

# UMA ARQUITETURA PARA SUBSISTEMAS DE COMPUTAÇÃO DE BORDO DE UM SIMULADOR DE SATÉLITES

Joaquim Pedro Barreto<sup>1</sup> e Ana Maria Ambrósio<sup>2</sup>

Instituto Nacional de Pesquisas Espaciais,  
Coordenação de Engenharia e Tecnologia Espacial,  
Curso de Sistemas Espaciais (CSE), São José dos Campos, SP, 12227-010  
<sup>1</sup>joaquim@dss.inpe.br e <sup>2</sup>ana@dss.inpe.br

**Resumo:** Um simulador de satélites é utilizado em diferentes fases do ciclo de vida da missão nas atividades de integração e testes do modelo de engenharia, do sistema de controle, avaliação de contingências e treinamento de operadores. A prática do seu reuso leva à redução de custos, prazos e complexidade, aliado ao aumento na confiabilidade. Todavia, para que haja uma efetiva reutilização dos simuladores, inclusive entre missões, é necessário que o projeto e desenvolvimento dos mesmos incluam flexibilidade. Uma solução para mitigação de custos, tempo e riscos é o uso de padrões para projeto e desenvolvimento. O SMP (Simulation Model Portability) foi proposto pela ESA (European Space Agency) como um padrão visando a uniformidade no desenvolvimento de simuladores. O padrão inclui instanciação e controle da simulação, serviços mandatórios e modelos que implementam as funcionalidades dos subsistemas do satélite, através de um conjunto de interfaces padronizadas para comunicação entre estes elementos. Este artigo apresenta a proposta de uma arquitetura baseada na interface de modelos do SMP a ser aplicada em subsistema de computador de bordo de um satélite (OBDH – On Board Data Handling). Os modelos para cada serviço do OBDH deverão ser implementados como componentes, para que se possa verificar a flexibilidade da arquitetura quanto à inserção e/ou retirada de componentes sem que isto exija alterações na estrutura geral do simulador, potencializando o reuso e a interoperabilidade.

**Palavras-chaves:** simulador de satélite, computador de bordo, interoperabilidade, reuso

## 1 Introdução

Um simulador de satélite é um software ou um sistema de software e hardware que substitui o satélite físico real e pode ser utilizado em diferentes fases do ciclo de vida de uma missão (EICKHOFF, 2009). Considerando-se que toda missão espacial requer a construção de simuladores de satélite, a questão da reutilização como uma forma de reduzir os custos da missão é de fundamental importância. A reutilização de simuladores de satélite pode se dar em duas vertentes diferentes. Na primeira, o simulador é reutilizado dentro de uma mesma missão, em várias fases de verificação e validação da missão. Na segunda, a reutilização se dá entre missões. Ambas as situações consideram que de um satélite para outro há vários pontos em comum (REGGESTAD et al, 2004). A abordagem de desenvolvimento de um simulador de satélites, a forma como seus elementos (subsistemas representados por modelos) interagem, o grau de fidelidade na representação, inclusão ou não de hardware “in the loop”, a definição de um acoplamento perfeito entre os elementos a serem simulados, bem como a possibilidade de inserção e retirada destes elementos a qualquer momento da simulação, de forma prática e rápida, são itens merecedores de um estudo aprofundado para uma melhor reutilização. Com o objetivo de padronizar o desenvolvimento de simuladores de satélites, a Agência Espacial Européia (ESA) propôs o padrão SMP (Simulation Model Portability), atualmente na versão 2. Este padrão utiliza técnicas e mecanismos para elaboração de uma arquitetura de software baseada em componentes, visando a independência de plataforma (portabilidade), a interoperabilidade e o reuso dos elementos do simulador. (ESOC, 2005). A modelagem do simulador é estruturada em um conjunto de componentes, padronizando as interfaces entre os mesmos e fortalecendo, desta forma, a interoperabilidade e o reuso dos componentes. Um dos subsistemas mais complexos do simulador de satélites é o subsistema para manipulação de dados em bordo (On-Board Data Handling - OBDH). Em termos de simulação, o computador de bordo pode estar no simulador tal qual estaria no satélite (hardware in-the-loop), pode ter o hardware emulado e ter o mesmo software de bordo, ou ser um software que implementa o conjunto completo ou parcial das funções do software e do hardware do OBDH (ARIAS et al, 2008).

