# Design process of a reusable Space Tug based on a Model Based approach

FABRIZIO STESINA, SARA CRESTO ALEINA, DAVIDE FERRETTO, NICOLE VIOLA
Department of Mechanical and Aerospace Engineering
Politecnico di Torino
Corso Duca Degli Abruzzi 24, Torino
ITALY
fabrizio.stesina@polito.it, sara.cresto@polito.it, davide.ferretto@polito.it, nicole.viola@polito.it

*Abstract:* - In the last years, space agencies are showing an increasing interest in space tug systems concept for a large range of future applications. The space tug is a spacecraft able to transfer payloads from Low Earth Orbit (LEO) to higher operational orbits. It allows the reduction of the spacecraft mass because some subsystems decrease in term of complexity (i.e the propulsion system), and the improvement of the spacecraft payload/platform ratio. The present paper deals with the design of a space tug involved in on-orbit satellite servicing missions. The design process is led following a proposed Model Based System Engineering (MBSE) tool-chain. This solution allows an effective classification, traceability and verifiability of requirements among the various phases of the design process, combining the main features of specific tools and software, such as Doors, Rhapsody, and Simulink. The crucial point is to guarantee by automatic exchange of information and models among the different phase of the product life cycle. The paper shows the design at different levels: mission and stakeholders analyses, functional analysis, concept of operations, the space tug logical and physical architectures and the sizing of the main on board subsystem. The details of the recursive process of requirements definition is provided highlighting how they derive from the mission scenario, the mission architecture, the concept of operations and the functional analysis and, in general, how the proposed sequence of tools simplifies and gives effectiveness to the design.

*Key-Words:* Model Based System Engineering, space tug, software tool-chain

## 1 Introduction

In the recent years, space tug received great attention into the aerospace field because it can increase the effectiveness of future space missions, in terms of cost reduction and resources saving. Traditional telecommunication and navigation for the positioning or repositioning of the satellites of a constellation, or space exploration missions for reach more convenient orbits that facilitate the escape manoeuvres, can take advantage from a Space Tug's capabilities.

A space tug vehicle is designed to rendezvous and dock with a space object; make an assessment of its current position, orientation, and operational status; and then either stabilize the object in its current orbit or move the object to a new location with subsequent release. The most actual example of Space Tug is SHERPA system [1] that will be scheduled for launch in 2017. It is a Spaceflight Inc. proposal for an orbital tug to be combined with SpaceX's Falcon 9 launch and could transfer small and secondary payloads to their operative orbits. This Space Tug consists of a ring structure hosting the payloads and of a VASIMR (Variable Specific Impulse Magnetoplasma Rocket), theoretically capable of carrying several tons of payloads from LEO to Low Lunar Orbit (LLO) in few months.

More in general, Space Tug can find application in a wide range of on orbit operations and missions (Figure 1): the most promising examples are the satellite servicing [2] and the support to space exploration.

In the satellite servicing missions, a Space Tug can be a key element for payloads retrieving [3], maintainability actions (i.e. take again the control of GEO satellite that have lost attitude control and repositioning or station keeping the satellite [4]) and refuelling [5], and cargo resupply service [6]. The use of this kind of system for orbital transfer manoeuvers allows significant simplifications in satellite design, especially considering the propulsion system, with a consequent mass and volume reduction of the satellite [7]. Analogous considerations can be applied for the on orbit refuelling because spacecraft can be launched without fuel, reducing sensibly the lift-off mass. In addition, small launchers can be optimized to reach LEO, increasing the mass available for the payload that is no more supposed to reach the operative orbit through the help of dedicated on-board systems or through launcher stages.

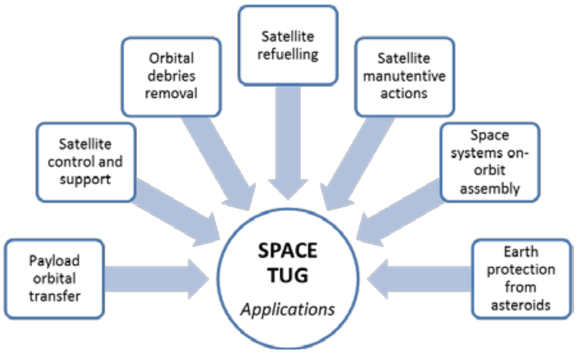

Figure 1: Possible missions based on a Space Tug.

A great improvement on the Space Tug mission can be provided applying the *reusability* concept. It means that the same Space tug can repeat more times their operations within the same mission. In [8], authors explore the reuse of in-space transportation systems, with focus on the propulsion systems. It begins by examining why reusability should be pursued and defines reusability in space-flight context. Moreover, reusable Space Tugs can contribute to solve the overcrowding of some Low Earth Orbit and the actual democratizing of the space tends to dramatically enlarge the problem [9]. A space tug could be the reusable chaser spacecraft catching the targets and transferring them on a parking orbit or deorbiting them.

Space agencies Roadmaps [10] pay attention on Space Tug concepts for the space exploration. The first application of a Space Tug in the space exploration context is the transfer of interplanetary probes from lower Earth orbits to escape orbits [11]. Space Tug destination could be an asteroid: in [12] authors analyse the scenario in which, supposing a potential asteroid impact with Earth, an unmanned space tug would rendezvous with an incoming asteroid, attach to its surface and slowly push the body. An additional application can be related not only to large space systems but also to smaller ones: indeed, it is possible to consider the use of small launcher combined to a Space Tug to deliver in higher orbits also CubeSats or other Small Satellites for interplanetary missions, so designed to operate in orbit not easily reachable by small launchers [13].

A second application is the on orbit assembly of larger spacecraft or planetary outposts. Aerojet Rocketdyne has demonstrated a significant cost reduction and logistics simplification in the use of a tug in the assembly and servicing of a cislunar deep space habitat [14], by comparing a cargo delivery mission to the Earth-Moon Lagrange points (EML1 or EML2) using a solar electric propulsion (SEP) stage (or “tug”) versus the all-chemical approach.

The present paper deals with the design of a Space Tug that should support satellites deployment on orbit. The design is one of the outputs of STRONG (Systems Technology and Research National Global Operations) project that has the objectives both to improve the national space operability in terms of access to space and to increase the Italian industrial capability to manufacture a Space Tug. The design is carried out through tools typical of the Model Based System Engineering (MBSE); in particular a tool-chain of software has been constituted in order to merge and share their main features tailoring an effective tool (in terms of portability and flexibility) and that provides fundamental design outputs, such as the mission analysis, the stakeholder needs analysis, the functional analysis, budgets, and requirements definition and management.

