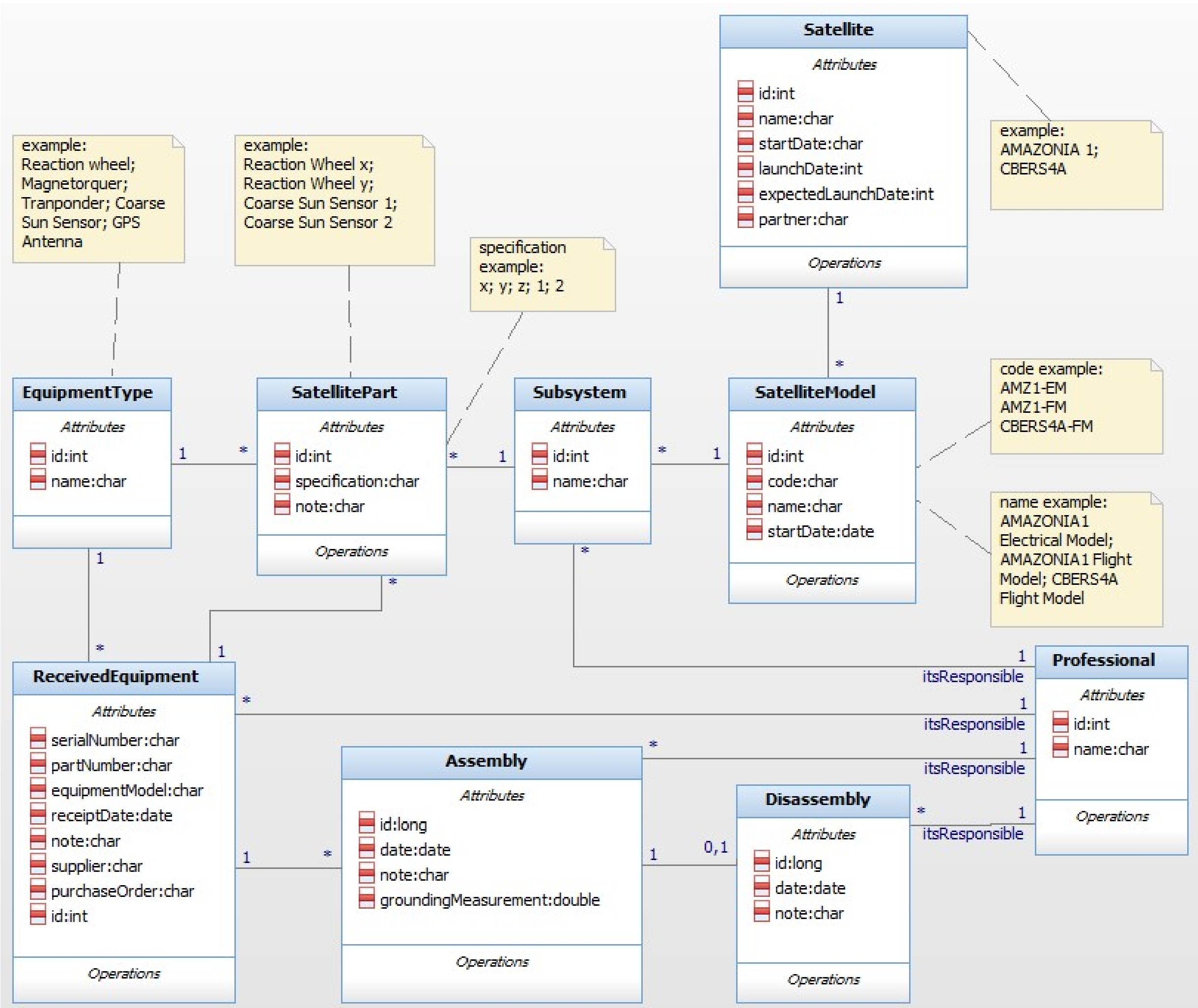

Fig. 3. Simplified UML Class Diagram representing part of the software structure for an IS for the LIT's AIT business process