O padrão SMP2 apresenta características necessárias para uma decomposição do OBDH em componentes reusáveis. Todavia, o mesmo apresenta uma arquitetura genérica para um modelo, sem, no entanto, especificar uma decomposição deste modelo. Uma decomposição para modelos específicos e complexos como o OBDH, que contemple as especificidades de reuso e fidelidade, bem como a comunicação entre os modelos e o ambiente de simulação, pode proporcionar grande versatilidade para simuladores.

O trabalho proposto visa a definição de uma arquitetura de software para o subsistema de computação de bordo de um simulador de satélite. Nesta arquitetura, o subsistema OBDH será baseado em componentes de software e deve obedecer as definições do padrão SMP2. Uma abordagem baseada neste padrão poderá permitir a

reutilização dos componentes gerados para diversas instâncias do subsistema em diferentes usos de um simulador durante uma missão, ou em diferentes missões. A divisão do OBDH em componentes deve tomar por base os serviços implementados pelo mesmo. Neste caso, duas situações são admissíveis: que os serviços do OBDH sejam específicos daquele satélite ou que sejam baseados em um padrão como o PUS (Packet Utilisation Standard)(CCSDS, 2010). No segundo caso ter-se-á mais probabilidade de reuso, porém a arquitetura a ser proposta deve abranger ambas as possibilidades.

## 2 Trabalhos relacionados

Como parte deste trabalho, foi feito um levantamento dos diversos simuladores de satélite no âmbito da ESA, foram analisados simuladores de missões como Galileo, LISA Pathfinder, XMM, etc. Observa-se nos trabalhos (HOMEM et al., 2006; CÔME et al., 1998; DELHAISE et al., 2006) uma confirmação da preocupação da ESA com o aspecto do reuso e padronização. Em (ESA, 2010) foi inserido um núcleo de simulação com modelos e serviços comuns aos simuladores chamado SIMSAT (LINDMAN et al, 2006). A mais recente versão deste núcleo contempla o padrão SMP2, fazendo, desta forma, com que todos os simuladores de satélite a serem desenvolvidos já sejam projetados e desenvolvidos obedecendo ao conjunto padronizado de interfaces entre modelos, serviços e ambiente de simulação.

As principais aplicações dadas aos simuladores estudados referem-se a treinamento de operadores, validação do segmento de controle da missão e validação do software de bordo antes de ser carregado a bordo. Há ainda outras aplicações específicas, para as quais também se desenvolveu simuladores, como segmento solo (análise de disponibilidade e carga).

Este estudo comprovou a importância da simulação em diferentes fases do ciclo de vida da missão, reforçando ainda a necessidade de reuso do simulador como elemento de redução de custos e prazos.

Também foi feito um estudo tendo por base o simulador de satélite SIMC3 em desenvolvimento na coordenação de Engenharia e Tecnologia Espaciais (ETE) quanto à similaridade entre sua arquitetura e a arquitetura do padrão SMP2, bem como os serviços oferecido por ambos (BARRETO et al., 2010).

## 3 Principais áreas aplicáveis

A seguir são detalhados os padrões, processos e tecnologias relacionados ao trabalho proposto.

### 3.1 Simulation Model Portability (SMP)

O SMP é um padrão proposto pela ESA visando promover a portabilidade e reuso de modelos entre aplicações e sistemas operacionais. Em 25 de Janeiro de 2011 o SMP foi publicado como um Memorando Técnico ECSS, dividido em 5 documentos, sob o código ECSS-E-TM-40-07. Vários trabalhos apresentados mostram a evolução da aplicação do mesmo no desenvolvimento de simuladores no âmbito da ESA (VRIES et al., 2006; SEBASTIÃO et al, 2008; FRITZEN, 2004). O referido memorando técnico engloba o padrão SMP2 e a norma ECSS-E-ST-40C, que define gerenciamento, engenharia e garantia do produto para aplicações e projetos espaciais

A arquitetura do SMP é decomposta em três camadas, sendo que a primeira camada representa o mundo real a ser modelado (reality), a segunda camada apresenta a especificação dos modelos através de uma plataforma independente (PIM - Platform Independent Model), não levando em conta como serão desenvolvidos e executados, e a terceira camada apresenta os objetos já em plataforma específica (PSM - Platform Specific Model) para desenvolvimento e execução.

