

PADRONIZAÇÃO DOS PROCEDIMENTOS DE TESTE PARA COMPUTADORES DE SUPERVISÃO DE BORDO DE SATÉLITES

Thiago Duarte Pereira¹, Fabricio de Novaes Kucinskis² e Mauricio Gonçalves Vieira Ferreira³

Instituto Nacional de Pesquisas Espaciais,
Coordenação de Engenharia e Tecnologia Espaciais,
Curso de Sistemas Espaciais (CSE), São José dos Campos, SP, 12227-010
¹thiago@dea.inpe.br, ²fabricio@dea.inpe.br e ³mauricio@ccs.inpe.br

Resumo: Este artigo descreve um trabalho relacionado à padronização e automatização dos procedimentos de teste aplicados a um Computador de Supervisão de Bordo, On-Board Data Handling (OBDH) do Instituto Nacional de Pesquisas Espaciais (INPE). Os procedimentos de teste criados durante seu desenvolvimento, se padronizados, podem ser reusados em fases posteriores, como a aceitação do computador, integração e operação do satélite. Através da automatização da execução de seus procedimentos de teste, reduziremos significativamente todo esforço gasto na aplicação. O principal interesse é reduzir o custo dos testes e, se possível, contribuir também para a criação dos procedimentos de operação para satélites.

Palavras-chaves: engenharia especial, engenharia de sistemas espaciais, verificação e validação de sistemas espaciais, procedimentos de teste.

1 Introdução

Em função da complexidade presente em um satélite, a execução de procedimentos de teste é uma das atividades mais realizadas durante seu desenvolvimento. Esta tarefa está inserida na área de verificação e validação de sistemas e tem como objetivo garantir a qualidade funcional de seus equipamentos e subsistemas, (Engel, 2010).

Procedimentos de teste são realizados em todas as fases do desenvolvimento de um satélite, principalmente durante sua montagem e integração. São criados diversos procedimentos, em diferentes níveis e linguagens, nas diversas fases de desenvolvimento. Para tanto, fatores como tempo e custo são extremamente altos, porque é exigido muito esforço para planejar, descrever, aplicar e manter um histórico dos procedimentos de teste. Embora estes procedimentos sejam aplicados em fases e contextos diferentes, todos têm um objetivo comum: operar o equipamento de forma a validar seu funcionamento e encontrar potenciais erros de fabricação ou integração.

A padronização da descrição e execução destes procedimentos pode nos trazer vários benefícios. Se padronizados, os procedimentos de teste podem ser reusados nos diversos níveis e etapas do desenvolvimento e conseqüentemente favorecer a redução de tempo e esforço, gastos nas suas reaplicações. Além disso, as ferramentas computacionais que são desenvolvidas para a execução e gestão dos testes também podem ser reusadas em diversas missões, desde que se use o mesmo padrão.

Nos últimos anos, padrões definidos pelo *European Committee for Space Standardization* (ECSS) vêm sendo adotados pelo INPE, com o objetivo de melhorar os processos e atividades de engenharia e gestão durante as fases de desenvolvimento de seus satélites.

O ECSS publicou alguns padrões que podem contribuir no projeto de soluções capazes de facilitar a verificação e validação de sistemas embarcados em satélites. Entre estes padrões, encontra-se o que especifica uma linguagem formal de geração de *scripts* para descrição e execução automática de procedimentos de teste, chamada *Procedure Language under Tests and Operations* (PLUTO). A linguagem é de fácil entendimento, pois tem características semelhantes à linguagem natural.

