Development process for automatically generated plans for satellites control based on UML

Charles-Edouard Winandy[1] and Mauricio Gonçalves Vieira Ferreira[2]
The Brazilian National Institute for Space Research – INPE, São José dos Campos, SP, Brazil

One of the main causes for the growing demand for satellites controllers and control centers is the increase number of satellites launched from our planet. Some of these needs are supplied by new controllers, but it will soon be a time on which this kind of monitoring will be overtaken by the growing number of satellites to control, without an auto-generated plan process. In this real scenario, it is important to have, before thinking of implementing a planner or even before create a Knowledge Base equipped with meta-models, a development process divided in very well defined steps. Each of these steps must have a target objective to reach and one or more than one software artifacts to generate, using for this techniques in fields like the knowledge acquisition and tools for successful software design, such as the Unified Modeling Language (UML), one of the most significant modeling tool used in commercial systems. A development process for an automatically generated plan, based on the satellite components status, the ground positions stations and its orbit frequency, must be used as soon as possible, in order to grant our capability to safely control all of our satellites. This paper object is to show in full detail the PRODESEA, an auto-generated plan development process useful when a satellite control planning is needed to reach the above goals. For didactic purpose, this paper also illustrates one case study in a satellites control domain.

Nomenclature

AI	=	Artificial Intelligence
CBERS	=	China-Brazil Earth Resources Satellite
CCS	=	Brazilian Satellites Control Center
ES	=	Expert System
ESA	=	European Space Agency
EUP	=	Enterprise Unified Process
FOP	=	Flight Operations Plan
INPE	=	Brazilian National Institute for Space Research
LIT	=	Brazilian Test and Integration Laboratory
LTP	=	Long Term Planning
MTP	=	Medium Term Planning
PMM	=	Multi-Mission Platform
GS	=	Ground Station
STP	=	Short Term Planning
TC	=	Telecommand
TM	=	Telemetry
UML	=	Unified Modeling Language
URD	=	User Requirements Document
XML	=	Extensible Markup Language

[1] Systems Analyst, Ground Systems Development Division of INPE, winandy@gmail.com.
[2] Researcher, Satellites Control Center of INPE, mauricio@ccs.inpe.br, AIAA Member.

I. Introduction

THE main activity of INPE's CCS (Brazilian Portuguese for Satellites Control Center) is monitoring and controlling the Brazilian satellites managed by INPE, as well as the satellites from foreign partnerships for which they take responsibility, in a coherent and safe way, aiming at benefiting the most from the potential of equipments and resources allocated. For next years, INPE has the challenge of launching an average of one satellite every year, due to Platform PMM (Brazilian Portuguese for Multi-Mission Platform) and Platform CBERS (China-Brazil Earth Resources Satellite), the last one already in process for some years in partnership with China. Therefore, an even stronger need for operations automation is demanded.

Due to the vast Brazilian territory and the concern of tracing the annual deforestation of Amazon Forrest, satellites have become important tools for the country. In spite of the limited governmental budget allotted to the Brazilian spatial sector, in a country that has financial problems in many other sectors like education and health care, INPE intends to project and build 13 satellites over the next 10 years. Exactly because of this budget scarceness, the optimization of computational and human resources becomes essential for INPE. Through automation and the use of AI (Artificial Intelligence), INPE awaits to reduce the costs in operation without interrupting the increasing number of satellites that they expect to control in next years.

The lack of a very well-defined development process focused on INPE's field of interest and applied to the automation of satellites operations can become an issue in the attainment of the ambitious outcomes desired. This bigger need was the motivation for this project.

Therefore, the focus and the scope of this paper center in the description of the development process of automatic plans for the operation of satellites, based in the UML needed to fulfill INPE's biggest aim of optimizing their scarce resources to the most through automation. This process was the object of a research project published in INPE's library in 2011[1] and is a continuation of the paper published in SpaceOps 2010 conference[2].

For the description of such process and the software artifacts to be generated during its execution, the 13 diagrams of version 2.2 from UML were used as maximum as possible, adapting their use and considering INPE's actuality. One of the aims targeted in the choice of UML for this project was promoting a bigger integration among stakeholders, considering a group as heterogeneous as the one in INPE, formed by researchers, professors, students, scholars, trainees and others. For that reason, nothing better than the election of tooling that appreciates the standardization and that is largely broadcasted in the academic environment and in the professional as well, as UML is.

II. The Process Overview

The process was named PRODESEA, a combination of syllables and first letters of the following words in their respective order in Brazilian Portuguese: Development Process of Automated Spatial Systems (*Processo de Desenvolvimento de Sistemas Espaciais Automatizados*).

PRODESEA does not ambition to attend other areas apart from the spatial. Since the beginning, it was focused and aimed on attending INPE's CCS. Divided in six main activities (separated in stages and sub-stages), PRODESEA starts with the definition of the problem domain, goes through the construction of the Knowledge Base, follows to the definition of goals and finishes with the generation, testing and publishing of FOP (Flight Operations Plan), as its scope ends before the execution of the plan created in an automatic way.

The strategy for the creation of an operational plan is encapsulated in its activities and stages, following techniques for the acquisition and the representation of knowledge as an example.

At first sight, PRODESEA simply seems to be a process of sequential development composed by six activities in which one follows the other. However, it is in part iterative, not only in the sub-stages of the second main activity, in which there is a continuous iteration of knowledge acquisition and its representation, but mainly from the activity labeled as Definition of the Goals, on which a constant iteration is found between the third and the sixth activities, besides the re-alimentation of the Knowledge Base that, despite being ready for use at a certain point, is never finished.

The construction of the plan depends on a planner and discussing how or what planner is more suited for the problem domain escapes the scope of this project. There are many planners available in the market for purchase, and the possibility of developing their own planner can not be discarded. The focus of this activity is the assessment and the prioritization of stages in the construction of the FOP. It is not dependable on the chosen planner solution.

The Plan Test activity concentrates on some specific areas, but does not encompass every type of test and can thus be improved in the future or even become the subject of a complementary research project.

The last activity consists of the publication of the plan, a task that is much more administrative and bureaucratic

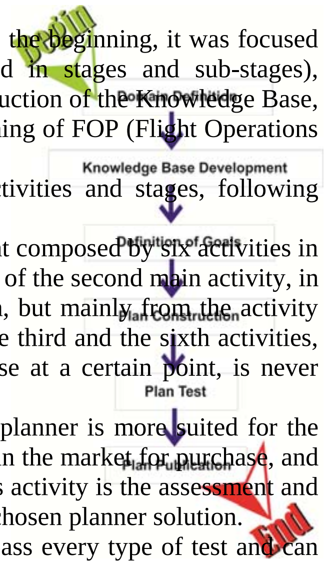

Figure 1. Process overview

than technical. It finishes the process, as long as the plan is accepted. In case it isn't, the generated plan would have to be rolled back to a previous activity or to a change activity, mandatorily passing by testing and publication activities again.

III. Revealing PRODESEA

Each of the stages and sub-stages from each of the six main activities of the development process called PRODESEA will be explored in details hereafter. For each stage presented, one or more software artifacts to be generated will be associated, when applicable. A software artifact may be mandatory or optional for a certain stage. Most of the software artifacts generated make use of UML. Later on, a case study in the domain of satellite control and tracking will be presented, exemplifying the software artifacts in practice.

A. Activity 1: Domain Definition

Starting with the activity labeled Domain Definition, PRODESEA has as a first target preparing the path to initiate the construction of the Knowledge Base. At first, a high-level approach is made with the elaboration of a text that is very close to natural language, to then invent in a lower level of abstraction, describing the problem in a more technical language like algorithms and diagrams.

Great knowledge on the business environment is needed to have a good-quality modeling of the problem. Exactly for this reason, the first designated stage in the first activity of PRODESEA was the Business Analysis, as seen in figure 2.

