

USO DE UM PADRÃO DE PORTABILIDADE PARA COMUNICAÇÃO DE SISTEMAS LEGADOS DE SIMULAÇÃO DE SATÉLITES

Miguel Adrian Carretero

INPE – Instituto Nacional de Pesquisas Espaciais
São José dos Campos – São Paulo, Brasil
adrian.carretero@gmail.com

Dra. Ana Maria Ambrósio

INPE – Instituto Nacional de Pesquisas Espaciais
São José dos Campos – São Paulo, Brasil
aambrosio27@gmail.com

Resumo: *A pesquisa em simulação vem permitindo uma constante evolução na área através do aumento de processamento dos computadores e da infra-estrutura de comunicação. Com o aumento da capacidade computacional e com o advento das novas tecnologias de software como a orientação a objetos, novos paradigmas para melhorar o desenvolvimento dos sistemas de simulação, sejam os seus ambientes ou os modelos simulados, tem surgido. A engenharia de software tem incentivado o re-uso ou o aproveitamento de programas computacionais. Na área de simulação, não apenas a construção de sistemas que permitirão o re-uso futuro é importante, mas também a utilização de sistemas desenvolvidos e validados em diferentes e antigas tecnologias de software devem ser agregadas. Estes últimos são chamados sistemas legados. O desenvolvimento dos modelos abstratos para um ambiente de simulação minimiza os impactos das mudanças, permite avaliar e garantir a sua qualidade. Este artigo apresenta uma proposta que aplica conceitos do Simulation Model Portability (SMP), um padrão da ESA para a integração de simuladores de satélites que incluem o uso de sistemas legados de simulação.*

Palavras-chave: *simulador de satélite, modelagem, sistemas legados, orientação a objeto, simulação paralela e distribuída*

1 Introdução

O uso de simulação em computadores é uma técnica que busca criar uma representação do mundo real (ambientes físicos, ações de pessoas e equipamentos físicos) em modelos computacionais, matemáticos, probabilísticos e estatísticos que possam ser testados e avaliados. Simulação é conveniente quando a experimentação com o sistema real é impossível ou não desejável, novos sistemas de alto custo e risco devem ser desenvolvidos, desejar realizar treinamento de pessoal e testar algum sistema utilizando uma escala de tempo maior ou menor (Gordon, 1969 e Shannon, 1975).

Um dos conceitos clássicos diz que a simulação é o processo de projetar um modelo de sistema real e conduzir experimentos com tal modelo com o propósito de entender o comportamento do sistema ou avaliar as várias estratégias, dentro dos limites impostos por um critério ou conjunto de critérios para a operação de um sistema (Shannon, 1975). Segundo Law (1991) um sistema é definido como a coleção de entidades (pessoas e veículos, por exemplo) que agem e interagem juntas para a realização de alguma tarefa. Na prática um sistema depende do objetivo de um estudo em particular (Marques e Abdurahiman, et al., 2001). Quando falamos de sistema ou sistemas em simulação estamos definindo uma agregação, união ou construção de vários objetos que possuem uma interação ou interdependência e que podem ser modelados (Gordon, 1969).

Um modelo é uma representação de um objeto, sistema ou idéia de alguma outra forma que não o da própria entidade (Gordon, 1969), os modelos podem ser confeccionados de várias formas como computacional, matemático, probabilístico e estatístico. Quando se fala de modelos algumas regras no desenvolvimento do modelo devem ser avaliadas como o grau de fidelidade, o nível de agregação e relevância em relação ao ambiente em que serão simulados. Definidos os modelos e caracterizados os procedimentos de simulação se torna necessário a criação de um ambiente de simulação que deve possuir os estados de execução da simulação. O ambiente de simulação deve ter como principal preocupação a necessidade de criar uma interface simples e intuitiva, para que o condutor do simulador tenha como principal a atenção os modelos simulados. A simulação é um objeto muito útil no aprendizado, reconhecimento e ensino, a qual permite fazer muitas experiências sem correr riscos ou pagar os custos de uma experiência real, pois tudo ocorre no mundo virtual e controlado.