Para utilizar esta linguagem na descrição e execução de procedimentos, é necessária a existência de um modelo capaz de representar o comportamento real do *System Under Test* (SUT). Para resolver este problema, outro padrão do ECSS define o *Space System Model* (SSM). O SSM é um modelo do sistema espacial organizado de forma hierárquica e pode abranger todos os equipamentos, componentes e subsistemas do SUT. Além de definir o SSM, este padrão também define os tipos de dados que devem compor este modelo. Com o uso deste padrão é possível modelar o SUT, incluindo todos os dados das atividades executadas em bordo, dados de relato e eventos que são gerados durante seu funcionamento. Através do SSM, é possível também, facilitar a aquisição de conhecimento através de históricos gerados durante a aplicação dos testes. Durante a execução, um procedimento PLUTO pode ler ou escrever dados no SSM, por isso, a criação e execução dos procedimentos de teste através desta linguagem devem fazer referência ao modelo.

Este trabalho está inserido no contexto da automatização da execução de procedimentos de testes em sistemas embarcados em satélites. Um de seus objetivos é a definição da arquitetura de um ambiente capaz de automatizar a execução dos procedimentos de teste em sistemas computacionais embarcados usando a linguagem de procedimentos PLUTO. Através desta arquitetura, será possível desenvolver um ambiente capaz de facilitar a verificação e validação dos equipamentos e subsistemas desenvolvidos pelo Grupo de Supervisão de Bordo (SUBORD), da Divisão de Eletrônica Aeroespacial (DEA) do INPE. O Grupo SUBORD desenvolve sistemas computacionais embarcados para os programas institucionais do INPE. Atualmente, o grupo está desenvolvendo um computador de *On-Board Data Handling* (OBDH) através do projeto chamado Sistemas Inerciais para Aplicação Aeroespacial (SIA) e será usado como estudo de caso.

2 O European Committee for Space Standardization (ECSS) e a Linguagem PLUTO

O ECSS foi criado em 1993, substituindo a *European Space Agency* (ESA) como autoridade européia responsável pela criação e publicação de padrões para projetos espaciais. Os membros do ECSS são a ESA, agências espaciais nacionais européias, como a ASI, CNES, DLR e outras, representando a indústria européia e a *Canadian Space Agency* (CSA). Desde sua criação, o ECSS já publicou dezenas de padrões organizados em três principais ramos: Gerenciamento de Projetos Espaciais; Garantia de Produtos Espaciais e Engenharia Espacial (ECSS, 2008 a).

Dentre os padrões pertencentes ao ramo da Engenharia Espacial, encontra-se o ECSS-E-ST-70-32C: *Test and Operations Procedure Language* (ECSS, 2008 c). Este padrão especifica uma linguagem formal para a descrição de procedimentos de teste e operação de satélites. A esta linguagem é dado o nome de *Procedure Language under Tests and Operations* (PLUTO). Este é um novo padrão de linguagem e atualmente está sendo usado pela ESA em seus ambientes de teste e operação, (Adamson, 2010).

De acordo com (ECSS, 2008 c), o padrão ECSS-E-ST-70-32C define um procedimento como um “meio de interagir com o sistema espacial de forma a se atingir um determinado objetivo ou conjunto de objetivos”. Por “meio de interagir com o sistema” entenda-se, na prática, um conjunto de comandos, em qualquer nível de abstração. Também é possível, inclusive, inserir procedimentos dentro de outro procedimento.

Procedimentos escritos nesta linguagem também podem ser usados para encapsular todo conhecimento operacional, que na maioria das vezes é complexo, (Koller, 2010). A linguagem PLUTO tem sintaxe similar a linguagens de programação como *Basic* e *Pascal*, e permite que se descreva o comportamento de um sistema espacial, esteja ele sob teste ou em operação, durante a execução de procedimentos. A estrutura de um procedimento PLUTO é apresentada na Figura 1 (ECSS, 2008c), e descrita a seguir.