A Figura 1 apresenta esta visão de alto nível do SMP. Na terceira camada, a sigla SA refere-se a painel solar (do inglês *Solar Array*) e CSA refere-se a painel solar concreto (do inglês *Concrete Solar Array*)

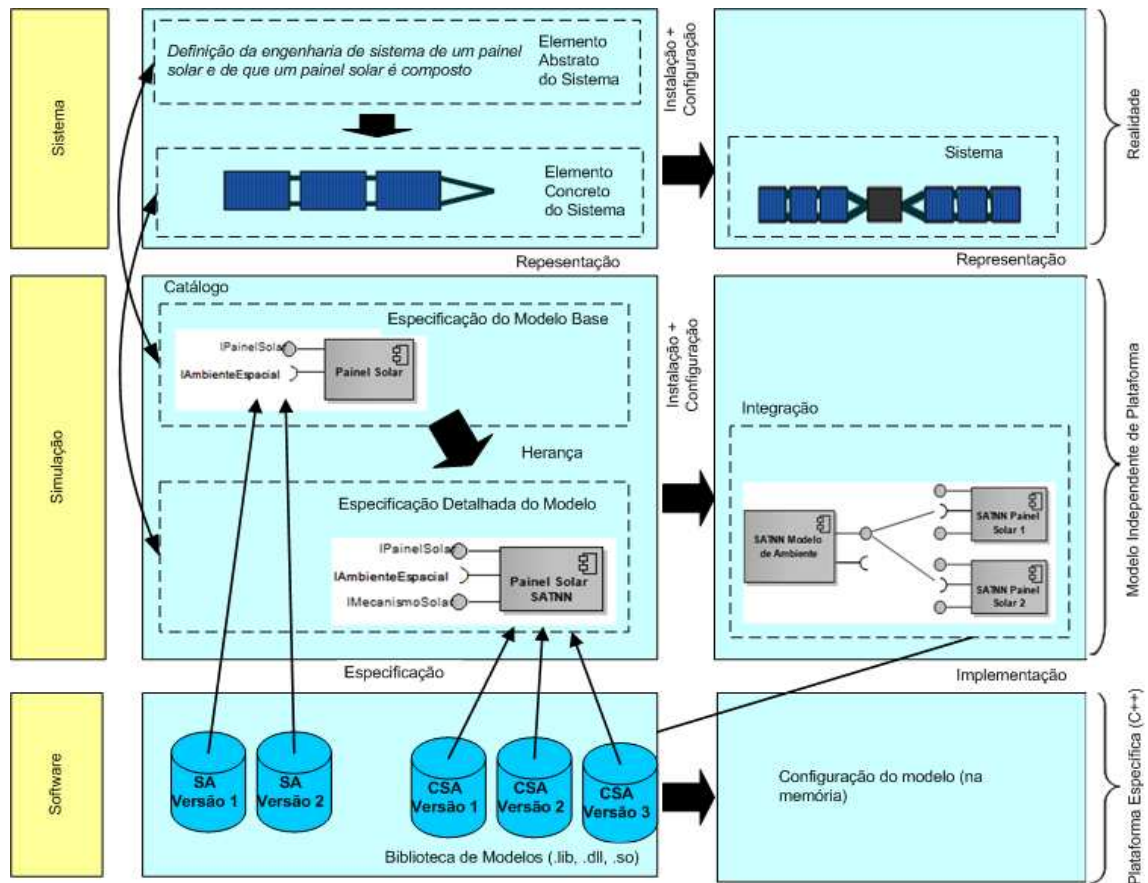


Figura 1 - Arquitetura do padrão SMP FONTE: Adaptado (ESOC, 2005)

Esta figura também apresenta duas colunas sendo que a primeira corresponde às definições dos artefatos e a segunda a instâncias dos mesmos, se tomarmos como analogia a Orientação a Objetos. O objetivo principal do SMP é a promoção da independência de plataforma para os modelos. Esta independência é obtida pela definição de um conjunto de modelos, independentes de plataforma, que depois possam ser mapeados para uma plataforma específica.

