IAA-BR-16-49

A SysML Reference Model to Satellite/Launcher Interface and its Instantiation to a CubeSat Project

Ricardo Franco, Walter Abrahão dos Santos**.*

Every satellite project needs to take into account all the concerns imposed to its interface to a real launcher vehicle, as it is the main interface of the satellite to the outer world during its launch lifecycle phase. This interface generates constraints that the satellite needs to comply. In this work a Model-Based Systems Engineering (MBSE) approach is taken to define a reference model to this interface which is later on instantiated to a CubeSat mission as a case study. MBSE is an emerging technology that provides many advantages to the classical and document-oriented Systems Engineering (SE) process. One advantage to this approach is solving the integration problem that each one of the individual systems domain disciplines have in the variety of their own well-established modelling methods and tools to support design, analysis, verification and validation. Since these disciplines are not, *a priori*, well connected, data exchange and sharing may become cumbersome. This is where MBSE comes into place as it primarily models the whole system from a holistic problem point-of-view. MBSE can support tools for early system verification and validation and can provide means to simulate and execute many parts or the whole system. Models are here written in SysML (Systems Modeling Language) which is a general-purpose graphical modeling language for specifying, analyzing, designing, and verifying complex systems that may include hardware, software, information, personnel, procedures, and facilities and can model systems architecture and behavior. Finally, this paper shows how the launcher-satellite interface reference model is effectively used during a launch lifecycle Systems Engineering (SE) process, how it supports many Systems Engineering (SE) processes and how it meets the constraints imposed to a CubeSat project taken as a satellite case study.

Introduction

Systems Engineering (SE) is an interdisciplinary approach which enables the

* Instituto de Aeronáutica e Espaço (IAE), Brazil, ricardorf@iae.cta.br
** Instituto Nacional de Pesquisas Espaciais (INPE), Brazil, walter.abrahao@inpe.br

realization of successful systems from satellites to launchers and the interface between them. This topic in particular is part of the list of concerns primarily for satellite designers and also for prospective providers of launching services at some extent. Interfacing things always has to deal with matching requirements, constraints, representations and, models from each entity it interfaces. In order to facilitate this task, the present work suggests a Systems Engineering approach which is driven by models (MBSE) [1].

SE integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production and then to operation [2]. A CubeSat project, like any other complex system, involves many different operational scenarios with several aspects studied by many different engineering disciplines and it must be conducted with the SE approach in mind.

SE processes takes into account not only the system that is being realized but all the people and organizations involved. These people and organizations are described by SE methodology as stakeholders who may also exert influence over the project, its deliverables, and the team members. [3].

The increase in system complexity demands more rigorous and formalized SE practices. In response to this demand, along with advancements in computer technology, the practice of systems engineering is undergoing a fundamental transition from a document-based approach to a model-based approach [4]. This work will look on how a SysML reference model to satellite/launcher interface and its instantiation to a CubeSat Project may contribute.

This paper is organized as follows, it first briefs the area of Model-Based Systems Engineering, then a SysML Reference Model to Satellite/Launcher Interface and its instantiation to a CubeSat project are discussed. Afterwards, the paper shows how MBSE can support concerns and constraints checking, as well as verification and validation and other SE Processes which is then followed by conclusions.

Model-Based Systems Engineering

Usually in a CubeSat project, the stakeholders are university, or other organization responsible for the project, students, suppliers, integration and test organizations, CubeSat generated data users, launch service organization and sponsor. There may be others depending on the CubeSat project and a way to have all the stakeholders expectations met is through good practices with SE process. Model-Based initiatives are one of such approaches and they describe the system as an abstract model and not only with words and static figures as the earlier document-based approach.

A model is an approximation, representation, or idealization of selected aspects of the structure, behavior, operation, or other characteristics of a real-

world process, concept, or system [5]. A model is a simplification of all the phenomena involved in a specific system to only the phenomena that really matters to an undergoing study through a SE process. A model can be an equation, a diagram, a table, an algorithm.
System specific domain engineering disciplines have been using models to describe its problems to be solved using Computer Aided Engineering (CAE) tools, e.g. SPICE models to describe circuits, finite element models to describe phenomena that are stated by differential equations like wave propagation, finite state machines to describe a machine behavior. Model-Based Systems Engineering (MBSE) is an approach that brings the advantages to have a problem described in a way computers can understand and solve.
In the model-based approach, the emphasis shifts from producing and controlling documentation to producing and controlling a coherent model of the system. MBSE can help to manage complexity, while at the same time improve design quality and cycle time, communications among a diverse development team, and facilitate knowledge capture and design evolution [4].
There are several advantages of MBSE over the classical document-based SE and they constitute the main motivation of this study. The three of most significant advantages to this study are: (1) the integration improvement between system specific domain disciplines, (2) improvements in configuration control and, (3) the advantages of having an executable model.
A standardized and robust modeling language is considered a critical enabler for MBSE [4]. During the last decade, SysML has evolved into an enabling technology for MBSE [6] and evolved as the main language used for it. SysML is intended to model systems from a broad range of industry domains such as aerospace, automotive, health care, and so on [4].
The case study presented by this paper used SysML because of its importance in the MBSE global current context as one of its main driver.

