



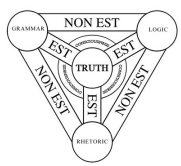
IEA-S – DEPARTAMENTO DE SISTEMAS ESPACIAIS
(SPACE SYSTEMS DEPARTMENT)

DOCUMENTOS

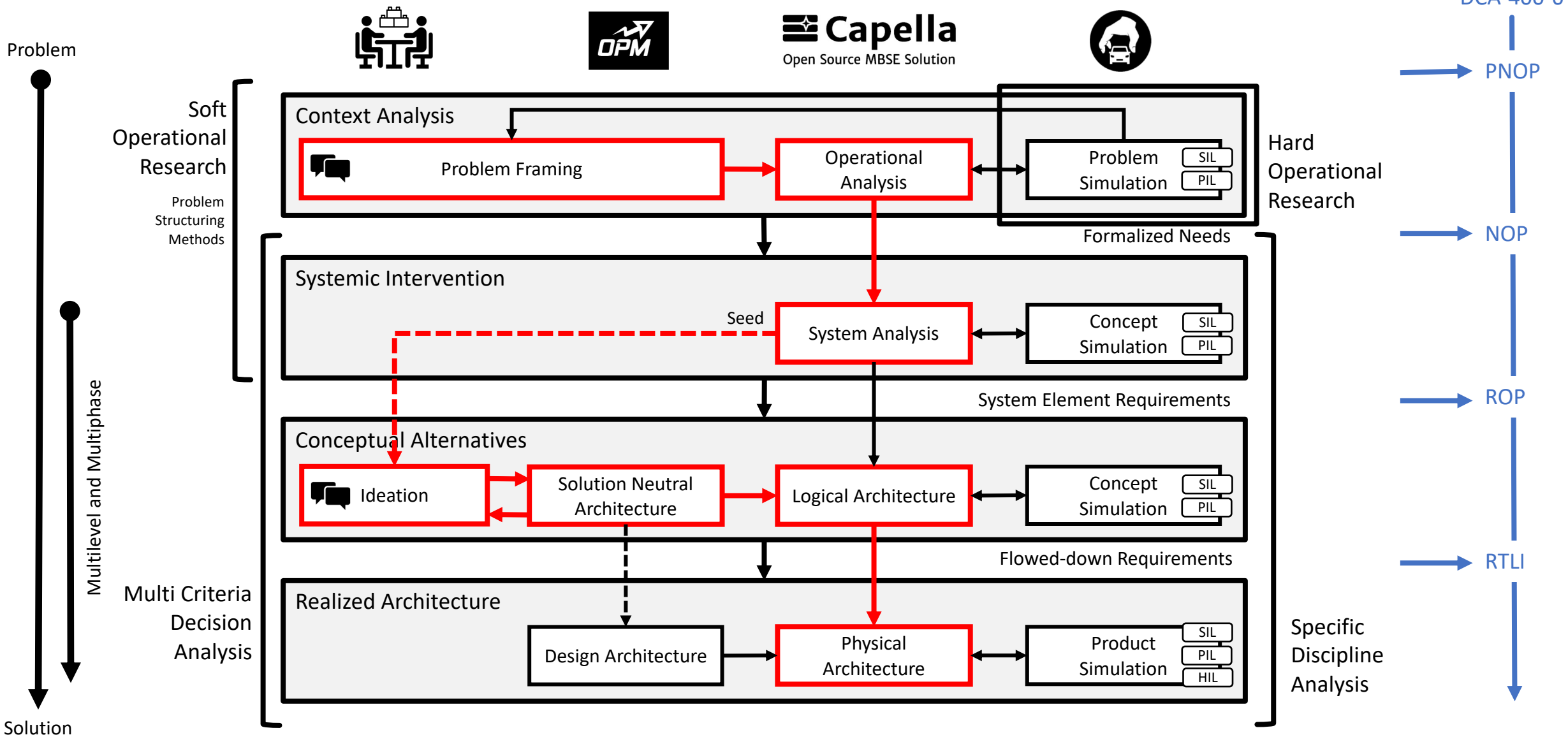


[TE-265 ENGENHARIA DE SISTEMAS BASEADA EM MODELOS] [2023]

| AULA | TEORIA | INDIVIDUAL | GRUPO | AULA | TEORIA | GRUPO |
|-------------|---|--------------------------------------|--|---|--|---|
| 1 06-Mar | Introdução e Apresentação de Engenharia de Sistemas | Resumo dos princípios | Definição do grupo. Montagem de apresentação do tema. | 9 08-May | Introdução ao Arcadia e Análise do Contexto | Análise do Contexto e apontamento de necessidades |
| 2 13-Mar | Pitch: Temas Frameworks e Stakeholders | Trabalho sobre MBSE | Elicitar stakeholders | 10 15-May | Pitch: Análise do Contexto - NOP Intervenção Sistemica | Intervenção Sistemica e requisitos da missão do sistema |
| 3 20-Mar | Arquitetura e Funções. Coesão e acoplamento. | Exercícios de análise estruturada | Mapa de interações | 11 22-May | Pitch: Análise do Sistema - ROP OPM e Exploração de Alternativas | Montagem de Alternativas |
| 4 27-Mar | Ciclo de Vida e CONOPs | | Ciclo de Vida e CONOPs | 12 29-May | Pitch: Alternativas Arquitetura Conceitual e desdobramentos | Arquitetura Conceitual e requisitos do sistema |
| 5 03-Apr | Pitch: Descrição livre da captura do problema Requisitos | Exercícios de correção de requisitos | Requisitos dos stakeholders | 13 05-Jun | Pitch: Arquitetura Conceitual - RTLI Revisita de Requisitos e o processo de Verificação e Validação | Apontamento de etividades de verificação |
| 6 10-Apr | Modelagem Estrutural da Arquitetura | Exercícios de fixação | Relatório e Gravação de 5min com explicação | 14 12-Jun | Pitch: Propostas de Verificação Arquitetura Concreta e Carta Morfológica | Arquitetura Concreta com decisões tecnologicas |
| 7 17-Apr | Modelagem de Comportamento da Arquitetura | Exercícios de fixação | | 15 19-Jun | Pitch: Arquitetura Final - Especificações Revisão e desdobramentos para especialidades | Relatório e Gravação de 5min com explicação. |
| 8 24-Apr | P1 - Questões conceituais e Mini-Case | | | 16 26-Jun | P2 - Apresentação final da relação entre as etapas | |
| | | | | EXAME Graduação (grupo): Desenvolvimento de um mini-case de um subsistema usando o Capella | | |
| | | | | 03-Jul Pós-graduação (grupo): Entrega de um artigo (Formato do SIGE) descrevendo seu case e atividades. | | |
| | | | | 17-Jul | | |



CONCEPTIO FRAMEWORK





REQUISITOS E VV



EXEMPLO DE MODELO DE ESTRUTURA

- Os requisitos são mais comumente expressos como declarações de linguagem natural, embora as linguagens de requisitos matemáticos gráficos e formais também sejam usadas.
- Usando linguagem natural, as métricas de qualidade de requisitos podem ser desenvolvidas por meio da análise de cada declaração de requisito nos elementos de um modelo estrutural de um bom requisito.



EXEMPLO DE UMA ESTRUTURA

- **Actor.** Este é o sujeito da sentença - a coisa que está sendo especificada. Exemplos (bons e ruins!) são: “the system”, “the interface”, “the function”,
- **Conditions of Action.** Isso define as condições durante as quais a ação deve ocorrer, por exemplo. “in Replay Mode”, e/ou as condições iniciais: “upon receipt of a message”, “power having been applied”,
- **Action.** Este é um verbo - a ação a ser tomada pelo ator (sujeito). Exemplos são: “shall calculate”, “shall display”, “shall fly”, “shall be displayed”
- **Constraints of Action.** Estes qualificam a ação, por exemplo. “at a resolution of 400 x 1000 pixels”, “within limits imposed by vehicle speed”,, e incluir desempenho
- **Object of Action.** Este é um substantivo, e é a coisa sobre a qual se age ao tomar a ação. São exemplos: “the message”, “the input signal”,
- **Refinement/Source of Object.** Estes qualificam o objeto, por exemplo (refinamento): “of flash priority”, por exemplo (fonte): “from DISCON”.
- **Refinement/Destination of Action.** Estes qualificam ainda mais a ação e podem ser adicionais às Restrições de Ação. “in accordance with IEEE 802.11g”, “to DISCON”.