The paper describes the Model Based System Engineering (MBSE) approach followed for the design of the Space Tug and the STRONG system (Section 2). In Section 3 the space tug design is described in details, showing the main outputs (functional analysis, block schemes, mission scenario and budgets) deriving from the application of the tool-chain. Section 4 concludes the article with remarks and suggestions for future improvement.

## 2 Method and toolchain

The Space Tug mission and system design was led using a tool-chain of commercial software that favours a MBSE approach. This approach provides advantages compared with that proposed in [15] due to:

- The capability to generate and trace requirements, avoiding repetitions, misunderstandings, conflict;
- The capability to produce models to share among the partners of a project instead long and boring documents;

- The possibility to automatically and consistently upgrade without loss of information.

The design process is described in Figure 2: the main output is the definition of specification through the identification of a step-by-step method that led the designed system to be compliant with stakeholders' needs and imposed constraints, key drivers and contour conditions.

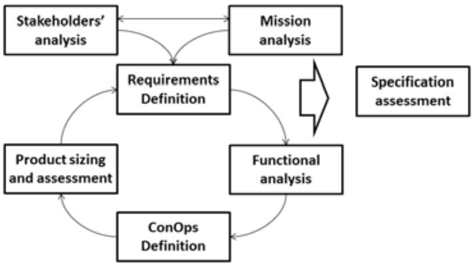

Figure 2: Design process

Mission Analysis and Stakeholders' Analysis concur to the definition of the primary and secondary objectives of the project: in particular, primary Mission Objectives and Constraints are directly derived from the defined Mission Statement while the stakeholders' needs and expectations define the Secondary Mission Objectives. The stakeholders have to be identified and categorized. As proposed in [16], they can be sponsors (i.e. people who establish mission statement and fix constraints on schedule and resources), operators (i.e. people in charge of controlling and maintaining the products), end-users (i.e. people that receive and use products and capabilities) and customers (i.e. users who pay fees to utilize a specific space mission's product). The output of the stakeholders' analysis and the mission analysis is a complete set of requirements that constitute highest level requirements that will be the parents of the major part of the requirements present in the final specification.

Functional analysis is the process for individuating the required functions at mission and systems level till the most basic functions. It provides a logical decomposition able not only to define technical requirements and the relationships among them, but also to decompose the parent requirement into a set of models and their associated lower level requirements. The main tool of the Functional Analysis is the Functional Tree that defines hierarchically the functions. Functional Flow Block Diagrams (FFDB), which is a particular kind of tool that gives further information about timing and functional logical sequences, are adopted very often [17]. Being related to functions and not to products, this kind of tool shows what has to happen in the system without referring to physical solutions.

Concept of operations (ConOps) is a way to show the physical solutions that can be applied to solve the Mission Statement is the ConOps analysis. In particular, the definition of the ConOps should consider all the aspects of the mission to be performed, including integration, test, launch and disposal. Typical ConOps information are the mission phases, modes of operation, mission timeline, Design Reference Mission (DRM) and/or operational scenarios, end-to-end communication strategy and/or command and data architecture, operational facilities, integrated logistic support and critical events. Usually, in preliminary phases of the design process, it is common to have one or more operational scenarios and architectures, but only one should be the preferred solution of the design.

The Functional Analysis and the Concept of Operation shall lead to the selection of the preferred design solution. This process starts from the definition of the products that perform the functions. Functions/Products Matrix is a valuable tool in this sense: checked cells of the matrix are used to identify connections between functions and products, drawing the Product Tree. Knowing the interfaces, it is possible to build the Functional/Physical Block Diagram, a graphical representation of the connections among all the products at each level of detail. Contextually, mass, power, link, data, and delta V budgets allow to size the system. Moreover, any elements of the diagram can be sized in terms of performance required and constraints from the budgets. The final output of the design process is the specification definition. Once the preferred solutions have been selected and the refinement level has been completed the design is translated in to the end product specifications that are used to drive the verification phases through the system models built during the design process just described. For this reason, the authors consider the capability to manage the requirements in a safe and effective way as an essential and crucial element within a project frame.

Technical reviews of the data and analysis, including technology maturity assessment [18], are an important part of the decision support packages. The preferred solution is taken evaluating the system in terms of functional capabilities, reached performance, safety and reliability. The taken decisions are generally used as input for the configuration management system, that changes them into the chosen system solution. The selection of a preferred solution has to be supported by trade studies, which help to complete the selection with much more confidence. It is important to remember that this process is recursive and iterative, thus aimed at increasing its detail and, consequently, the resolution of the design, from the highest mission and system level until the component level.

## 2.1 Toolchain description

In order to produce a complete specification and the models used (and reused among any product lifecycle phase), a toolchain of commercial software was defined (Figure 3). This toolchain is thought in order to take advantages from the main features of each element and enhance their capabilities of sharing information and models and updating them, reducing the risk of loss of information and of errors due to misunderstanding.

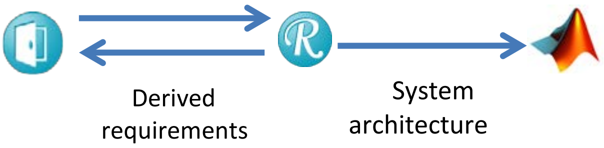

Figure 3: Toolchain structure

The process is mainly based on SysML modelling in Rhapsody® for what is related to Stakeholders and Mission analyses, Functional analysis and ConOps analysis, where the main diagrams are created to describe system functional, operational and physical architecture. The iterative requirements definition is jointly performed in Rhapsody®, where requirements are defined, and in DOORS®, where they are stored, classified and ordered. Functional and ConOps analyses are also divided in sub-phases in order to maintain consistency with the system characterization, notably STRONG level (top level), segment level, system level (where the Tug is defined), subsystem level and equipment level. Finally, the opportunity to integrate Simulink with both DOORS® and Rhapsody® allows completing the design loop, from requirements up to the implementation of the simulation for the system design, after the definition of functional and operational aspects, back to requirements specification again.