A SysML Reference Model to Satellite/Launcher Interface

The current Brazilian space activities are supposed to be driven by the *Programa Nacional de Atividades Espaciais* (PNAE) which stands for “The National Program of Space Activities” in English. The activities that are being currently promoted by the PNAE involve satellites and rocket launch vehicle systems and subsystems development, manufacturing, integration and operation.[7]
As the MBSE approach is a methodology for dealing with complex systems such as aerospace projects, studies are needed to implement and improve this methodology in Brazilian space context.
The International Council on Systems Engineering (INCOSE) Space Systems Working Group (SSWG) established the Space Systems MBSE Challenge team in 2007. The SSWG Challenge team has been investigating the

applicability of MBSE for designing CubeSat's since 2011. They have been developing a CubeSat Model-Based System Engineering reference model. The reference model is a starting point that teams can use for their specific CubeSat models and it establishes a nomenclature for incorporating stakeholder analysis and defining architecture down to the logical systems [8]

As a mean to support MBSE implementation in the current Brazilian space context it was proposed to model the satellite/launcher interface using MBSE methodologies and the SysML language.

A reference model in addition can serve as a basis for setting standards, and can serve as an example for MBSE tools and infrastructure tradeoff studies and the implementation of MBSE approach itself.

In a MBSE approach development environment, the satellite/launcher interface model is supposed to be the main link between the launcher service provider and the satellite developer organization in terms of SE.

A reference model for this interface is made and it is described in this paper. The satellite/launcher interface is modeled as a system itself. The context diagram presented as a simplified SysML internal block diagram in Figure 1 is supposed to summarize all constraints and phenomena involved through this interface during the launch lifecycle phase.

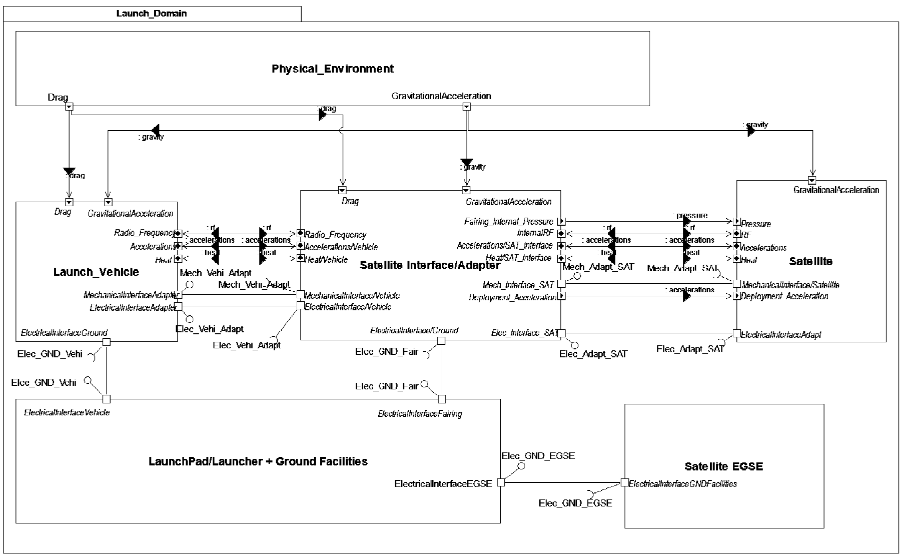

Fig. 1 Simplified context diagram of the SysML reference model to satellite/launcher interface.

The blocks, modeled in the satellite/launcher interface reference model, were the launch vehicle, the satellite/launcher interface (which can be any CubeSat interface or a main satellite adapter), the satellite itself, the physical environment, ground facilities (which englobes launcher or Launchpad, ground cables, buildings, etc.) and finally the satellite Electronic Ground Support Equipment (EGSE), (which englobes hardware, software and people).

The flows entities in the model represents physical phenomena involved in the launch context. The flows from the physical environment modeled were drag imposed by launch vehicle displacement through atmosphere and gravity acceleration.

The phenomena modeled, as flows exchanged between launch vehicle and satellite, are radio frequency, accelerations (quasistatic, random, shock and vibroacoustic) and heat.

