

# **ESTRATÉGIA DE INTEGRAÇÃO DE MBSE COM SISTEMAS DE INFORMAÇÃO QUE APOIAM PROCESSOS DE CICLO DE VIDA DE SATÉLITES**

Engenharia de Sistemas, organização, complexidade

Ana Claudia de Paula Silva

Instituto Nacional de Pesquisas Espaciais - INPE

[ana.claudia@inpe.br](mailto:ana.claudia@inpe.br)

Geilson Loureiro

Instituto Nacional de Pesquisas Espaciais - INPE

[geilson.loureiro@inpe.br](mailto:geilson.loureiro@inpe.br)

## **RESUMO**

Este trabalho apresenta uma estratégia para integração automatizada de modelos da Engenharia de Sistemas Baseada em Modelos (MBSE) com sistemas de informação que apoiam processos de ciclo de vida de satélites. Os modelos de engenharia construídos nas fases iniciais de desenvolvimento do satélite são fonte de informação para os processos das fases posteriores. Dessa forma, é de interesse que informações contidas nesses modelos estejam disponíveis nos sistemas de informação. Dentro desse contexto, a integração automatizada tem um papel facilitador. O trabalho foi demonstrado com a integração de modelos do satélite Amazonia 1 com um sistema de informação para apoio ao processo de montagem, integração e testes de satélites. A estratégia de integração proposta neste trabalho consiste na definição de uma convenção de modelagem e de um mapeamento entre os modelos da MBSE e dos sistemas de informação. A convenção e o mapeamento viabilizam a integração automatizada entre os modelos. Concluímos que a estratégia apresentada aponta um caminho possível para integração automatizada de modelos da MBSE com sistemas de informação que apoiam processos do ciclo de vida de satélites.

## 1. INTRODUÇÃO

A Engenharia de Sistemas baseada em modelos (MBSE) é uma opção para desenvolver sistemas complexos, como satélites, em substituição à abordagem tradicional baseada em documentos.

Na MBSE, modelos são a principal fonte de informação e o principal meio de troca de informação entre os envolvidos no processo de engenharia, substituindo nessas funções os documentos utilizados na abordagem tradicional de engenharia de sistemas (FRIEDENTHAL et al., 2014). Modelos são construídos para representar requisitos, estrutura e comportamento dos sistemas (FRIEDENTHAL et al., 2014).

Friedenthal et al. (2014) cita como benefícios da MBSE: a melhoria na comunicação, a redução no risco de desenvolvimento, o aprimoramento da qualidade, o aumento da produtividade e a melhoria na transferência de conhecimento.

Este trabalho apresenta uma estratégia para integração automatizada de modelos da MBSE com sistemas de informação que apoiam processos de ciclo de vida de satélites.

Os processos do ciclo de vida de um satélite vão da sua concepção até o seu descarte. Considerando uma abordagem baseada em modelos para a engenharia do satélite, os modelos construídos nas fases de desenvolvimento do produto são fonte de informação para os processos de todas as fases posteriores. Dessa forma, é de interesse que informações contidas nos modelos de engenharia estejam disponíveis nos sistemas de informação que apoiam os processos do ciclo de vida de satélites. Por essa razão uma integração entre os modelos do satélite e os sistemas de informação mostra-se desejável.

A estratégia de integração proposta neste trabalho é demonstrada com o caso de um sistema de informação em desenvolvimento no Laboratório de Integração e Testes (LIT) do Instituto Nacional de Pesquisas Espaciais (INPE) que tem o objetivo de apoiar o processo de AIT (*Assembly, Integration and Testing* ou Montagem, Integração e Testes) de satélites.

Muitas informações importantes para AIT vem do processo de desenvolvimento do satélite. Sendo esse processo baseado em modelos, torna-se viável buscar uma integração que possibilite que essas informações sejam automaticamente inseridas no sistema de informação.

Embora até agora os satélites que passaram pelo LIT tenham sido desenvolvidos com a abordagem baseada em documentos, há uma tendência na Engenharia de Sistemas Espaciais de migração para a MBSE. Grandes atores da área espacial têm investido nessa abordagem, como NASA, ESA, Airbus, Thales Alenia Space e o próprio INPE (SILVA; LOUREIRO, 2020).

Os benefícios que a adesão a MBSE e a integração dos modelos com o sistema de informação podem trazer para o processo de AIT e, possivelmente, para outros processos do ciclo de vida de satélites, são a motivação para este trabalho.