The IBM Dynamic Object Oriented Requirements System (DOORS®) [19] was selected as main hub for the requirements, due to its wide application in MBSE and the capabilities of supporting connection with multiple design tools and software. DOORS® is a robust database organized in a predefined way, with projects, folders and modules, that permit to collect, classify and link requirements. Modules are the most important objects. Formal Modules contain the list of requirements, organized following the design phases, while Link Modules map the links among the requirements inside the specification, in order to specify the derivation structure of low level objects from high level ones. With this hierarchy, each requirement can be easily identified and traced during the whole design process. Particularly, the traceability which is internal to the specification is already guaranteed by the Link Modules, whilst the traceability with the system architecture (external) can be implemented through the deployment of the MBSE tool used for the design, which shall be interfaced with DOORS®. This is the case of IBM Rational Rhapsody® product, a general purpose modeling environment for UML and SysML. It is independent from the adopted System Enginering methods and allows the coverage and impact analysis of requirements. Within the toolchain, Rhapsody is used to characterize functional and operational aspects of the any element in the different phases, from stakeholders and mission analysis to the lower levels of system definition (subsystems, components, devices etc.) using SysML. Rhapsody is adopted to create the fundamental views and diagrams, to trace and allocate requirements to functional and physical architecture, to establish the interfaces within the subsystems and components of the system in order to prepare the numerical simulation in terms of block diagrams and system breakdown. The two-ways link with the requirements database allows synchronizing and updating the specification either from DOORS® or from Rhapsody® itself, guaranteeing an effective integration of requirements and system elements. Moreover, the possibility of preparing the data for further types of analyses, as simulation, creates a seamless oriented toolchain, reducing the time related to models set-up in separate environments. Particularly, the connection with Simulink® is available with a dedicated import/export facility even if the tool is also able to support the interoperability standard Functional Mock-up Interface (FMI) for model exchange [20]. The final choice was also driven by the availability of some in-house tools aimed at Model-Based verification that are currently supporting integration with Matlab/Simulink® [21] and which will be used in future works for the verification campaigns.

## 3 Space Tug design

The toolchain has been applied to the STRONG space tug design. Starting from the mission statement definition and the stakeholders' analysis, the high level requirements are derived. They addressed the functional analysis and the concept of operations related to the STRONG Space Tug [15]. Budgets, products interfaces and product sizing have been defined and the specification have been obtained at the end of the first design iteration.

Following the method described in Section 2, the first step is the mission analysis that consisted in the mission statement definition, and the definition of the primary objectives. Going into details, the mission statement is written as: *To improve the national space operability in terms of access to space by providing a transportation system capable to transfer satellites platforms from Low Earth Orbits to operational orbits and back, relying on Italian space assets.*

This allowed to derive the Primary Mission Objective as "To perform satellites taxi between Low Earth Orbit (LEO) and the operational orbit" constrained by "To use Italian space assets"

The Stakeholder analysis has been performed: the relations among stakeholders and primary and secondary mission objectives have been represented in a Use Case Diagram (UCD) that stands as graphical representation of the mission statement. Figure 4 shows an example where the use cases specify the objectives that the stakeholders want to reach by using the system, whose borders are sketched by the boundary box in the centre of the figure, while the stakeholder themselves are clearly shown outside of it. In particular, the Primary Mission Objective, placed in the centre of the box (in bold), is connected to the Primary Mission Constraint (small grey box) and to the other use cases that represent the Secondary Mission Objectives, summarized as follows:

- To explore new mission concepts for future space exploration
- To validate selected critical technologies (enabling this operative scenario)
- To enhance the cooperation between industries and universities
- To enhance reusability
- To interface with international space facilities
- To enhance modularity in interface segments
- To increase the Vega usage
- To have standardized interfaces
- To receive data and transmit commands from/to ground
- To exploit existing Ground facilities

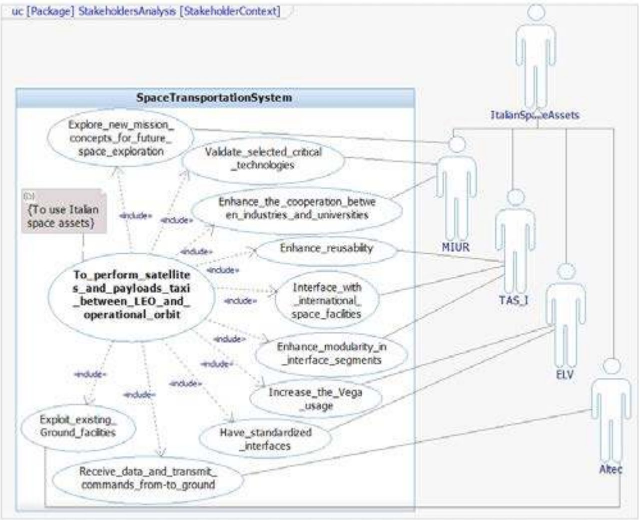

Figure 4: Use Case Diagram for Stakeholders Analysis

SysML dependencies ("include") are used to state the relations between primary and secondary objectives, whilst generalization is used to specify the stakeholders belonging to Italian space assets (Italian Ministry for education and university – MIUR, Thales Alenia Space Italy – TAS-I, European Launch Vehicle – ELV and Altec S.p.A).

From these objectives, the first set of mission requirements can be derived through the Requirements Diagram (RD), where use cases are traced onto them. The summary of these "trace" links can be represented through a dedicated matrix, where the rows indicate the objectives and columns stand for the requirements.

Requirements are then synchronized to DOORS and stored in a dedicated Formal Module, which is a model-based version of a common requirements document.

This first set of elements and relations is fundamental for the next Functional and ConOps analyses since it provides the basis for the traceability links that will be populated and extended during the process.

Functional analysis highlights the functionalities through the creation of the functional tree, which is implemented thanks to a Block Definition Diagram (BDD). This diagram represents the functional breakdown for the specific level of analysis, showing the hierarchy levels among the blocks. For example, Figure 5 represents the BDD for the breakdown at segment level. Following a “top-down” approach the functional tree is progressively populated after any iteration of the design process. In this case, the Top Level Function “To improve the national space operability in terms of access to space for satellite platforms” is the main SysML block placed at the top of Figure 5. Segment level functions derive directly from this block, and they have been defined as:

- To reach LEO
- To perform satellites transfer from LEO to operational orbit
- To retrieve satellites from operational orbit to LEO
- To reenter on Earth payloads loaded on board satellites once completed their operative cycle
- To perform refueling on orbit
- To support mission execution

Each block is described by an Activity Diagram (AD) where the relations among the functions are highlighted and the sequence of their execution is presented. The diagram helps the derivation of the low level functions and shows an important sketch for the further definition of ConOps architecture. In fact, the defined functions generate new functional requirements that are transferred to DOORS® Formal Modules, and linked to functional block to keep the traceability path unambiguous.