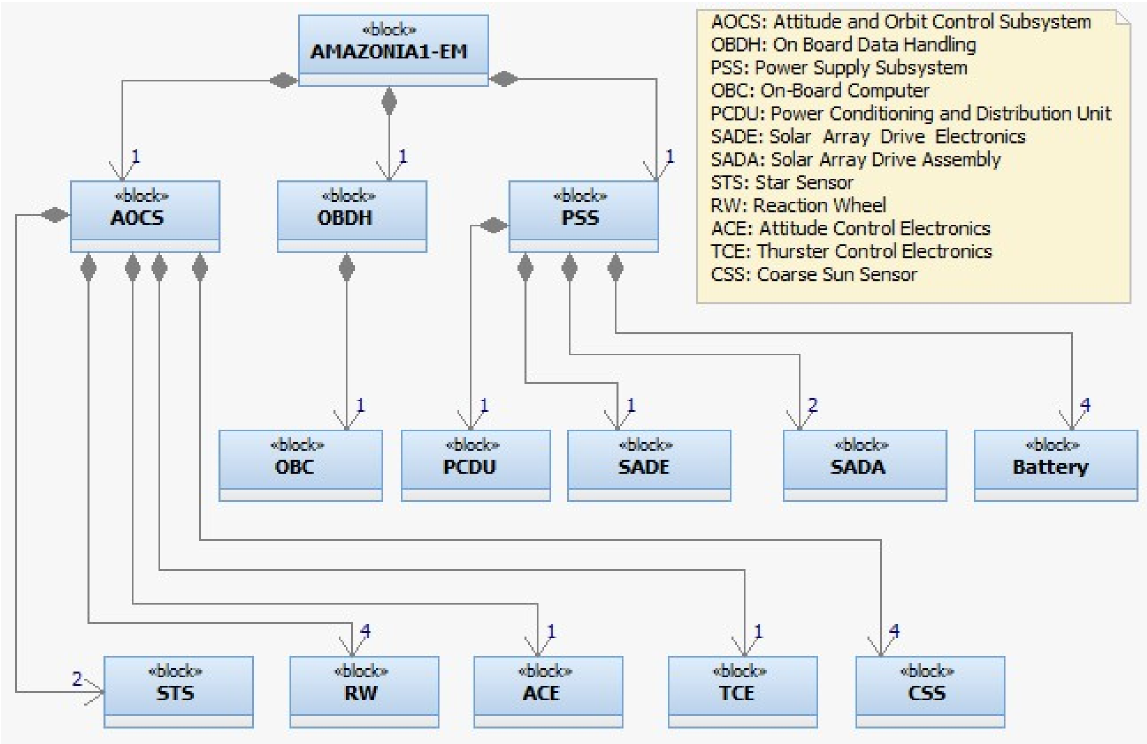

Fig. 4. Simplified SysML BDD for part of the electrical model of Amazonia-1 satellite

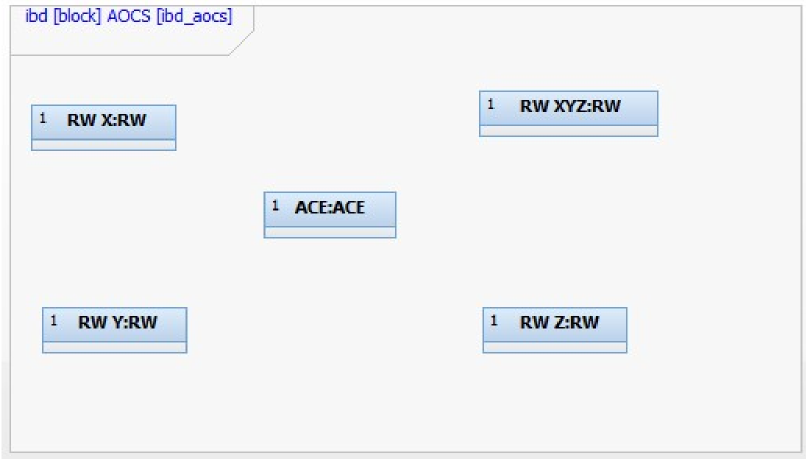

Fig. 5. Simplified SysML IBD for part of Amazonia-1 satellite

In the IBD, the types of the Parts are equivalent to instances of the *EquipmentType* Class of the Class Diagram (Fig. 3), and it is possible to relate each Part of the IBD with instances of the *SatellitePart* Class. Similarly, in an IBD of the *Satellite* Block, the subsystems would appear as Parts and would correspond to the instances of the *Subsystem* Class.