EXEMPLO UTILIZANDO O TEMPLATE

| Element | Text |
|--|--|
| Actor: | The system, |
| Condition: | upon receipt of a message, |
| Action: | shall switch |
| Object of Action: | that message |
| Constraints of Action: | within 10 milliseconds of receipt |
| Refinement of Object: | for messages in ACP128 format having a valid routing indicator |
| Source of Object: | from the message input port, |
| Destination of Object: | to a message output port, |
| (Further) Refinement of Action: | corresponding to the routing indicator in the message. |



METADADOS

TABLE 4.2-2 Requirements Metadata

| Item | Function |
|---------------------|---|
| Requirement ID | Provides a unique numbering system for sorting and tracking. |
| Rationale | Provides additional information to help clarify the intent of the requirements at the time they were written. (See “Rationale” box below on what should be captured.) |
| Traced from | Captures the bidirectional traceability between parent requirements and lower level (derived) requirements and the relationships between requirements. |
| Owner | Person or group responsible for writing, managing, and/or approving changes to this requirement. |
| Verification method | Captures the method of verification (test, inspection, analysis, demonstration) and should be determined as the requirements are developed. |
| Verification lead | Person or group assigned responsibility for verifying the requirement. |
| Verification level | Specifies the level in the hierarchy at which the requirements will be verified (e.g., system, subsystem, element). |



VERIFICAÇÃO

- **A VERIFICAÇÃO** de um produto mostra **prova de conformidade com os requisitos** - o produto pode atender a cada declaração "deve" como comprovada através do desempenho de um teste, análise, inspeção ou demonstração (ou combinação destes).
- **VERIFICAÇÃO é um processo de confirmação de que um requisito ou sistema está em conformidade.**
 - Em outras palavras, a verificação do sistema faz a pergunta: **o sistema atende aos seus requisitos?**
 - A verificação de requisitos faz a pergunta: **o sistema realmente atende a esse requisito específico?**



VALIDAÇÃO

- VALIDAÇÃO refere-se ao documento **ConOps**.
- A **VALIDAÇÃO** é conduzida em **condições realistas (ou condições simuladas)** em **produtos finais** com o objetivo de determinar a eficácia e adequação do produto para **uso em operações de missão por usuários típicos**.
- A **VALIDAÇÃO** pode ser realizada em cada fase de desenvolvimento usando produtos de fase (por exemplo, modelos) e não apenas na entrega usando produtos finais.



VALIDAÇÃO X VERIFICAÇÃO

- **VERIFICANDO UM SISTEMA:**

Construindo o sistema corretamente: garantindo que o sistema esteja em conformidade com os requisitos do sistema.



- **VALIDANDO UM SISTEMA:**

Construindo o sistema certo: certificando-se de que o sistema faz o que deve fazer em seu ambiente operacional.

TABLE D-1 Requirements Verification Matrix

| Requirement No. | Document | Paragraph | Shall Statement | Verification Success Criteria | Verification Method | Facility or Lab | Phase ^a | Acceptance Requirement? | Preflight Acceptance? | Performing Organization | Results |
|--|--|---|---|--|--|--|--|---|--|---|--|
| <i>Unique identifier or each requirement</i> | <i>Document number the requirement is contained within</i> | <i>Paragraph number of the requirement</i> | <i>Text (within reason) of the requirement, i.e., the "shall"</i> | <i>Success criteria for the requirement</i> | <i>Verification method for the requirement (analysis, inspection, demonstration, test)</i> | <i>Facility or laboratory used to perform the verification and validation.</i> | <i>Phase in which the verification and validation will be performed.</i> | <i>Indicate whether this requirement is also verified during initial acceptance testing of each unit.</i> | <i>Indicate whether this requirement is also verified during any pre-flight or recurring acceptance testing of each unit</i> | <i>Organization responsible for performing the verification</i> | <i>Indicate documents that contain the objective evidence that requirement was satisfied</i> |
| P-1 | xxx | 3.2.1.1 Capability: Support Uplinked Data (LDR) | System X shall provide a max. ground-to-station uplink of... | 1. System X locks to forward link at the min and max data rate tolerances 2. System X locks to the forward link at the min and max operating frequency tolerances | Test | xxx | 5 | Yes | No | xxx | TPS xxxx |
| P-i | xxx | Other paragraphs | Other "shalls" in PTRS | Other criteria | xxx | xxx | xxx | Yes/No | Yes/No | xxx | Memo xxx |
| S-i or other unique designator | xxxxx (other specs, ICDs, etc.) | Other paragraphs | Other "shalls" in specs, ICDs, etc. | Other criteria | xxx | xxx | xxx | Yes/No | Yes/No | xxx | Report xxx |

^a Phases defined as: (1) Pre-Declared Development, (2) Formal Box-Level Functional, (3) Formal Box-Level Environmental, (4) Formal System-Level Environmental, (5) Formal System-Level Functional, (6) Formal End-to-End Functional, (7) Integrated Vehicle Functional, (8) On-Orbit Functional.

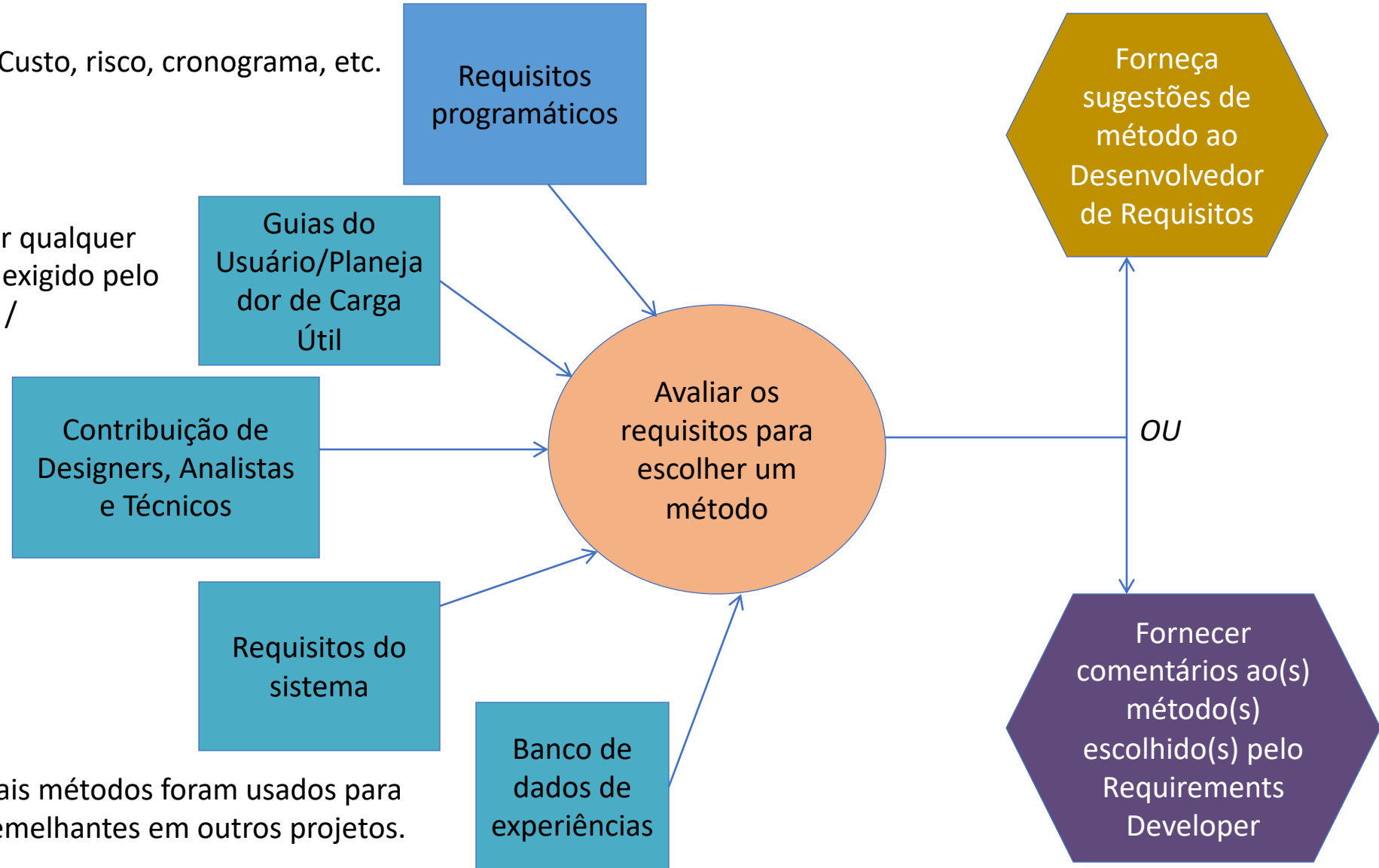


ESCOLHENDO UM MÉTODO V&V

➤ Custo, risco, cronograma, etc.