Existem muitos ambientes de simulação disponíveis para as mais variadas áreas de conhecimento, uma revisão sobre simuladores de satélites comerciais e não comerciais pode ser obtida no trabalho do Turner (2003). Uma vasta gama de ambientes de simulação é caracterizada pelo desenvolvimento nos mais variados níveis de propriedade como computadores, sistemas operacionais, linguagens de programação, metodologias e empresas contratadas para o desenvolvimento dos simuladores e de modelos simulados. A mudança de um ambiente de simulação de sua plataforma atual para uma nova plataforma tecnológica pode ser parcialmente ou totalmente impraticável, pois o esforço gasto esta atualização poderá ser igual ou superior ao já gasto na versão atual em uso.

As complexidades dos sistemas computacionais adicionada a vários métodos e ferramentas de criação, geração e produção de software tornam a integração, a interoperabilidade, o re-uso e portabilidade uma tarefa árdua para um ambiente de simulação proprietária ou legado de outro desenvolvimento. Para não perder os gastos feitos no desenvolvimento de um ambiente de simulação a idéia é prover uma arquitetura comum, ou seja, um framework de simulação, que permita a execução de modelos independentes de plataforma. **Alta à capacidade** de interoperabilidade para diferentes satélites integrados a partir das interfaces comuns com o ambiente de simulação legado criará uma estrutura a qual suportará a reutilização das capacidades disponíveis em diferentes simulações e reduzindo os custos e o tempo necessário para criar um novo ambiente artificial de simulação.

Para reduzir e minimizar os impactos das mudanças é necessário seguir padrões bem definidos, criados por órgãos de padronização, para novos projetos de ambientes de simulação ou integração em ambientes de simulação legados, tornando os ambientes e modelos portáveis, reutilizáveis e de fácil entendimento pelos desenvolvedores. Um padrão para desenvolvimento de simuladores de satélite proposto pela ESA (European Space Agency) é o SMP Simulation Model Portability (ESA, 2001 e ESA, 2005).

Este artigo está organizado da seguinte maneira: a seção 2 apresenta os conceitos básicos do SMP; a seção 3 descreve a integração do ambiente de simulação legado com o SMP; seção 4 a conclusão do uso do SMP.

2. Simulation Model Portability - SMP

Com o objetivo de melhorar as necessidades endereçadas à padronização uma API chamada de Simulation Model Portability SMP foi definida pela ESA. A proposta principal da SMP é promover a portabilidade e o re-uso dos modelos em simulação. A SMP define um conjunto de especificações para codificação, uso da interface e a geração da documentação as quais são compartilhadas entre a ESA e a indústria. Os resultados são alcançados via quatro objetivos que são minimizar as interações dos modelos com o ambiente, padronizar as interfaces usadas pelos modelos, criar novos modelos com interfaces simples e que são facilmente entendidos pelos desenvolvedores (ESA, 2005). Em 1999 iniciou as atividades de documentação das especificações para um padrão em simulação e modelagem gerando a primeira liberação em 2001 com a sigla SMP1. Entretanto, a SMP1 possuía uma falta de suporte a moderna tecnologia de engenharia de software, como orientação a objetos (herança, métodos, etc.). Visando aperfeiçoar este padrão foi desenvolvida uma nova versão aprimorada chamada SMP2, cuja meta fundamental foi promover o desenvolvimento em plataforma independente, interoperável e reutilizável de modelos de simulação, baseados em projetos de componentes e em Model Driven Architecture (OMG, 2001). O SMP2 utiliza os padrões abertos de UML e XML e formula dois importantes conceitos (ESA, 2005):

Common Concepts. O desenvolvimento do modelo deve ser feito em níveis de abstração, para ser independente de plataforma e ser reutilizável.

Common Type System. Define que diferentes modelos tenham em comum: uma base sintática e semântica, o que é essencial para a interoperabilidade dos diferentes modelos.

Os modelos devem conviver entre estas duas camadas conforme mostrado na Figura 1.