One can notice that these flows are flowing, in the model, in both directions in and out the interface. All of the flows modeled represents concerns imposed to the satellite and consequently to the satellite project. In SE processes these bidirectional flows represents concerns about physical coupled phenomena and demands coupled analysis and obviously qualification tests e.g. Coupled Load Analysis (CLA) for vibration, Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) analysis and tests, and thermal load analysis. These analysis and tests demands data and generates data. These data can be exchanged using this reference model language.

One of the main uses of the satellite/launcher interface system is “to deploy the satellite into a predetermined orbit”, to do so the interface imposes an acceleration over the satellite at a certain time, this acceleration is modeled too. SysML models interfaces as ports from blocks there are two types of ports in SysML: flow ports that are ports providing flow interface from block, and standard ports that represents interfaces between the blocks.

The context diagram shows all interface, standard ports of Satellite Interface/Adapter block, that are supposed to exist in a reference model. In the MBSE these interfaces carry all information about and definition of the interface. These modeled interfaces takes the role of the Interface Control Documents (ICDs) of a document-based SE approach.

A CubeSat Instantiation for the SysML Reference Model

The reference model presented earlier in Figure 1 represents the concerns to be studied and processed by SE processes. The reference model is now instantiated as a CubeSat/Launcher Vehicle interface as part of a research activity to shows how MBSE approach and the SysML helps the system (interface) accomplish its mission. A real CubeSat project is chosen, but the most important system modeled in this study is not the CubeSat itself but it is the interface with the launcher vehicle. This interface is a standard deployment system, usually called POD (Picosatellite Orbital Deployer). The reference model was then changed, to model one of these PODs.

CubeSats usually "takes rides" into a main satellite launcher, taking advantage from free spaces inside launcher vehicle fairing envelope, and are always encapsulated by PODs. CubeSats and their PODs are not allowed to influence the main satellite launch. The satellite block showed in Figure 1 was replaced by a POD playing a role of a CubeSat package (POD encapsulating a CubeSat).
The POD used in this modelling effort is the Poly Picosatellite Orbital Deployer (P-POD).mk II [9]. This is the Cal Poly (California Polytechnic State University) standard deployment system which is based on Poly Picosatellite Orbital Deployer (P-POD).mk I, changed to allow this POD to be used with the Dnepr launcher. The Poly Picosatellite Orbital Deployer (P-POD) is a square, tubular structure capable of deploying up to three CubeSat's into space.

1. The Cal Poly P-POD mk II model

The reference model is changed to represent the Cal Poly P-POD mk II POD, the main model blocks are still: launch vehicle, the satellite/launcher interface (which is in this case the main satellite adapter), ground facilities, the satellite Electronic Ground Support Equipment (EGSE) and now the satellite block is changed to represent the CubeSat plus POD package. The main satellite to be launched is not modeled since all constrains to the POD, in this case Cal Poly P-POD, are imposed by the main satellite adapter.
The block definition diagram, presented in Figure 2 shows the P-POD subsystems hierarchy. The P-POD release system is presented expanded with its own subsystems showed up; the reason of this presentation is that one of the main uses of the P-POD is to deploy the CubeSat into its orbit and is allocated to this system.

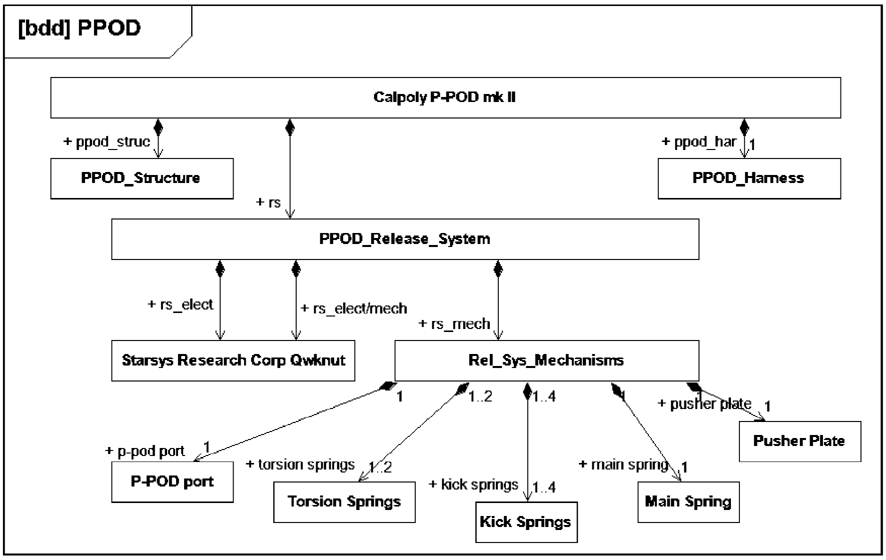

Fig. 2 P-POD block definition diagram and subsystems hierarchy.