➤ Revise-os para procurar qualquer método de verificação exigido pelo veículo de lançamento / transportador

➤ Pesquise quais métodos foram usados para requisitos semelhantes em outros projetos.





ESCOLHENDO UM MÉTODO

Teste é usado quando:

- * Técnicas analíticas não produzem resultados adequados
- * Existem modos de falha que comprometem a segurança, os sistemas espaciais ou os objetivos da missão
- * Componentes associados a interfaces críticas do sistema

Demonstração é usada quando:

- * A verificação das funções projetadas pode ser realizada através da observação
- * Os resultados tendem a ser "aprovação/reprovação", "sim/não"
- * Natureza mais subjetiva, como fatores humanos ou capacidade de manutenção
- * Usa sistemas de voo reais ou representativos

A inspeção é usada quando:

- * Desenhos, documentos ou dados podem ser verificados visualmente para verificar se as características físicas foram projetadas no produto
- * Normalmente usado para características de projeto, métodos de construção, mão de obra, dimensões e registros

A análise é usada quando:

- * Técnicas analíticas precisas são possíveis
- * O teste não é uma opção econômica
- * A verificação por inspeção não é adequada
- * Técnicas e modelos analíticos foram validados



Design Requirement

3.2.2.15.34 Recovery Force Communications
The product shall provide a communications system capable of communicating with the recovery forces pre- and post- landing

Design Verification

| Verification Objective | Pass / Fail (Success Criteria) |
|--|---|
| Perform Integrated System Test of the communications system capability to provide a voice communications and beacon with recovery forces pre and post landing within an integrated hardware / software environment | Testing will show that the communications system can transmit and receive audio at frequencies and ranges (power) represented by standard ground recovery force communications devices as defined in TBD |
| Perform a demonstration of the communications systems capability to provide voice and beacon communications with recovery forces pre and post landing while within a representative environment and using a production equipment configuration | Demonstration will show the ability for the communications systems to verbally communicate with the on-board communication production configuration equipment. The demonstration will also show beacon tracking within communication ranges established by TBD. |

Traceability

| Verification Cross Reference Matrix | | | | | | |
|-------------------------------------|-----|---|---|-----|-------|-------|
| Paragraph # | N/A | I | A | M/S | D | T |
| 3.2.2.15.34 | | | | | | VR-5T |
| 3.2.2.15.34 | | | | | VR-5D | |

SE – Translates Operational Objectives into Design Requirements

Design – Provides assessment of requirements implementation

Test – Provides assessment of requirements verifiability

SE – Provides compliance of the design requirement

Test / Implementation Group –

Ensures Verification Implementation Feasibility

Advises alternatives to support programmatic

Assesses completeness

Provides verifiability assessment

SE – Verification Allocation and Traceability Assurance



- **VR-5T: Prove that the product's communications system is capable of communicating with the ground command team by performing an integrated system test within an integrated hardware/software environment. Testing will show that the product can transmit and receive to standard ground recovery forces audio at frequencies represented by communications devices defined in (TBD).**

Verification Objective

Verification Method

Environment

Note – there are no
Special Conditions

Success Criteria



FERRAMENTAS DE VERIFICAÇÃO

- O programa de verificação deve ser executado com o apoio de instrumentos de verificação adequados (ou seja, equipamento de apoio em terra, simuladores, ferramentas analíticas, instalações de ensaio, etc.).
 - Deve ser dedicado o máximo cuidado à concepção e verificação destas ferramentas, uma vez que os resultados da verificação dependem estritamente da sua qualidade.
 - Os instrumentos de verificação **devem ser normalizados** para melhorar a comunalidade e a reutilização a diferentes níveis.



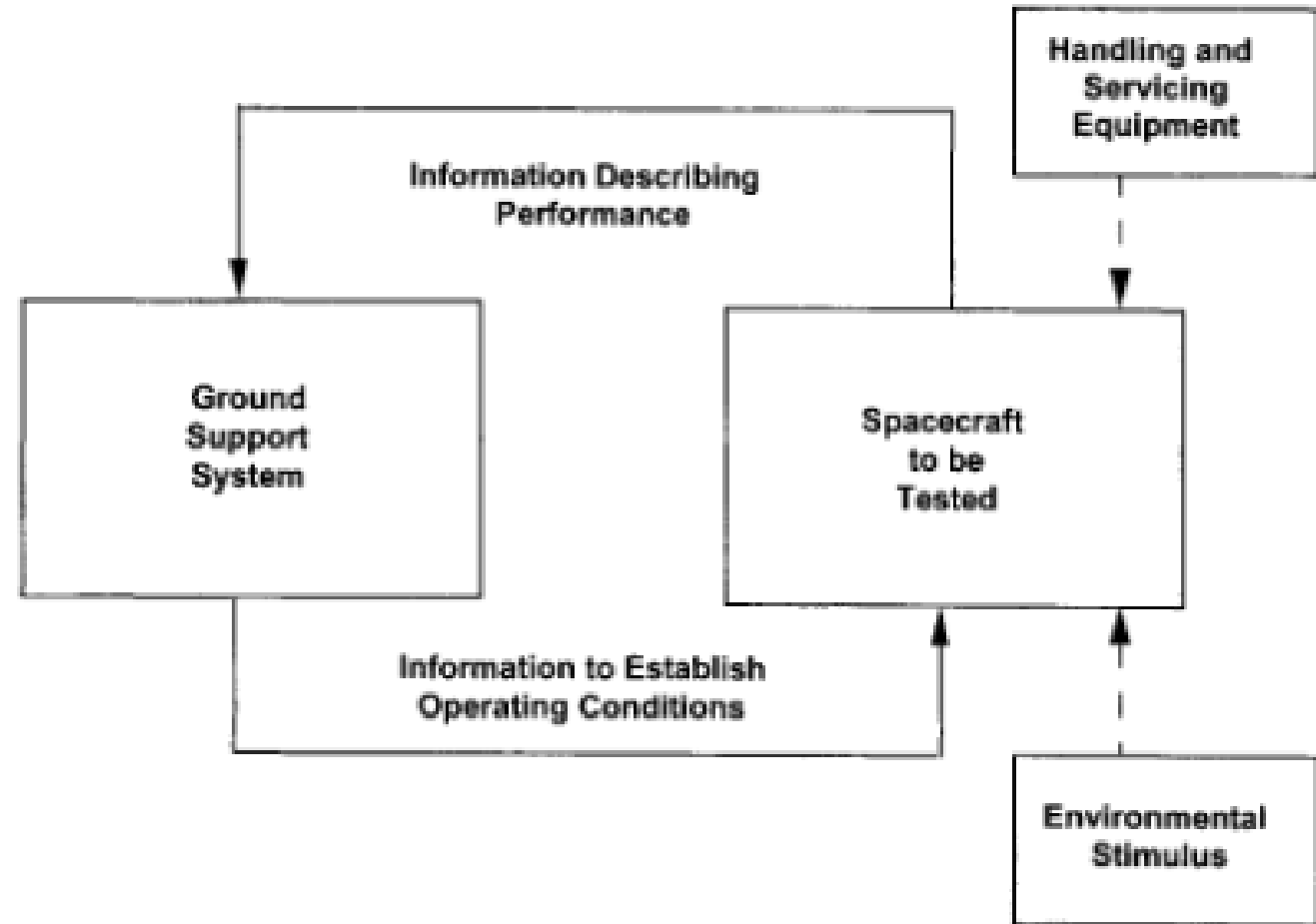
SISTEMA DE APOIO TERRESTRE GROUND SUPPORT SYSTEM (GSS)