Outros fundamentos usados no SMP que reforçam o reuso e a portabilidade são: (i) a separação entre projeto e execução (demonstrado nas colunas da figura anterior); (ii) configuração dinâmica; (iii) extensivo uso de interfaces, componentes e herança (recurso de orientação a objeto).

A arquitetura típica de um simulador baseado no padrão SMP possui uma camada para os modelos do satélite que serão simulados, uma camada de serviços obrigatórios e uma camada de controle da simulação (ambiente de simulação). A Figura 2 mostra esta arquitetura.

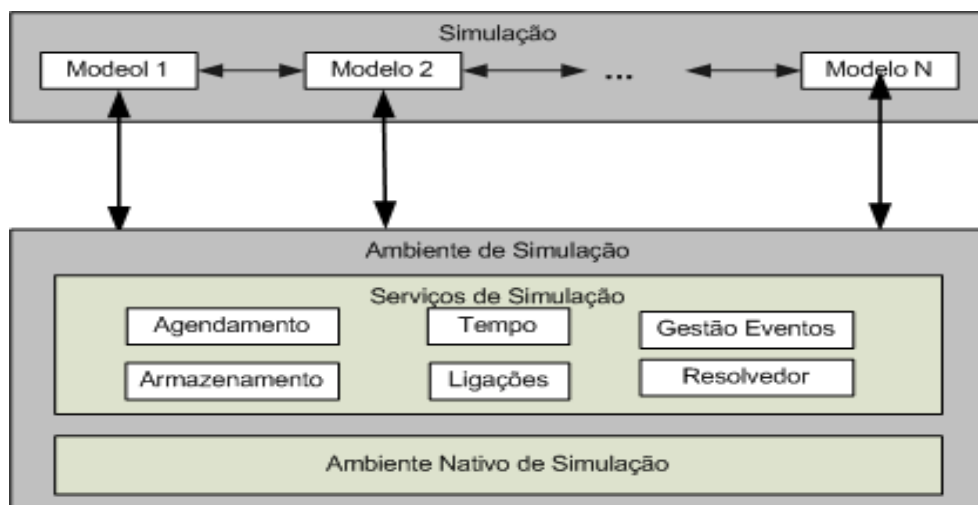


Figura 2 - Arquitetura típica de um simulador SMP FONTE: Adaptado (ESOC, 2005)

O núcleo da simulação é representado pelos modelos. Os Serviços de Simulação fazem parte da camada Ambiente de Simulação, provendo facilidades para os modelos. Ainda nesta camada de Ambiente, eventuais recursos nativos de simulação já existentes podem ser reaproveitados, através de um encapsulamento dos mesmos, para que se adaptem ao padrão de interfaces dos componentes SMP

O SMP baseia sua arquitetura em componentes. Tanto modelos como serviços são vistos como componentes, o que permite um tratamento igualitário entre eles. Todas as conexões entre componentes e entre serviços e componentes são realizadas através de um conjunto predefinido de interfaces. Todo componente pode abrigar uma árvore hierárquica abaixo de si, através de composições. A interface de componente provê métodos para navegação através da hierarquia construída, o que permite recuperar todas as informações dos componentes das composições. A Figura 3 mostra o diagrama de componentes.