The Cal Poly P-POD is modeled as a composition of three subsystems represented by the blocks: PPOD_Structure, that is the structure itself, the PPOD_Release_System, this system is the composition of all components, electronic, mechanic and electromechanic components, involved into the deployment use case and finally the PPOD_Harness that is the main electronic interface with the Satellite Adapter and the Launcher vehicle as consequence.

The Release System is composed by the Starsys Qwknut, the P-POD port, torsion springs, kick springs, main spring and pusher plate [9]. The Starsys Qwknut is an electromechanic device, which restrains the P-POD port while the CubeSats are not supposed to be deployed, and it is responsible for starting the deployment sequence. The Starsys Qwknut component plays the role of the release system electronics and is the actuator of the mechanisms release system, it restrains all the deployment force until the time of deployment.[9]

The P-POD port opens in the exact time of deployment permitting the set of one to three CubeSats to exit into its orbits. Torsion springs are responsible for providing torque to help the P-POD port to open.[9]

Finally, the kick springs helps the set of one to three CubeSats to start its deployment movement throughout the POD, this movement receives potential elastic energy from the main spring turned into kinetic energy. The pusher plate supports the set of one to three CubeSats over the main spring.[9]

2. The Cal Poly P-POD mk II Internal Block Diagram

The Figure 3 presents the model of the P-POD in the form of an internal block diagram. This internal block diagram shows how the flows are exchanged between internal blocks (P-POD subsystems). As said before these flows represents physical phenomena to be studied and processed by the SE processes.

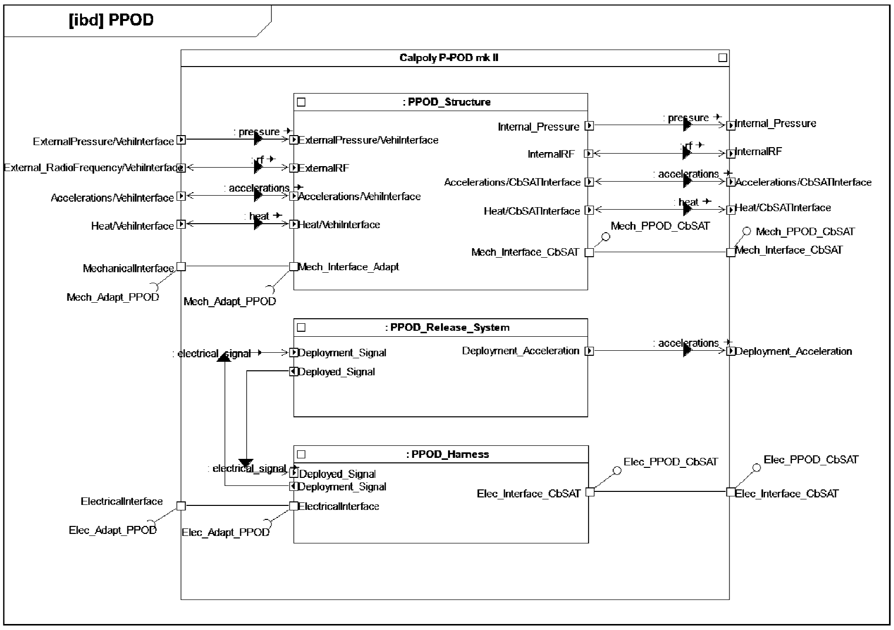

Fig. 3 P-POD internal block diagram

Some simplifications were made in for the instantiated model, the P-POD model presented in Figure 2 and Figure 3 and its flows from the reference model presented in Figure 1.

It is assumed that, as the CubeSat is supposed to be turned off during the launch lifecycle phase, it doesn´t irradiate any electromagnetic energy to its environment so the first simplification made, is to remove the RF flow flowing from satellite, in this case the CubeSat, to the vehicle.

The same approach is taken to the heat flow, as CubeSat is turned off during the launch lifecycle phase it doesn´t generate any heat.

Finally the last simplification is related to the acceleration Flow where the approach is the same of other Flows simplification, since the mass of the P-POD plus CubeSat is negligible. Nevertheless, one needs to keep in mind that coupled analysis is still necessary since the P-POD is subject to vibration and has its own vibration modes which can interfere geometrically with surroundings.

Other flow modeled is the pressure here described as the pressure inside the fairing (payload envelope) which is the pressure outside P-POD.
There are two electrical signals: (1) the deployment signal that is the deployment command from the launch vehicle and runs through the main satellite adapter to the electrical interface, and (2) the deployed signal, which is the signal to vehicle telemetry indicating that the release system was activated.
The two interfaces modeled are the mechanical and electrical. The mechanical interface carries all flows except the electrical signals which are carried through electrical interface. The both interfaces imposes constrains to the P-POD system, and are inputs in the SE processes.