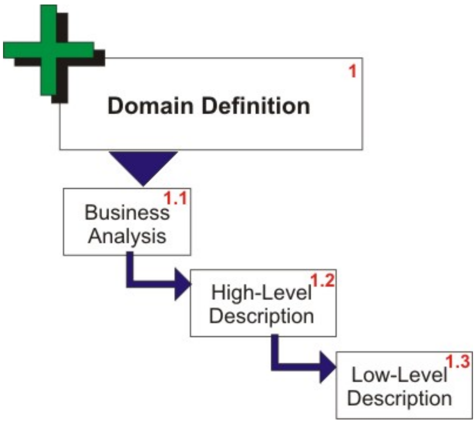

Figure 2. First activity of the PRODESEA

Each of the three stages in this activity can be separated in sequential sub-stages. The first of them, labeled as Business Analysis and numbered as 1.1 in the hierarchy (first stage of first activity), is subdivided in four other stages, the three firsts taken from the EUP (Enterprise Unified Process), as seen in figure 3.

The first stage of the activity named Business Analysis, numbered as 1.1.1, exists to acknowledge the environment in which the business is inserted. This first contact with the customer's context is important for the integration of the subjects involved in the problem domain. It is only the first stage to obtain the business rules later.

In the second stage, 1.1.2, the business analyst will start to assess the target organization processes. In this stage, the expertise in the business domain will be of great help and it can be obtained through a specific individual that is a specialist in the subject or though the organization knowledge as a whole[3].

Finally, in the third stage, the business critical rules are assessed, based on business model or even in case of use to that end[3]. The first PRODESEA mandatory software artifact comes up exactly in this stage, numbered as 1.1.3 and labeled as Business-Critical Rules Assessments. Such artifact is the UML Use Case Diagram, and it must be used to identify the pre-conditions of processes assessed in the previous stage.

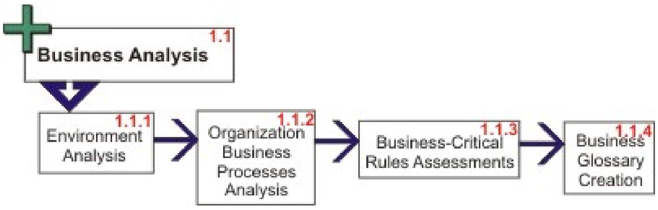
Business Analysis
1.1
Environment Analysis
1.1.1
Organization Business Processes Analysis
1.1.2
Business-Critical Rules Assessments
1.1.3
Business Glossary Creation
1.1.4

Figure 3. First step of the first activity of the PRODESEA

In the fourth and last stage of the first activity, numbered as 1.1.4 as seen in figure 3, a technical terms or business key-words dictionary is created. At this point in the process, it is expected that enough knowledge on the business will be held for the elaboration of a glossary in a relatively good size. The addition of this stage happened to favor the job when there is a very heterogeneous group of professionals involved in the project or when there are high turnover rates. The glossary, dictionary or lexical created in this stage is itself a mandatory software artifact to be created.

Once these four sub-stages of the first stage of the first PRODESEA activity are finished, it is possible to move to the stage labeled High-Level Description, designated with number 1.2 in the hierarchy (second stage of first activity). It is subdivided in two other stages, as seen in figure 4.

In the attainment of the problem general view, the UML Use Case Diagram is used, showing the limits of the future system and their relation with external entities or legacy system. This software artifact is optional and, therefore, is not mandatory in PRODESEA. Another possibility for this stage is the creation of a brief description of the problem, with few words.

After that, a textual description of the problem details is created, but not too long. Numbered as 1.2.2 and labeled as Brief Description of the Problem, this stage will generate a textual description in natural language as a mandatory software artifact.

The last stage of the first activity, labeled as Low-Level Description, designated with number 1.3 in the hierarchy, is subdivided in four sub-stages and has a more technical feature than the High-Level Description, as observed in figure 5.

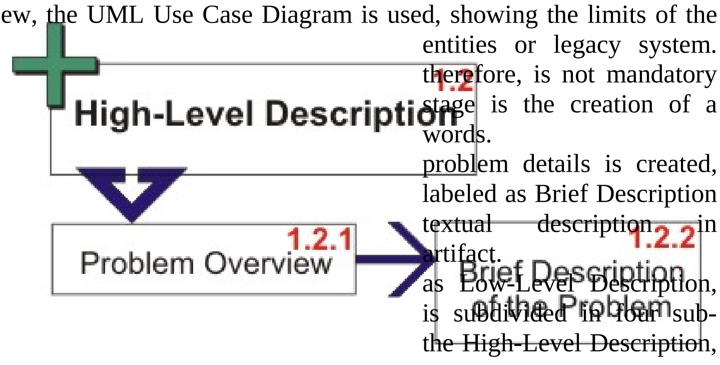

Figure 4. Second step of the first activity of the PRODESEA

In the first sub-stage of this stage, 1.3.1, the seizure of the essence of the interest domain functionalities takes place. Name of functions or methods must come up. The processes analyzed in stage 1.2.2 (the second sub-stage of the previous stage) will very likely be the main functionalities, but others may come up and most of them will be derived from these functionalities. The UML Use Case Diagram must be used in this stage with the relationship types include and extend.

After that, the roles and respective actors are defined. For that end, the nomenclature of the UML Use Case Diagram is also used, where actor is nothing more than a pattern stereotype to define a role. The functionalities defined in 1.3.1 will be performed by actors.

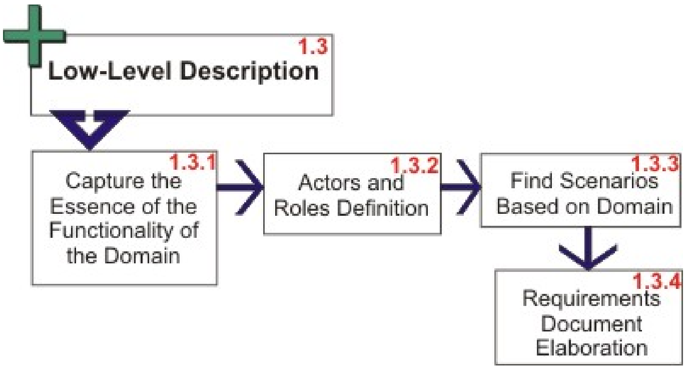
1.3
Low-Level Description
1.3.1
Capture the Essence of the Functionality of the Domain
1.3.2
Actors and Roles Definition
1.3.3
Find Scenarios Based on Domain
1.3.4
Requirements Document Elaboration

Figure 5. Third step of the first activity of the PRODESEA

In 1.3.3 occurs the assessment of possible scenarios. It is more appropriate to employ the UML Sequence Diagram in this stage, as seen in the case study presented in the next topic. In any case, the creation of this kind of artifact in the stage 1.3.3 of PRODESEA is not mandatory, but at least the main scenarios must be described on a textual way.

As for stage 1.3.4 of the process, a document containing the requirements must be generated. It is the URD (User Requirements Document), a mandatory software artifact in the process. The requirements do not come up in this stage, they are only formalized (it is assumed that, in this sub-stage, the knowledge needed to clearly describe the requirements is already held). With this, the first activity of PRODESEA is concluded.

B. Activity 2: Knowledge Base Development

With the problem domain well studied, it is possible to move to the activity of creating a Knowledge Base, because the construction of an ES (Expert System) is only possible after a good understanding of the problem domain and its relationships[4].

Creating a Knowledge Base implicates in applying practically the aspects studied in Knowledge Engineering, like the acquisition and representation of the knowledge of a previously studied domain. In this topic, the second main activity of PRODESEA, labeled as Knowledge Base Development, is to develop the data base where the knowledge will be stored.

In PRODESEA, Martin and Oxman[5] techniques and the three usual sub-stages of the knowledge acquisition process exposed by Cordingley[6] were used in this activity. Each one of the sub-stages was detailed and can be verified in figure 6. The process of knowledge acquisition is an iterative process that is only finished when the Knowledge Base is complete (when it is considered ready, in the eyes of the knowledge engineer). The dotted line in figure 7 means that the (iteration) path is optional, and can be executed zero or more times.