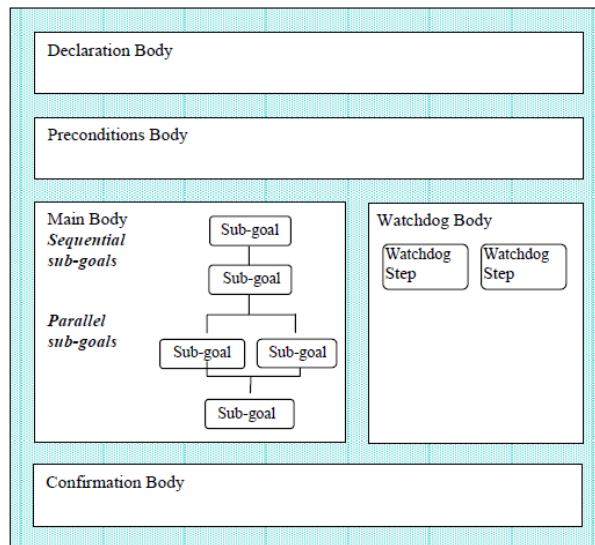


Figura 1 – Estrutura de um Procedimento PLUTO.

Um “*Declaration Body*”, opcional, descreve quais eventos podem ocorrer durante a execução deste procedimento. Um “*Preconditions Body*”, opcional, descreve quais condições devem ser atendidas pelo sistema antes que o procedimento seja executado. Por exemplo, pode-se deixar explícito que um equipamento esteja ligado antes de se iniciar a execução do procedimento. O “*Main Body*” é a única parte da estrutura de um procedimento não-opcional. É nela que os passos que compõem o procedimento são descritos. Os passos podem ser: um comando, uma ação do operador (ex: “regule manualmente a tensão da fonte”), ou mesmo um outro procedimento (com seus próprios “*declaration body*”, “*preconditions body*”, “*main body*”, etc). Os passos, de um procedimento, podem ser executados de forma seqüencial ou paralela. O “*Confirmation Body*” contém as leituras a serem feitas no equipamento sob testes ou em operação, para determinar se todos os passos do “*Main Body*” foram executados corretamente. Finalmente, é no “*Watchdog Body*” que são descritas as medidas de contingência a serem tomadas caso algum problema seja detectado durante a execução dos passos do “*Main Body*”.

3 A infraestrutura necessária para implementar a linguagem PLUTO

O Grupo SUBORD está iniciando suas primeiras pesquisas sobre padronização e execução automática de procedimentos através da linguagem PLUTO. Pretende-se construir uma infraestrutura computacional para padronizar a criação e automatizar a execução dos procedimentos de teste durante o desenvolvimento de seus sistemas e equipamentos. O primeiro passo é fazer uma análise para definir o que é necessário para construir esta infraestrutura. Devem ser considerados três elementos fundamentais: o SUT, o modelo SSM e a linguagem PLUTO. Neste trabalho o computador de OBDH é o SUT.

Durante a execução, um procedimento PLUTO interage com o SUT através de um modelo do mesmo. O padrão ECSS-E-ST-70-31C: *Ground systems and operations – Monitoring and control data definition* (ECSS, 2008 b) especifica o modelo de um sistema espacial, através do conceito de um *Space System Model* (SSM). Além disso, este padrão também especifica quais os dados, e seus diversos tipos, devem compor este modelo.

O SSM é um modelo hierárquico de todo o sistema espacial, podendo compreender os segmentos solo, espacial, e todos os equipamentos e subsistemas que o compõe, incluindo aí a infra-estrutura de testes e de lançamento. Em função do grande número de dados e elementos existentes em uma missão espacial, a norma define o conceito de “*domain-specific-view*”, ou seja, visão específica de domínio.

Uma visão específica de domínio é composta por uma parte do SSM, necessária para determinada aplicação. Durante um teste de unidade aplicado a um *payload* do satélite, por exemplo, apenas o modelo deste *payload*, e talvez dos subsistemas com os quais ele se comunica, seria necessário, e não o modelo de todo o sistema espacial.