## **2. ABORDAGEM**

Com base em documentos do satélite Amazonia 1 (INPE, 2022), que foi integrado e testado no LIT/INPE, foram construídos modelos exemplificando a abordagem MBSE para esse satélite. A ferramenta utilizada para a construção dos modelos foi a IBM Rhapsody (IBM, 2023).

Esses modelos foram comparados com o modelo de dados do sistema de informação, em desenvolvimento, que visa apoiar o processo de AIT de satélites do LIT.

A comparação dos modelos foi realizada com a intenção de identificar as informações dos modelos do satélite que são de interesse do sistema de informação, ou seja, informações que constam nos modelos do satélite e que também são previstas de serem armazenadas no sistema de informação.

Identificadas as informações de interesse, foi desenvolvida uma estratégia para integração dos modelos. Definida a estratégia, foi desenvolvido um código em linguagem groovy (APACHE, 2023) para automatizar a integração.

### 3. RESULTADOS

O processo de montagem, integração e testes (AIT na sigla em inglês) realizado no LIT/INPE pode ser resumido em cinco macro subprocessos (INPE/LIT, 2017):

Definir AIT: são definidas as atividades que serão executadas (desde as macro atividades até o nível de procedimento) e a sequência em que essas atividades devem ser executadas.

Preparar AIT: detalham-se as atividades, passo a passo, identificando e apontando os recursos necessários aos operadores para executá-las, como *setup*, equipamentos do satélite e GSEs (*Ground Support Equipment*).

Gerenciar AIT: o gerente de AIT organiza a execução das atividades, elabora o cronograma, libera tarefas para execução e trata das questões e problemas que ocorrem durante o processo.

Executar AIT: as atividades definidas são efetivamente realizadas e sua realização é documentada.

Gerenciar recursos: os recursos necessários para o processo de AIT, como materiais, ferramentas, pessoal, GSEs, são gerenciados e mantidos em condições de uso.

Para apoiar esse processo de ponta a ponta, está em desenvolvimento no LIT um sistema de informação que visa prover informações confiáveis para tomada de decisão e ser base de conhecimento para AITs futuros.

A estratégia de integração proposta será exemplificada com a integração desse sistema de informação com modelos do satélite Amazonia 1 que foi integrado e testado no LIT.

A linguagem utilizada nos modelos do sistema de informação é a UML. UML é uma linguagem utilizada para modelar softwares baseados no paradigma da orientação a objetos (OMG, 2017). Neste trabalho é apresentado um diagrama de Classes da UML. Uma Classe é um modelo para instanciar objetos (OMG, 2017).

A linguagem dos modelos do satélite utilizado como estudo de caso é a SysML. SysML é uma linguagem de modelagem gráfica de uso geral que pode ser base para diferentes métodos da MBSE (INCOSE, 2015).

Neste trabalho são apresentados um Diagrama de Definição de Blocos (BDD) e um Diagrama de Blocos Internos (IBD) da SysML. Esses são diagramas utilizados para representar a estrutura do sistema (FRIEDENTHAL et al., 2009). O bloco é a unidade básica de estrutura em SysML e pode ser usado para representar hardware, software, instalações, pessoal ou qualquer outro elemento do sistema. (FRIEDENTHAL et al., 2009).

Um BDD foca na definição e descreve a hierarquia do sistema (IBM, 2012). O IBD foca no uso (FRIEDENTHAL et al., 2014) e descreve a estrutura interna de um sistema em termos de suas partes, portas e conectores. Enquanto um Bloco é a definição de um elemento, o uso ou instância desse elemento é chamado Parte (IBM, 2012).

### 3.1. CLASSES DO SISTEMA DE INFORMAÇÃO

A Figura 1 mostra algumas Classes do sistema de informação com alguns dos seus atributos.

As classes referenciadas para demonstração da estratégia de integração são explicadas a seguir.

SatelliteModel: representa modelos do satélite (como modelo elétrico e modelo de voo, não confundir com modelos da MBSE).

Subsystem: representa subsistemas do satélite.

EquipmentType: representa tipos de equipamento que compõe o satélite, como roda de reação ou sensor solar.

SatelliteEquipment: representa um tipo de equipamento numa posição específica do satélite, como uma roda de reação na posição x ou y.