The functional requirements at each level are also linked to higher level requirements in Rhapsody® to highlight the derivation process, whilst it is possible to replicate this kind of relations in DOORS® thanks to a dedicated Link Modules. The Link Module is a powerful tool to trace the relations among requirements directly within the database, exploiting the so-called internal traceability. Different Link Modules, which are represented as a sort of matrices, have been defined by establishing proper link sets between the Formal Modules related to the several phases.

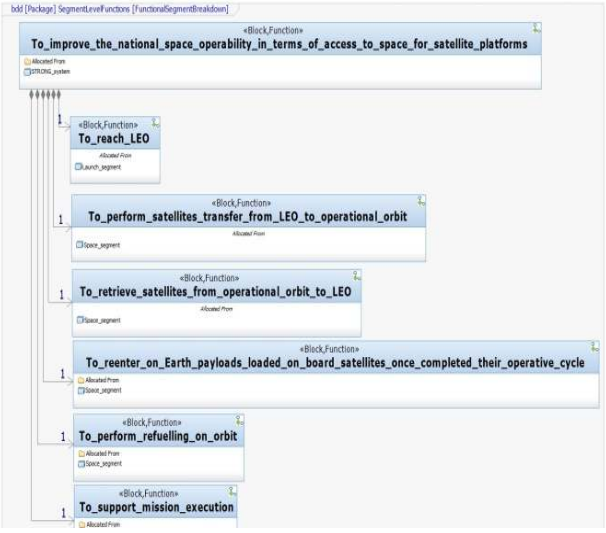

Figure 5: BDD for segment functional breakdown of STRONG system

Link Modules allow browsing the derivation structure directly within the requirements database, from mission requirements to device requirements. Figure 6 shows an example of Link Module, established between mission and segment requirements: the first mission requirement generates six son-requirements at segment level, immediately visible from the matrix thanks to the blue square tags.

The same views can be created also in Rhapsody® where dedicated matrix layouts can be configured in order to obtain a visual summary of requirements derivation and functions-requirements coverage. This process has been replicated for each level building a high number of relations among model elements and requirements, and constituting a solid multi-tools platform for traceability.

The ConOps analysis defines mission scenario at different level of depth. For this application, use cases represent the phases, sub-phases or operational situation where the system shall be able to work, while the small boxes are the parts of the system involved during these phases, defined coherently with the level of the ConOps analysis (segment, system, subsystem etc...).

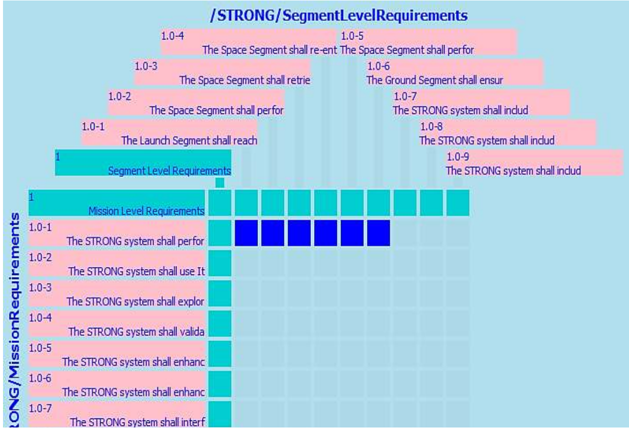

Figure 6: Link module used to establish relations between Mission and Segment Requirements.

Figure 7 shows the UCD for segment level ConOps, where the links between STRONG mission and the other use cases are highlighted, together with the associations with the products at segment levels (which are inside the boundary box because they are part of STRONG system). In particular, the STRONG mission use case “include” integration and tests, launch operations, science operations and disposal operations. Launch segment, space segment and ground segment are then associated to the different use cases.

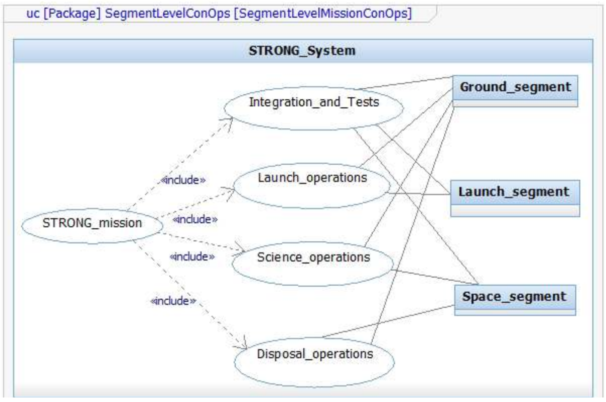

Figure 7: Example of Use Case Diagram for segment level ConOps

ConOps analysis represents the most interesting field of the modelling activity, since the use of different behaviour diagram of the SysML allows representing several aspects and views of the system in operations. Each use case can be, in fact, characterized through Sequence Diagrams (SD) and/or State Machine Diagrams (SMD) to specify sequences and modes of operations of the system, defining the so-called use case realization. These diagrams can be used to describe the sequence of tasks that the systems and subsystems shall be able to accomplish, together with the sketch of the input/output relationships as function of phase time. In order to present a more specific example of this implementation through behaviour diagrams, Figure 8 shows an overview of SD related to the Space Tug refuelling phase, which is a sub-phase of the Science Operations use case (i.e. at system level).

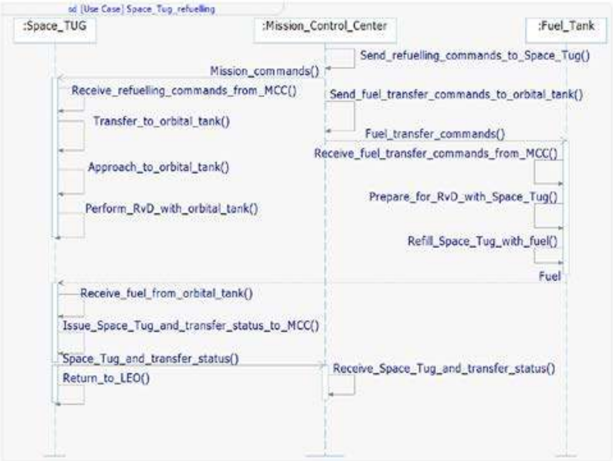

Figure 8: Sequence Diagram describing the Space Tug refuelling phase

In the SD, the time axes is vertical (with positive direction from the top to the bottom of the diagram), so it is possible to indicate the different tasks (close loop arrows) operated by the system products and the massages or information exchanged among them (horizontal arrows). In this case, the Space Tug is responsible for approaching the Tank, performing RvD and receiving the fuel, while the Tank has to actively collaborate for the RvD and to provide fuel flow. The MCC provides commands and receives feedback.