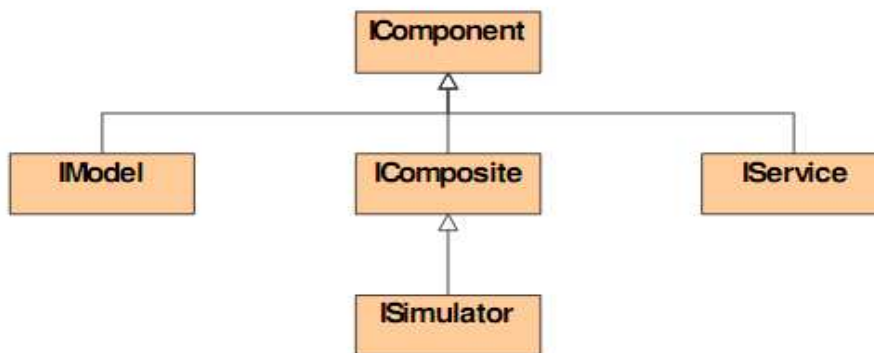


Figura 3 - Diagrama hierárquico de componentes do SMP FONTE: Adaptado(ESOC, 2005)

Os modelos (interface IModel) e serviços (interface IService) se equivalem, o mesmo ocorrendo com as composições (interface IComposite), que são agrupamentos de modelos. A instância do simulador (interface ISimulator) não possui um componente pai, sendo ela a raiz da árvore. Este padrão define quatro serviços como obrigatórios e outros opcionais. Também é possível a criação de serviços pelo usuário, desde que respeitando a interface padrão para serviços. Os serviços mandatórios são:

- Tempo: provê quatro tipos de tempo para a simulação - tempo da simulação, tempo da missão, tempo da época (epoch time) e tempo Zulu (computador)
- Eventos: provê um mecanismo para disparo de eventos globais assíncronos
- Armazenamento: grava todas as mensagens geradas por modelos ou serviços
- Agendamento: executa eventos de forma cíclica ou em intervalos de tempos específicos.
- Referências: permite resolver a referência entre componentes, seja usando caminho relativo ou absoluto
- Registro de vínculos: mantém a lista de relacionamentos entre todos os componentes, permitindo avaliar se a exclusão de determinado componente da simulação é segura.

A criação do ambiente de simulação, com os componentes, seus relacionamentos e escalonamentos para execução pode ser feita de duas formas: estática e dinâmica.

Na criação estática, todos os passos para criação do ambiente, até o momento em que o simulador esteja pronto para execução, são feitos com base em código. Ou seja, qualquer modificação que venha a ser feita posteriormente implicará em modificação no código de criação e recompilação.

A criação dinâmica permite que os passos para a criação do ambiente de simulação sejam definidos em arquivos externos, escritos em uma linguagem própria, baseada em XML. Desta forma, os passos da criação podem ser modificados em tempo de execução. Com isso, fica facilitada a tarefa de substituir, por exemplo, um dado componente, em tempo de execução. O SMP usa uma linguagem de definição de modelos para simulação (SMDL - Simulation Model Definition Language) que permite criar catálogos de modelos, relações de dependências entre modelos, escalonamentos, integração, empacotamento e documentação.

### 3.2 Arquitetura Orientada a Modelos (MDA)

A arquitetura orientada a modelos é um processo de desenvolvimento (OMG, 2003) criado em 2001 pelo Object Management Group (OMG) que propõe uma solução que separa os detalhes de sua implementação. A idéia central é a definição de um conjunto de modelos independentes de plataforma. A UML (Unified Modeling Language) é usada para descrever os modelos. Para representar um conjunto de modelos é necessária uma abstração ainda superior à modelagem, que é a metamodelagem, já que um metamodelo é um modelo de um modelo. A arquitetura MDA usa uma metalinguagem definida como Meta Object Facility (MOF). Esta família de metamodelos define um repositório de artefatos. Também é usado o Common Warehouse Metamodel para definição de metamodelos.

Uma vez criada a família de metamodelos independentes, um conjunto de regras de transformação é usado para a conversão em uma plataforma específica. A Figura 4 mostra a arquitetura MDA, sendo que o anel externo mostra as plataformas para as quais o núcleo central de modelos pode ser convertido.