- O (GSS) compreende o **equipamento usado para operar o Sistema por meio da integração de subsistemas**, testes ambientais e operações no local de lançamento.
- O GSS é usado para **verificar se o Sistema opera conforme o esperado e atende às suas especificações de desempenho**. O projeto do GSS a ser usado durante a integração e teste **começa com uma compreensão da missão** e do design de seus subsistemas e instrumentos.
- Isso significa que é necessário que a equipe de integração e teste (e os projetistas do sistema GSS) estejam extremamente familiarizados com os vários subsistemas que compõem o Sistema e a operação do software de voo.



FERRAMENTAS DE IMPLEMENTAÇÃO

- (1) estabelecer as condições de funcionamento dos subsistemas dos sistemas;
- (2) coletar e processar dados do Sistema que indicam seu desempenho em resposta a vários estímulos;
- (3) fornecer o ambiente simulado sob o qual o Sistema deve ser avaliada;
- (4) fornecer manuseio seguro do sistema.





EQUIPAMENTOS DE APOIO TERRESTRE

GROUND SUPPORT EQUIPMENT

- Os equipamentos de suporte AIT de sistemas espaciais são geralmente classificados em mecânicos e elétricos.
 - **Equipamento de apoio mecânico - Mechanical support equipment (MGSE)** são todas as ferramentas específicas do satélite necessárias para realizar todas as operações mecânicas no e com o satélite.
 - **Equipamentos de apoio elétrico - Electrical support equipment (EGSE)** consiste em todo o equipamento elétrico e acessórios correspondentes necessários para realizar os testes funcionais do satélite.



[HTTPS://WWW.CLEMESSY.CH/EGSE/](https://www.clemessy.ch/EGSE/)

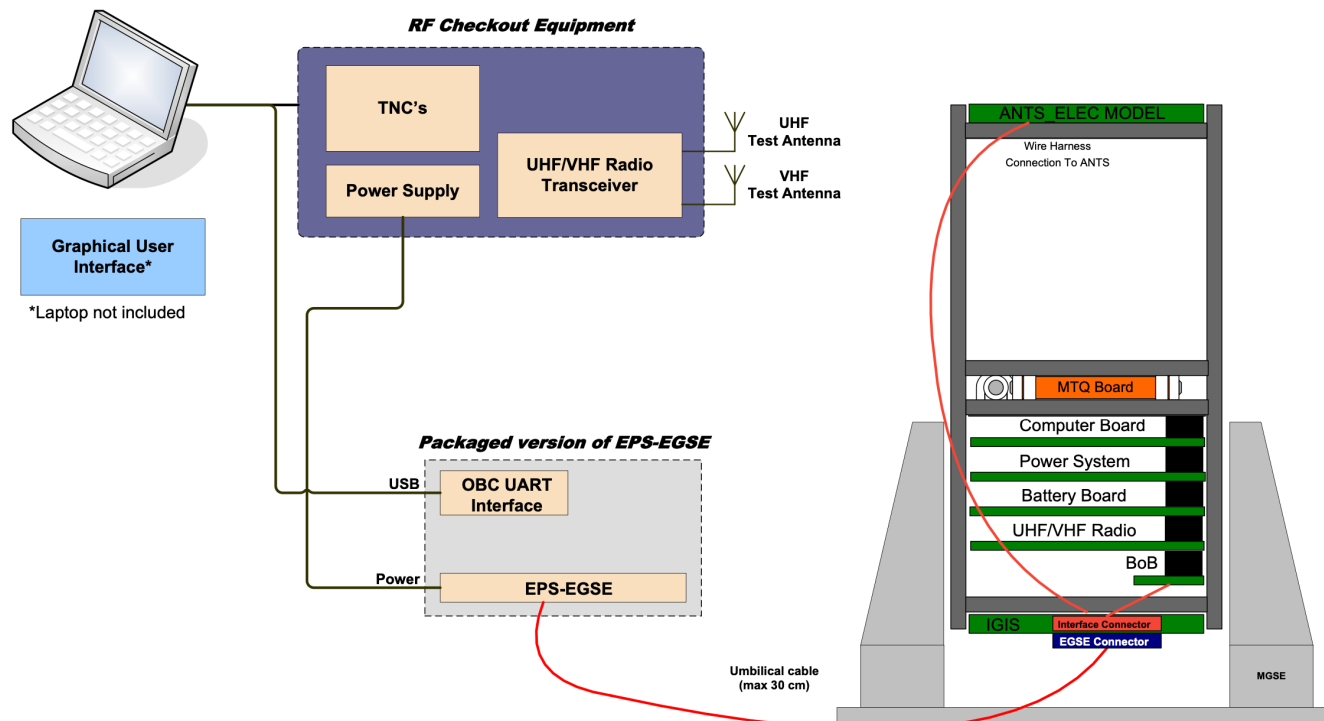


Satellite battery conditioning equipment made
@Clemessy Switzerland





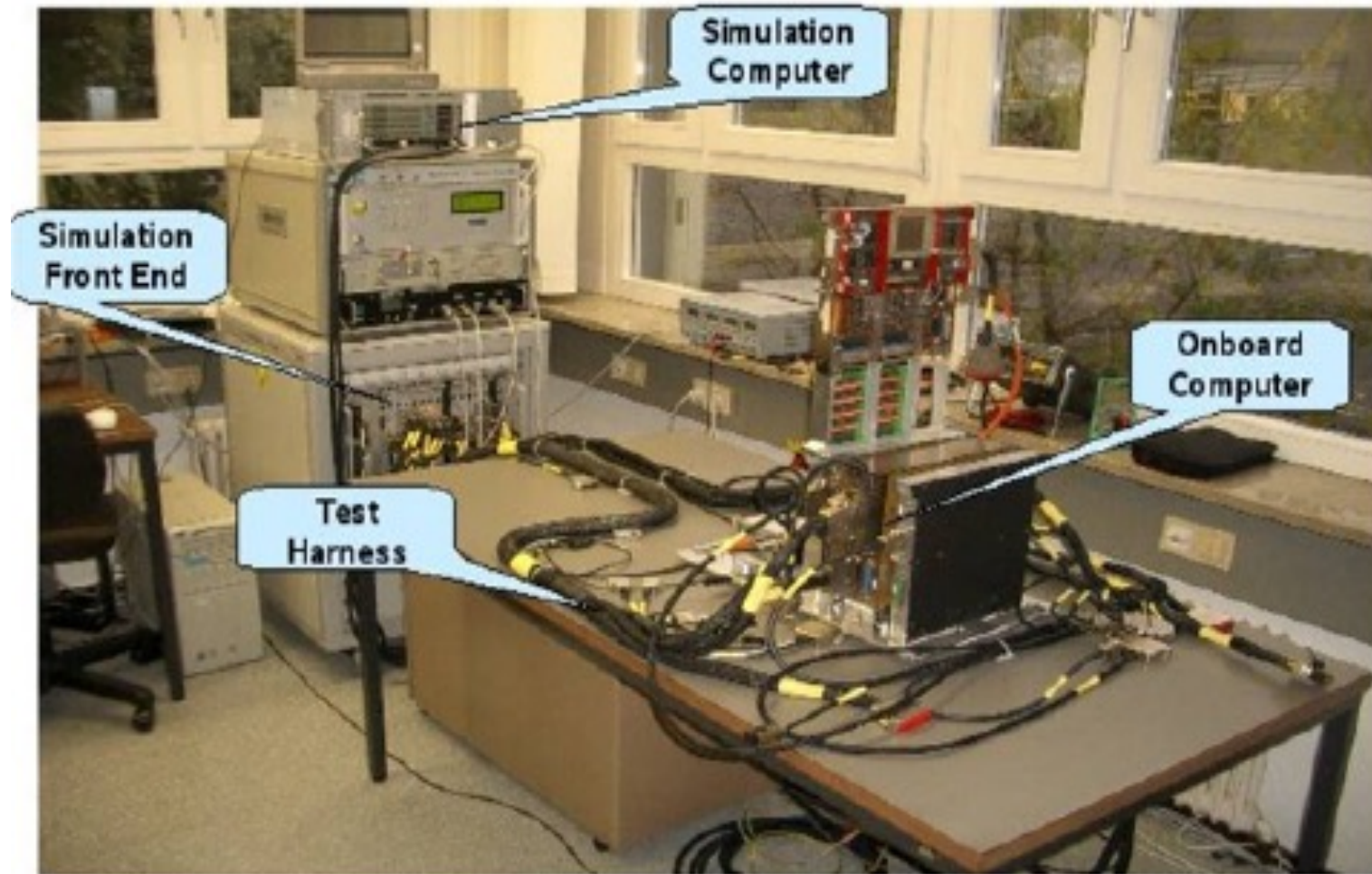
[HTTPS://WWW.ISISPACE.NL/WP-CONTENT/UPLOADS/2016/02/ISIS-GEM-GENERIC-ENGINEERING-MODEL-OVERVIEW-V1.3.PDF](https://www.isispace.nl/wp-content/uploads/2016/02/ISIS-GEM-GENERIC-ENGINEERING-MODEL-OVERVIEW-V1.3.PDF)