In these examples, each element in BDD corresponds to one or more instances (depending on the multiplicity) of the *SatelliteModel*, *Subsystem* or *EquipmentType* Class.

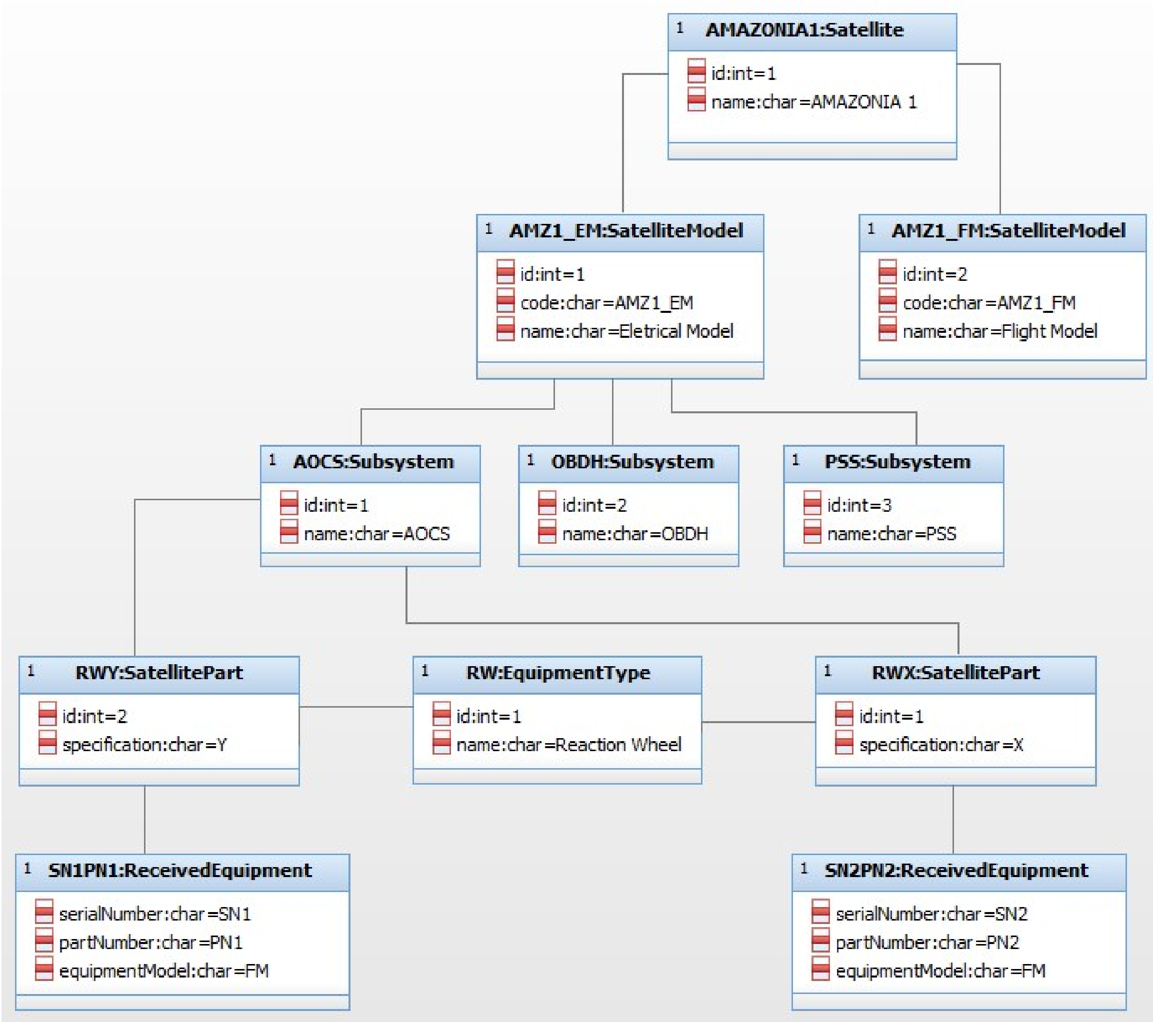

Fig. 6. Simplified UML Object Diagram representing instances of the Class Diagram showed in Fig. 3

Fig. 7 illustrates an approach to model these examples in the Systems Engineering domain in a way that the model could be reused in the field of Software Engineering for the construction of the IS. Using a super type for the Blocks would allow you to identify where your instances should be allocated to the IS domain: *EquipementType*, *Subsystem*, or *SatelliteModel* Class. For a modelling transition, the super type would define the Classes in the IS domain, and the subtypes the Objects. As not all Systems Engineering domain super types will necessarily influence the IS, a higher-level type (or Block), like *ISLink*, could be created to identify super types of interest to the Software Engineering domain.

