67th International Astronautical Congress (IAC), Guadalajara, Mexico, 26-30 September 2016.

IAC-16- 34891

A FRAMEWORK FOR OVERSIGHT OF SOFTWARE´S SUPPLIERS OF SAFETY-CRITICAL SPACE SYSTEMS BASED ON CIVIL AVIATION BEST PRACTICES

Benedito M. Sakugawa[a]*, Ana M. Ambrosio[b], Geilson Loureiro[b], Carlos H. N. Lahoz[c]

[a] *Agência Nacional de Aviação Civil (ANAC),Avenida Laurent Martins, 209, São José dos Campos,SP,Brazil,CEP12242-431,* benedito.sakugawa@anac.gov.br

[b] *Instituto Nacional de Pesquisas Espaciais (INPE), Avenida dos Astronautas, 1758, São José dos Campos, SP,Brazil, CEP12227-010,* ana.ambrosio@inpe.br, geilson@lit.inpe.br

[c] *Instituto de Aeronáutica e Espaço (IAE), praça Marechal Eduardo Gomes, 50, São José dos Campos,SP, Brazil, CEP12228-901,* lahozchnl@iae.cta.br

* Corresponding Author

Abstract

The Brazilian Program of Space Activities for the period 2012-2021 has among its priorities: to engage industry at all stages of the space project development, the standardization and certification, and mastering of critical technologies. Considering the outsourcing growth of increasingly complex systems, the certification demand tendency, the relevance and critical role of software for embedded space systems, the commonality between space and aviation domains, and the current maturity level of Brazilian space industries, this paper presents a framework for oversight of software's supplier of safety-critical space systems based on metrics and best practices of the civil aviation. The metrics are used for evaluation of the oversight and decision making support. They are generated by using the civil aviation past twelve years oversights' results. Those oversights have been performed by the National Civil Aviation Agency (ANAC), some jointly with the Federal Aviation Administration (FAA) and the European Aviation Safety Agency (EASA), mostly on-site at the supplier's facilities, and comprises systems for flight controls, brake, landing gear, electrical generation and distribution, pressurization, cockpit displays, flight management, etc. Software safety systematic comparison between space and aviation domains was performed in order to identify the potential reuse level from aviation and adjustments due to space specific necessities. The comparison shows a great amount of aviation reuse, but due to the space necessities many additions covering different topics are needed (e.g., delivery and acceptance, inflight modification), but the identified differences do not preclude the framework viability. The framework is built on the standards of the European Cooperation for Space Standardization (ECSS) as base, and focuses on relevancies of company, process and product for software safety impact, together with a reduced set of activities. The authors believe this approach can better suit to the current stage of Brazilian space industry (small companies), and can help in reducing to an acceptable level the presumed inherent risk that space systems software outsourcing has in adversely impacting safety, by identifying project problems and product potential problems at earlier stages of software development.

Keywords: software safety; supplier oversight; space system; civil aviation certification

1. Introduction

In line with the world tendency, the Brazilian Program of Space Activities (PNAE) [1] for the period 2012-2021 has included, among the priorities:

- Engage industry at all stages of the space project development - from equipment conception and construction to complete space systems;
- Standardization and certification to ensure the quality and safety of space activities in the country.

In such scenario, supplier oversight activity, especially of complex space systems, is growing in importance either by increased outsourcing and its scope, or eventually for compliance verification with certification regulations.

The PNAE also highlights among its priorities, "master critical technologies and restricted access technologies, with the industry's participation, and with the expertise and talent in universities and national research institutes". The embedded software can be considered one of the critical technologies. According to Leveson and Weiss [2], software is quickly becoming a major part of and a major concern in space applications. It is also playing an increasing role in space accidents [3].

The National Institute for Space Research (INPE), where this research has been carried out, is responsible for the development of main Brazilian satellites and has followed the European trend of standardization since its first Space mission. The ECSS documents focus on creating a common language and a standard means of development between customer and supplier, and allow the customer-supplier contract to define the mandatory requirements of the ECSS standards. This characteristic makes the ECSS standards unclear regarding to what requirements are mandatory [4], and unfeasible to apply in the original to certification activities. Moreover, the assessment and improvement of software processes for the European space industry [5] may not be adequate to the current maturity stage of Brazilian space industries.