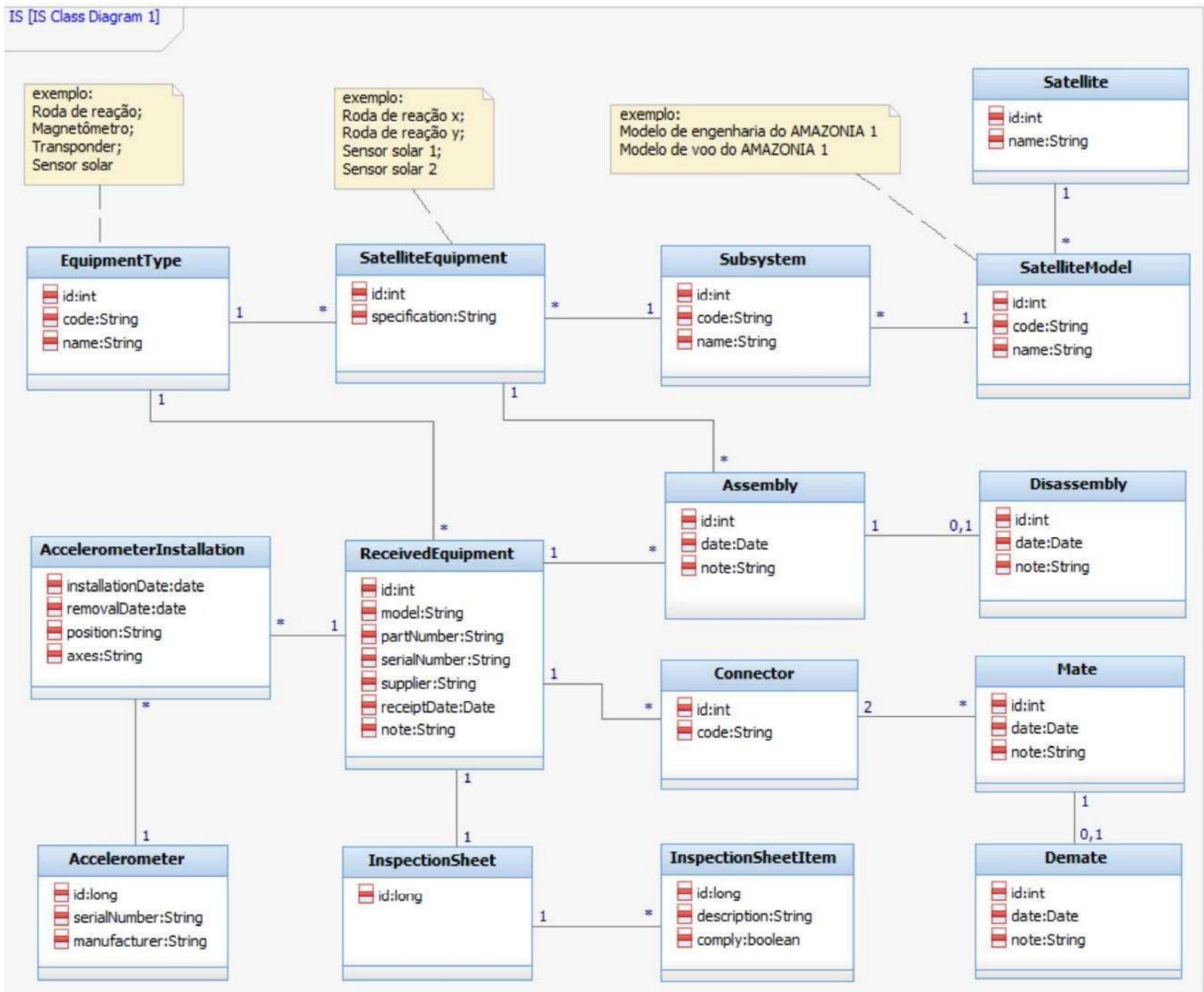


Figura 1- Diagrama de Classes do sistema de informação

### 3.2. MODELOS DO SATÉLITE

Uma pequena parte do modelo de arquitetura de implementação do satélite está representada nos diagramas das Figura 2 e Figura 3.