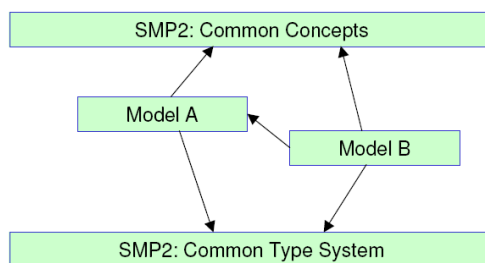


Figura 1. Modelos entre camadas.

Utilizando padrões abertos o SMP2 pode ser também usado como uma interface para um ambientes de simulação já existente, aproveitando-se de todas as vantagens que o padrão SMP2 tem e com poucas modificações no ambiente de simulação legado.

2.1 Arquitetura

A arquitetura do SMP2 se divide em dois componentes básicos, um é a simulação das instâncias dos modelos e o outro é o ambiente de simulação que providencia os serviços de simulação para a aplicação, esta arquitetura é mostrada na Figura 2.

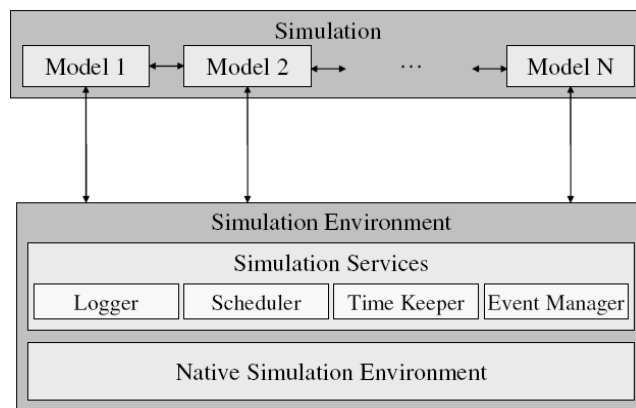


Figura 2. Arquitetura do SMP2.

A comunicação entre as entidades instanciadas no modelo e o ambiente é feita através das interfaces, e duas interfaces são identificadas na arquitetura: uma é a interface entre os modelos simulados e o ambiente de simulação e outra é a interface entre modelos, a chamada comunicação inter-model.

2.2. Metamodel SMP2

O Simulation Model Definition Language SMDL é uma linguagem usada para descrever os elementos e tipos que compõe o SMP2, enquanto o SMP2 Component Model (ESA, 2005) define os mecanismos de interação entre os modelos e entre outros componentes do SMP2, esta linguagem formaliza as regras básicas de colaboração entre os modelos. O SMDL gera documentos que descrevem os modelos e que são validados sintaticamente sobre um conjunto de esquemas implementados em XML. Com esse mecanismo existe a certeza de que as informações que foram modeladas podem ser trocadas com segurança entre diferentes stakeholders. O SMDL é composto de três partes principais: (i) *catalogue*, (ii) *assembly* e (iii) *schedule*. O *catalogue* possui informações que definem os modelos como tipos de dados e interfaces, enquanto o *assembly* descreve como as instâncias dos modelos serão interconectadas, e por fim o *schedule* define o agendamento das tarefas descritas no *assembly*. Em apoio existe o package que agrupa as implementações dos modelos e o workspace que une vários documentos em uma única unidade. A Figura 3 mostra a estrutura do meta-modelo SMP2 .