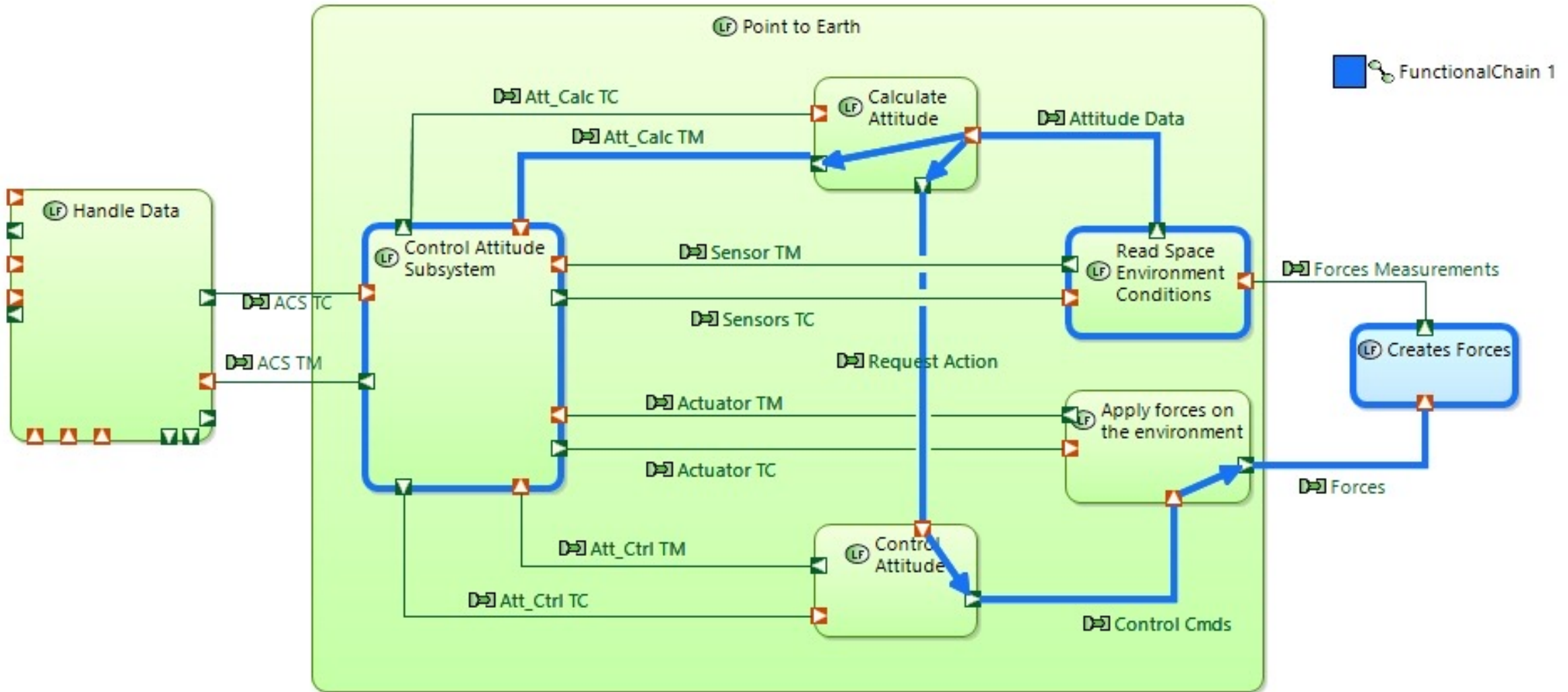
APLICAÇÕES DO MDVE

- Real-time testbed (TerraSAR)