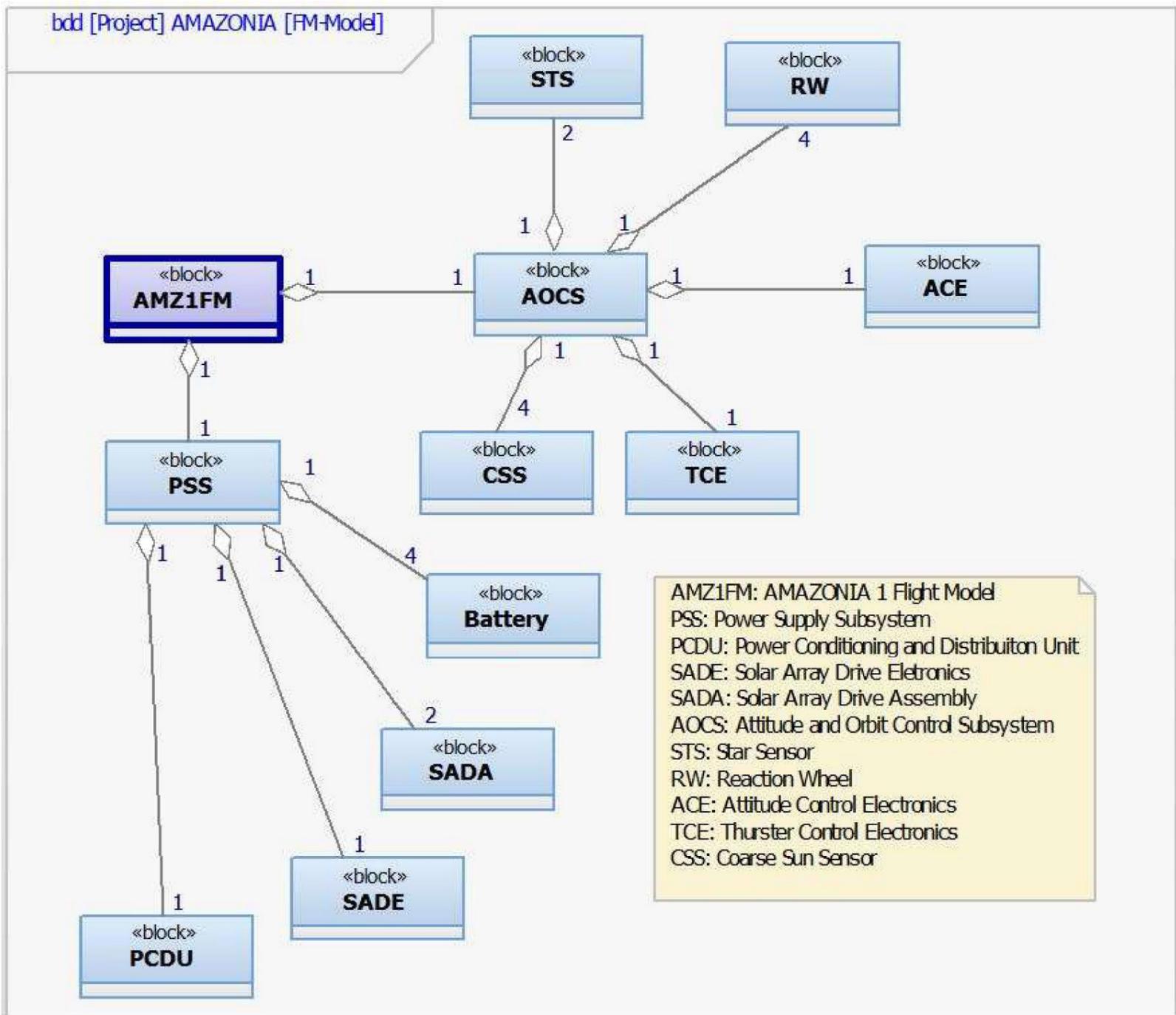


Figura 2 – BDD simplificado do satélite

No BDD da Figura 2 o modelo de voo do satélite (Bloco AMZ1FM) é representado como uma composição dos seus subsistemas e seus subsistemas são representados como uma composição de seus equipamentos. O diagrama apresenta dois subsistemas do satélite: PSS (*Power Supply Subsystem* ou subsistema de suprimento de energia) e AOCs (*Attitude and Orbit Control Subsystem* ou subsistema de controle de atitude e órbita) e alguns dos equipamentos que compõe esses subsistemas, como RW (*Reaction Wheel* ou roda de reação) e ACE (*Attitude Control Electronics* ou Eletrônica de controle de atitude).

De acordo com a notação SysML, no BDD as caixas representam os Blocos; as linhas com um losango em uma extremidade conectando os Blocos representam uma Associação de Partes (que são comumente usadas para decompor um Bloco em suas Partes) e os números na outra extremidade das linhas representam a quantidade de instâncias do Bloco relacionado. Por exemplo, o subsistema PSS é decomposto em 1 PCDU, 1 SADE, 2 SADA e 4 baterias.