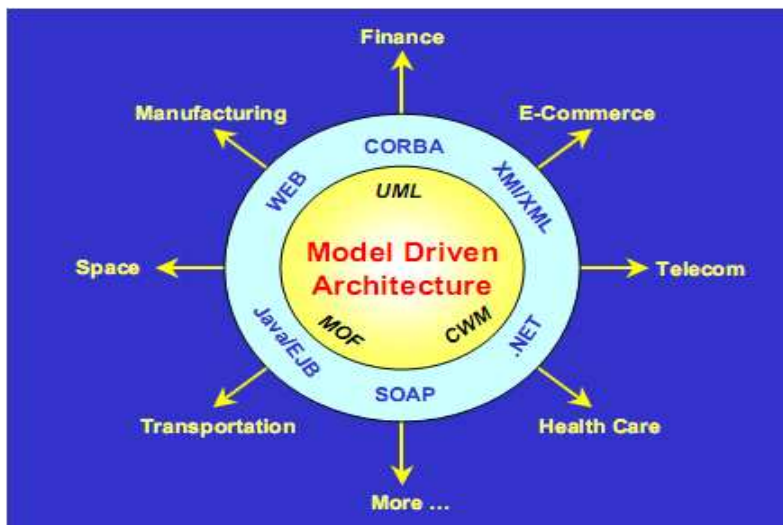


Figura 4 - Arquitetura MDA FONTE: (OMG, 2003)

Este direcionamento a modelos guia todo o processo de entendimento, projeto, desenvolvimento, distribuição, operação, manutenção e modificação dentro da MDA.

O padrão SMP trata o conceito de independência de plataforma de maneira similar ao MDA. No SMP, é definido um catálogo de modelos independente de plataforma, que é mapeado posteriormente para uma plataforma específica.

### 3.3 Packet Utilisation Standard (PUS)

O PUS (ECSS, 2003) define um padrão de comunicação entre os segmentos solo e bordo, que é composto por um conjunto de serviços a serem implementados pelo computador de bordo. Estes serviços são então acessados pelo segmento solo através de um conjunto predeterminado de chamadas, que por sua vez acionam os processos internos do OBDH (em bordo) responsáveis pelos serviços. Os dados obtidos são então retornados através de estruturas de dados igualmente padronizadas.

O padrão PUS (ECSS, 2003) define uma interface de comunicação solo-bordo, visando atender requisitos de interface elétrica, testes e operações de voo. Esta interface é definida através de um conjunto padrão de serviços cuja implementação é definida na fase de projeto, quando então os serviços são customizados de acordo com os requisitos da missão. A implementação destes serviços pelo computador de bordo requer uma infra estrutura em solo para monitoração e controle dos dados. O PUS possui um amplo leque de serviços que dificilmente será totalmente implementado pelo OBDH, devendo, portanto, ser adaptado de forma específica para uma dada missão, dependendo de suas necessidades.

### 3.4 Computadores de bordo (OBDH)

Computadores a bordo de satélites são responsáveis pelo gerenciamento das operações em bordo. Entre as funções desempenhadas por um OBDH (On Board Data Handling) estão: (i) comunicação interna de dados, (ii) detecção e diagnóstico de falhas, (iii) processamento de dados em bordo, (iv) controle e armazenamento de dados, (v) obtenção, formatação, armazenamento e envio das telemetrias dos subsistemas e da carga útil, (vi) recepção, decodificação, processamento e distribuição dos telecomandos para os subsistemas e cargas úteis.

As versões de um OBDH podem variar desde um modelo básico responsável apenas por telemetria e telecomando até um modelo mais complexo capaz de capturar e analisar erros, interagir com o sistema de controle de atitude e órbita e gerar relatórios detalhados sobre as operações e procedimentos executados em bordo.

## 4 Resultados Esperados

Com os resultados desta pesquisa espera-se elaborar uma arquitetura para um subsistema OBDH de simuladores de satélites baseada no padrão SMP. Esta arquitetura deve ser focada em modelos especializados a partir do modelo genérico do SMP e implementados posteriormente como componentes.



Cada modelo deve representar um serviço típico implementado pelo computador de bordo. Para o PUS, a lista de serviços padrões já está documentada. Já para satélites que não usem este padrão, os serviços deverão ser derivados das funcionalidades do OBDH.

A arquitetura deverá permitir que um OBDH seja estruturado a partir da interligação de um grupo de componentes (implementados a partir dos componentes). Deverá, ainda, permitir que componentes sejam inseridos ou retirados na arquitetura (aumento ou decréscimo de funcionalidades) ou então que componentes possam ser substituídos por versões melhoradas (aumento da fidelidade).