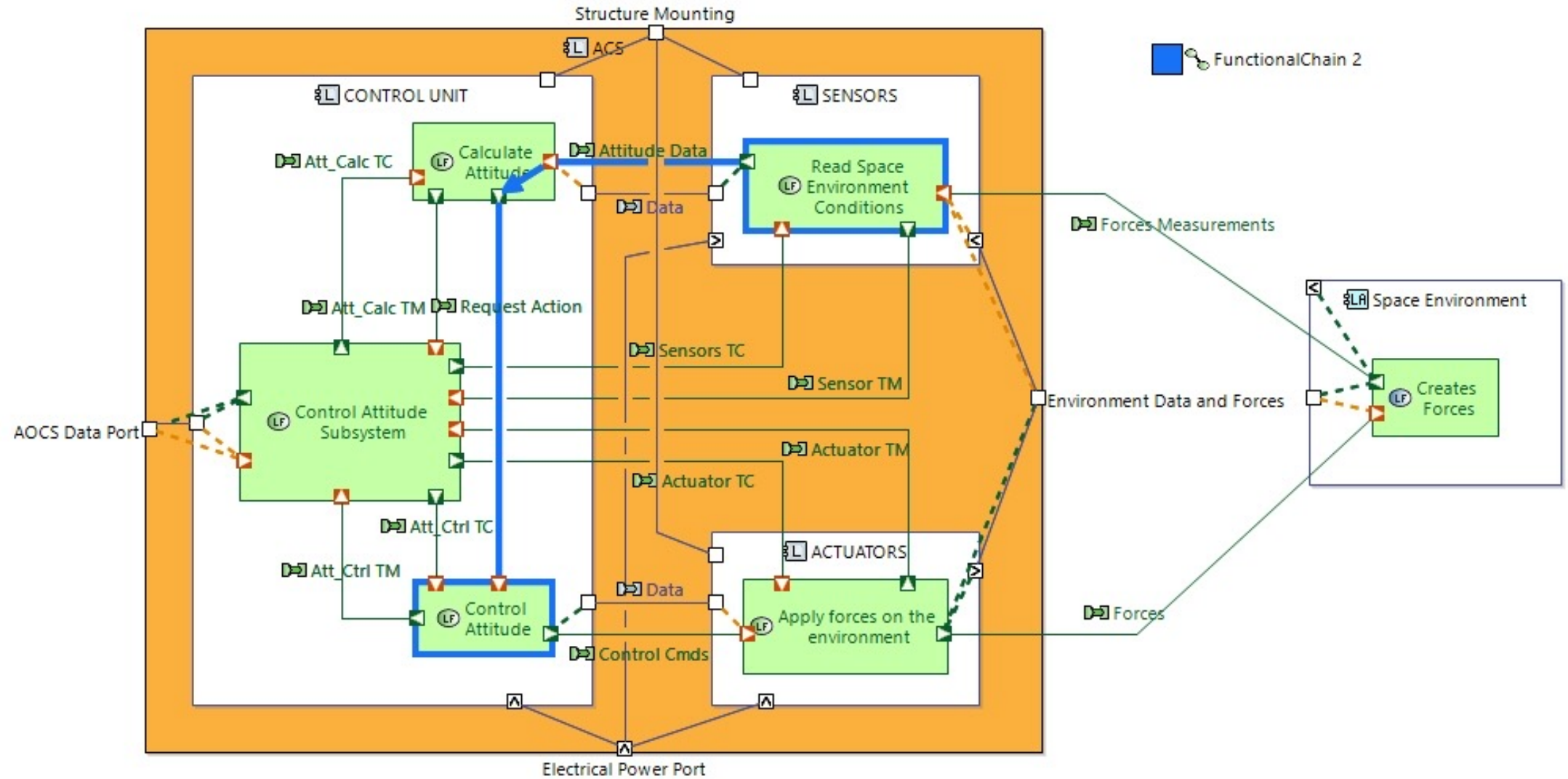


ARQUITETURA FUNCIONAL



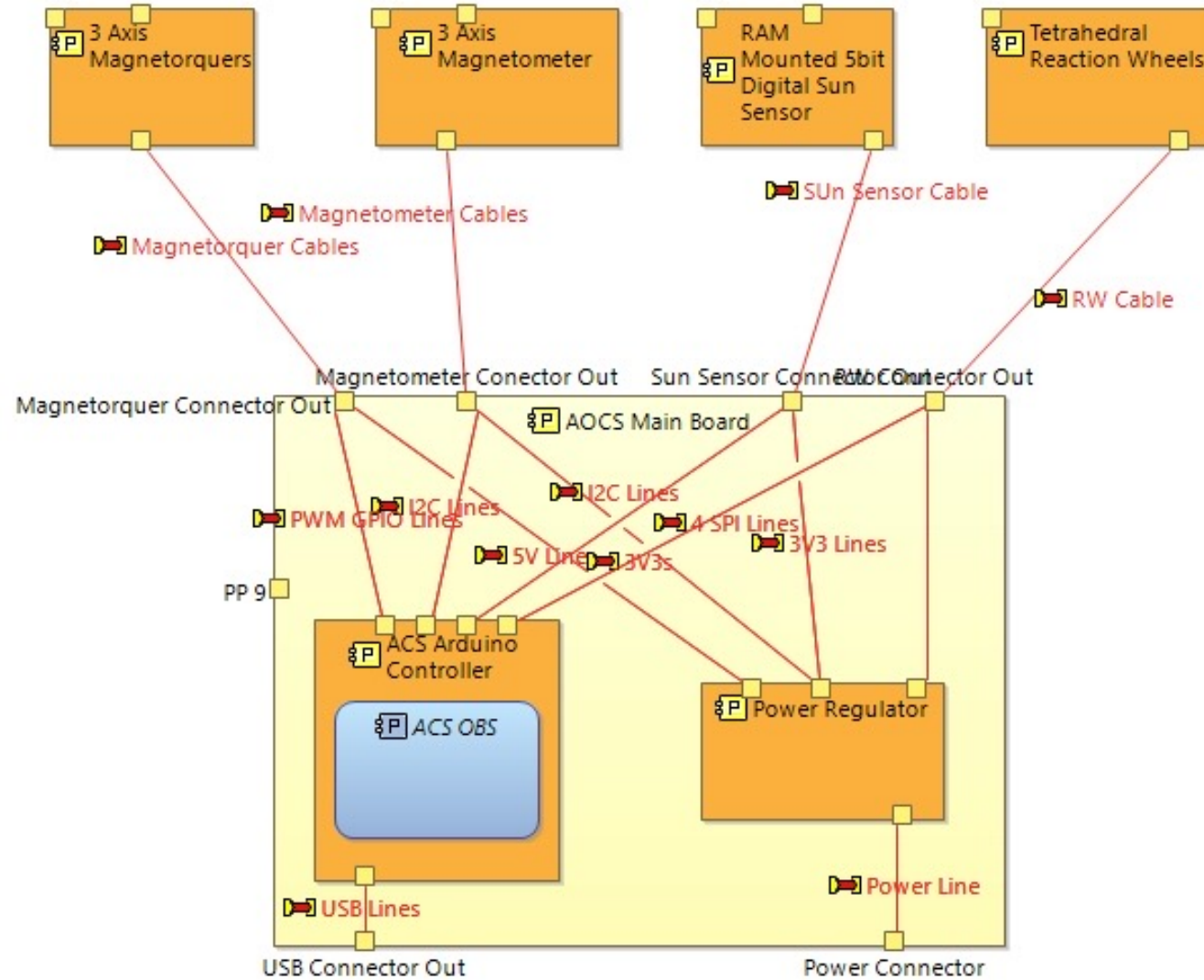


COMO FOI ESTRUTURADO (LAB):