3. The Cal Poly P-POD mk II Use Cases

The Use-Case diagrams indicate the uses of the system, and are refinements of behavioral system requirements. The Figure 4 presents the use case diagram modeled for the P-POD system.

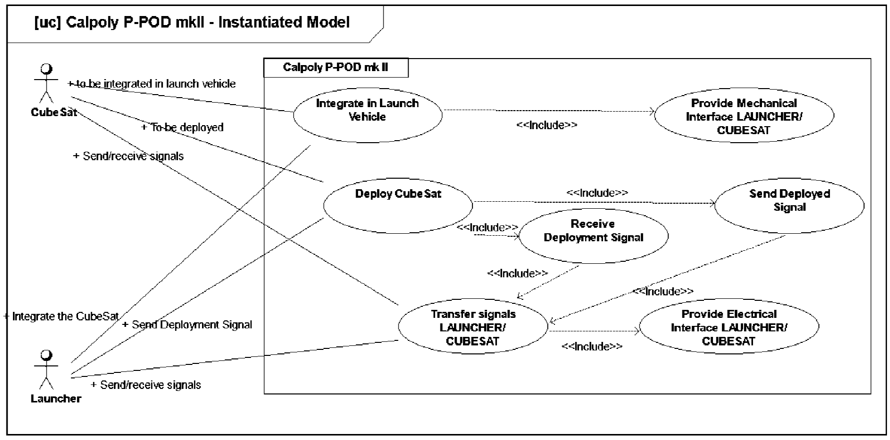

Fig. 4 Use case diagram for the Cal Poly P-POD mkII.

The actors are entities specifying roles played by a person or other system interacting with the system modeled. For the launch lifecycle phase of the P-POD system, the actors are the CubeSat and the Launcher.
The use case "*Integrate in Launch Vehicle*", represents the use of providing capabilities of the CubeSat to be integrated in the Launcher, including "*Provide Mechanical Interface LAUNCHER/CUBESAT*".
To deploy the CubeSat into its orbit the use cases are, "*Deploy CubeSat*", "*Receive Deployment Signal*", "*Send Deployed Signal", "Transfer signals LAUNCHER/CUBESAT"* and *"Provide interface LAUNCHER/CUBESAT".*

4. Parametric Diagrams

Parametric diagrams can represent relations between parameters. All flow parameters are supposed to be modeled through parametric diagrams, it can be useful for simulation, and interface constrains checking and coupled analysis.
The Figure 5 shows how the deployment signal parameters, voltage, current and power are interconnected and it is used for harness, and deployment signal driver dimensioning.

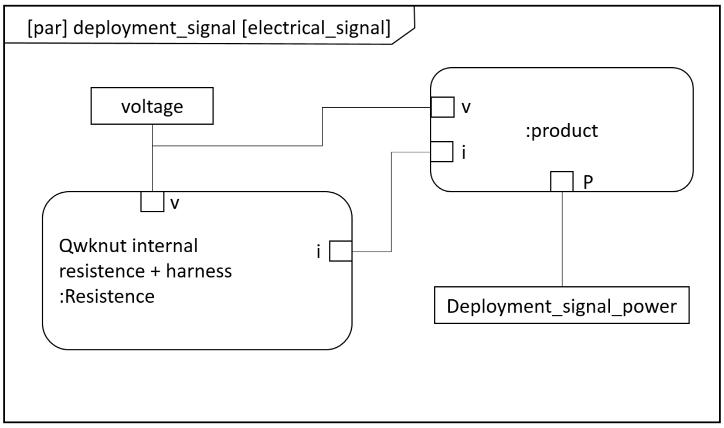

Fig. 5 The parametric diagram for the P-POD deployment signal

MBSE Support to Concerns and Constraints Checking, Verification and Validation and other SE Processes.

Another facet that can benefit from the MBSE approach is described in this session. To achieve the next level of SE processes improvements, it is necessary an integrated, consistent, and executable model. Such models and its associated data would be enablers of earlier verification and validation (V&V) of the systems [1] and they could help the regular V&V automation process and preventing it from human error.
The Figure 6 is presented from deKoning et al. (2010) [1] and it summarizes the concept of how early verification and validation (V&V) improves the SE processes allowing early system correction. Figure 6 shows the engineering V model and the "mini-V-s" cycles enabled by MBSE and shows also the European Space Agency (ESA) systems engineering lifecycle process phases (0, A, B, C, D and E).