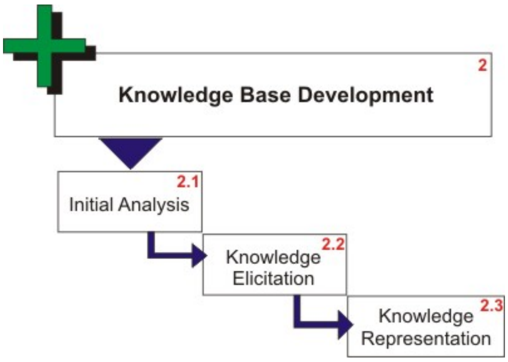

Figure 6. Second activity of the PRODESEA

From an initial domain knowledge, specifically obtained in stage 2.1.1 (first sub-stage of the first stage of the second PRODESEA activity), it is possible to move to the Knowledge Prototyping with the aim of putting it to test. In certain cases, it may be necessary to effectively build a physical prototype or even a human interface prototype for that in stage 2.1.2 and, once there is a prototype, the test is made in stage 2.1.3 as a task or an example if the prototyping knowledge is valid. It will be up to the knowledge engineer in stage 2.1.4 to analyze the outcome obtained with this test to decide if the knowledge is pertaining or not to the Knowledge Base.

Then the Knowledge Elicitation occurs in stage 2.2, with sub-stage 2.2.1 that contains the specialist's inference over the analyses made in the previous stage. From this point, in stage 2.2.2 of PRODESEA, the acquisition of the needed knowledge (considered pertaining to the solution of the problem by the specialist) to the Knowledge Base occurs.

At the end the knowledge must be represented in some way. The Knowledge Representation is not restricted to a single technique. There is a big quantity of techniques that can be used to represent knowledge and it will be up to the knowledge engineer to choose the one that better fits their problem. For this project the option was made for UML, with its Class Diagram.

Representing the knowledge in stage 2.3.1 of PRODESEA demands the creation of at least one software artifact. There are complementary ways of representing part of the knowledge in this stage. Assuming, for example, that the UML Class Diagram is the mandatory software artifact created in this stage, there could be one or more UML State Diagrams as optional software artifacts and, in certain cases, even the UML Package Diagram to organize the represented knowledge (also optional).

The organization of objects into categories is a vital part of the knowledge representation because, although the interaction with the world occurs in the level of individual objects, a big part of the thinking takes place in the level of categories[7]. The UML Package Diagram fits very well in this concept.

The cycle repeats through sub-stage 2.3.2 that allows new knowledge raised to go through prototyping to be acquired and represented, repeating the process from sub-stage 2.1.2 until sub-stage 2.3.1 (this constructive process repeats itself until the specialist considers the Knowledge Base as ready).

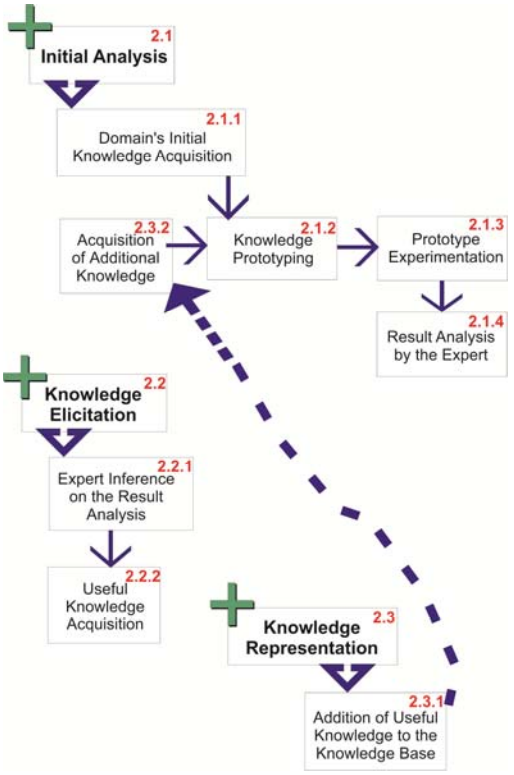
2.1
Initial Analysis
2.1.1
Domain's Initial Knowledge Acquisition
2.3.2
Acquisition of Additional Knowledge
2.1.2
Knowledge Prototyping
2.1.3
Prototype Experimentation
2.1.4
Result Analysis by the Expert
2.2
Knowledge Elicitation
2.2.1
Expert Inference on the Result Analysis
2.2.2
Useful Knowledge Acquisition
2.3
Knowledge Representation
2.3.1
Addition of Useful Knowledge to the Knowledge Base

Figure 7. Inside PRODESEA second activity

C. Activity 3: Definition of Goals

When the third main activity of PRODESEA is initiated, it is expected that the Knowledge Base is ready for use. The Definition of Goals is concentrated in this activity, but this does not mean that new goals can not be accomplished later. The goals that perchance come up after the end of this activity can be accomplished in a subsequent plan, for example, by re-using all the work from the two first activities.

A goal is a quantified aim. It can be considered that an aim is a set of states existing in the real world, in which the states of the set represent the states by which it is necessary to transit to reach the aim satisfaction[7].

The third activity is subdivided in three sequential stages, as seen in figure 8. Each goal or aim must pass by each one of these three stages, starting by the descriptive one, 3.1, labeled Goal Description. Once the goal is described, in stage 3.2 the goal classification can start in one of the following time scales: STP (Short Term Planning), MTP (Medium Term Planning) and LTP (Long Term Planning).

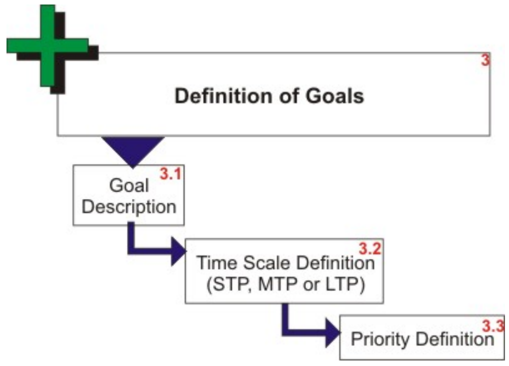

Figure 8. Third activity of the PRODESEA

Every mission has its own planning time definitions. For example, in Venus Express mission, the space probe from ESA (European Space Agency), launched in November 9, 2005 to explore planet Venus, the planning time definitions from the STP type consisted of activities planned for 3 to 5 days ahead of operations, the MTP type for 7 to 8 weeks in the future and the LTP type for 19 to 20 weeks ahead[8]. However, nothing prevents the use of a different classification. There could indeed be only one defined interval for MTP, something like 4 to 7 days, and everything bigger than that would be classified as LTP and smaller as STP.

The critical goals (frequently present in the beginning of the mission, like the first maneuvers after launching) will be grouped in type STP, because a critical event like launching, landing or insertion of the spatial vehicle in the planetary orbit needs to have a bigger priority than a routine event. In general, the moments of elevated priority concentrate in the initial phase of a spatial mission[9].

Once it is defined in which time scale the goal is framed, it is possible to move to stage 3.3 of PRODESEA. Labeled as Priority Definition, stage 3.3 is nothing more than a second classification for the goal. If the biggest priority is the smaller number or the bigger number is not of much importance to the process. The important is having a pre-defined priority scale to favor the job of the planner in the following activity.

D. Activity 4: Plan Construction

With the relation of goals defined and classified, it is possible to build a flight plan for an satellite. As verified in figure 9, the construction of a plan is based in the idea that the relations of operations from type STP precede the ones from type MTP that, on the other hand, precede the ones from type LTP. After that, the definition of the order in which the operations or activities must be performed comes into effect through the value of the priority of each associated goal.