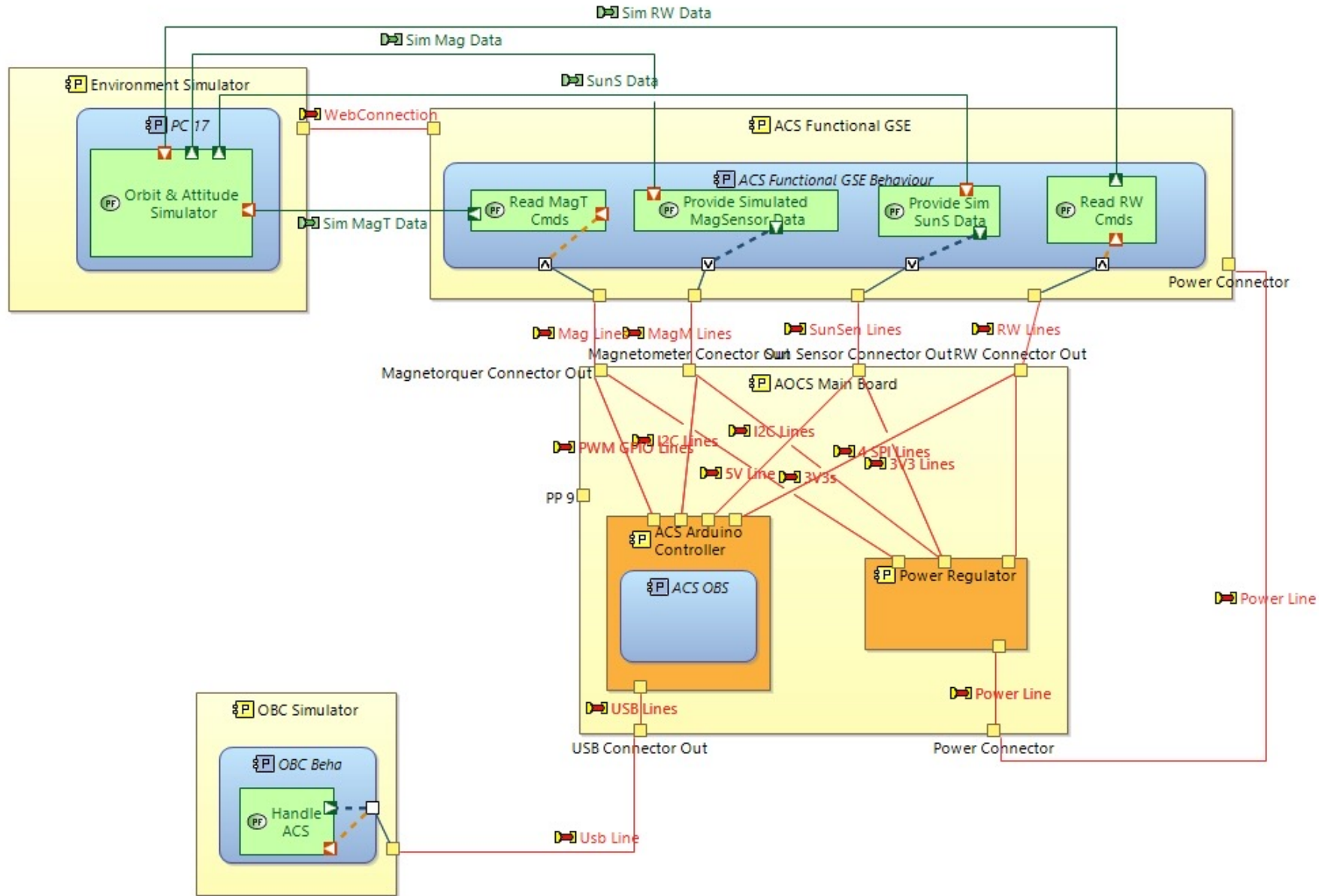


COMO FOI ENTREGUE





ESTRATÉGIA DE INSTRUMENTAÇÃO





CICLO DE VIDA DA PRÓPRIA MODELAGEM

Table 2—M&S Life-Cycle Phase Descriptions

| Phase | Name | Brief Description |
|-------|---------------------------|---|
| Pre-A | Model Initiation | The process of determining the scope of the RWS on which to apply an M&S and defining the intended use of the M&S. |
| A | Model Concept Development | The process of compiling all relevant information about the scoped RWS and beginning to develop general modeling concepts and requirements for representing the RWS. |
| B | Model Design | The typically iterative process of creating the detailed, verifiable and validated specification of an M&S for an intended use, using the relevant information regarding the RWS, the conceptual model, and other defined objectives/criteria. The model design should be conceptually validated prior to commencing with Model Construction (Phase C). |
| C | Model Construction | The process/activity of implementing (generating or building) a usable model, as defined by its requirements, specifications (some of which may be embodied in a conceptual model/diagram), and intended use. |
| D | Model Testing | <p>Verification is the process of determining the extent to which an M&S is compliant with its requirements and specifications as detailed in its conceptual models, mathematical models, or other constructs.</p> <p>Validation is the process of determining the degree to which a model or a simulation is an accurate representation of the real world from the perspective of the intended uses of the M&S.</p> <p>Model Release is the process of establishing the baseline and controlled version of the model and associated key documentation for use. <i>After release, changes to the baseline are to be evaluated, justified, and authorized with traceability prior to implementing and releasing the revision.</i></p> |
| E | Model Use | The application of an M&S to the purpose for which it is intended. <i>This Phase begins with assessing a proposed use, preparing the model and scenarios for use or otherwise integrating the model into the simulation, using (e.g., running) the model, gathering and post-processing the output, and assessing and reporting the results.</i> |
| F | Model & Analysis Archival | The process of storing and cataloging all M&S and designated development and use artifacts for retrieval and use. |

| | | |
|----------|--------------------|--|
| | | <p>the typically iterative process of refining the RWS, and creating and validated specification of an M&S for an intended use, using the relevant information regarding the RWS, the conceptual model, and other defined objectives/criteria. The model design should be conceptually validated prior to commencing with Model Construction (Phase C).</p> |
| C | Model Construction | <p>The process/activity of implementing (generating or building) a usable model, as defined by its requirements, specifications (some of which may be embodied in a conceptual model/diagram), and intended use.</p> |
| D | Model Testing | <p>Verification is the process of determining the extent to which an M&S is compliant with its requirements and specifications as detailed in its conceptual models, mathematical models, or other constructs.</p> <p>Validation is the process of determining the degree to which a model or a simulation is an accurate representation of the real world from the perspective of the intended uses of the M&S.</p> <p>Model Release is the process of establishing the baseline and controlled version of the model and associated key documentation for use. <i>After release, changes to the baseline are to be evaluated, justified, and authorized with traceability prior to implementing and releasing the revision</i></p> |