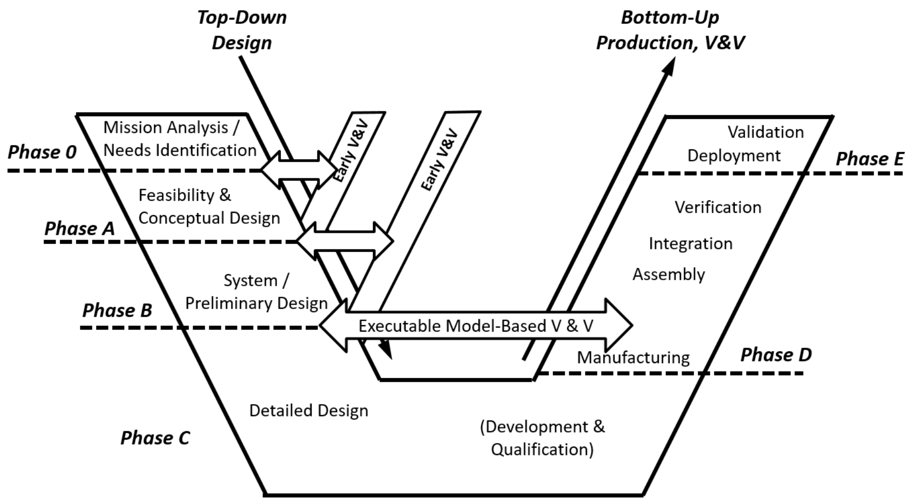

Fig. 6 An enhanced systems engineering V-model with early model-based verification and validation [1]

In the context of a MBSE, all teams must be able to access and interact with the underlying model and this model must be stored in a common repository. The model is supposed to contain all concerns and constraints of the system; because only with a consistent model, we can have a good early V&V.

1. SysML Flows and Phenomena Involved

The internal block diagram is supposed to model all physical phenomena involved on the system operation. As an example, we can see how this flow model can be used by SE, and a specific domain discipline, namely Structure studies.
For this example, we can analyze how accelerations flows into the P-POD system according to the definition presented in the Figure 7.

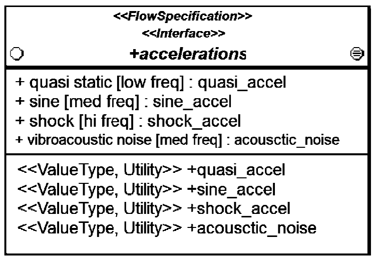

Fig. 7 Modeling accelerations flow in a SysML definition block for Structure studies

In the abstraction presented, accelerations flows have four attributes: *"quasi static [low freq]"*, *"sine [med freq]"*, and *"shock [hi freq]"*. Each one of these attributes represents one acceleration range and may serve later as inputs of verification studies.
For this study, we will concentrate only in the sine or random vibration, which are medium frequency vibration. This vibration is imposed by the launcher vehicle and has always a well-known profile and P-POD must meet this vibration profile. Each vehicle that takes the "launch vehicle" role has its own intrinsic random vibration profile.
The Dnepr Launch vehicle is taken as an example and its expected random vibration levels [10] are listed in the Table 1. These levels were modeled with the sine_accel type, as a vector. The sine_accel type is the sine [med_freq] type.

Frequency [Hz]	Acceleration Spectral Density (ASD) [g^2/Hz]
20	0,0175
40	0,0175
80	0,0175
160	0,055
320	0,0875
640	0,0875
1280	0,0425
2000	0,0125

Table 1. Dnepr Launch Vehicle Random Vibration Levels.

Now we have the random vibration model complete and this model can be used in many systems engineering processes, one example is the launch vehicle trade study and P-POD qualification levels definition. There are two studies that random acceleration studies were made and the results are showed in Figure 8, presented by A.Toorian et. al. (2005)[11] and Figure 9 [12]. For these two studies the model could be used.
It is possible to automate these studies with a MBSE approach and the model presented in this paper. These two studies were made for P-POD mk II changes project[11] and UbatubaSat picosat qualification levels definition study[12]. These both studies aimed to define vibration qualification levels allowing the system to be used with launcher vehicles previewed.

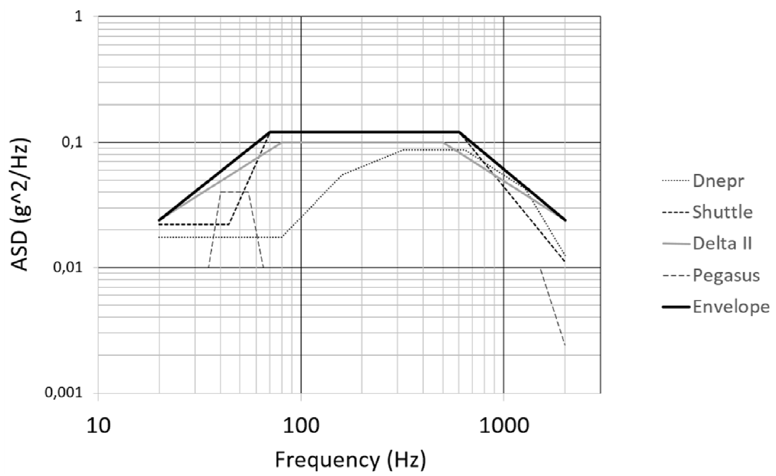

Fig. 8 Various Random Vibration Profile – P-POD qualification [11]