5. Discussion

The examples explored in this paper are only a small part of all the work that needs to be done to work with a model-based approach and to have an IS that supports the AIT process. The work is vastly larger.

We could see that some models of information that compose the structure of the IS are determined by models that would be constructed to engineering the satellites, considering a model-based approach. Although one has different focus when modelling to engineering the product and when modelling to engineering an IS, there is a relationship between these two activities.

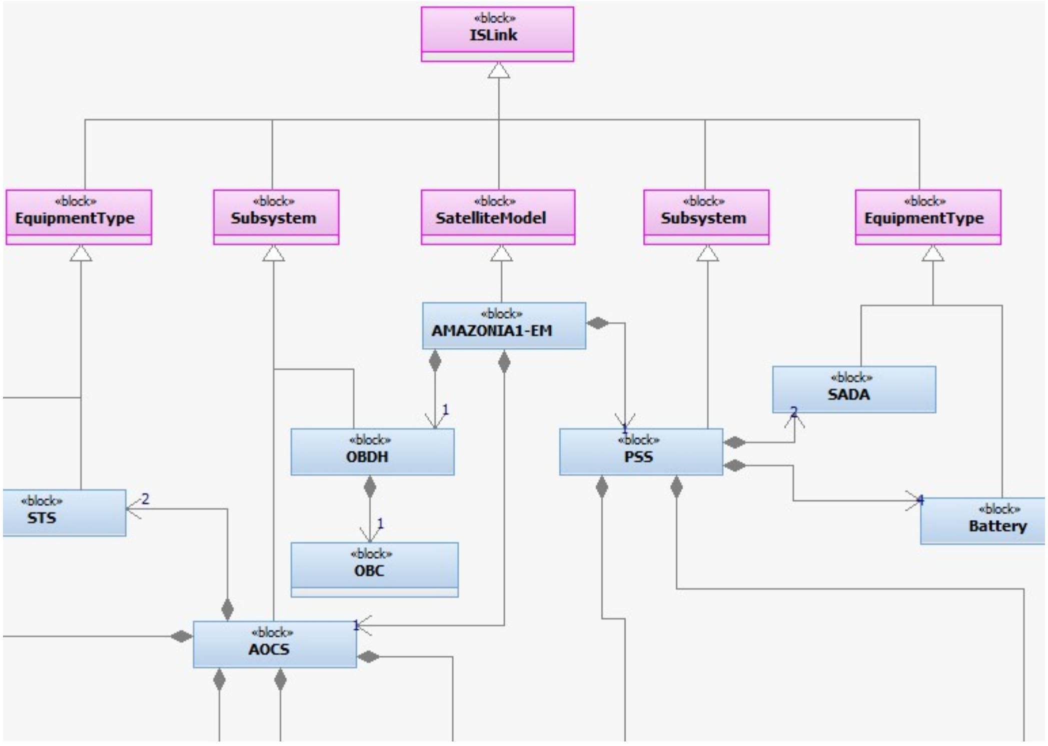

Fig. 7. An approach to model in the Systems Engineering domain in a way that the model could be reused in the Software Engineering domain

We must consider that both to build models that faithfully represent a space product, and to build Information Systems that faithfully meet the needs of the business processes to perform each space product life cycle phase, demands effort and skilled workforce. Looking at the examples, there is a sense that much of the information required for the IS is determined by the Systems Engineering domain models and that, therefore, collaboration between these two activities can be useful in reducing the development effort.

Finding a link between the models built in the field of Systems Engineering and those built in the field of Software Engineering can also bring other benefits. For example, having a structural model for satellite engineering, and knowing that part of this model is a source of information for the IS, the ideal would be that the IS was directly affected by changes in this model. This would contribute to the consistency of the information and to eliminate rework when feeding the IS with information.

It seems interesting to explore this relationship and the possible collaboration between these activities.