Since the different tasks are organized directly on the products, represented by the vertical lines in Figure 8, the allocation is already guaranteed. Moreover, the connection between ConOps and Functional analysis is based on dedicated traceability links established between the use cases (mission phases) and functions, which are conceived to point out where a specific functionality is used during the mission operations.

The ConOps include the following phases: Space Tug deployment, Satellite platform deployment, Space Tug refuelling (Figure 9). In detail, considering this particular scenario as reference and the listed systems to be used, the first missions starts with the launch, through VEGA, of the space tug at a launch orbit (350 km of altitude and 5° of inclination). After being released in orbit, the tug is supposed to move autonomously in its waiting orbit (500 km of altitude and 5° of inclination) and remain there till the launch of a satellite platform. On the contrary, the tank is launched directly to the waiting orbit with a Soyuz launch. Consequently, VEGA launcher will bring P/Ls to be transferred, in the same launch orbit, while the tug has to reach the P/L. The maximum mass for a single P/L to be transferred is 1000 kg (from stakeholders' analysis). Once in the same orbit, the P/L is then docked to a Space Tug for the manoeuvres, thus allowing minimizing the propulsion on the platform and maximizing the P/L mass. Launch orbit and waiting orbit are supposed to be different. Once the tug has docked with the satellite platform at the launch orbit, the transfer towards the P/L final operational orbit begins. From stakeholders' analysis the maximum operative orbit to be reached is a Geostationary Earth Orbit (GEO) of 36000 km of altitude and 0° of inclination. After having released the P/L, the tug moves to the waiting orbit to perform the first refuelling. After that refuelling operations have been completed, a second mission can start. In particular, 4 P/L transfers are supposed before a new Orbital Tank has to be provided.

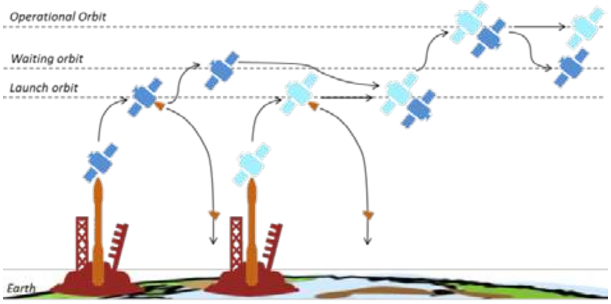

Figure 9: Nominal electric space tug mission concept (refuelling phase is not reported)

The next step is to characterize the system through the definition of the products, their interface and the calculation of the budgets. Product tree was again represented as BDD. In this paper, more details are provided, as example, at Space Tug subsystem level.

Figure 10 shows the BDD concerning the Space Tug (main block) product breakdown. The Space Tug is a product block at system level, while the other represent its Sub-systems. The different blocks contain the information about the functions that they are responsible to accomplish since functions have been allocated to them through proper dependency links. Functions/products matrices represent the mutual relations between functions and products.

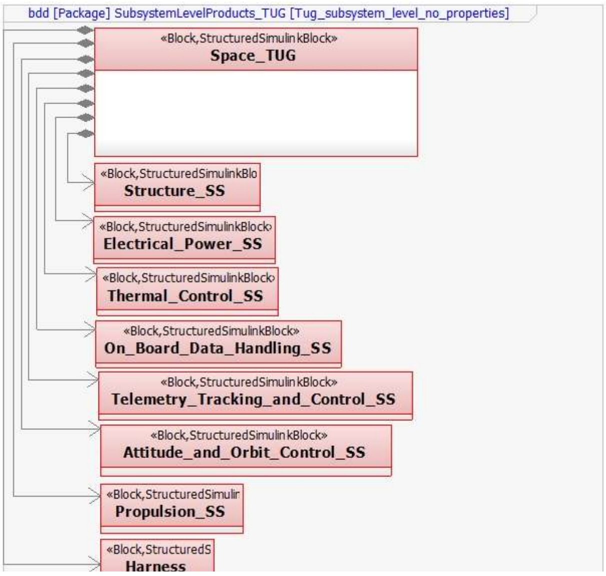

Figure 10: Block Definition Diagram for the Space Tug product breakdown

Requirements are then connected to products through a stronger type of relation, the so-called satisfaction, meaning that the specific aspect stated by the requirement will be formally accomplished by the related product, even if a real verification is not yet present. This type of dependency concludes the path of the requirement, which started from the derivation, passed through the trace link onto the function and ends now onto a product. Another important aspect to be considered within product architecture definition is related to the formalization of the internal interfaces among the product themselves, that can be useful not only to sketch the topology of a specific layer of the system, but also to introduce other types of analysis, like simulation and verification campaigns early in the design process, and to raise the automation level related to data sharing among tools. For these reasons, Internal Block Diagrams (IBD) have been created to specify the internal structure of the blocks and to define the proper interfaces among them. Figure 11 shows the IBD for the Space Tug block. In particular, the Space Tug will be equipped with a certain number of sub-systems, including Propulsion Sub-system, Electrical Power Sub-system (EPS), Thermal Control Sub-system (TCS), Attitude and Orbit Control Sub-system (AOCS), On-Board Data Handling (OBDH) Sub-system, Telemetry Tracking and Control Sub-systems (TT&C), Structures Sub-system, Harness Sub-system [8]. These sub-systems can be directly seen within the IBD of Figure 11 since the diagram is conceived at this specific level, thus representing the internal layout of the Space Tug with its interfaces. These are implemented through directional flow ports (small arrows) following SysML notation. It is interesting to note that the interfaces on the boundary of the diagram (right and bottom of the IBD), related to commands I/O, structural coupling with launcher, satellite, orbital tank and refuelling system, are the ports exposed outside the Space Tug. On the other hand, the interfaces defined on the main subsystems will need to be managed also at equipment level. In particular, the Propulsion sub-system includes the thrusters, the reaction control system, propellants tanks, all the interface and feeding devices needed to provide propellant to the thrusters and the active refuelling system to interface with the Orbital Tank. In particular, electric thrusters with a power of 9.4 kW will provide a constant thrust equal to 480 mN and an Isp of 2500 s. In addition, thrusters' power ratio is assumed to be of about 50 mN/kW. A very impacting sub-system is the EPS, since the tug is equipped with electric thrusters and, this system is in charge of providing, storing and distributing power to the other sub-systems. EPS mainly includes solar arrays (with an area of 75 $m^2$) and batteries (with a capacity of 9.6 kWh and a specific energy of 175 Wh/kg). Another enabling sub-system is the AOCS, aimed at stabilizing the system and orienting it in desired directions during the mission despite of external disturbance torques.