A Figura 3 mostra instâncias do Blocos RW e ACE em um IBD do subsistema AOCS. No IBD as caixas representam as Partes e em seu primeiro compartimento aparecem o nome da Parte e o Bloco do qual a Parte é um tipo, no formato: Nome da Parte: Nome do tipo.



Figura 3 - IBD simplificado de um subsistema do satélite

### 3.3. ESTRATÉGIA DE INTEGRAÇÃO

Podemos observar que a estrutura do satélite é modelada nos diagramas estruturais do satélite, BDD (Figura 2) e IBD (Figura 3), e também no Diagrama de Classes (Figura 1) do sistema de informação, mas de maneiras diferentes.

Os elementos do BDD e IBD não correspondem diretamente aos elementos do Diagrama de Classes (um Bloco não corresponde a uma Classe), mas existe um relacionamento entre os modelos.

Elementos do BDD e IBD correspondem a instâncias das Classes do sistema de informação. O Bloco AMZ1FM corresponde a uma instância da classe SatelliteModel. Os Blocos dos subsistemas (AOCS e PSS) correspondem a instâncias da Classe Subsystem. Os Blocos que aparecem na decomposição dos subsistemas correspondem a instâncias da classe EquipmentType.

Instâncias da Classe SatelliteEquipment não aparecem no BDD, mas aparecem no IBD dos Blocos de subsistema.

Embora esses modelos do satélite e o modelo de Classes não tenham relação direta, é possível definir uma estratégia de integração. A estratégia apresentada neste trabalho inclui uma convenção de modelagem e um mapeamento entre os modelos.

## CONVENÇÃO DE MODELAGEM

Deve ser definida uma convenção de modelagem para a construção dos modelos do satélite que possibilite a identificação dos elementos de interesse do sistema de informação.

Neste trabalho são utilizados estereótipos da SysML para implementar o link entre os modelos. Os estereótipos permitem classificar elementos com “algo em comum”. Com este recurso, podemos classificar os elementos dos modelos SysML, identificando a qual Classe do sistema de informação eles estão relacionados.

Foi definido que os estereótipos seriam iniciados com o prefixo “is\_” (referência a *Information System*) para indicar que os elementos com eles classificados são de interesse do sistema de informação. A Figura 4 mostra IBDs com Partes classificadas com estereótipos. As propriedades, ou atributos, dos elementos também são apontadas nessa figura porque são referenciadas na Tabela 1.

Os seguintes estereótipos foram utilizados: <<is\_satellite\_equipment>>, <<is\_satellite\_model>> e <<is\_subsystem>>.