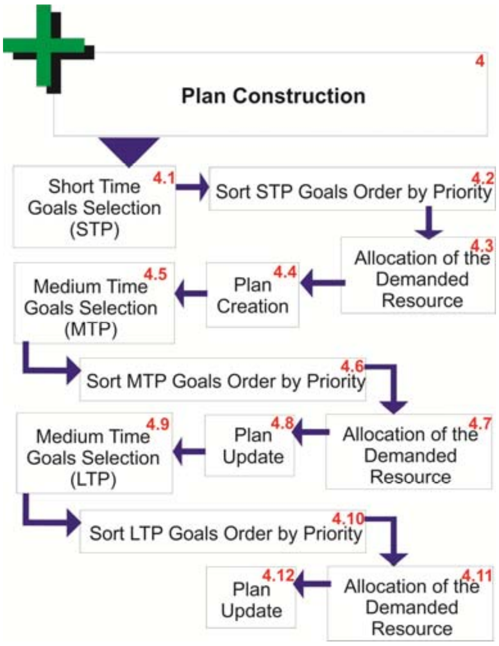
4
Plan Construction
Short Time Goals Selection (STP)
4.1
Sort STP Goals Order by Priority
4.2
Allocation of the Demanded Resource
4.3
Plan Creation
4.4
Medium Time Goals Selection (MTP)
4.5
Sort MTP Goals Order by Priority
4.6
Allocation of the Demanded Resource
4.7
Plan Update
4.8
Medium Time Goals Selection (LTP)
4.9
Sort LTP Goals Order by Priority
4.10
Allocation of the Demanded Resource
4.11
Plan Update
4.12

Figure 9. Fourth activity of the PRODESEA

Nevertheless, high priority does not guarantee the allocation of the demanded resource[9]. Placing the goals classified as STP in front only increases the chances of accomplishing the mission critical events.

When it is about creating a new plan for an satellite, its planning cycle is taken into account. For example, the planning cycle of the Earth Observation satellite ADM-Aeolus from ESA, with launching planned for the end of 2013, is weekly-based, i.e., the activities of a certain week are planned with two weeks in advance[10].

The constructive process of the fourth activity will be finished with a complete plan involving all goals reachable within the available resources through searches and reservations of resources in the Knowledge Base. The goals that are excluded can be included in a future plan.

Stages 4.1, 4.2, 4.3 and 4.4 exist to build a first version of the plan, which contemplates only the goals from type STP. As for stages 4.5, 4.6, 4.7 and 4.8, they add the goals from type MTP to the plan generated in 4.4 and generate a second version of the plan. Finally, stages 4.9, 4.10, 4.11 and 4.12 complete the plan with the goals from type LTP, generating the final version of the flight plan for an satellite.

A software artifact that perfectly fits this activity is the UML Activity Diagram. It is applicable in an optional way in stages 4.4 and 4.8, but mandatorily in stage 4.12 of PRODESEA. In this stage, in face of such software artifact, a clear continuity relation can be found with the scenarios from the Sequence Diagram which, on the other hand, are based in the pre-conditions of the Use Case Diagram.

E. Activity 5: Plan Test

Just because the plan was generated with data from the Knowledge Base considered concrete, it does not mean that it is correct. In the fifth activity of PRODESEA the integrity of the plan and its completeness are verified. Also, the plan can be run in some simulator with orbit propagation and its coherence can be assessed.

It's not even needed to say how important this activity is, since at its conclusion the plan enters its last activity before execution in the CCS's scheduling. The aim of this testing activity is to identify

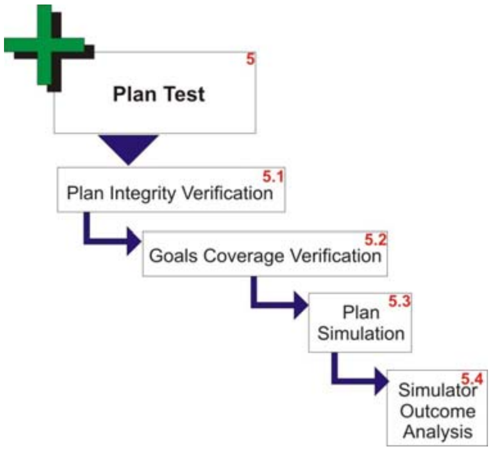
5
Plan Test
Plan Integrity Verification
5.1
Goals Coverage Verification
5.2
Plan Simulation
5.3
Simulator Outcome Analysis
5.4

Figure 10. Fifth activity of the PRODESEA

problems that can sometimes exist in a plan generated automatically before its publication.

The first stage of the fifth activity was labeled as Plan Integrity Verification, as seen in 5.1 in figure 10, and is aims at identifying if there is no conflict of resources, considering time and space.

A software artifact considered as mandatory for this activity is the UML Timing Diagram. It perfectly fits to demonstrate the use of resources during the timeline, therefore allowing the comparison of more than one plan for the same satellite and the identification of resource conflicts, in case they exist.

In the next stage, 5.2, labeled as Goals Coverage Verification, it is verified which goals will be accomplished by the plan generated and which will be excluded (it will not always be possible to accomplish all goals in a single plan). The aim of this stage is to enhance the plan coverage, identifying the goals not accomplished by the plan.

The Plan Simulation occurs in stage 5.3 and the analysis of the simulator outcome will be used in the next stage. The Simulator Outcome Analysis is stage 5.4 of PRODESEA and it depends entirely on a specialist. The problems that are perchance identified in this stage escape the scope of this project, but will undoubtedly decisive for the approval or rejection of the plan in the next activity.

F. Activity 6: Plan Publication

The final activity of PRODESEA consists of forwarding the plan generated in the fourth activity, once the tests are concluded, for its execution. It contains a decisive stage, a stage of the publication itself and a stage or re-alimentation.

The decisive stage, labeled as Plan Approval and numbered as 6.1, is the first stage of the sixth activity, as observed in figure 11. The person responsible for this decision must be well-informed of the tests results from the previous activities. This is considered the non-returning point, since once the plan is approved, it moves to the publication stage. In case it is rejected, the plan can either be changed manually to fit actuality or simply be discarded. The ideal in the first situation would be to restate the changed plan to the fifth and sixth activity of PRODESEA, but this is something to be discussed and defined by the final users of the process.

Stage 6.2 is responsible for the publication of the plan. In the case of the FOP, it must enter CCS's system to be executed for a specific satellite.

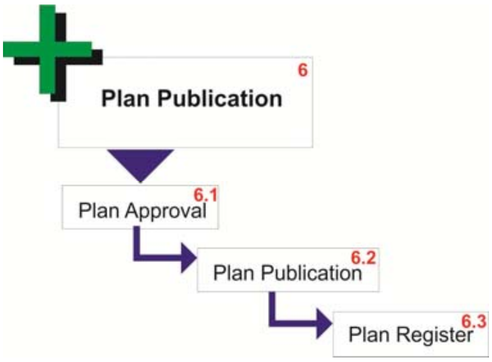

Figure 11. Final activity of the PRODESEA

The last stage of PRODESEA exists to fulfill the re-alimentation of the Knowledge Base. Labeled as Plan Register, stage 6.3 is responsible for the persistence of the approved and published plan in the Knowledge Base. This stage is important to avoid the construction of plans that are redundant or with resources in use.

An example of the importance of the Knowledge Base re-alimentation occurs when a plan approved and published to be executed in the domain of satellites control and tracking needs to set apart resources from the related satellite to capture images from an area of the planet, like one of its cameras.

In case a new plan was generated for the same satellite and for the same time frame, a competition for resources would occur and this could lead to and undesired conflict. That's why it's fundamental to register each plan approved for execution, avoiding that two plans are generated with the same resources and even that two plans focus on fulfilling the same goal.

The whole development process was thereby detailed step-by-step in this topic theoretically. The next topic will show a practical example of the use of PRODESEA shaped as a case study on the domain of satellites control and tracking.

IV. Case Study

The goal of the case study presented next is to show the practical appliance of each of the activities of the development process shown in the beginning of this paper, with practical examples of software artifacts to generate in each stage. It is important to emphasize that, even if the case study seems real, it is fictitious and it was created as a hypothetic example of the use of PRODESEA. The context is partly based in reality on purpose to make the reading more pleasant. The satellite control and tracking domain was chosen as a topic, which is the exact niche to which PRODESEA was initially designed for. For such case study, a fictitious name was chosen for the artificial satellite: SatF, acronym for Fake Satellite.