Although the 1967 Space Treaty states that each country is internationally accountable for its national space activities and responsible for the damage caused to other countries, the certification activity in the space field is being built, so there is no definite standard and consensus among certifiers in different countries [6].

Aviation and Space share many concerns, needs and solutions in terms of processes, methods and techniques [7,8,9]. Particularly for software, the Civil Aviation performs oversight-like activities throughout the development for verifying compliance with the certification regulation. The civil aviation contains harmonized regulations among the various member nations of the International Civil Aviation Organization (ICAO), added by a vast technical material open for consultation, as the result of long certification history. The ANAC is responsible for certification in Brazil and adopts the rules, standards and guidelines used by the FAA.

Considering the outsourcing growth of increasingly complex systems, the certification demand tendency, the relevance and critical role of software for embedded space systems, the proximity between space and aviation, and the current maturity stage of Brazilian space industries, this work presents a framework for oversight of software supplier of safety-critical embedded space systems, based on metrics and best practices of the civil aviation. The Oversight Framework focuses on key relevancies of company, processes and product for software safety impact, together with a reduced set of activities. This approach can better suit to the current stage of Brazilian space industry.

The paper is organized as follow: section 2 presents the Civil Aviation best practices, section 3 provides an overview of the Oversight Framework, section 4 explains the construction of the Oversight Framework, section 5 shows the results and discussion, and finally the section 6 presents the conclusion.

2. Civil Aviation Best Practices

The major Civil Aviation certification agencies recognize the RTCA-DO-178C [10] as an acceptable means of compliance for approval of software in airborne systems and equipment. The DO-178C comprises process of planning, development, verification, quality assurance, configuration management and certification. A list of 71 objectives is provided, and if the developer can show compliance with the applicable objectives and related activities, the software will be approved for use in the aircraft under certification. The DO-178C states that the certification authority may review the software life cycle processes and data for compliance verification. The FAA Order 8110.49 [11] provides guidelines on those reviews, and the figure-1 illustrates when they occur during the software life cycle.

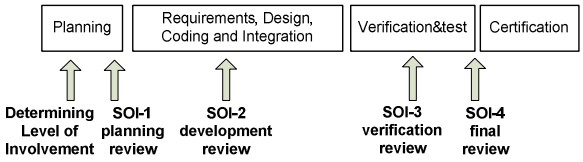

Fig.1, Certification reviews in lifecycle

The first activity determines the authority level of involvement by evaluating company experience, use of subcontractors, level of reuse, use of new technologies, required safety level, system complexity, etc., and the result defines which subsequent activities are necessary, which may be from none to all reviews called *"Stage of Involvement"* (*SOI*), and also informally known as audits. The *SOI-1* is usually a desktop review of planning documents. The *SOI-2* and *SOI-3* are usually on-site reviews of the implemented processes and related life cycle data. The *SOI-4* is to ensure that planned activities have been satisfactorily accomplished and there is no pending that could adversely impact safety. In order to assist in performing those reviews, the FAA has created a Jobaid [12] to be used as a reference tool. Although the Job aid is not all inclusive of all possible situations that need to be reviewed, it provides a good picture of the scope of the *SOIs*. Although Order 8110.49 and Job aid refer to DO-178B [13], their contents are still applicable in the scope of this work, as the basic characteristic has been preserved from DO-178B to DO-178C, and the main differences are on the supplements that provide specific technology-dependent guidance [14,15,16,17].

Although determining the level of involvement and performing related reviews are under certification authority scope, aviation companies usually do similar activities in order to mitigate certification risk and adverse safety impact. In this case, it is in the scope of

supplier oversight. The Oversight Framework presented in this paper is based on it.

3. An overview of the Oversight Framework

3.1 – General context and scope

The motivation base to construct the Oversight Framework is the "*space tendency*", "*civil aviation maturity in certification*" and "*similarity between aviation and space*." The space domain tendency is for more oversight, either due to the outsource growth of increasingly complex parts or the need for regulation and consequent certification activity. In this scenario, the civil aviation high level of maturity in certification comes as a potential source of contribution, because there are many similarities between these two domains, particularly regarding to software-intensive critical embedded systems. The figure-2 shows the general context of the Oversight Framework.