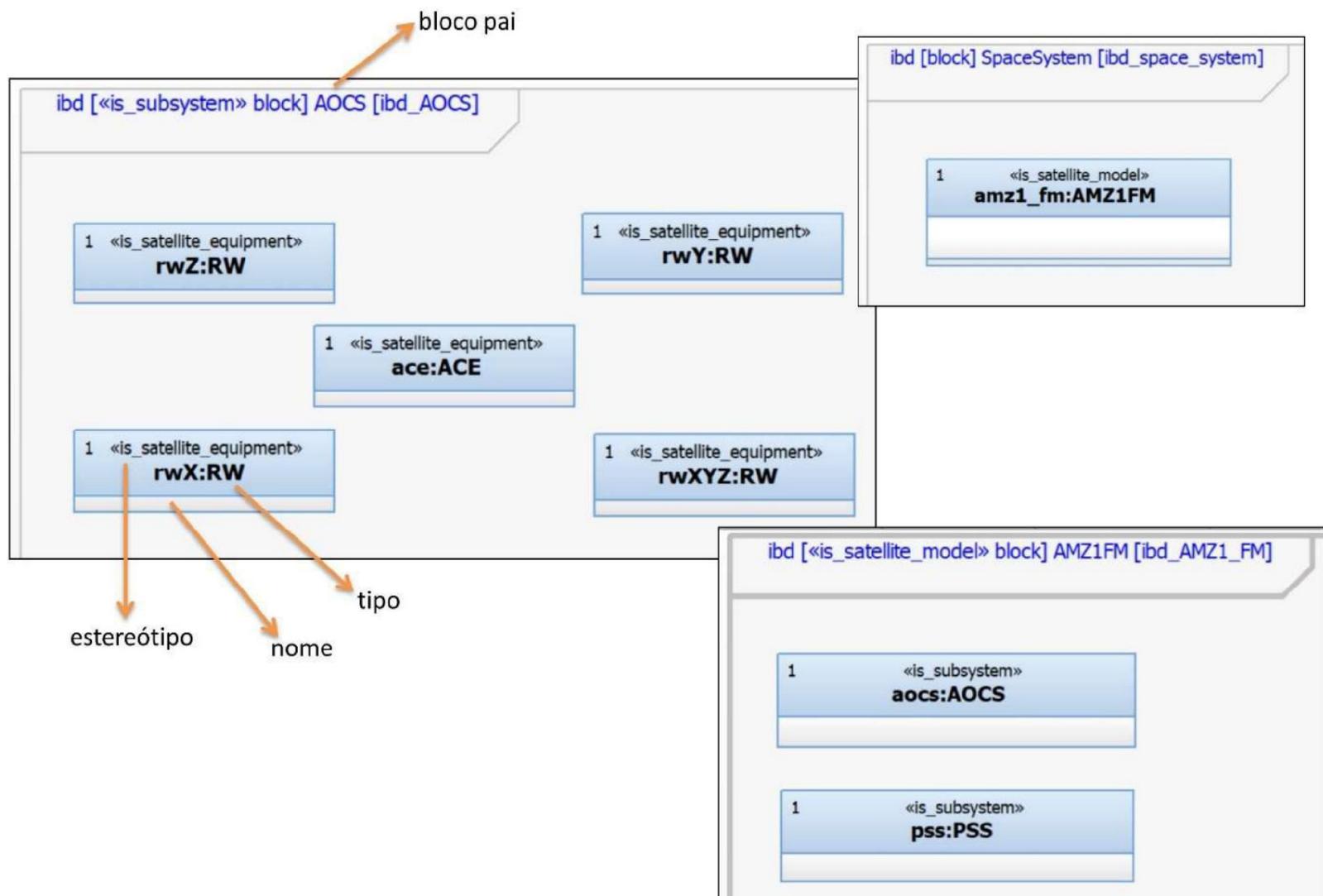


Figura 4 - IBDs com Partes classificadas com estereótipos

## MAPEAMENTO

Definida a convenção, é necessário mapear o destino de cada informação do modelo SysML que interessa ao sistema de informação, conforme exemplificado na Tabela 1, na qual é feito um mapeamento para leitura de IBDs para popular Classes do sistema de informação.

Nessa tabela, na coluna “Atributo da Parte”, aparecem os nomes dos atributos apontados na Figura 4 e os termos entre parênteses correspondem ao nome dos atributos na ferramenta Rhapsody (IBM, 2023), que foi utilizada para criar os modelos SysML.

De acordo com o mapeamento apresentado na Tabela 1, os elementos do modelo do satélite identificados com o estereótipo “is\_subsystem” devem ser alocados na Classe Subsystem do sistema de informação, sendo que, o valor do atributo “nome” do elemento será atribuído ao atributo “code” e o valor do atributo “bloco pai” será atribuído ao atributo “satelliteModel” dessa Classe.

Os elementos do modelo do satélite identificados com o estereótipo “is\_satellite\_equipment” devem ser alocados nas classes EquipmentType e SatelliteEquipment. Sendo que, o valor do atributo “tipo” do elemento será atribuído ao atributo “code” da Classe EquipmentType. O valor do atributo “nome” será atribuído ao atributo “specification” da Classe SatelliteEquipment, o valor do atributo “tipo” será atribuído ao atributo “equipmentType” e o valor do atributo “bloco pai” será atribuído ao atributo “ subsystem” dessa mesma classe.

Os elementos do modelo do satélite identificados com o estereótipo “is\_satellite\_model” devem ser alocados na Classe SatelliteModel, sendo que o atributo “nome” do elemento será atribuído ao atributo “code” dessa Classe.