The attitude control is also particularly critical for the rendezvous and docking manoeuvres required. Finally, another compelling sub-system is the structure one, that supports all the other sub-systems and includes the attachment interfaces with the launcher and the ground support equipment interfaces.

The system architecture shown in Figure 11 can be eventually replicated in Simulink® (Figure 12), since the ports, the signals and the variables can be exported after a dedicated set up procedure. Blocks are replicated with the same interfaces defined within the IBD. In this case, the interfaces with external elements are highlighted by the numbered ports, while the internal ones are similar to the SysML notation.

In general, IBD can be used as Model-Based version of Physical Block Diagram and they can be customized at user discretion for multiple purposes.

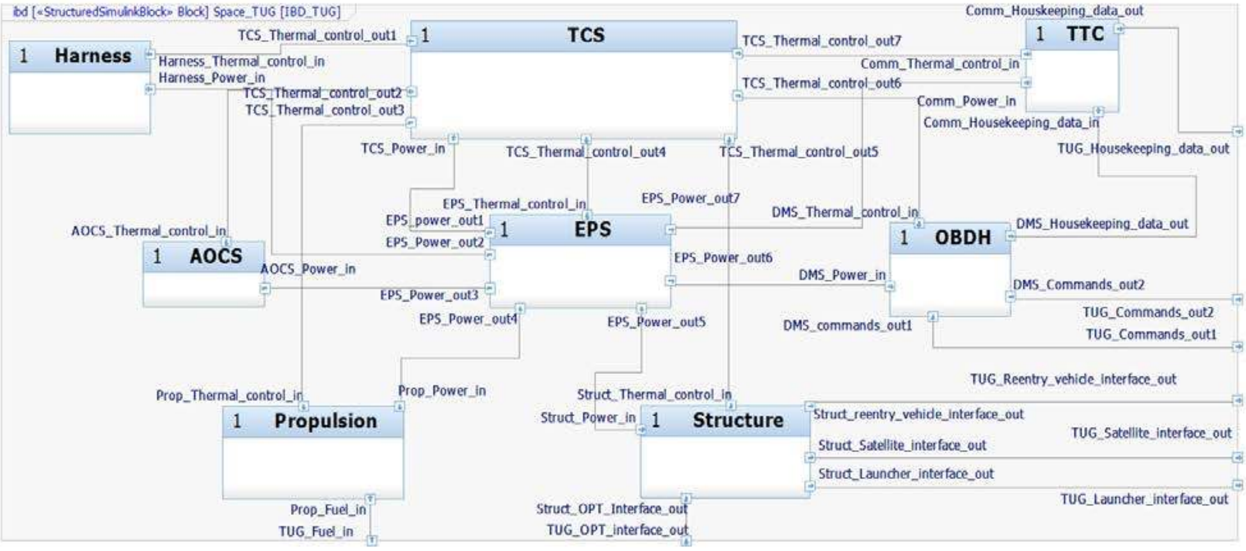

Figure 11: Internal Block Diagram for the Space Tug

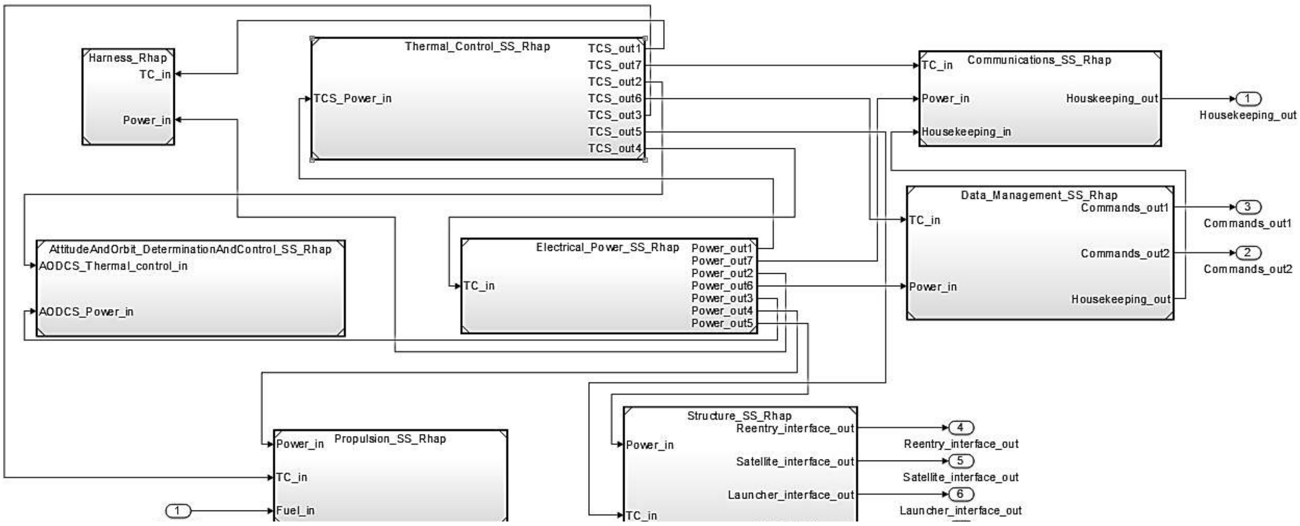

Figure 12: Simulink model of the Space Tug based on the IBD defined in Rhapsody

As it was described for functions, the analysis of products breakdown and architecture is replicated onto the different phases up to device level, since the hierarchy defined within the functional model is preserved. The different blocks can be enriched with proper dynamic behaviour to perform simulations as well as with updated interfaces and ports. These data can be synchronized back to Rhapsody® in order to maintain consistency between the two models, using the reverse connection process. Moreover, the allocation of the requirements is possible also in Simulink, since each element is represented as referenced block, capable of interfacing with the Requirements Management Interface (RMI) toolbox, properly configured. This toolbox, embedded in Matlab, allow the connection of the elements of a Simulink model with the requirements listed in the DOORS database using the proprietary DOORS eXtension Language (DXL). The information of the link can be shown both in Simulink and in DOORS, allowing the traceability.