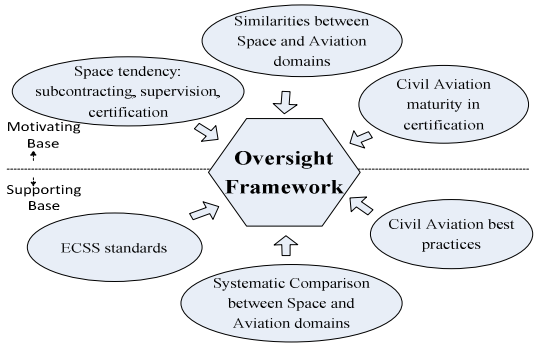

Fig. 2, General context of the Oversight Framework

The supporting base for the Oversight Framework construction comprises "*ECSS standards*", "*Civil Aviation best practices*" and "*Systematic Comparison.*" The Systematic Comparison identifies similarities and differences between space and civil aviation in order to apply the civil aviation best practices customized for the space domain to build the Oversight Framework. The figure-3 shows the Oversight Framework scope in the different phases of the space mission development. The Oversight Framework covers mainly the phases B, C and D.

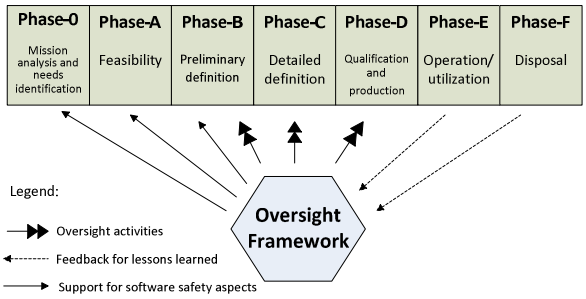

Fig. 3, The Oversight Framework general scope

The emphasis of the oversight application is when the software supplier is defined (phase B), the software is developed (phase C), verified and delivered (phase D). However, the Oversight Framework also works in the earlier stages (stage 0 and A) providing support regarding software safety concerns, as well as in later phases (E and F) evaluating feedbacks from operational and disposal difficulties, and their impacts in the Oversight Framework as part of the lessons learned process.

3.2 – Main activities

The oversight activities begin with a risk assessment in the software supplier. The result of the risk assessment will define which subsequent oversight activities are necessary, starting from desktop review of key documents, e.g., the development plan and software delivery document (the lowest critical), up to a permanent staff on supplier's site (highest critical case), and may perform up to five formal reviews (intermediate cases) as follow:

Planning review: usually a desktop review of planning documents like development plan, verification plan, configuration management plan, quality assurance plan, and any standard documents to be adopted (e.g., requirements standard, coding standard), in order to ensure compliance to the software safety criticality level.

Requirements and architecture review: usually an on-site review of the processes implemented (tools, procedures, etc.) as well as the quality of the requirements, preliminary architecture and related life cycle data, in order to ensure compliance to the planning documents and adopted standards.

Design and implementation review: usually an on-site review of the processes implemented as well as the quality of the detailed architecture, source and object code, and related life cycle data, in order to ensure compliance to the planning documents and adopted standards.

Verification review: usually an on-site review of the processes implemented as well as the quality of the verification activities (e.g., reviews, analysis, inspections, testing) and related life cycle data, in order to ensure compliance to the planning documents and adopted standards.

Final review: usually a desktop review to ensure that planned activities have been satisfactorily accomplished and there is no pending that could adversely impact safety.

The figure-4 shows the main oversight activities of the Oversight Framework in relation to the mission phases and reviews.

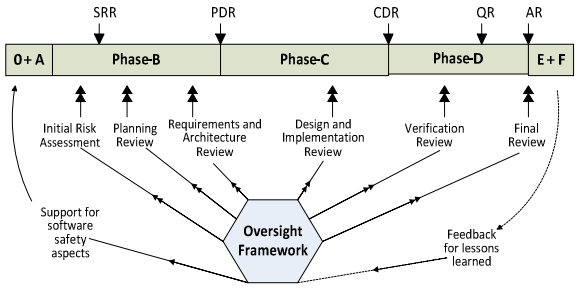

Fig. 4, Oversight Framework main activities

Figure-5 shows at what periods of the software life cycle the initial risk assessment and formal reviews occur. The figure uses as reference the software life cycle process defined by the ECSS [18].