A Figura 5 mostra um diagrama de contexto para a arquitetura, relacionando serviços, OBDH e simuladores.

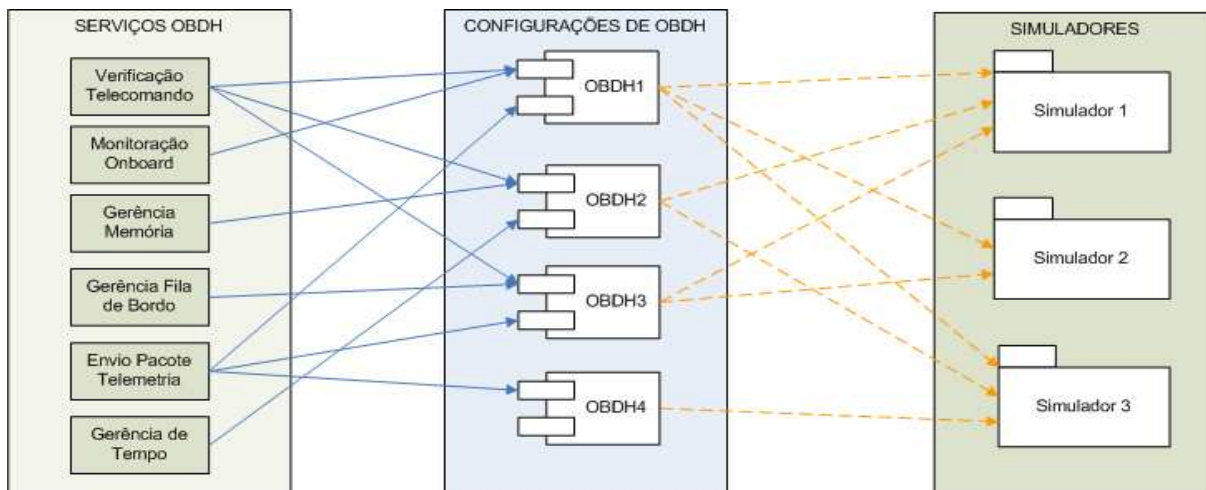


Figura 5 - Diagrama de contexto de utilização

A primeira coluna representa os componentes que serão gerados a partir dos modelos, para cada serviço do OBDH. A coluna de configurações indica que diferentes OBDHs poderão ser construídos a partir da composição de serviços que o mesmo deverá fornecer, ou seja, pode-se ter computadores de bordo simples ou complexos. Já a última coluna indica que um OBDH pode ser utilizado em diversos simuladores de satélites, dependendo dos requisitos. Portanto, a padronização das interfaces dos componentes e suas interligações com as camadas de controle da simulação e de serviços de simulação é que dotarão a arquitetura da flexibilidade almejada.

Espera-se que o aperfeiçoamento do modelo genérico previsto no SMP para uma arquitetura detalhada aplicada aos serviços de um OBDH possa, eventualmente, ser estendida a outros subsistemas de um satélite.

O trabalho também pretende trazer para o INPE elementos que possam auxiliar nos estudos e projetos futuros envolvendo simuladores de satélites. Uma vez que o SMP vem se consolidando cada vez mais no ambiente da ESA (FRITZEN, 2010), que produz simuladores de forma distribuída, através de diversas empresas contratadas, com fortes esforços no sentido de implantação de padrões (WALSH et al, 2010), entende-se que também o INPE pode também se beneficiar da aplicação destes conhecimentos.

## 5 Conclusão

Considerando-se a importância de simuladores no ciclo de vida de desenvolvimento e operação de satélites, seja para análise de contingências, para apoio aos testes em fases preliminares de integração do satélite ou no treinamento de operadores, um estudo voltado para o aprimoramento de técnicas, metodologias ou padrões no contexto de simuladores pode contribuir para a melhoria dos projetos de simuladores.