At each level, the design budgets have been recalculated, following the preliminary budget assessment proposed in [22]. As example, Table 1 show respectively the mass budget, the power budget, the link budget and the delta V budget at the Space Tug Subsystem level; **Errore. L'origine riferimento non è stata trovata.** summarizes the mass, volume and power budgets. As expected, the propulsion sub-system is predominant in the power and mass budget, being the main purpose of the space tug under design the transfer of a non-cooperative space system at its operative orbit. Another important design solution that has significant influence on the budgets is the use of an electrical power sub-system: such a sub-system has to be supported by the EPS system increasing the Solar Array area and this influence can be seen in the volume allocation, where the EPS sub-system is the predominant one.

Table 1: Space Tug mass breakdown

| **Sub-system** | **Mass fraction [%]** | **Mass Margin [%]** | **Mass [kg]** | **Volume [m$^3$]** | **Power consumption [W]** |
|---|---|---|---|---|---|
| *Propulsion* | 26 | 10 | 238 | 0,56 | 9433 |
| *EPS* | 25 | 15 | 224 | 3,81 | - |
| *TCS* | 5 | 20 | 49 | 0,31 | 20 |
| *AOCS* | 4 | 5 | 34 | 0,02 | 151 |
| *OBDH* | 1 | 20 | 6 | 0,01 | 5 |
| *TT&C* | 3 | 15 | 25 | 1,25 | 103 |
| *Structures* | 22 | 20 | 202 | 0,21 | 80 |
| *Harness* | 5 | 20 | 49 | 0,30 | - |
| **TOTAL** | **-** | | **911** | **6,47** | **9791** |

The same data of Table 1 can be included in Rhapsody model as well (Figure 13).

Functional analysis, ConOps analysis and product definition and sizing were strictly connected and they were updated after any iteration. At the end of any iteration a new list of traced requirements constitutes the specification.

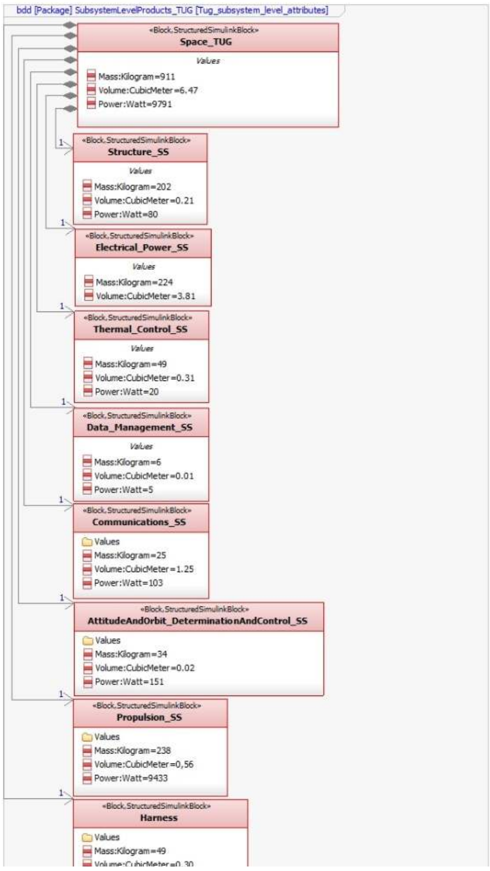

Figure 13: BDD of the Space Tug with mass breakdown

A good way to check the correct implementation and the consistency of the design is looking at the web of traceability links that is recorded thanks to the coverage analysis features of the Rhapsody Gateway®. This tool provides a view of the design space where the relations of requirements can be analysed and navigated from the point of view of both functional and physical architecture of the STRONG system. In addition, Rhapsody® allows navigating the whole set of relations of an element even without the dedicated analyzer embedded in the Gateway. Each element of the design is characterized by a set of properties as the diagrams where the element itself appears, the dependency links and the allocations. This huge amount of information is updated live during the design process, enhancing considerably the quality of traceability and solving those problems related to data classification for Document-Based procedures. Figure 14 summarizes graphically the kind of relations that can be navigated in the model, by showing the example of the type of traceability links instantiated among objectives, requirements, functions and products.

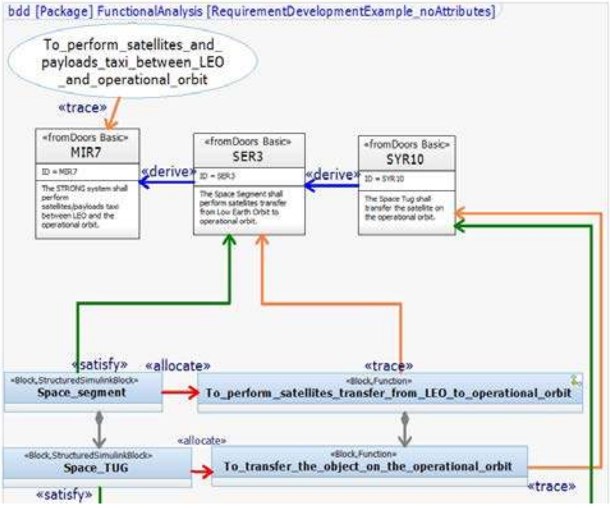

Figure 14: Example of traceability links for the development of the primary mission objective

In particular, the Mission Requirement (MIR7) is connected through "trace" dependency to the mission objective that has generated it (orange lines) and through the "derive" link (blue arrow) to the son segment requirement (the link has positive direction when expressed as "derive from" ) and so on. The functions (right bottom blue blocks) are connected through "trace" links to the requirements that they generated and are the target of the "allocations" coming from the products responsible from their accomplishment (red arrows). "Satisfaction" links (green) are instead established between products and requirements. It has to be noted that requirements, functions and products are always referred to the level of the analysis (low level function and product are part of the high level ones, as established by the "directed composition" links shown in grey). This sort of meta-model has been used within the whole Rhapsody® project to guarantee consistent data management.

## 4 Conclusion

The present paper shows the design of a Space Tug using a method based on a tool chain of tools that improve the design activities. The interest in the development of this kind of system derives from several applications that a space tug could have, from the satellite servicing to debris removal and large spacecraft assembly. In particular, the main mission scenario in which the here presented space tug is employed is to support the transfer of satellites from a generic low orbit, where the launchers release them, to their final operational orbits. Indeed, one of the main benefits of this particular mission scenario is to avoid the need of a dedicated propulsion system in the target spacecraft, in favour of a larger amount of mass available for the payload.