The initial risk assessment should occur:

- after the software supplier selection and during the early stage of the "software management process";
- during the final stages of the software product definition (the second half of " software related system requirement process"), already with the supplier participation in the software definition finalization;
- After starting the planning of development and V&V.

The planning review should occur:

- After defining the software product, i.e., after the system requirements allocated to software have been reviewed and baselined ,i.e., conclusion of the System Requirements Review (SRR);
- After finishing the planning of development and Verification and Validation (V&V);
- Before starting development and V&V activities.

The requirement and architecture review should occur:

- After more than 50% of the requirements and architecture have been defined, verified and validated;
- Before starting the software design and implementation;

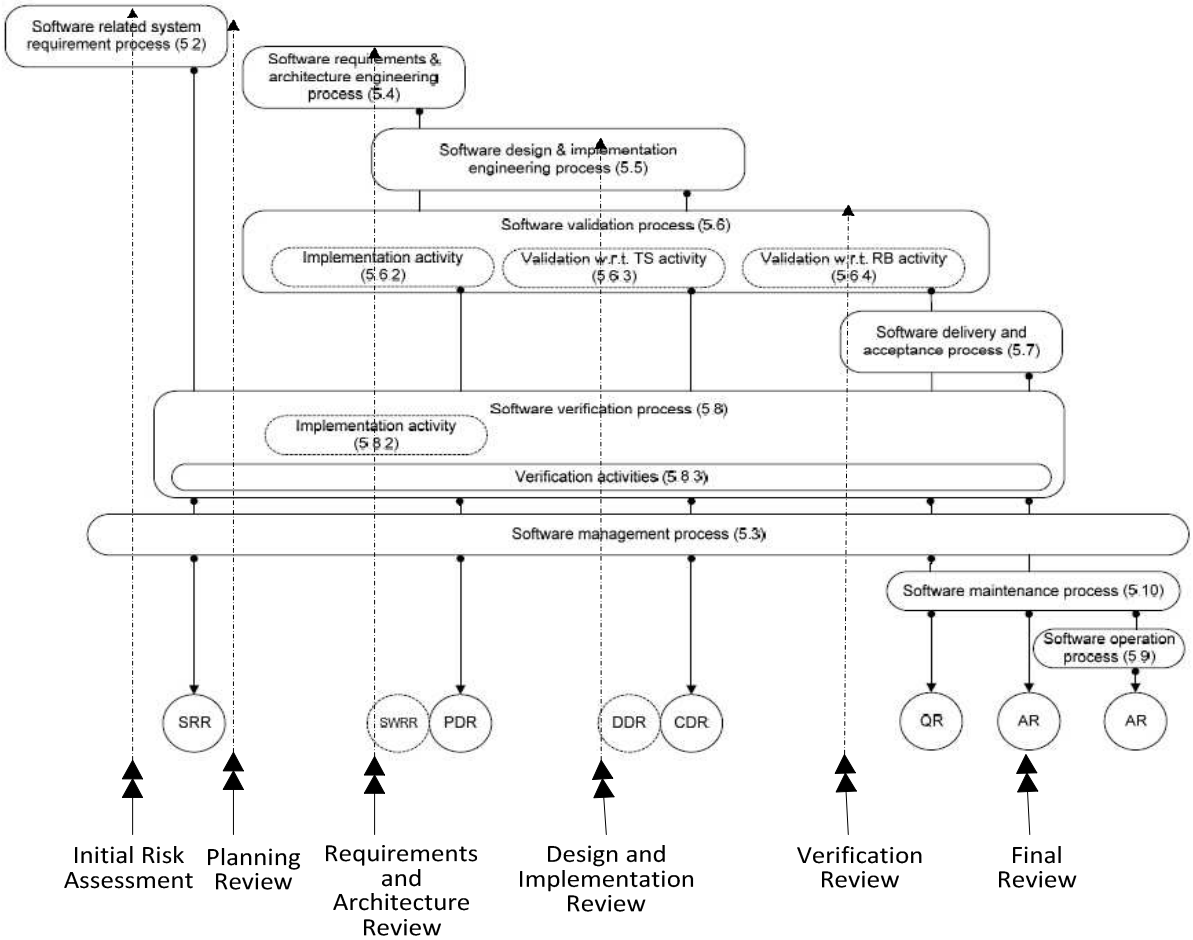

Fig. 5, Oversight Framework activities in the ECSS software life cycle

- Before starting the Software Requirements Review (SWRR), as the focus is on the assessment of the processes and of a representative sample of requirements and architecture. It is an important mitigation for the Preliminary Design Review (PDR).