6. Further work

A small part of the Amazonia-1 satellite structural model was used as an example for the scope of this article. Many more models, other structural models and models of other aspects of the system, still need to be explored in future work.

Aiming to arrive at a collaborative approach between modelling of systems for engineering of space products and modelling IS to support the business processes that perform the phases of the life cycle of the space products, as next steps we intend to explore:

- the unfolding of the business process model that performs AIT in LIT;
- the unfolding of the information that flows in this business process;
- the unfolding and complementation of the structural models presented in this work;
- other MBSE models and their relationship to the information flowing in the AIT process.

7. Conclusions

With a focus on Space Systems Engineering, exemplified by the satellite AIT process performed at LIT/INPE, this work presents and compares a model (Class Diagram), usually built in the domain of Software Engineering, to model the structure of software systems that use the object orientation paradigm, and two models (BDD and IBD), usually built in the field of Systems Engineering considering a model-based approach, to model the structure of systems. From the analysis of the relationship between these models it was possible to observe that the collaboration between the modelling activities performed in the field of Software Engineering and those performed in the field of Systems Engineering can be helpful to reduce development effort.

This is an initial study; further work is needed to arrive at a collaborative approach to model space products to engineering them, and to model IS to support the business process that perform the phases of the life cycle of the space products.

Acknowledgements

The authors would like to thank the LIT's AIT team that provided expertise that greatly assisted the research; and also Fundep (Research Development Foundation) for sponsoring Dr. Loureiro’s participation at IAC 2018.

References

[1] P. M. Fischer, D. Ludtke, C. Lange, F. C. Roshani, F. Dannemann, A. Gerndt, Implementing model-based system engineering for the whole lifecycle of a spacecraft, CEAS Space J (2017) 351-365.

[2] S. Bonnet, J. Voirin, D. Exertier and V. Normand, Not (strictly) relying on SysML for MBSE: Language, tooling and development perspectives: The Arcadia/Capella rationale, 2016 Annual IEEE Systems Conference (SysCon), Orlando, FL, 2016, 18-21 April, pp.1-6.

[3] D. Kaslow, B. Ayres, P. T. Cahill, L. Hart and R. Yntema, A Model-Based Systems Engineering (MBSE) approach for defining the behaviors of CubeSats, 2017 IEEE Aerospace Conference, Big Sky, MT, 2017, 4-11 March, pp.1-14.

[4] W. A. dos Santos, B. Leonor, S.Stephany, A Knowledge-Based and Model-Driven Requirements Engineering Approach to Conceptual Satellite Design, ER '09 Proceedings of the 28th International Conference on Conceptual Modeling, Gramado, RS, 2009, 9-12 November, pp. 487-500.

[5] INCOSE, Systems Engineering Handbook A Guide for System Life Cycle Processes and Activities, fourth ed., John Wiley & Sons, New Jersey, 2015.

[6] S. Friedenthal, A. Moore, and R. Steiner, A Practical Guide to SysML: The Systems Modeling Language, 3rd ed. Morgan Kaufmann, Waltham, 2015.

[7] M. Rosing, A. W. Scheer, H. Scheel, A Practical Guide to SysML: The Complete Business Process Handbook Body of Knowledge from Process Modeling to BPM. Morgan Kaufmann, Waltham, 2015.

[8] R. Stair, G. Reynolds, Fundamentals of Information Systems, Ninth Edition. Cengage Learning, Boston, 2018.

[9] G. T. A. Guedes, UML 2 Uma abordagem prática, 3rd ed. Novatec Editora Ltda., São Paulo, 2018.

[10] OMG, OMG Unified Modeling Language Version 2.5.1, OMG Document Number formal/2017-12-05, 2017.

[11] A. C. Melo, Desenvolvendo Apicações com UML 2.2, Third Edition. Brasport, Rio de Janeiro, 2011.

[12] IBM Corporation, Essentials of IBM Rational Rhapsody v761 for Systems Engineers Student Manual vol 1, 2012.

[13] ECSS, Space project management - Project planning and implementation, ECSS-M-ST-10C rev, 1, 2009.

[14] INPE/LIT, Minutes of the meeting held on September 21, 2017 at LIT/INPE, LIT25-LIT05-MR-001, 2017.

[15] INPE, Missão Amazonia 1, http://www3.inpe.br/amazonia-1/, (accessed 08.09.18).