Therefore, we will suppose that a satellite called SatF has to capture images from part of the Amazon Forest to follow the rhythm of deforestation and that suddenly there is another goal for the same equipment. To make our case study more vivid, we will consider the hypothesis that governments from Brazil and France want to follow the progress of the bridge over Oiapoque River supranational project. In this context, it is assumed that Brazil is in charge of capturing images on the work site in the border with Guyane (the French Guyana) to discover if the bridge that will connect Brazil to France, through this ultramarine French department, will be ready on time. It is also assumed that there is a certain pressure from the Brazilian government to have these pictures in a short period of time, because they want to know if the bridge will be finished before the end of the current presidential term, so that the president of Brazil can cross de 200-meter space of the bridge along with his French equivalent in the work opening.

Thus, by applying stage 1.1.1 of PRODESEA (Environment Analysis) to the case study, it is stated that:

- The environment is formed by two GS (Ground Station), one located in Cuiabá and the other in Alcântara, a Satellite Control Center, located in São José dos Campos, and a Mission Center, located in Cachoeira Paulista;
- Artificial Satellite SatF is located, on the other hand, at an altitude of 778 km, in low orbit.

By applying stage 1.1.2 of PRODESEA (the of the Organization Business Processes Analysis) to the case study, it is stated that:

- The only process identified is the one of providing images of the Brazilian territory.

By applying stage 1.1.3 (the Business-Critical Rules Assessment) of PRODESEA to the case study, it is stated that:

- It is not possible to capture images from different locations in the same space of time;
- It is not possible to transmit data that is out of the reach of an GS;
- It is not possible to transmit a volume of data that surpasses the 80GB quote per day;
- It is necessary to connect the CCS to a GS to process a TM (Telemetry) or a TC (Telecommand);
- To process a TM, it is necessary to have the onboard computer of the artificial satellite turned on.

For this stage, UML Use Case Diagram can be used, showing the pre-conditions according to example shown in figure 12.

By applying stage 1.1.4 of PRODESEA (Business Glossary Creation) to the case study, it is stated:

- CCS is an acronym for Satellite Control Center, responsible for the tracking and control of INPE's satellites;
- GS is the abbreviation for Ground Station;
- Passage is the term used to refer to the interval of time in which a given satellite in its orbit is in the range of some GS to transmit telemetries and receive telecommands;
- Ground Segment is composed by tracking and control antennas, a control center with proper computers and programs to monitor and control satellites, including their state and orbit (data and voice means of communication are also part of this segment);
- TC is the abbreviation for Telecommand;
- TM is the abbreviation for Telemetry.

By applying stage 1.2.1 of PRODESEA (Problem Overview) to the case study, it is stated that:

- A problem of resource concurrence, since it will be necessary to capture images from the bridge site without disturbing the usual activities of satellite SatF (following the Amazon Forest deforestation).

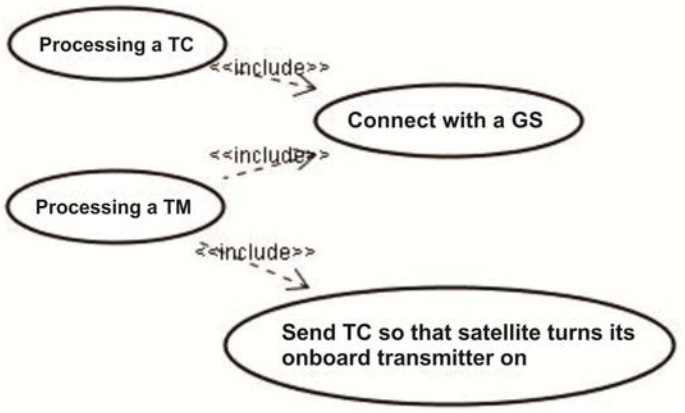

Figure 12. UML Use Case Diagram showing the pre-conditions

By applying stage 1.2.2 of PRODESEA (Brief Description of the Problem) to the case study, it is stated that:

- The set of problems in this case of remote sensing is to try to insert a specific task, like capturing images one single time, by taking advantage of the satellite passage during its routine activities.

By applying stage 1.3.1 of PRODESEA (Capture the Essence of the Functionality of the Domain) to the case study, it is stated that:

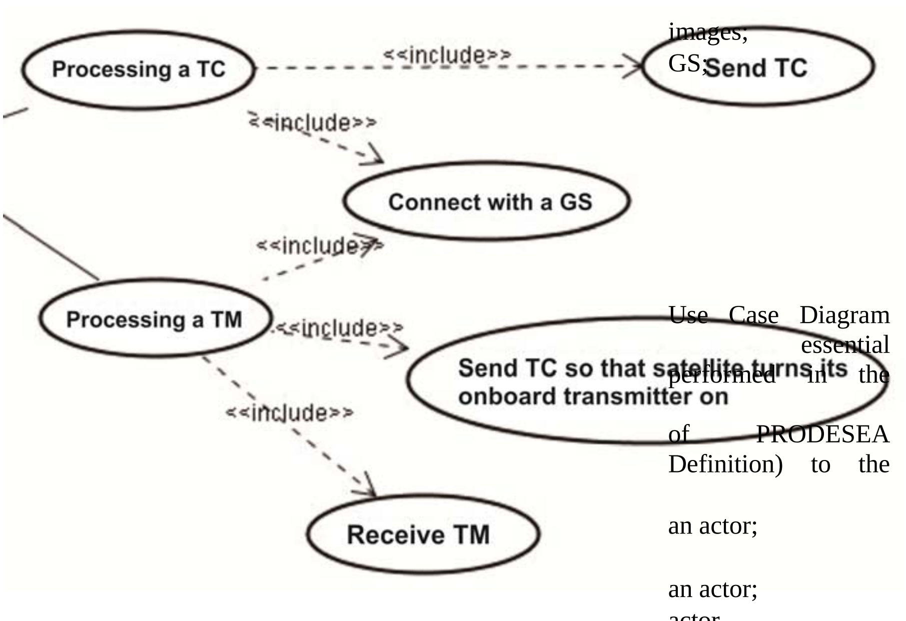

Figure 13. Assessing the essential functionalities with the UML Use Case Diagram

- Capturing images;
- Connect with GS;
- Transmit TM;
- Receive TC;
- Process TM;
- Process TC;
- Send TC;
- Receive TM;
- Run diagnosis.

For this stage the UML Use Case Diagram can be used, assessing the essential functionalities as performed in the example of figure 13.

By applying stage 1.3.2 of PRODESEA (Actors and Roles Definition) to the case study, it is stated that:

- Satellite SatF is an actor;
- CCS is an actor;
- Alcântara GS is an actor;
- Cuiabá GS is an actor.

For this stage the UML Use Case Diagram can also be used, as observed in figure 14, in the separation of actors with a single identification to be applied in each of the Use Case Diagrams created, grouped by perspective (in the view of CCS, for example, a TC is sent while in the view of SatF a TC is received).

SatF
CCS
Cuiabá GS
Alcântara GS

Figure 14. UML Use Case Diagram to find the actors

By applying stage 1.3.3 of PRODESEA (Find Scenarios Based on Domain) to the case study, a series of scenarios or scenes in this problem domain are stated, as detailed below:

- Satellite SatF sending TM to Cuiabá GS:
  - Connect with Cuiabá GS;
  - Prepare data package for sending;
  - Transmit data.
- Satellite SatF sending TM for Alcântara GS:
  - Connect with Alcântara GS;
  - Prepare data package for sending;
  - Transmit data.
- Satellite SatF receiving telecommand from CCS:
  - Sign of contact with Alcântara GS received;
  - TM received.
- CCS processing a TC:
  - Connect with Cuiabá GS;
  - Send TC.
- CCS processing a TM:
  - Connect with Cuiabá GS;
  - Send TC so that satellite turns its onboard transmitter on;
  - Receive data.
- Satellite SatF performing a passage;
- Satellite SatF not responding.