O SSM é decomposto em elementos de sistema, em atividades que podem ser executadas nestes elementos, em dados de relato que refletem o estado destes elementos e os eventos que podem ser levantados e manipulados para o controle destes elementos, das atividades ou dos dados de relato. A Figura 2 (ECSS, 2008b), ilustra como são estruturados os itens de um SSM.

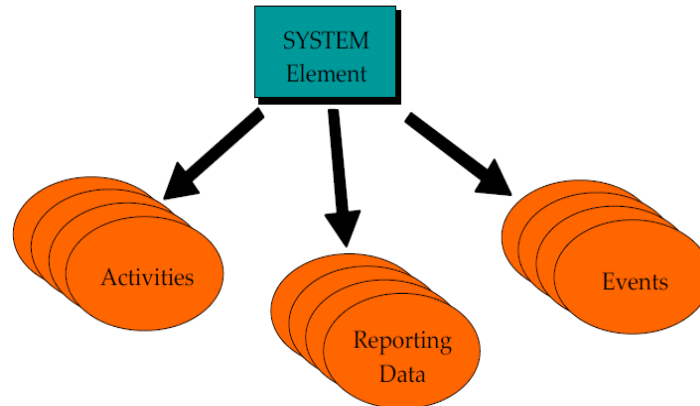


Figura 2 – Estrutura dos itens de um modelo SSM.

Uma atividade tem como função o controle e monitoramento do sistema espacial e pode ser definida como um telecomando ou telemetria, um procedimento, de teste ou operação, ou qualquer outro tipo de comando ou conjunto de comandos que seja específico para uma dada implementação do sistema.

Os dados de relato são informações que um elemento de sistema fornece. Dados de relato podem compreender medidas que refletem o estado do elemento associado ou saída de um produto cujo objetivo é ser utilizado por outro elemento de sistema.

Um evento é uma ocorrência de uma condição ou grupo de condições de significância operacional. Eventos são amplamente utilizados no sistema espacial para acionar a execução de funções, como por exemplo, a aquisição de sinais, que pode iniciar a tarefa de geração de uma telemetria a ser enviada à estação de solo. Usuários também podem definir “eventos específicos da missão” e associá-los a um elemento de sistema para uso em procedimentos ou operações de voo.

O SSM se comunica de um lado com os procedimentos e do outro com o sistema real para atuar e receber relatos que representam o estado atual do sistema, como mostra a Figura 3.

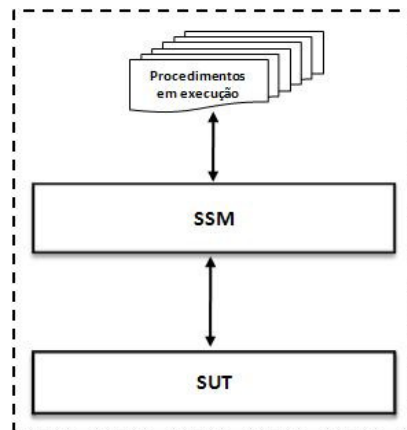


Figura 3 – Relação entre procedimentos PLUTO, o SSM e o SUT.

Durante a execução, um procedimento PLUTO pode ler ou escrever dados no SSM. Desta forma, a criação e execução dos procedimentos de teste através desta linguagem devem fazer referência ao SSM.

4 Arquitetura para execução de procedimentos de teste em sistemas espaciais

Para definição da arquitetura, deve ser necessário fazer uma análise para definir os principais elementos que devem estar presentes para que o sistema seja desenvolvido. Os principais passos para o desenvolvimento deste sistema são:

1. Definição da arquitetura do sistema;
2. Criação de uma interface para auxiliar na criação do SSM;
3. Criação de um SSM (visão específica de domínio) e definição das interfaces de comunicação entre o SSM e os procedimentos PLUTO e vice-versa;
4. Criação de uma *Integrated Development Environment* (IDE) para a criação de procedimentos PLUTO, e
5. Criação de um interpretador e executor dos procedimentos.