Tabela 1- Mapeamento

<b>Estereótipo</b> (no modelo SysML)	<b>Classe</b> (no modelo do sistema de informação)	<b>Atributo da Parte</b> (no modelo SysML)	<b>Atributo da Classe</b> (no modelo do sistema de informação)
is_subsystem	Sybsystem	nome (name)	code
		bloco pai (ofClass.name)	satelliteModel
is_satellite_equipment	EquipmentType	tipo (otherClass.name)	code
	SatelliteEquipment	nome (name)	specification
		tipo (otherClass.name)	equipmentType
		bloco pai (ofClass.name)	subsystem
is_satellite_model	SatelliteModel	nome (name)	code

### 3.4. INTEGRAÇÃO AUTOMATIZADA

A ferramenta Rhapsody possui uma biblioteca, chamada JAVA API (IBM, 2009), que permite ler e escrever nos modelos. Utilizando essa biblioteca, foi criado um script na linguagem groovy (APACHE, 2023) para ler os modelos SysML, interpretá-los de acordo com o mapeamento definido e gerar entradas para o sistema de informação a partir dos modelos do satélite.

A Figura 5 apresenta o código escrito em groovy que faz uso da biblioteca JAVA API para a leitura dos modelos do Rhapsody. A Figura 6 apresenta o código groovy que cria objetos, a partir dos elementos do Rhapsody, para popular o banco de dados do sistema de informação.

## 4. DISCUSSÃO E CONCLUSÃO

Este trabalho apresentou uma proposta de convenção de modelagem e mapeamento que possibilitam a integração dos modelos de engenharia de um satélite com modelos de um sistema de informação que apoia um processo do ciclo de vida desse satélite e demonstrou ser viável a automatização dessa integração via software.

Embora a estratégia de integração automatizada apresentada tenha sido demonstrada com um caso de desenvolvimento de um satélite e de um sistema de informação específicos, ela pode ser aplicada a outros casos que envolvam MBSE e sistemas de informação para apoio a processos do ciclo de vida de satélites.

Visto que informações contidas nos modelos do satélite são de interesse dos sistemas de informação que apoiam os processos do seu ciclo de vida, a integração proposta neste trabalho mostra-se um facilitador para o compartilhamento das informações, criando um vínculo que minimiza retrabalho e inconsistências, já que informações dos modelos do satélite podem ser automaticamente inseridas no sistema de informação e mudanças podem ser refletidas automaticamente.

O padrão da convenção de modelagem a ser utilizada e do mapeamento vai depender da linguagem de modelagem escolhida para a engenharia do satélite e da linguagem de modelagem e paradigma adotados para o sistema de informação.

Para a integração automatizada é necessário que sejam utilizadas ferramentas de modelagem que ofereçam recursos que possibilitem a leitura dos modelos. É preciso destacar que atualmente as ferramentas para a prática da MBSE ainda são limitadas e não oferecem muitas facilidades, fator que pode trazer certa dificuldade para a aplicação prática da estratégia proposta.

De qualquer forma, concluímos que a estratégia apresentada neste trabalho aponta um caminho possível para integração de modelos da MBSE com sistemas de informação que apoiam processos do ciclo de vida de satélites.

Como trabalho futuro tem-se a perspectiva de desenvolvimento de um *framework* para padronizar e direcionar a integração automatizada de modelos de engenharia de sistemas com modelos de engenharia de software.

```

/*Classe que lê o projeto do Rhapsody e armazena os elementos de interesse do SI */
class ReadRhapsody {

    private IRPProject prj //projeto do Rhapsody

    /*Listas para armazenar os elementos de interesse do SI */
    List satelliteEquipments = []
    List subsystems = []
    List satelliteModels = []

    ReadRhapsody(){
        /*Pega o projeto ativo do Rhapsody.
        * Precisa usar o java 32-bit para rodar, porque a biblioteca rhapsody.dll é 32-bits*/
        prj = RhapsodyAppServer.activeRhapsodyApplication.activeProject()
        initialize()
    }

    /* Lê os modelos ativos no Rhapsody e popula as listas com os elementos de interesse do SI */
    private void initialize(){
        //Pega todos os elementos do projeto ativo
        List allElements = prj.nestedElementsRecursive.toList()

        allElements.each { element ->
            //armazena todos os esteriótipos do elemento
            List stereotypes = element.stereotypes.toList()*.name
            /*verifica se o elemento está classificado com esteriótipo existente no mapeamento
            acordado entre o engenheiro de software e o engenheiro de sistemas (Tabela 6.12 da tese)
            e o inclui na respectiva lista
            */
            if ("is_subsystem" in stereotypes)
                subsystems << element
            else if ("is_satellite_equipment" in stereotypes)
                satelliteEquipments << element
            else if ("is_satellite_model")
                satelliteModels << element
        }
    }
}