For this stage, the most appropriate is to use the UML Sequence Diagram, as observed in figure 15, which demonstrates one of the scenarios described above.

The relevance of the use of the UML Sequence Diagram in this stage occurs for the fact that this diagram defines the order of actions in timeline. The Use Case Diagram does not respect the sequence and does not show the message exchange, that’s why it is not applicable here.

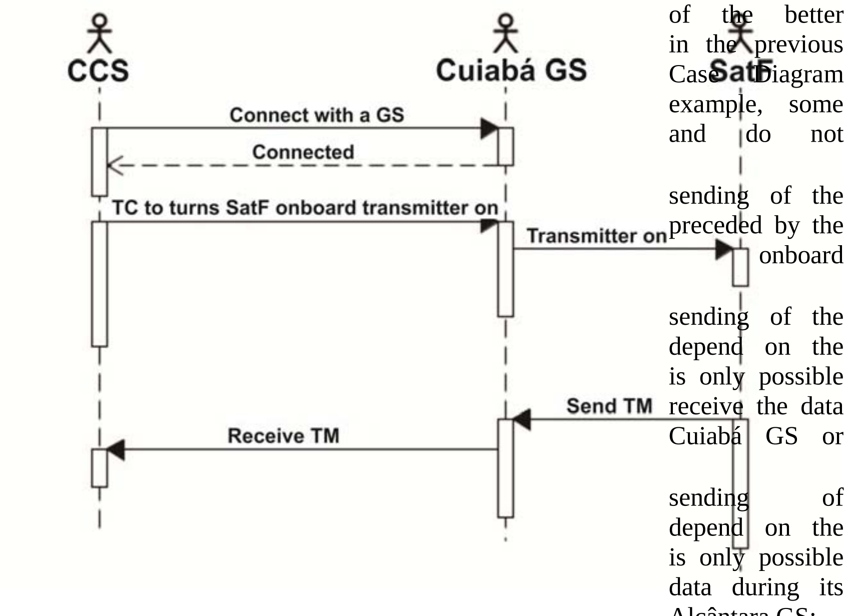

Figure 15. UML Sequence Diagram to define the order of actions in timeline

By applying stage 1.3.4 of PRODESEA (Requirements Document Elaboration) to the case study proposed, the requirements elicited in virtue of the better understanding of the problems in the previous stages by the use of the Use Case Diagram are stated (in the following example, some items are project requirements and do not depend on that):

- URD01 – The sending of the satellite's TM to CCS must be preceded by the TC that turns the satellite onboard transmitter on;
- URD02 – The sending of the satellite's TM to CCS will depend on the moment of its orbit, because it is only possible for the Ground Segment to receive the data during its passage through Cuiabá GS or through Alcântara GS;
- URD03 – The sending of CCS's TC to the satellite will depend on the moment of its orbit, because it is only possible for the satellites to receive the data during its passage through Cuiabá GS or Alcântara GS;
- URD04 – The transmission band is 80GB per day;
- URD05 – The satellite must satisfy the demand for photos of the concerned public in a time frame smaller than 27 days;
- URD06 – The satellite's coverage must be 80% of South America and 100% of Brazil.

By applying stage 2.1.1 of PRODESEA (Domain's Initial Knowledge Acquisition) to the case study, it is stated that:

- As an initial knowledge for the Knowledge Base in this particular case study, it can be expected that there already is some information on the artificial satellite's orbit and also on the functioning of its sub-systems.

By applying stage 2.1.2 of PRODESEA (Knowledge Prototyping) to the case study, it is stated that:

- A complementary knowledge for the Knowledge Base needs to be prototyped before, because this way the knowledge (also called sentence in this case) is put to test before being added to the Knowledge Base (it it not necessarily needed to build a physical prototype for this, a new sentence can be put to test by a logical test).

By applying stage 2.1.3 of PRODESEA (Prototype Experimentation) to the case study, it is stated that:

- Taking as example a failure discovered in a component of one of the sub-systems of INPE's LIT (Brazilian Test and Integration Laboratory) satellite earth replication that occurs in a given situation, this limitation is a complementary knowledge that must be proved by logical tests or through a task executed by a prototype.

By applying stage 2.1.4 of PRODESEA (Result Analysis by the Expert) to the case study, it is stated that:

- The specialist analyses the test outcome with the proposed task.

By applying stage 2.2.1 of PRODESEA (Expert Inference on the Result Analysis) to the case study, it is stated that:

- The inference made by the specialist on the outcome attained with the test occurs.

By applying stage 2.2.2 of PRODESEA (Useful Knowledge Acquisition) to the case study, it is stated that:

- Knowledge considered as useful by the specialist is acquired.

By applying stage 2.3.1 of PRODESEA (Addition of Useful Knowledge to the Knowledge Base) to the case study, it is stated that:

- Representation of the knowledge attained in the way the specialist considers best.

There are various ways of representing knowledge, be it graphically, like the UML diagrams, or descriptively, like with the XML (Extensible Markup Language). For this stage, the UML Class Diagram, for example, can be used, with the application of the Object-oriented paradigm to show the knowledge through entities composed by attributes (the features) and methods (the functionalities), as shown in example of figure 16.

By applying stage 2.3.2 of PRODESEA (Acquisition of Additional Knowledge) to the case study, it is stated that:

- New knowledge raised to be prototyped, repeating the process from stage 2.1.2 until stage 2.3.1 (this process is repeated until the specialist considers the Knowledge Base ready).

Moving to stage 3.1 of PRODESEA (Goal Description) in the exercise proposed for this case study, it is stated that:

- As the goal is to capture images of the River Oiapoque bridge work site in the borderline with Guyane, the goal description would bring instructive data like numbers of pictures and captured area.

By applying stage 3.2 of PRODESEA (Time Scale Definition) to the case study, it is stated that:

- A goal that is not as urgent as a correction maneuver or as deviation from spatial junk, but that needs to be reached in a relatively short time frame could be classified as MTP.

By applying stage 3.3 of PRODESEA (Priority Definition) to the case study, it is stated that:

- The definition of the goal priority can serve as urgency attribution and, as in our case study it is interesting to have the bridge site photos even before the pictures already scheduled in the satellite's routine, in this case a high priority would be defined for the goal described in stage 3.1 of PRODESEA.

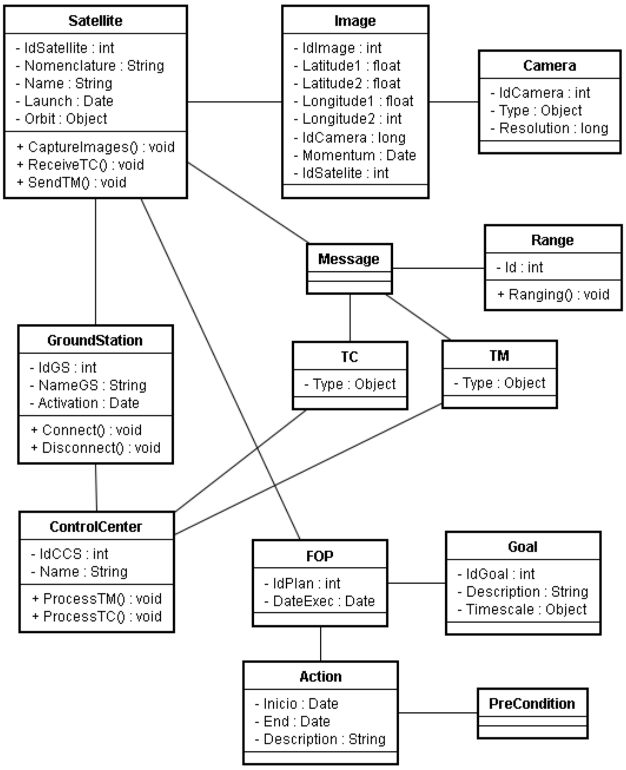

Figure 16. UML Class Diagram representing knowledge

By applying stage 4.1 of PRODESEA (Short Time Goals Selection) to the case study, it is stated that:

- The assessment of the goals previously classified as STP (the relevant in this stage of PRODESEA is to separate critical goals, that must be accomplished in a short time frame) is performed.