Atualmente, está em desenvolvimento a primeira e terceira etapas. A arquitetura do sistema já se encontra em sua versão preliminar, como mostra a Figura 4.

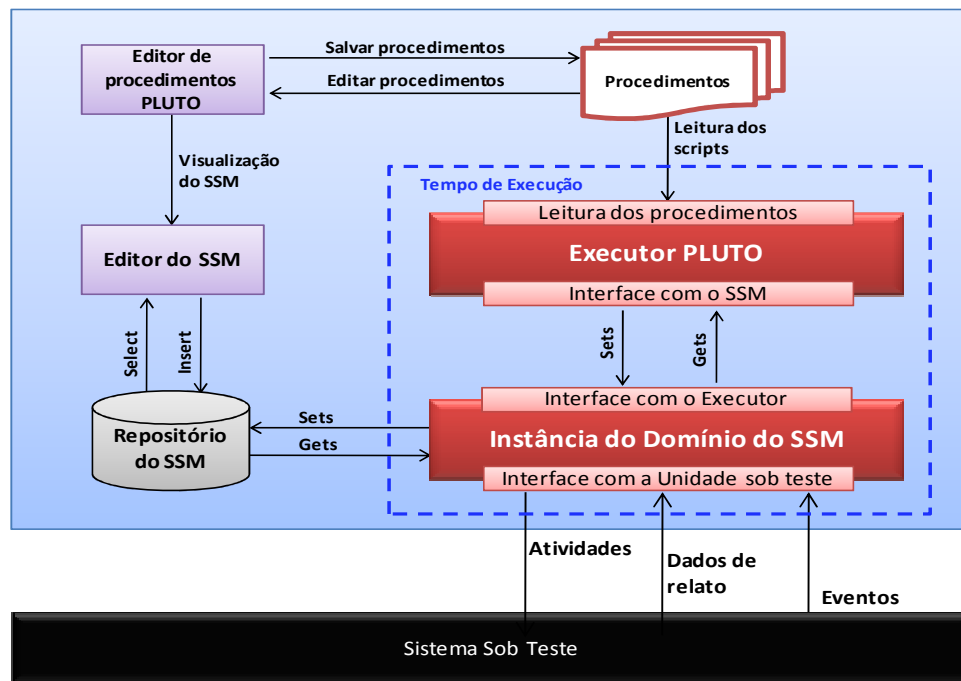


Figura 4 – Arquitetura do ambiente de execução automática de procedimento de teste.

A descrição de cada elemento presente na arquitetura é feita logo abaixo:

- **Editor do SSM:** interface usada para a criação e edição do modelo do SUT. Esta também permitirá a visualização do modelo e seus componentes durante a descrição dos procedimentos e visualização dos resultados provenientes da execução dos testes.
- **Editor de procedimentos PLUTO:** interface que será usada para criação e edição dos procedimentos de teste na linguagem PLUTO. Tem como objetivo verificar a sintaxe da linguagem.

- Repositório do SSM: banco de dados relacional que será usado para armazenar os resultados dos testes e manter seu histórico.
- Executor PLUTO: um dos elementos essenciais da arquitetura. A interpretação e execução dos *scripts* de teste serão feitas pelo Executor PLUTO. Este deve conter interfaces bem definidas para comunicação com a “Instância do Domínio”.
- Instância do Domínio do SSM: visão específica de domínio instanciada para a execução dos procedimentos de teste. Durante a execução dos procedimentos, o modelo instanciado deverá se comunicar de um lado com o executor da linguagem e do outro com o SUT. Por isso a necessidade de definição de interfaces de comunicação entre estes elementos.
- Procedimentos: arquivos salvos com os procedimentos escritos na linguagem PLUTO. Serão usados pelo interpretador da linguagem ao dar início à execução dos testes.
- Sistema sob Teste: o sistema sob teste poderá ser o sistema real ou um simulador.