The design and implementation review should occur:

- After more than 50% of the design and implementation have been completed, verified and validated;
- Before starting the Detailed Design Review (DDR), as the focus is on the assessment of the processes and of a representative sample of design and implementation.

The verification review should occur:

- After more than 50% of the requirements have been V&V through testing;
- Before starting the delivery/acceptance process.

The final review can occur simultaneously with the Acceptance Review (AR) or be part of it as a complement, in order to verify previous reviews pending.

3.3 – Main components

The Oversight Framework comprises the following components illustrated in the figure-6.

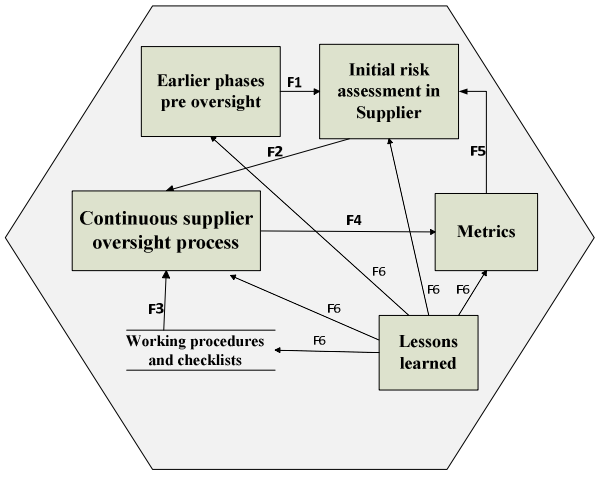

Fig.6, The Oversight Framework main components

The components are summarized below:

a- Earlier phases pre-oversight: Support for software safety aspects in phase-0 and phase-A.

b- Initial risk assessment in supplier: evaluation of company experience, use of subcontractors, level of reuse, new technologies, required safety level, system complexity, etc., to determine the oversight activities.

c- Continuous supplier oversight process: planned activities commensurate to the risk assessment outcome (e.g., permanent on-site staff for training and supervision, periodic on-site reviews, periodic desktop reviews, etc).

d- Working procedures and checklists: for supporting the planned activities.

e- Metrics for oversight evaluation and decision making support.

f- Lessons learned from phase-E and phase-F to improve the Oversight Framework.

The flows among the components are summarized below:

F1: specific software safety concerns detected during phase-0 and phase-A;

F2: oversight activities to be performed as a result of the risk assessment;

F3: set of working procedures and checklists to be used during oversight activities;

F4: oversight results for generation of measurement by applying the metric;

F5: measurement evaluation for continuous risk assessment;

F6: process improvement.

4. The Framework construction

The main activities for the construction of the Oversight Framework are as follow:

a- Systematic comparison between space and civil aviation domains to identify commonalities that may allow for the use of aviation best practices, or identifies differences between both due to the specific needs of aviation and space;

b- Metric generation for supplier oversight in civil aviation certification, followed by analysis and customization for using in space domain;

c- Customization of civil aviation best practices to space, applied to ECSS standards and focusing on supplier oversight, based on the systematic comparison result;

d- Case study using INPE's projects and applying different phases of the Oversight Framework;

4.1 The Systematic Comparison Process

In order to have confidence that the systematic comparison provides a representative result, first an investigated work was performed on software safety comparison, where it was possible to identify some assumptions and limitations, as well as a set of comparison criteria. Then a Systematic Comparison Process was specified assuming the assumptions identified and using the set of comparison criteria. Taking into account the limitations identified, the

Systematic Comparison Process had to cover the following concerns:
a-Ensure domains' comparison at adequate level, regardless of standards scope;
b-Identify clearly the differences and similarities between both domains;
c-Ensure software safety coverage of both domains;
d-Facilitate identifying reuse level and adjustments from Civil Aviation.