By applying stage 4.2 of PRODESEA (Sort STP Goals Order by Priority) to the case study, it is stated that:

- Among the goals raised in the previous stage, a priority order is established in this stage of PRODESEA.

By applying stage 4.3 of PRODESEA (Allocation of Demanded Resource) to the case study, it is stated that:

- The separation of resources considered as necessary for the accomplishment of each of the goals classified as STP, respecting the classification of the previous stage.

By applying stage 4.4 of PRODESEA (Plan Creation) to the case study, it is stated that:

- The first version of the plan is generated in this stage of the process, and contains only the STP goals (usually, the critical goals).

For this stage (and for stages 4.8 and 4.12), the UML Activity Diagram can be used, by way of showing the necessary stages to reach the goal with the plan, as shown in the example of figure 17.

By applying stage 4.5 of PRODESEA (Medium Time Goals Selection) to the case study, it is stated that:

- The assessment of goals classified as MTP occurs in this stage of PRODESEA.

By applying stage 4.6 of PRODESEA (Sort MTP Goals Order by Priority) to the case study, it is stated that:

- Among the goals raised in the previous stage, a priority order is established, by placing a more urgent but not critical goal in front of the others in the MTP group.

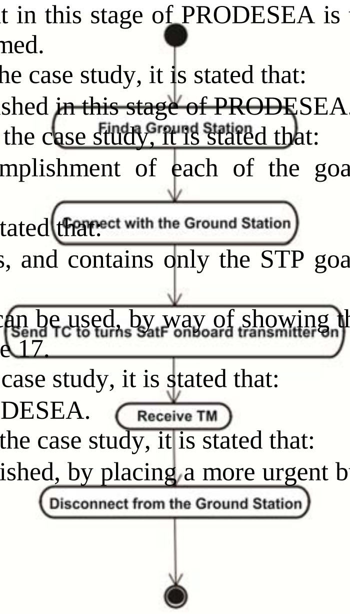
Find a Ground Station
Connect with the Ground Station
Send TC to turns SatF onboard transmitter on
Receive TM
Disconnect from the Ground Station

Figure 17. UML Activity Diagram

By applying stage 4.7 of PRODESEA (Allocation of Demanded Resource) to the case study, it is stated that:

- The separation of resources considered as necessary to accomplish each of the goals classified as MTP, respecting the classification of the previous stage.

By applying stage 4.8 of PRODESEA (Plan Update) to the case study, it is stated that:

- The generation of a second version of the plan occurs in this stage of the process, and contains the MTP and STP goals.

By applying stage 4.9 of PRODESEA (Long Time Goals Selection) to the case study, it is stated that:

- The assessment of goals classified as LTP finally occurs, with the purpose of placing the long-term goals in the end of the plan.

By applying stage 4.10 of PRODESEA (Sort LTP Goals Order by Priority) to the case study, it is stated that:

- Among the goals raised in the previous stage, a priority order is established.

By applying stage 4.11 of PRODESEA (Allocation of Demanded Resource) to the case study, it is stated that:

- The separation of resources considered necessary to accomplish each of the LTP goals, respecting the classification of the previous stage.

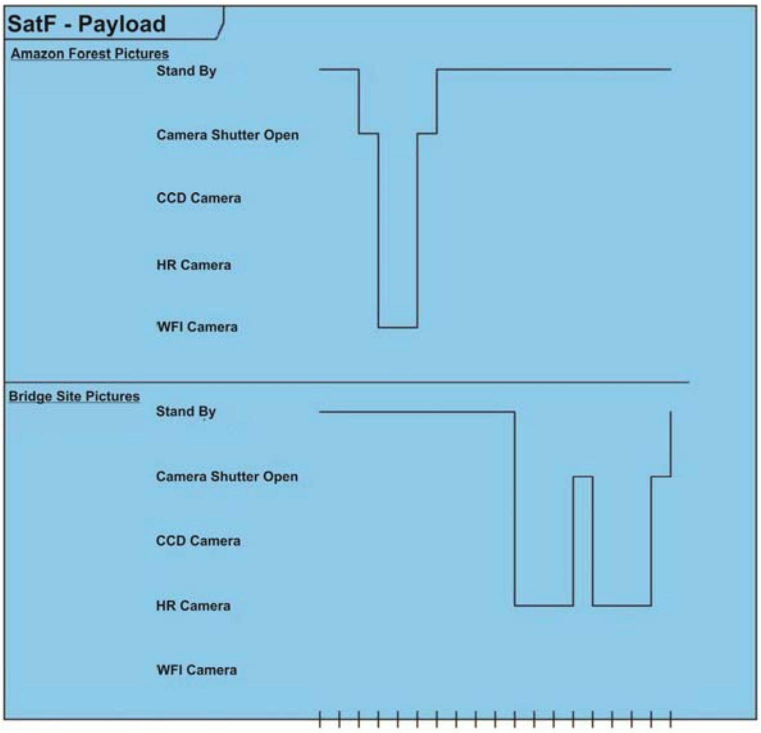

Figure 18. UML Timing Diagram to show the concurrence of plans

By applying stage 4.12 of PRODESEA (Plan Update) to the case study, it is stated that:

- The final version of the plan is generated, and contains all the reachable goals.

By applying stage 5.1 of PRODESEA (Plan Integrity Verification) to the case study, it is stated that:

- Testing of the plan steps in time to check the occurrence of one or more conflicts with the plans published in the execution line.

For this stage, the UML Timing Diagram can be used in order to show the concurrence of plans for resources in timeline, as shown in example of figure 18.

By applying stage 5.2 of PRODESEA (Goals Coverage Verification) to the case study, it is stated that:

- The assessment of goals to check if they will be accomplished with accordance to the plan.

By applying stage 5.3 of PRODESEA (Plan Simulation) to the case study, it is stated that:

- The execution of the plan in the Simulator, which must generate data to be analyzed in the following stage.

By applying stage 5.4 of PRODESEA (Simulator Outcome Analysis) to the case study, it is stated that:

- The analyses of the material produced by the Simulator during plan execution.

If we apply stage 6.1 of PRODESEA (Plan Approval) to the case study, it is stated that:

- The approval or rejection of the plan after the analyses of the plan simulation outcome.

By applying stage 6.2 of PRODESEA (Plan Publication) to the case study, it is stated that:

- The plan publication, its charging on the line or list of plans with commands to be sent to the satellite.

By applying stage 6.3 of PRODESEA (Plan Register) to the case study, it is stated that:

- Record of the plan in the Knowledge Base, because the knowledge applied to the plan must then return to the Knowledge Base, completing the cycle with this re-alimentation.

IV. Conclusion

A development process in the area of automatic planning was shown in this paper with an example in the form of a case study, in the domain of satellites control and tracking, to illustrate such process.

For that matter, various aspects of the planning in the AI area were exploited, always bearing in mind the aim of defining a development process for spatial systems that use the automatic planning in the search for optimization of the scarce resources in INPE.

This paper also points out how and when is the moment of generating fourteen software artifacts, considering the plan itself (the FOP) as one of the artifacts generated in the process.

From these fourteen software artifacts (eleven different ones), nine were framed as mandatory and five as optional, as observed in figure 19. They are:

- UML Use Case Diagrams in 1.1 and in 1.3 (mandatory);
- Business Glossary in 1.1 (mandatory);
- Brief Description of the Problem in 1.2 (mandatory);
- Textual Description of the Scenarios in 1.3 (mandatory);
- UML Sequence Diagrams in 1.3 (optional);
- URD in 1.3 (mandatory);
- UML Class Diagram in 2.3 (mandatory);
- UML Package Diagram in 2.3 (optional);
- UML State Diagram in 2.3 (optional);
- UML Activity Diagrams in 4.4 (optional), in 4.8 (optional) and in 4.12 (mandatory);
- FOP in 4.12 (mandatory);
- UML Timing Diagram in 5.1 (mandatory).