As vantagens da arquitetura proposta constam de facilidades para: Criar e editar um SSM; Extrair do SSM visões específicas de domínio; Criar e editar procedimentos PLUTO; Interpretar e executar os procedimentos de teste automaticamente, e; Monitorar os resultados provenientes da aplicação dos testes.

Durante a preparação, o usuário especialista no domínio deve criar a visão específica de domínio através do Editor do SSM. Os procedimentos em linguagem PLUTO devem ser criados através do Editor de procedimentos PLUTO. Como um procedimento PLUTO deve fazer referência a um item do SSM, o Editor do SSM poderá ser utilizado como auxílio à edição dos procedimentos.

Uma sessão de testes deve ser iniciada para a execução de um ou mais procedimentos. O ambiente deverá fazer a leitura dos procedimentos gravados em arquivos, carregá-los em memória e, durante a execução, o Executor PLUTO fará a comunicação direta com a Instância do Domínio, também carregada em memória, para ler ou escrever dados. Ao término da execução de cada procedimento ou conjunto de procedimentos, uma sessão de teste poderá ser gravada em banco de dados. Portanto, será possível manter um histórico da execução dos procedimentos de teste.

5 Implementação inicial do sistema

O início da implementação do sistema será através dos elementos mais importantes para execução dos procedimentos de teste. Os elementos mais importantes são: Instância do Domínio do SSM, Executor PLUTO e as interfaces de comunicação entre eles e o SUT. Para o desenvolvimento do Executor PLUTO, os principais comandos da linguagem serão selecionados de modo a atenderem aos casos de teste que serão definidos para o OBDH. Os casos de teste serão definidos como parte deste trabalho. O conjunto de comandos selecionado será implementado na forma de um emulador, que será capaz de interpretar os procedimentos pré-definidos na sintaxe da linguagem. O mesmo será feito para o desenvolvimento SSM, ou seja, um subsistema ou módulo do OBDH do SIA será escolhido para ser modelado como uma visão específica de domínio de um SSM. Sua modelagem poderá ser representada através de arquivo(s) *Extensible Markup Language* (XML). Arquivos XML possuem a mesma estrutura hierárquica de um SSM. Estrutura que é definida pelo padrão do ECSS. Pretende-se também utilizar alguns recursos do EGSE do OBDH (hardware ou software) como apoio a execução este trabalho.

6 Implementação inicial do sistema

O grupo SUBORD está desenvolvendo um computador de *On-Board Data Handling* (OBDH) para ser embarcado em futuros satélites do INPE. Para que este computador seja verificado e validado, um *Electrical Ground Support Equipment* (EGSE) está sendo desenvolvido em paralelo para auxiliar tanto no desenvolvimento quanto na aplicação de seus testes funcionais. Como parte deste equipamento, encontra-se o Software de Monitoramento e Controle Remoto (SMC), responsável pelo gerenciamento e execução dos procedimentos de teste do OBDH. Pretende-se integrar o ambiente de teste proposto neste trabalho ao SMC. A Figura 5 ilustra uma das telas do SMC.