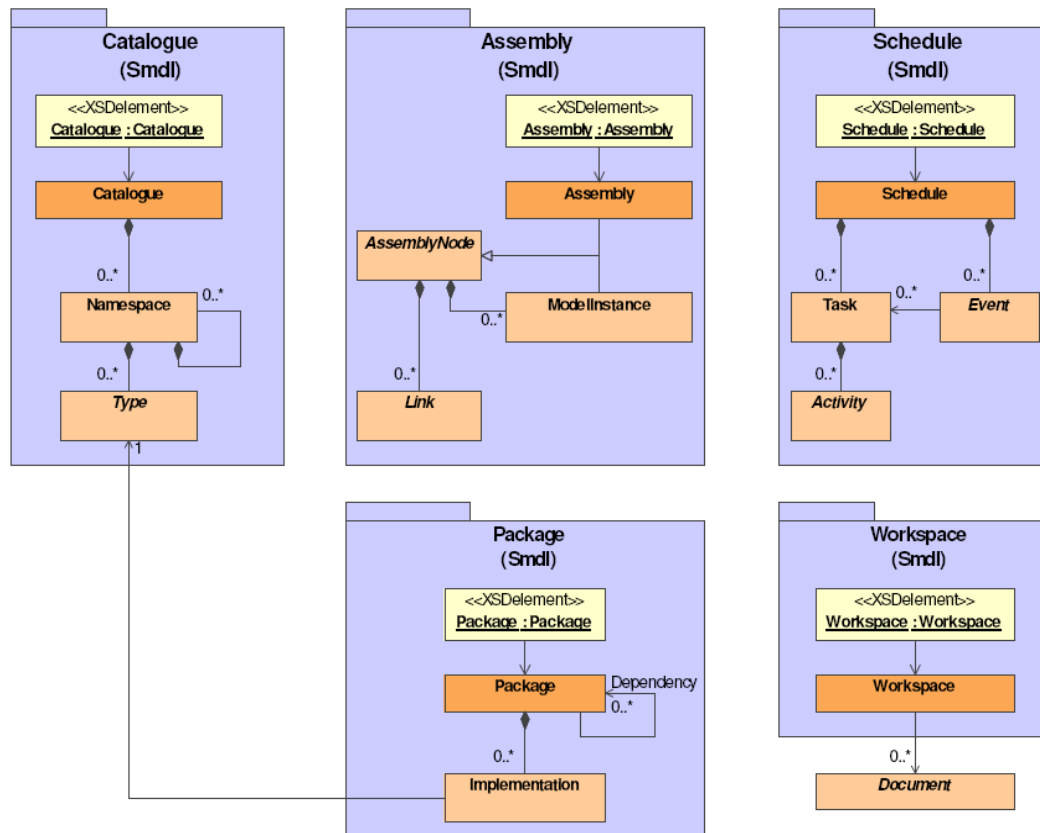


Figura 3. SMP2 Metamodel.

Por possuir um conjunto de documentos complexos o SMDL não pode ser gerado manualmente, sendo, então necessário um programa que ofereça um suporte a geração dos documentos em formato XML.

2.3. Model Development Kit MDK

O MDK é um conjunto de classes escritas em C++ com os componentes do ambiente de simulação estabelecidos e padronizados pelo SMP2 para poder instanciar os modelos e os serviços que foram formalizados via SMDL. Por usar uma linguagem de programação portátil o ISO/ANSI C++ o MDK pode ser reutilizado simplesmente compilando o código no sistema operacional desejado. Para que o MDK possa instanciar os modelos é necessário fazer o mapeamento dos componentes do modelo, dos serviços de simulação e os documentos SMDL para uma linguagem de programação a qual é definida pelo SMP2 como a ISO/ANSI C++ e pelas suas interfaces que são escritas na linguagem IDL/CORBA.

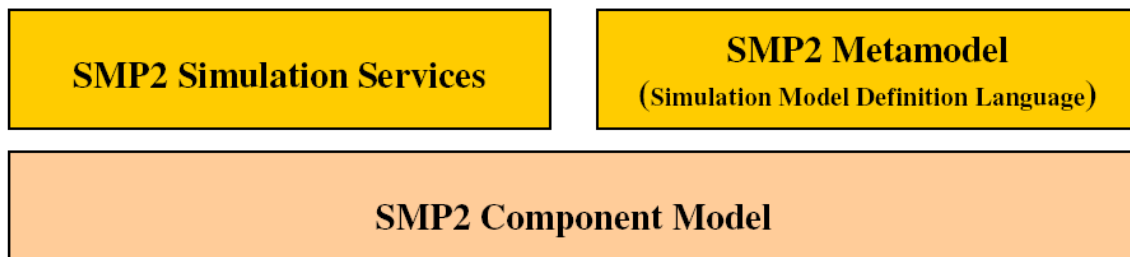


Figura 4. Partes que compõe a plataforma SMP2 usada pelo MDK.