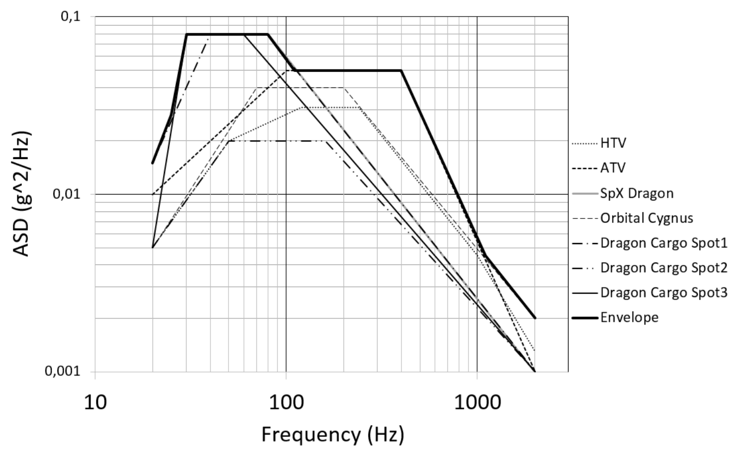

Fig. 9 Random vibration level for the UbatubaSat [12]

Another early verification process, which can be made with this model, is a coupled load analysis of the P-POD. One can simulate the system dynamic using this flow combined with a finite element model of the P-POD.
Since one can have an early made geometric computer aided design (CAD) model, one can made model in preliminary stages of the project, one can generate a finite element model from this CAD model and do this simulation.
One can use this approach with any flow modeled in this model. Then Early Validation can happen with any phenomena modeled by flows and computational tools available to simulate it.

2. System Use Cases Validation

For system validation, the system itself must meet all use cases since they are behavioral requirements refinements. The modeled use cases generates the system validation matrix presented in Table 2. This matrix considers early V&V as presented in this paper.
For this system SE process, this matrix was not supposed to be necessary, since there are not many use cases, but with systems that are more complex, this matrix helps to generate test cases. Finally, the model can helps to automate all steps of this process to avoid human interference and consequent human error.

Use - case	Early Validation Method	Phase	Validation Method	Phase
Integrate in Launch Vehicle	Mechanical Integration - Virtual Assembly Electrical Integration - Components Interface Simulation	C	Integration and Tests	D
Provide Mechanical Interface LAUNCHER/CUBESAT	Mechanical Integration - Virtual Assembly and analysis	C	Integration and Tests	D
Deploy CubeSat	Simulation	B, C	Integration and Tests	D
Send Deployed Signal	Simulation	C	Integration and Tests	D
Receive Deployment Signal	Simulation	C	Integration and Tests	D
Transfer signals LAUNCHER/CUBESAT	not possible		Integration and Tests	D
Provide Electrical Interface LAUNCHER/CUBESAT	Simulation	C	Integration and Tests	D

Table 2. Validation Matrix generated from use cases

3. The Relation Between Requirements, V&V and the Model

The P-POD system mission statement is comprised of three basic philosophies [9] modeled into the requirements diagram presented in Figure 10