O padrão SMP proposto pela ESA vem sendo implantado no núcleo SIMSAT, de simulação de satélites, daquela agência. Empresas contratadas para desenvolver simuladores para esta agência vem adotando o padrão SMP. Portanto, trata-se de um padrão a ser estudado e explorado, na medida em que sua utilização vem ganhando cada vez mais destaque.

O ganho a ser percebido pelo estudo é a possibilidade de se implementar uma arquitetura baseada em modelos, dentro da filosofia proposta pelo padrão SMP, trazendo uma padronização de projeto. Além disso, uma arquitetura validada para um OBDH poderá ser adaptada para outros subsistemas de um satélite, tornando-a ainda mais genérica e reutilizável.

## 6 Referências

Arias, R.; Kucinskis, F. N., Alonso, J. D. D. Lessons Learned from an Onboard ECSS PUS Object-Oriented Implementation, Proceedings of the 9th Conference on Space Operations. Heidelberg, Alemanha, 2008

Barreto, J. P., Hoffmann, L. T., Ambrosio, A. M., Using SMP2 Standard in Operational and Analytical Simulators, Proceedings of the 10th Conference on Space Operations. Alabama, United States, 2010

CCSDS, The Consultative Committee for Space Data Systems, 2010, Disponível em <<http://public.ccsds.org/default.htm>>

CÔME, H.; IRVINE, M. The XMM Simulator - The Technical Challenges, ESA Bulletin, v 96, 1998.

ECSS, Simulation Model Portability 2, Disponível em: <[http://www.esa.int/TEC/Modelling\\_and\\_simulation/TEC2DCNWTPE\\_0.html](http://www.esa.int/TEC/Modelling_and_simulation/TEC2DCNWTPE_0.html)>

ECSS, Ground Systems and Operations - Telemetry and Telecommand Packet Utilisation, ECSS-E-70-41A, ESA Publications, 2003.

Eickhoff, J., Simulating Spacecraft Systems, Springer Aerospace Technology, 1st ed., 2009.

ESA, SIMSAT - Simulation Infrastructure for the Modeling of SATellites, Disponível em: <<http://www.egos.esa.int/portal/egos-web/products/Simulators/SIMSAT/>>

ESOC. European Space Operations Centre, SMP 2.0 Handbook, Issue 1, Revision 2, 2005, Disponível em: <[http://www.eurosim.nl/support/manual\\_4\\_1/html/](http://www.eurosim.nl/support/manual_4_1/html/)>

Fritzen, P.; Segneri, D.; Pignède, M., SWARMSIM – The first fully SMP2 based Simulator for ESOC, SESP Workshop, Noordwijk, The Netherlands, 2010

Fritzen, P., Model Re-use through the SMP2 Standard, Proceedings on Simulation of On-board Systems, The Royal Aeronautical Society, London, 2004

HOMEM, M. T; PIGNÉDE, M; MERRI, M.; REGGESTAD, V; PIDGEON, A.; MATUSSI, S., The GALILEO simulator: a major step in software technology from single spacecraft to constellation simulators, Proceedings of the 8th Conference on Space Operations, Rome, Italy, 2006.

Lindman, N., Di Nisio, N., Sebastião, N.: The Implementation of the Simulation Model Portability 2 Specification at ESOC, Proceedings of the 8th Conference on Space Operations, Rome, Italy, 2006  
OMG, MDA Guide Version 1.0.1, 2003. Disponível em: <<http://www.omg.org/mda/>>

Reggestad, V.; Guerrucci, D.; Emanuelli, P.P; Verrier, D., Simulator Development: the flexible approach applied to Operational Spacecraft Simulators, Proceedings of the 7th Conference on Space Operations, Montreal, Canada, 2004

Sebastião, N. et al, A Reference Architecture for Spacecraft Simulators, Proceedings of the 9th Conference on Space Operations, Heidelberg, Alemanha, 2008

Vries, R., Moelands, J., SMP2 Developments in EuroSim, Eurosim Workshop, 2006, The Netherlands

Walsh, A.; Wijnands, Q.; Lindman, N.; Ellsipsen, P.; Segneri, D.; Eisenmann, H.; Steinle, T, The Spacecraft Simulator Reference Architecture, SESP Workshop, Noordwijk, The Netherlands, 2010