Cada modelo desenvolvido via SMDL do SMP2 e cada serviço associado à simulação requer um componente associado. Um dos maiores melhoramentos feitos no SMP2 foi a padronização dos componentes usados na simulação, permitindo quebrar as funcionalidades em componentes individuais promovendo a reutilização dos componentes e facilitando as atualizações. O SMP2 tem três tipos de componentes principais, que fazem parte da execução da simulação, a saber: o ambiente de simulação, os serviços de simulação e os modelos definidos em SMDL. As bases estabelecidas para cada um dos três tipos de componentes são: a) para o ambiente de simulação, todas as interfaces são documentadas; b) para os serviços de simulação, a interface base, bem como

as interfaces para a simulação individual é documentada; c) para os modelos, a interface de base é obrigatória e documentada juntamente com os componentes do modelo; d) o mecanismo de aquisição genérica de serviços que envolvem todos os três tipos de componente é padronizado e documentado.

2.4. Elementos do SMP2 organizados em camadas

Para executar os modelos e realizar as simulações é necessário um ambiente de simulação, o qual em geral requer um grande esforço de desenvolvimento.

A visão geral e padronizada do SMP2 é composta por três camadas conforme mostrado na Figura 5, mas a camada descrita como software pode ser apresentada como um ambiente de simulação legado, a comunicação com a camada assembly será feita a partir das interfaces comuns entre as camadas se utilizando de troca de mensagens com o formato padrão XML.

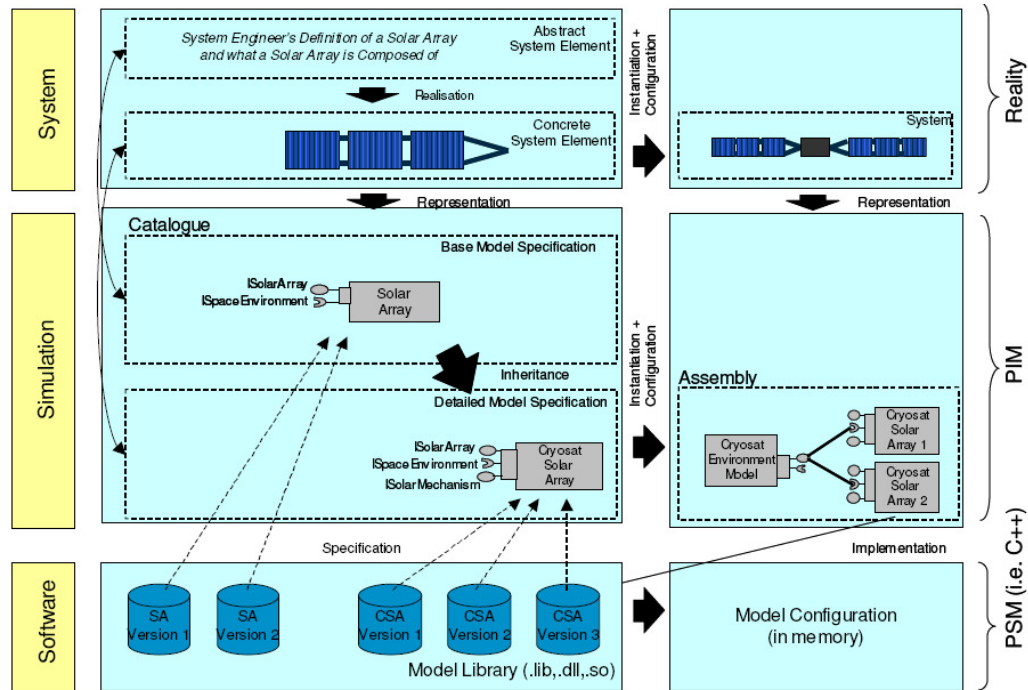


Figura 5. Camadas de elementos do SMP2.

3. Integração do ambiente de simulação legado com o SMP2

Os modelos e serviços do SMP2 devem ser assentados sobre um ambiente nativo de simulação, com isso é possível fazer uma integração entre o SMP2 e um ambiente de simulação legado. **A Erro! Fonte de referência não encontrada.** ilustra a integração de um sistema legado, representando o Simulador de Satélites e um conjunto de modelos novos.