For facilitating analysis of reuse and adjustment (taking the civil aviation best practices as reference), the comparison results were classified in significant, major, or minor impacts (additions, deletions) and major or minor reuses**.** If no adjustments are possible or they are of such magnitude that makes it unfeasible to reuse the Aviation best practices, then the impact is considered significant.

4.2 Metric generation

Metric is generated by analyzing the civil aviation past twelve years audit's results, which have been performed by ANAC, some jointly with the FAA and EASA, mostly on-site at the supplier's facilities, and comprising systems for flight controls, brake, landing gear, electrical generation and distribution, pressurization, cockpit displays, flight management, etc. A survey is performed with software safety specialists from ANAC and Brazilian aviation industry, in order to obtain in quantitative terms the relevance of the criteria used for the metric generation, as well as to obtain a score (based on experience, feeling), for some selected audit issues based on real cases. The result of the survey is used for adjustment of the metric. The adjusted metric is applied to the twelve years audit's results, and the resulting measurements are evaluated against the pre and post certification history of the related software.

4.3. Customization of civil aviation best practices

First, the aviation framework architecture is built based on the best practices described in section 2. Then, an Impact Analysis is performed on that framework by using the classified results of the Systematic Comparison Process. In case any significant impact is found (or combination of major impacts that lead to a significant impact), then the Oversight Framework construction is cancelled. Otherwise, the reuses are identified (first major, then minor), followed by major adjustments (first deletions, then additions) and completing with minor adjustments. The resulting architecture is the base for the Space Oversight Framework construction.

4.4. Case study

Case studies for evaluation of the Space Oversight Framework should use on going INPE's projects due to the characteristics of the phases performed on-site the development environment, which focus on the actual implementation of the process, i.e., not limited to documents review. However, INPE's past projects can also be used for the case of the first phase (planning), which is basically a desktop review focused on documentations. The past projects can also be used for evaluation of the metric by applying to records of past audits or reviews.

5. Results and discussion

Concerning the **Systematic Comparison** results for the Aviation best practices, the great majority is for potential reuse, but some remaining were considered Aviation-only not applicable to the Space, and need to be deleted. For the Space the majority is equivalent to the reusable content from Aviation, but a considerable amount need to be added. Only a few was considered not applicable to the scope of the Space Oversight Framework. It was not found any impact considered Significant.

Concerning the **metric generation**, the analysis of the past twelve years audit's results have identified the following criteria to consider:

a. the purpose of the recorded audit item;
b. the type of life cycle data where the audit item was recorded against;
c. the root cause in terms of human error;
d. the distance between the recorded audit item and the final executable code;
e. the amount of data impacted by the recorded audit item;
f. the time adequacy of the recorded audit item (i.e., whether the item scope and audit scope are same or not).

These criteria have been submitted to software safety specialists for survey, which is still on going.

Concerning the **customization of civil aviation best practices to space,** it is still in preliminary stage and there is no relevant result at this time.

Concerning the **case study**, it was selected an INPE project called "QSEE – Qualidade do Software Embarcado em Aplicações Espaciais" [19], which focuses on the quality of the embedded software in Space applications. But as the project is already concluded, it can be used mostly to exercise the Oversight Framework activities related to desktop reviews (i.e., documents evaluation), and also to exercise the metric by applying it in the review records. For the Oversight Framework on-site activities, which assess the quality of the implemented process in the software development environment, no projects have been selected yet, but due to time constraints, it may be necessary to select more than one project in order to exercise different phases in parallel (e.g., project-A for Requirements and Architecture Review, and project-B for Verification Review.

6. Conclusion

The increase of complex systems outsourcing, the certification demand tendency, and the relevance and critical role of software, support the importance of oversight activities in software suppliers of safety critical space systems. The systematic comparison between space and civil aviation has not shown any result that makes it unfeasible to reusing the Aviation best practices, though a considerable amount of adjustments are need due to specific characteristics of space and aviation domains. The criteria identified for the metric generation is independent of the domain because do not use any specific aviation characteristics that are not applicable to space. The fact that the criteria do not depend on the domain contributes for possible reuse in the Space Oversight Framework. The selection of projects to use as case study is an important concern, especially regarding to exercising on-site activities, which allow for assessing the quality of implemented process in the development environment. For those cases, on-going projects are necessary.