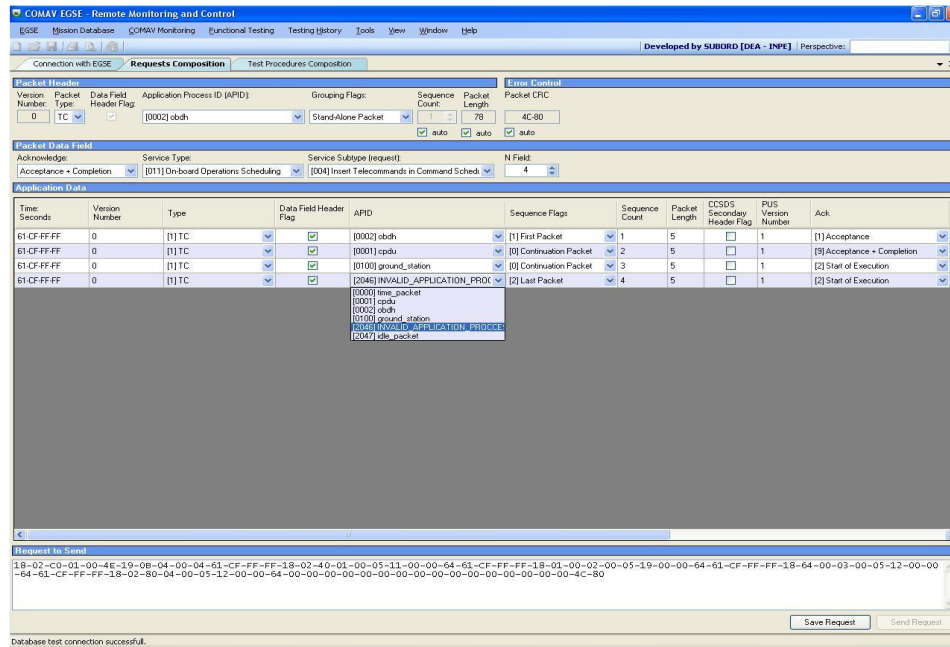


Figura 5 – Software de Monitoramento e Controle Remoto (SMC).

A idéia de integrar a uma ferramenta de teste do OBDH é permitir que os testes unitários sejam reutilizados nos procedimentos de teste para aceitação do computador de OBDH, na integração do satélite e, em menor grau, na operação do mesmo. Acredita-se que procedimentos de testes descritos durante a fase de desenvolvimento possam ao menos servir como *templates* para procedimentos operacionais, se estiverem expressos em uma mesma linguagem.

7 Perspectivas e Considerações Finais

Um ambiente de execução automática de procedimentos de teste que faz uso de uma linguagem de *scripts* tem potencial para reduzir o tempo de planejamento, criação e execução dos diferentes procedimentos, nas diferentes fases de desenvolvimento de um satélite. Além disso, permitirá a redução dos custos destas atividades, que atualmente são muito altos.

Atualmente, o grupo SUBORD carece de recursos para execução automática de procedimentos de teste através de *scripts*. Por esta razão, a linguagem de procedimentos PLUTO contribuirá nas atividades de verificação e validação durante o desenvolvimento dos sistemas desenvolvidos pelo grupo. Além de ser possível utilizá-la durante as etapas de integração e testes dos futuros satélites do INPE.

Agradecimentos

Agradecemos à Financiadora de Estudos e Projetos (FINEP) pelo apoio financeiro e aos membros do Grupo de Supervisão de Bordo (SUBORD) da Divisão de Eletrônica Aeroespacial (DEA) do INPE.

8 Referências

Adamson, K., Bargellini, P., Nogueira, T., Nett, H., Caspar, C., ADM-AEOLUS: Mission Planning Re-Use, Autonomy and Automation, SpaceOps Conference, 25-30 April, Huntsville – Alabama, EUA, 2010.

Engel, A., Verification, Validation, and Testing of Engineered Systems, Library of Congress Cataloging-in-Publication, New Jersey, 2010.

European Committee for Space Standardization (ECSS), “Document Tree and Status (ST, HB, TM)”, ECSS-DOC-ST-001/ES, 20 October, Noordwijk, Netherlands, 2008 a.

European Committee for Space Standardization (ECSS), “Space Engineering: Ground Systems and Operations”, ECSS-E-ST-70-31C, 31 July, Noordwijk, Netherlands, 2008 b.

European Committee for Space Standardization (ECSS), “Space Engineering: Test and Operations Procedure Language”, ECSS-E-ST-70-32C, 31 July, Noordwijk, Netherlands, 2008 c.

Koller, M., Reggestad, V., Adamson, K., Kay, R., ESOC Earth Observation Missions and the Automation of Operational Routine Tasks, SpaceOps Conference, 25-30 April, Huntsville – Alabama, EUA, 2010.