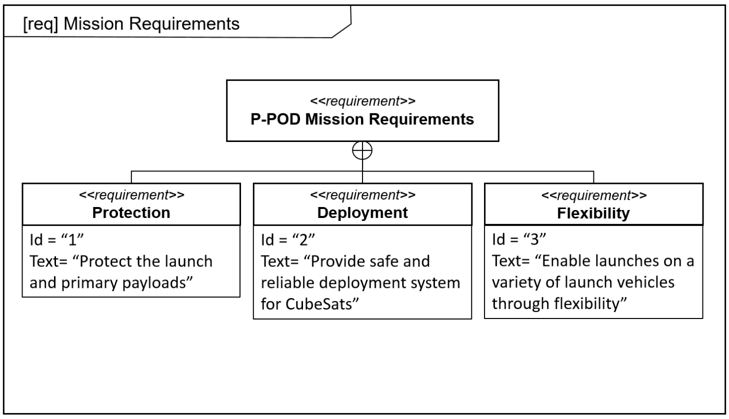

Fig. 10 Mission Requirements – Requirement Diagram

The Figure 11 presents the "Deployment" requirement relations modeled

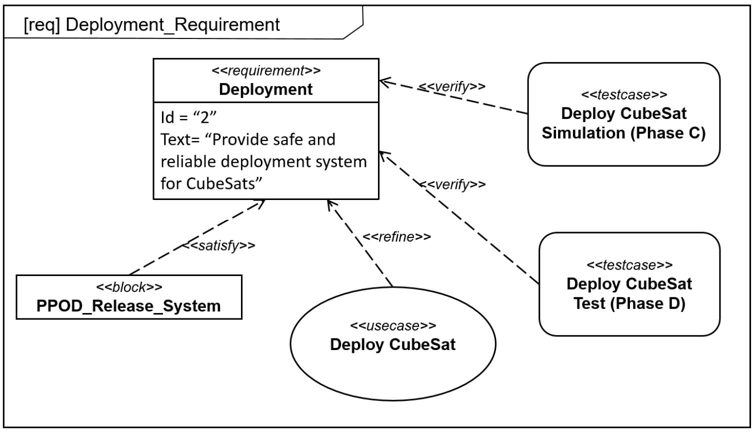

Fig. 11 Deployment Requirement – Requirement Diagram

Every stakeholder's necessity derives one or more requirements. The system validation means to check if the system meets all stakeholders' necessities. It is important to verify if the system is meeting every requirement and in order to do so one must choose V&V methods.
Verification has its own necessities, for example, for a test we must have the system to be tested and the test procedure, for a simulation we must have the model, the software and the data involved. As the model is supposed to have all relations of a requirement, this model supports programmatic, project schedule and data sharing.

The models presented are still not executable models but it may help in early V&V of the project saving money and time. Further increments to the presented SysML modeling and its application with proper metrics collections will drive future work.

Conclusion

Dealing with systems´ complexity demands sometimes rigorous and formalized SE practices and space systems are a natural candidate due to the complexities it has to deal with. This work glimpsed how a MBSE approach for SE processes is capable to discover, to describe, to record and deal with some concerns and constraints of the satellite/launcher interface.
A CubeSat project was taken as a case study since these projects are always executed by different discipline students, and by different institutes. In this case, a MBSE approach is needed as an effective approach in terms of data recording and sharing.
In order to take advantage of MBSE all project participants must share a consistent model. The MBSE approach is not only based on the model itself, but it is based on the MBSE processes, tools that implement it, the necessary infrastructure and people. Therefore, a simple reference model to the underlying interface problem was presented as a basis for MBSE approach implementation especially on modeling some project concerns and constraint as well as standardizing data exchanges between tools and people.
Future work will focus in two aspects: (1) The models presented still need refinements and (2) Proper metrics for evaluating effectiveness.

References

[1] H. Peter de Koning, H. Eisenmann, M. Bandecchi, *Evolving Standardization Supporting Model Based Systems Engineering*, 2010, SECESA Conference Lausanne, Swiss, p. 17.-19.

[2] INCOSE, *What is Systems Engineering*, accessed 31 December 2015, and available at http://www.incose.org/AboutSE/WhatIsSE.

[3] P. Staff, P. Staff, *Guide to the Project Management Body of Knowledge (PMBOK Guide)* (4th Edition), 2000, Project Management Institute, Newtown Square, USA.

[4] S. Friedenthal, A. Moore, R. Steiner, *A practical guide to SysML*, 2011, Morgan Kaufmann, Amsterdam,

[5] IEEE, I*EEE Std 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology*, 1990, IEEE.

[6] Sysml.org, SysML.org: *SysML Open Source Specification Project*, 2015, http://sysml.org/ (accessed 31 December 2015).

[7] Agência Espacial Brasileira - AEB, *Programa Nacional de Atividades Espaciais - PNAE: 2012 - 2021*, 2012, Agência Espacial Brasileira - AEB e Ministé-

rio da Ciência, Tecnologia e Inovação - MCTI, Brasília - DF.
[8] D. Kaslow, L. Anderson, S. Asundi, B. Ayres, C. Iwata, B. Shiotani et al., *Developing a CubeSat Model-Based System Engineering (MBSE) Reference Model - Interim Status*, in: 2015, IEEE Aerospace Conference.
[9] A. Toorian, *Redesign of the Poly Picosatellite Orbital Deployer for the Dnepr Launch Vehicle,* 2007, Master of Science in Aerospace Engineering, California Polytechnic State University.
[10] International Space Company Kosmotras, *Space Launch System DNEPR User's Guide*, 2001, ISC Kosmotras, Moscow.
[11] A. Toorian, E. Blundell, J. Puig Suari, R. Twiggs, *CubeSats as Reponsive Satellites*, 2005, in: 3rd Responsive Space Conference, American Institute of Aeronautics and Astronautics, Los Angeles, p. 5.
[12] Dos Santos, W. A. et al., *A Langmuir Probe Payload Adaptation for CubeSats-TubeSats*, 2014, 1st Latin-American CubeSat Workshop, Brasilia, Brazil.