Disclaimer

Although one author is an ANAC employee, this paper does not represent the official ANAC position, but solely the opinion of its authors.

Acknowledgements

The authors are thankful to the colleagues of ANAC and INPE for the time dedicated to review the technical material related to this work.

Carlos Lahoz gratefully acknowledge the financial support of the Ciência sem Fronteiras / Conselho Nacional de Desenvolvimento Científico e Tecnológico (CsF/CNPq) and Fundação Lemann/Brasil.

.

References

[1] Programa Nacional de Atividades Espaciais - PNAE - 2012 – 2021, Agência Espacial Brasileira, Ministério da Ciência, Tecnologia e Inovação, Brasília.

[2] N.G. Leveson, K.A. Weiss, Safety Design for Space Systems, Chapter 15, Software System Safety, Butterworth-Heinemann, 2009.

[3] N.G. Leveson, The Role of Software in Spacecraft Accidents, AIAA Journal of Spacecraft and Rockets, 2004.

[4] J.P. Blanquart, J.M. Astruc, P. Baufreton, J.L. Boulanger, H. Delseny, J. Gassino, G. Ladier, E. Ledinot, M. Leeman, J. Machrouh, P. Quéré, B. Ricque, Criticality categories across safety standards in different domains, ERTS-2012, Toulouse, France, 2012, 1 - 3 February.

[5] European Cooperation for Space Standardization, ECSS-Q-HB-80-02 part 1A, Space Product Assurance – Software process assessment and improvement – Part 1: Framework, 2010.

[6] A.R.S. Carvalho, J.H. Damiani, A.O.N. Follador, M.G.O. Guimaraes, An Overview of the Certification of VSB-30 with Emphasis on Technological Inovation, Journal of Aerospace Technology and Management – JATM, 2012.

[7] P. Baufreton, J.P. Blanquart, J.L. Boulanger, H. Delseny, J.C. Derrien, J. Gassino, G. Ladier, E. Ledinot, M. Leeman, J. Machrouh, P. Quéré, B. Ricque, Multi-domain comparison of safety standards, ERTS-2010, Toulouse, France, 2010, 19-21 May.

[8] J. Machrouh, J.P. Blanquart, P. Baufreton, J.L. Boulanger, H. Delseny, J. Gassino, G. Ladier, E. Ledinot, M. Leeman, J.M. Astruc, P. Quéré, B. Ricque, Cross domain comparison of System Assurance, ERTS-2012, Toulouse, France, 2010, 1-3 February.

[9] E. Ledinot, J. Gassino, J.P. Blanquart, J.L. Boulanger, P. Quéré, B. Ricque, A cross-domain comparison of software development assurance, ERTS-2012, Toulouse, France, 2012, 1-3 February.

[10] RTCA, Inc. RTCA/DO-178C: Software Considerations in Airborne Systems and Equipment Certification, 2011.

[11] Federal Aviation Administration, Order 8110.49 chg1, Software Approval Guidelines, 2011.

[12] Federal Aviation Administration, Aircraft Certification Service, Job Aid, Conducting Software Reviews prior to certification, 2004.

[13] RTCA, Inc. RTCA/DO-178B: Software Considerations in Airborne Systems and Equipment Certification, 1992.

[14] RTCA, Inc. RTCA/DO-330**:** Software Tool Qualification Considerations, 2011.

[15] RTCA, Inc. RTCA/DO-331**:** Model-Based Development and Verification Supplement to DO-178C and DO-278A, 2011.

[16] RTCA, Inc. RTCA/DO-332**:** Object-Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A, 2011.

[17] RTCA, Inc. RTCA/DO-333**:** Formal Methods Supplement to DO-178C and DO-278A, 2011.

[18] European Cooperation for Space Standardization, ECSS-E-ST-40C: Space Engineering – Software, 2009.

[19] A.M. Ambrosio, F.M. Francisco, E. Martins, An Independent Software Verification and Validation Process for Space Applications, AIAA SpaceOps, 2008.