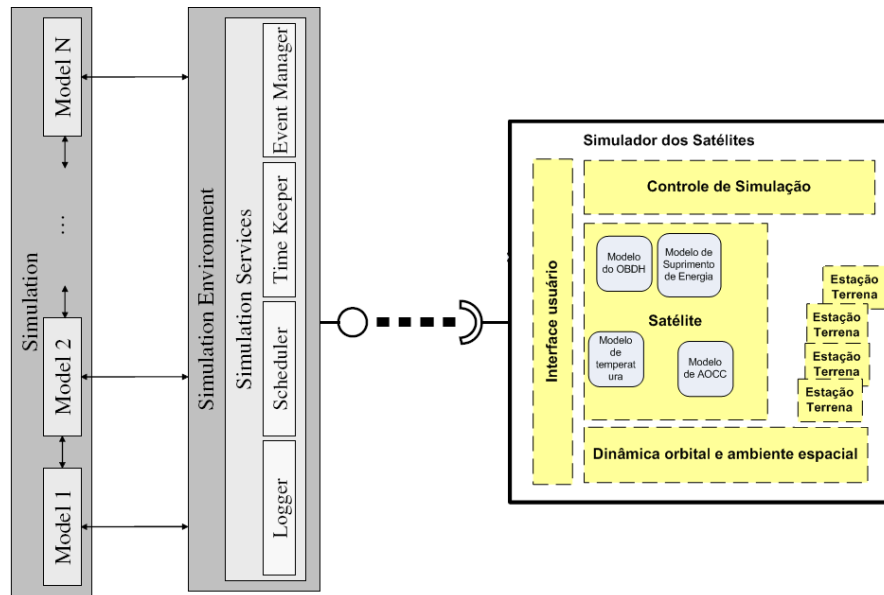
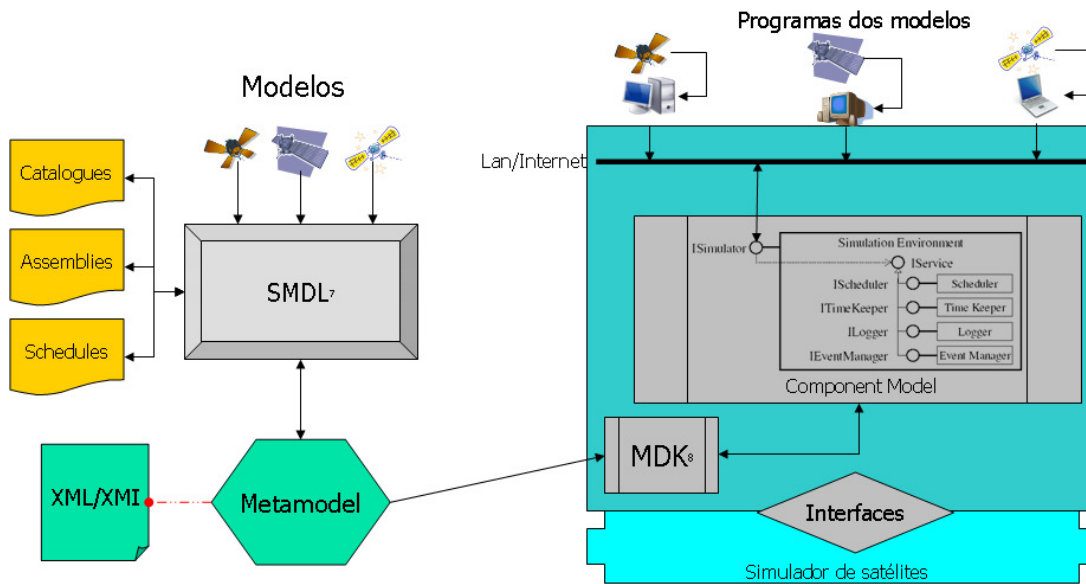


Figura 6. Esquema de Integração de um simulador legado a um conjunto de modelos novos.

A comunicação do ambiente de simulação legado será feita a partir das características e funcionalidades que o Simulation Model Portability possui conforme mostrado na Figura 7.



[7] SMDL – Simulation Model Definition Language, [8] MDK – Model Developer Kit

Figura 7. Ambiente de Integração.