The mandatory software artifacts of PRODESEA are represented in red, whilst the optional software artifacts are shown in blue, as presented in figure 19.

Besides the advantage attained with the standardization of the software artifacts created, the use of UML favors the comprehension of the material produced simply because it s a known and largely employed modeling tool, either in the academic environment or in the professional market.

PRODESEA contributes as a guide to be followed in the production of software artifacts originated in the automation of operations that make use of planning.

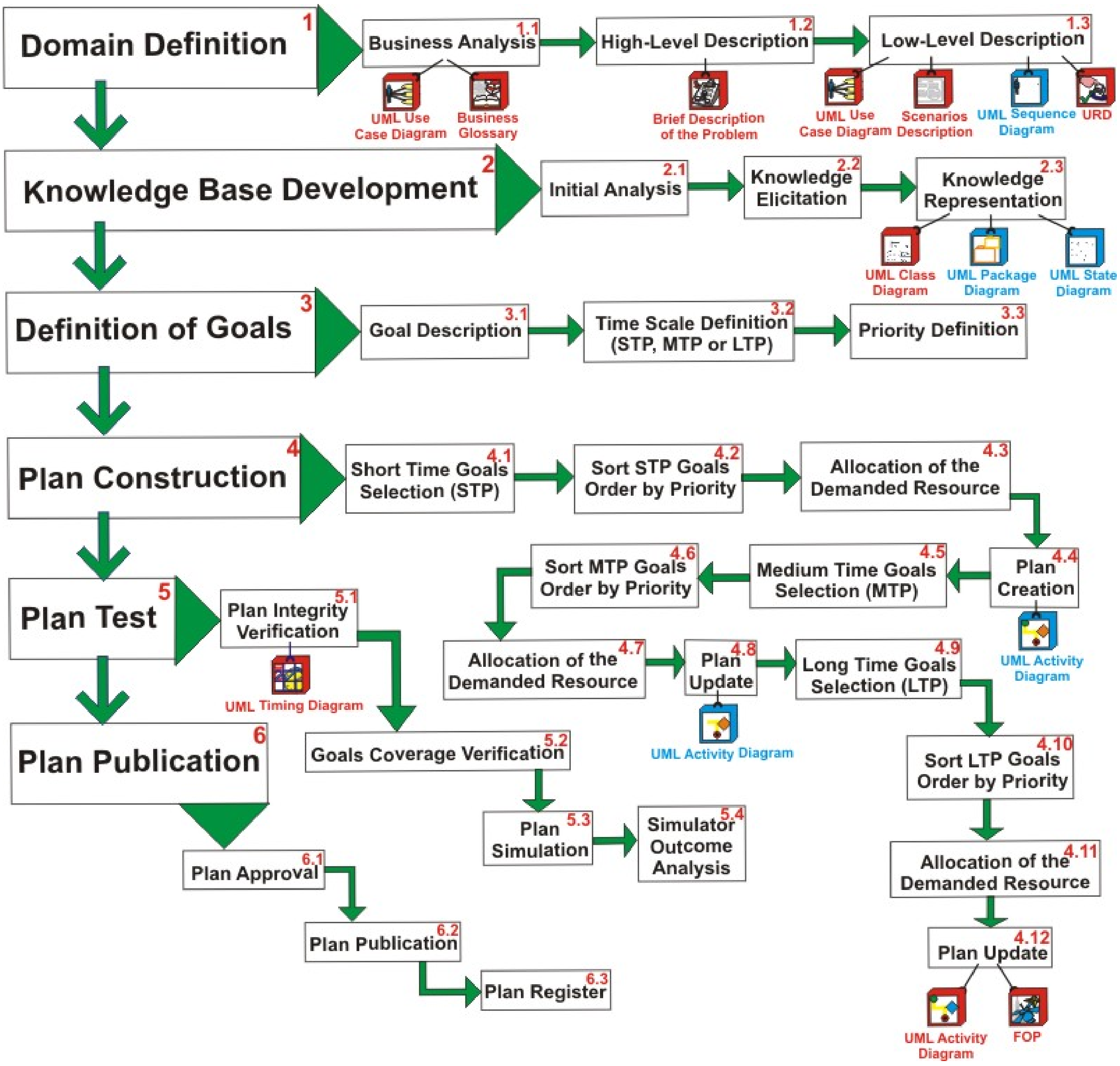
Domain Definition 1
Business Analysis 1.1
UML Use Case Diagram
Business Glossary
High-Level Description 1.2
Brief Description of the Problem
Low-Level Description 1.3
UML Use Case Diagram
Scenarios Description
UML Sequence Diagram
URD
Knowledge Base Development 2
Initial Analysis 2.1
Knowledge Elicitation 2.2
Knowledge Representation 2.3
UML Class Diagram
UML Package Diagram
UML State Diagram
Definition of Goals 3
Goal Description 3.1
Time Scale Definition (STP, MTP or LTP) 3.2
Priority Definition 3.3
Plan Construction 4
Short Time Goals Selection (STP) 4.1
Sort STP Goals Order by Priority 4.2
Allocation of the Demanded Resource 4.3
Plan Creation 4.4
UML Activity Diagram
Medium Time Goals Selection (MTP) 4.5
Sort MTP Goals Order by Priority 4.6
Allocation of the Demanded Resource 4.7
Plan Update 4.8
UML Activity Diagram
Long Time Goals Selection (LTP) 4.9
Sort LTP Goals Order by Priority 4.10
Allocation of the Demanded Resource 4.11
Plan Update 4.12
UML Activity Diagram
FOP
Plan Test 5
Plan Integrity Verification 5.1
UML Timing Diagram
Goals Coverage Verification 5.2
Plan Simulation 5.3
Simulator Outcome Analysis 5.4
Plan Publication 6
Plan Approval 6.1
Plan Publication 6.2
Plan Register 6.3

Figure 19. The mandatory artifacts of PRODESEA are represented in red, whilst the optional are shown in blue

In the end, it is expected that the development process created is used as a development and a pattern guide of software artifacts to be generated during the whole development process of new products with automated planning features within INPE, and in particular in the CSS.

References

[1]Winandy, C-E., “Processo de desenvolvimento de planos automáticos para operação de satélites baseado na UML”, INPE 05.27.13.45-TDI, 2011.

[2]Winandy, C-E., and Ferreira, M.G.V., *Development process for applications of automated planning for satellites control*, 11th SpaceOps, Huntsville – AL, AIAA, 2010.

[3]Fujita, H. S., “MAPOS: Método de Análise e Projeto Orientado a Serviços”. POLI-USP, 2009.

[4]Nikolopoulos, *Expert Systems: Introduction to First and Second Generation and Hybrid Knowledge Based Systems*, Dekker, New York, 1997.

[5]Martin, J. and Oxman, S., *Building expert systems*, Engle wood Cliffs, Prentice Hall, NJ, 1988.

[6] Cordingley, E. S., "Knowledge elicitation techniques for knowledge-based systems", *Knowledge elicitation: principle, techniques and applications*, Springer-Verlag New York, Inc., New York, NY, 1989.

[7] Russel, S. and Norving, P. *Artificial intelligence: a modern approach*, New Jersey - NJ, EUA: Prentice-Hall, 1995.

[8] Haddow, C.R. and Adamson, K. and Sousa, B. and Whitehead, G., *Mission planning – Establishing a common concept for ESOC's mission*, 11th SpaceOps, Huntsville – AL, AIAA, 2010.

[9] Johnston, M. D., *Request-driven schedule automation for the Deep Space Network*, 11th SpaceOps, Huntsville – AL, AIAA, 2010.

[10] Koller, M., Reggestad, V., Adamson, K. and Kay, R, *ESOC earth observation missions and the automation of operational routine tasks*, 11th SpaceOps, Huntsville – AL, AIAA, 2010.