The design of the space tug mission and system has been led using a toolchain that merges and shares the main advantages of both classical System Engineering processes and a Model Based approach, with the future purpose of simplify verification processes in late design phases. This tool is addresses at the resolution of an important limit in classical System Engineering processes: the results obtained through these processes are, indeed, not supported by software related tool and so are not able to achieve effective classification, traceability and verifiability of requirements among the various phases of the design process.

The application of the proposed toolchain to a known case study (i.e. STRONG space tug) has preliminary demonstrated the possibility to simplify the application of classical System Engineering processes, increasing the classification and traceability of requirements among the design activity. In addition the use of this kind of toolchain will also increase the verifiability of requirements during and at the end of the design loop, allowing a simplification of the system design and the continuous verification of the requirements. It confers a higher confidence in the formal correctness and effectiveness to the design process. Indeed, future developments of this work should address to complete the toolchain implementation and validation, extending it to the verification phases at each level.

Important for this phase is the high integration of the chosen tools with simulation environments such as Matlab® and Simulink® or ad hoc developed tool such as the simulator [23], which are easily configurable to be a dynamic link between the design and the verification phases. For the design of the space tug, the future works should focus on further iterations of the design process in order to improve and complete the specification at all the levels (i.e the part and the equipment level).

*References:*

[1] J. Andrews, “Spaceflight Secondary Payload System and SHERPA Tug – A New business Model for Secondary and Hosted Payloads, SSC12-V-6”, 26th Annual AIAA/USU Conference on Small Satellites, 2012.

[2] M.A. Viscio, N.Viola, E.Gargioli, A space tug for satellite servicing, 22nd AIDAA Conference, Naples, November 2013

[3] K. Galabova, G. Bounova, O. de Weck, and D. Hastings. "Architecting a Family of Space Tugs Based on Orbital Transfer Mission Scenarios", AIAA Space 2003 Conference & Exposition, AIAA SPACE Forum, http://dx.doi.org/10.2514/6.2003-6368

[4] M.Richards, P. Springmann, M. Mcvey, “Assessing the Challenges to a Geosynchronous Space Tug System”, Proceedings of the SPIE, Volume 5799, p. 135-145 (2005), doi 10.1117/12.603975

[5] V. Manzo, L. Yambrick, M. Elaine Bunn, “Toward a Theory of Spacepower: Selected Essays”, National Defense University, 2007, ISBN-10: 0160877865, ISBN-13: 9780160877865

[6] N. Sevastiyanov, N.Bryukhanov, "Clipper Reusable Space Transport System", 57th International Astronautical Congress, International Astronautical Congress (IAF), 2006, http://dx.doi.org/10.2514/6.IAC-06-D2.4.01

[7] E.F.Harrison, E.B. Pritchard, “Use of Space Tug to increase payload capability of Space Shuttle”, NASA Langley Research Center; Hampton, VA, United States, 1971

[8] S.A. Jefferies and R.G. Merrill. "Viability of a Reusable In-Space Transportation System", AIAA SPACE 2015 Conference and Exposition, AIAA SPACE Forum, (AIAA 2015-4580), doi: 10.25146.2015-4580

[9] J.C. Liou, D.T. Hall, P.H. Krisko, J.N. Opiela, “LEGEND, a 3-dimensional LEO to GEO Debris Evolutionary Model”, Adv. Space Res., Advances in Space Research, Volume 34, Issue 5, 2004, Pages 981–986, http://dx.doi.org/10.1016/j.asr.2003.02.027

[10] ESA, *Technologies for Space Exploration Roadmaps*, 2nd Edition, November 2015

[11] J. Mason, "Space Tug Performance Optimization", Journal of Spacecraft and Rockets, Vol. 9, No. 7 (1972), pp. 491-492. doi. 10.2514/3.61728

[12] R. L. Schweickart, E. T. Lu, P. Hut, C. R. Chapman, The Asteroid Tugboat, Scientific American, Scientific American 289(5):54-61 · December 2003

[13] F. Stesina, M. A. Viscio, N. Viola, S. Corpino, S. Fineschi, F. Fumenti, C. Circi, Interplanetary CubeSats system for space weather evaluations and technology demonstration, Elsevier Ltd Acta Astronautica,

Volume 104, Issue 2, November 2014, Pages 516-525. ISSN: 0094-5765, http://dx.doi.org/10.1016/j.actaastro.2014.06.005

[14] L. DeMaster-Smith, Scott Kimbrel, Christian Carpenter, Steve Overton, Roger Myers and David King, *Solar Electric Propulsion (SEP) Benefits for Near Term Space Exploration*, IEPC-2013-45, The 33rd International Electric Propulsion Conference, The George Washington University, USA, 6–10 October 2013

[15] S. Cresto Aleina, N. Viola, F. Stesina, M. Antonietta Viscio, S. Ferraris, Reusable space tug concept and mission, Acta Astronautica, Volume 128, November–December 2016, Pages 21-32, ISSN 0094-5765, http://dx.doi.org/10.1016/j.actaastro.2016.07.003

[16] W. J. Larson, L. K. Pranke, Human Spaceflight: Mission Analysis and Design, Space Technology Series, McGraw-Hill, 1999, 1035 pg

[17] NASA Systems Engineering Handbook, NASA/SP-2007-6105 Rev1, National Aeronautics and Space Administration, NASA Headquarters Washington, D.C. 20546, December 2007.

[18] S. Cresto Aleina, N. Viola, R. Fusaro, G. Saccoccia, *Technology Prioritization In Support Of Moon Initiatives In The Framework Of ESA Exploration Technology Roadmaps*, IAC-16-D4.1.2 (34073), 67th International Astronautical Congress (IAC), Guadalajara, Mexico, 26-30 September 2016.

[19] IBM Rational DOORS, www.ibm.com/software/products/it/ratidoor (accessed 06.09.2016).

[20] E. Brusa, D. Ferretto, A. Calà, Integration of heterogeneous functional-vs-physical simulation within the industrial system design activity, Proc. 1st IEEE-ISSE Conf. Rome, Italy, pp. 303-310, 2014.

[21] Mathworks, Simulink, http://us.mathworks.com/products/simulink/, (accessed 06.09.2016).

[22] W. J. Larson, L. K. Pranke, Human Spaceflight: Mission Analysis and Design, Space Technology Series, McGraw-Hill, 1999, 1035 pg

[23] S. Corpino, F. Stesina, “Verification of a CubeSat via hardware-in-the-loop simulation.” *IEEE Transactions on Aerospace and Electronic Systems*, 50(4), 2807–2818, 2014