No contexto deste trabalho de pesquisa, foi desenvolvido um programa, chamado SMP2 - SMDL, cujo objetivo é gerar documentos, formatados conforme as especificações do SMP2, a serem usados no ambiente de simulação. O foco principal deste programa é poder ser usado como sistema aberto em vários ambientes de simulação legados.

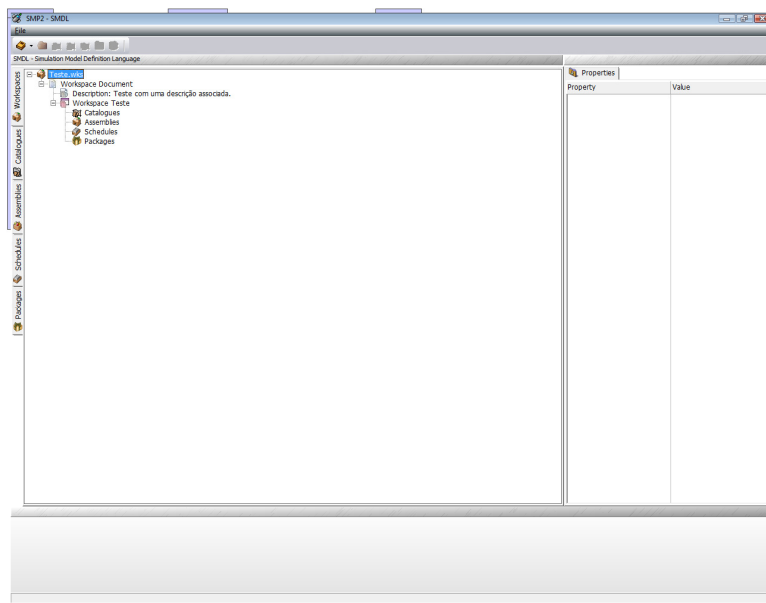


Figura 8. SMP2 - SMDL.

4. Conclusão

Com o grande número de ambientes de simulação legados para diferentes satélites, o uso dos conceitos do SMP para integração de novos modelos abstratos aos ambientes legados parece ser de grande valor. Esta solução agrega um aumento potencial de suas funcionalidades e não requer grandes alterações estruturais no código do sistema de simulação legado. Adicionalmente, com o avanço das tecnologias de computação é quase pequena a necessidade de mudanças na implementação do código do SMP, pois tem como principal característica o re-uso e a interoperabilidade dos modelos abstratos programados para a simulação.

5. Referências Bibliográficas

- Sommerville, Ian. Engenharia de Software, 497-513, Addison-Wesley, 2003
- Bisbal, J., Lawless, D., Wu, B. & Grimson, J. (1999). Legacy Information Systems: Issues and Directions. IEEE Software, 16, 103-111.
- SHANNON, Robert E., System Simulation: The art and science. Englewood Cliffs, N.J. Prentice Hall, 1975.
- GORDON, M. G. System Simulation Englewood Cliffs, N.J., Prentice Hall, 1969.
- Object Management Group; MDA guide version 1.0.1; OMG; <http://www.omg.org/docs/omg/03-06-01.pdf>.
- XML Database; http://en.wikipedia.org/wiki/XML_database
- An introduction to the XML: DB API; Kimbro Staken; January 09, 2002; http://www.xml.com/pub/a/2002/01/09/xmldb_api.html
- ESA, Simulation Model Portability Handbook, EWP-2080, Issue 1, 2002/02/01.
- TURNER, A. J. An Open-Source, Extensible Spacecraft Simulation And Modeling Environment Framework. Dissertação (Mestrado) | Virginia Polytechnic Institute and State University, 2003.
- ESA, Simulation Model Portability Handbook, EWP-2080, Issue 1, 2002/02/01.
- ESA, SMP 2.0 Handbook, EGOS-SIM-GEN-TN-099. Issue1.2, 28-Oct-2005.
- M.H, MacDougall, Simulation Computer Systems. MIT Press, 1987.
- Law, A. M.; Kleton, W.D. "Simulation Modeling & Nalysis". 2nd Internacional. Ed. Cingapura, SGP: McGraw-Hill, 1991.