```

Figura 5 - Código para ler os modelos do Rhapsody

```

1  /* Classe que cria objetos para popular o banco de dados do SI
2  * a partir dos elementos lidos do modelo Rhapsody */
3  class InformationSystem {
4      List<SatelliteModel>    satelliteModels    = []
5      List<Subsystem>        subsystems        = []
6      List<EquipmentType>    equipmentTypes    = []
7      List<SatelliteEquipment> satelliteEquipments = []
8
9
10     public InformationSystem(){
11         println "Reading Rhapsody Model ..."
12         ReadRhapsody rhapsody = new ReadRhapsody()
13
14         //popula subsystems
15         rhapsody.subsystems.each{subsystem ->
16             //cria o satelliteModel que contém o subsystem
17             SatelliteModel satelliteModel = new SatelliteModel(code:subsystem.ofClass.name)
18             //verifica se o satelliteModel já está na lista de satelliteModels, se não adiciona
19             if (! satelliteModels.find{ SatelliteModel model-> model.code == subsystem.ofClass.name })
20                 satelliteModels << satelliteModel
21             //cria o subsystem e o adiciona na lista de subsystems
22             subsystems << new Subsystem(code: subsystem.name, satelliteModel: satelliteModel)
23         }
24
25         //popula satelliteEquipment
26         rhapsody.satelliteEquipments.each { satelliteEquipment ->
27             //cria o equipmentType do satelliteEquipment
28             EquipmentType equipmentType = new EquipmentType(code: satelliteEquipment.otherClass.name)
29             //verifica se o equipmentType já está na lista equipmentTypes, se não, adiciona
30             if (! equipmentTypes.find{ EquipmentType it -> it.code == satelliteEquipment.otherClass.name})
31                 equipmentTypes << equipmentType
32             //obtem o subsystem do satelliteEquipment
33             Subsystem subsystem = subsystems.find{ Subsystem it -> it.code == satelliteEquipment.ofClass.name }
34             //cria o satelliteEquipment e o adiciona na lista de satelliteEquipments
35             satelliteEquipments << new SatelliteEquipment(specification: satelliteEquipment.name,
36                                                         type:equipmentType , subsystem: subsystem)
37         }
38     }
39 }

```

Figura 6 - Código para popular o sistema de informação

## AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

## REFERÊNCIAS BIBLIOGRÁFICAS

APACHE. **Groovy Language Documentation**. 2023. Acessado em 21 de abril de 2023.

Disponível em: <<http://www.groovy-lang.org/single-page-documentation.html>>.

FRIEDENTHAL, S.; MOORE, A.; STEINER, R. **OMG Systems Modeling Language (OMG SysML™) Tutorial**. [S.l.]: INCOSE, 2009.

FRIEDENTHAL, S.; MOORE, A.; STEINER, R. **A Practical Guide to SysML: The Systems Modeling Language**. 3th. ed. [S.l.]: Morgan Kaufmann, 2014.

IBM. **Rational Rhapsody API Reference Manual**. [S.l.]: IBM Corporation, 2009.

IBM. **Essentials of IBM Rational Rhapsody for Systems Engineers v.7.6.1 - Student Manual vol 1**. [S.l.]: IBM Corporation, 2012.

IBM. **IBM Engineering Systems Design Rhapsody**. 2023. Acessado em 21 de abril de 2023.

Disponível em: << <https://www.ibm.com/products/systems-design-rhapsody>>>.

INCOSE. **Incose Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities**. 4th. ed. [S.l.]: Wiley, 2015.

INPE. **Amazon Mission**. 2022. Accessed July 13, 2022. Disponível em:

<<http://www.inpe.br/amazonia1/en/>>.

INPE/LIT. **Minuta da reunião realizada em 21 de setembro de 2017 no LIT/INPE, LIT25-LIT05-MR-001**. 2017. Disponível no centro de documentação do LIT.

OMG. **Unified Modeling Language - OMG Document Number formal/2017-12-05**. [S.l.]: OMG, 2017.

SILVA, A. C. d. P.; LOUREIRO, G. Iniciativas de adesão à mbse na área espacial. In:

RODRIGUES, A. C.; BARBOSA, A. L.; TEIXEIRA, A. F.; BATISTA, C. L. G.; OLIVEIRA, C. M. d.;

CUELLAR, D. A. (Ed.). **Anais... do 11º Workshop em Engenharia e Tecnologia Espaciais**. São

José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), 2020. ISSN 2177-3114.