AIVV PARA A MET. ARCADIA

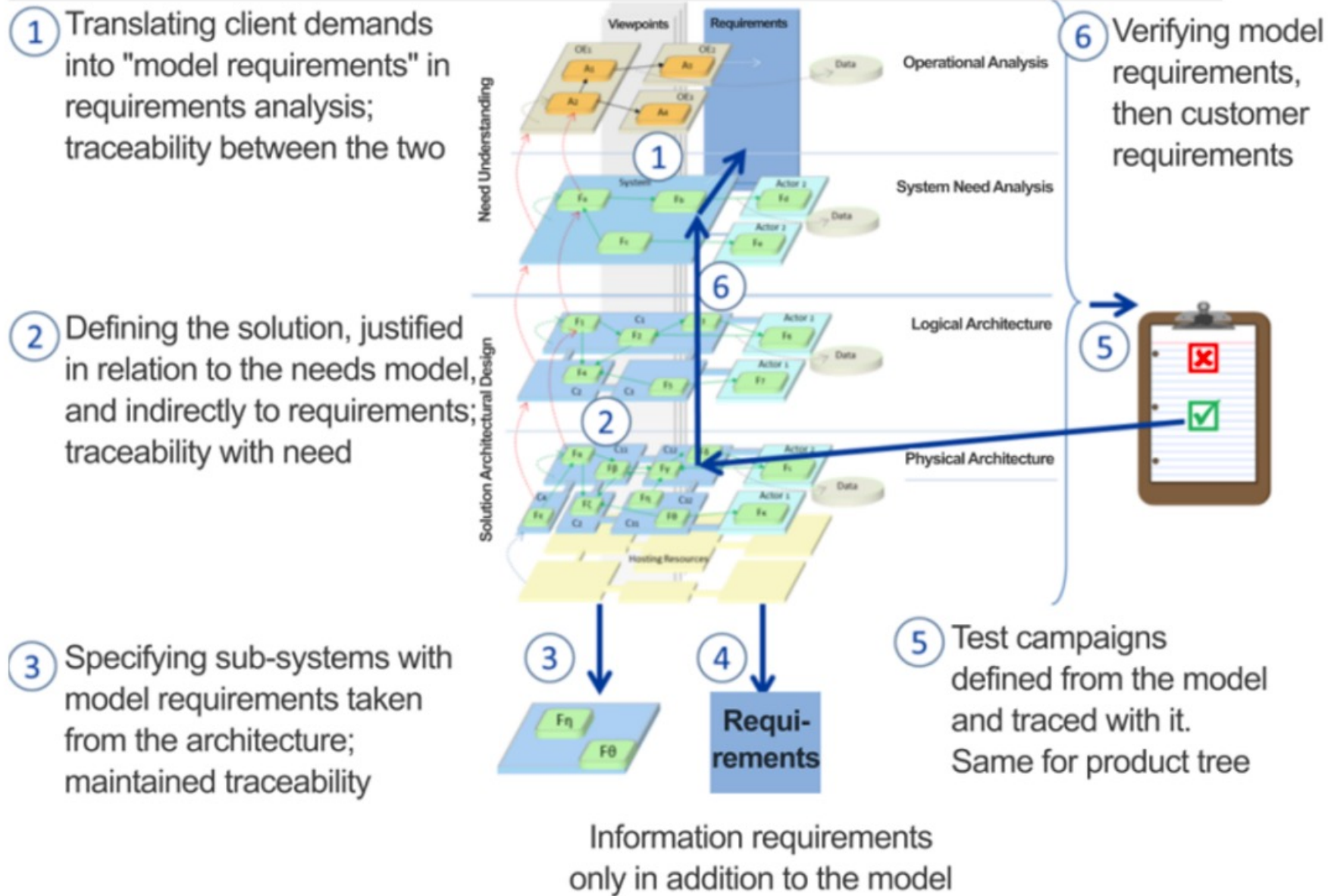


Figure 11.1. *The model, the principal support for requirements*

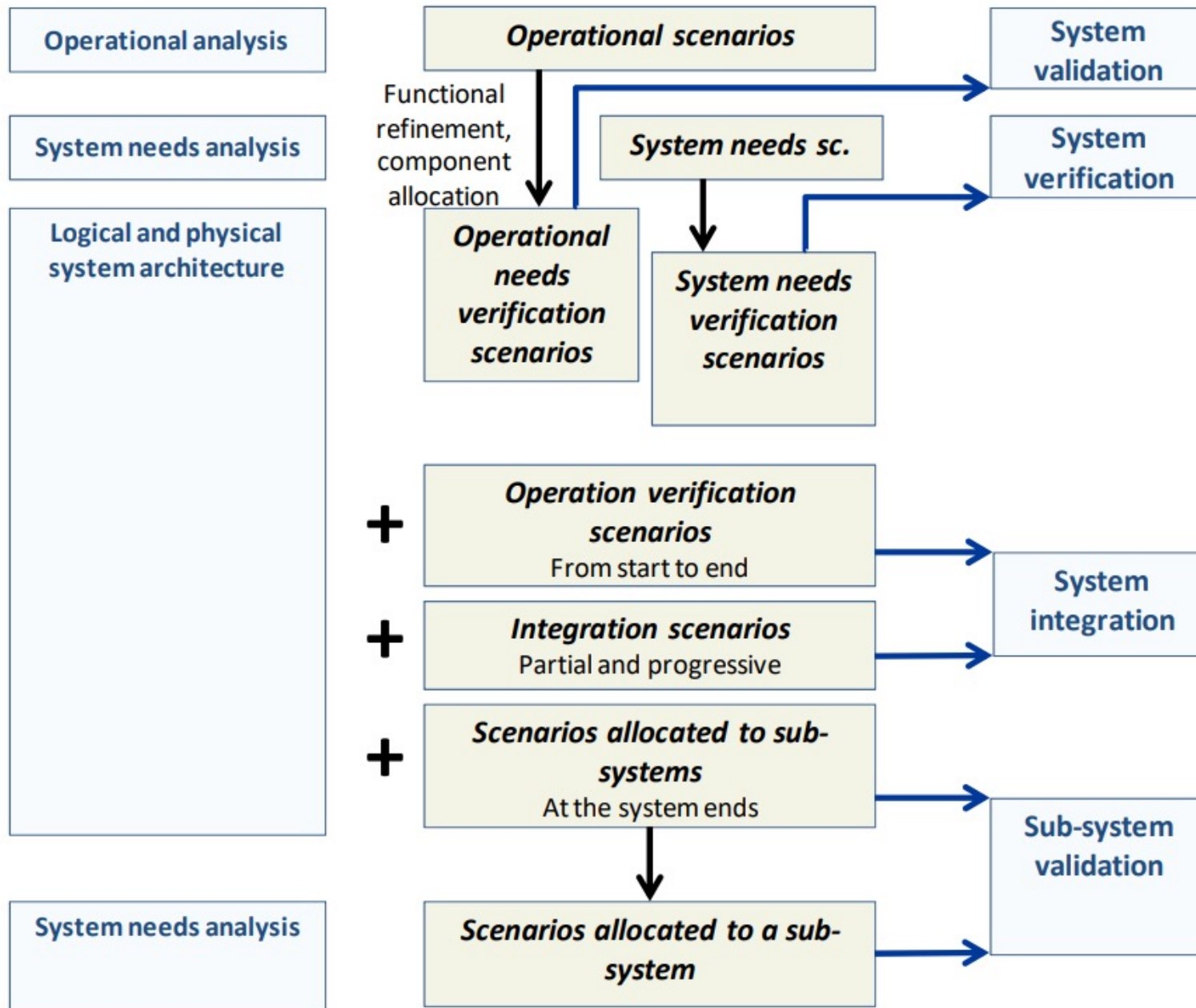


Figure 12.2. *Different types of scenario and their uses*



If you are designing a complex system, its **Validation & Verification** will probably be complex as well, distributed over :

- early validation virtual benches
- equipment level Test benches
- full system integration bench

In a context of tight plannings and costs optimization, it is crucial to define a proper **V&V Strategy** that identifies what shall be tested, when and which means are required.

And you probably can't afford to wait for the system to be built to start thinking of this strategy.

Our viewpoint allows you to start defining your V&V strategy and the different required Test Means as soon as Logical Architecture phase and later on in Physical Architecture.

Define the V&V strategy



NO FIM...

- Arcadia sugere basear o IVV principalmente no **uso de modelos como uma descrição de necessidades** e **soluções** para orientar e estruturar a estratégia IVV e sua implementação.
- Os requisitos textuais (funcionais) são "traduzidos" em requisitos de modelo em OA e SA. Estes são rastreados, através da construção do modelo, a partir da solução (LA e PA) por links de justificção.
- Capacidades, cenários e cadeias funcionais **estruturam versões de integração e entrega e campanhas de teste**, e garantem a verificação dos requisitos do modelo.
- Os requisitos textuais (não funcionais) permanecem quando o modelo não é apropriado para representá-los.



CONCEITOS

<https://www.m2doc.org/>



CONCEITOS

- O projeto M2Doc fornece a **geração de documentos** do Word (arquivos .docx) com base em um modelo de documento e modelos EMF.
- A abordagem geral consiste na criação de modelos no formato **OOXML** em que a criação de texto estático se beneficia dos recursos WYSIWYG do Microsoft Word. Partes dinâmicas são inseridas usando um vocabulário dedicado de código de campos OOXML.
- Os campos são usados principalmente para inserir números de página, referências, etc. O M2Doc faz uso de diretivas de geração de documentação. Isso permite uma separação total entre o documento e as diretivas M2Doc.





Tables

1.2 Tables-au-niveau-du-modèle

[m.for table | db.allTables]

1.2.1 Table- [m.table.name]

1.2.1.1 Table- [m.table.name]-description

| | |
|---------------|------------------------|
| Name | [m.table.name] |
| SGDB | [m.db.DBLibrary] |
| Record-number | [m.table.recordNumber] |

1.2.1.2 [m.table.name]-columns-list

| Name | Type |
|------|------|
|------|------|

[m.for column | table.columns]

| | |
|-----------------|---------------------|
| [m.column.name] | [m.column.typeName] |
|-----------------|---------------------|

[m.endfor]

[m.for column | table.columns]

1.2.1.3 Column- [m.column.name]-from-table- [m.table.name]

1.2.1.3.1 Column- [m.column.name]-description

| | |
|--------------------------|-------------------------|
| Nom | [m.column.name] |
| Type-de données | [m.column.typeName] |
| Obligatoire | [m.column.isMandatory] |
| Commentaire | [m.column.comments] |
| Clé-primaire | [m.column.isPrimaryKey] |
| Clé-étrangère | [m.column.isForeignKey] |
| Nombre-d'enregistrements | [m.column.recordNumber] |

[m.endfor]

.....Saut de page

Tables

1.2 Tables-au-niveau-du-modèle

1.2.1 Table-GS_SERIE

1.2.1.1 Table-GS_SERIE-description

| | |
|---------------|---------------|
| Name | GS_SERIE |
| SGDB | physicalTypes |
| Record-number | 10000 |

1.2.1.2 GS_SERIE-columns-list

| Name | Type |
|----------------------|----------|
| GS_SERIE_ID | INTEGER |
| RF_GENRE_ID | INTEGER |
| RF_PAYS_ID | INTEGER |
| GS_SERIE_NOM | VARCHAR2 |
| GS_SERIE_ANNEECRE | VARCHAR2 |
| GS_SERIE_ANNEEFIN | VARCHAR2 |
| GS_SERIE_DESCRIPTION | VARCHAR2 |
| GS_SERIE_LOGO | VARCHAR2 |
| GS_XTOPSUP | VARCHAR2 |
| GS_XDMAJ | DATE |

1.2.1.3 Column-GS_SERIE_ID-from-table-GS_SERIE

1.2.1.3.1 Column-GS_SERIE_ID-description

| | |
|--------------------------|-------------------------|
| Nom | GS_SERIE_ID |
| Type-de données | INTEGER |
| Obligatoire | Oui |
| Commentaire | PK-de la table-GS_SERIE |
| Clé-primaire | Non |
| Clé-étrangère | Oui |
| Nombre-d'enregistrements | 5000 |

1.2.1.4 Column-RF_GENRE_ID-from-table-GS_SERIE

1.2.1.4.1 Column-RF_GENRE_ID-description

| | |
|-----------------|-------------|
| Nom | RF_GENRE_ID |
| Type-de données | INTEGER |
| Obligatoire | Oui |

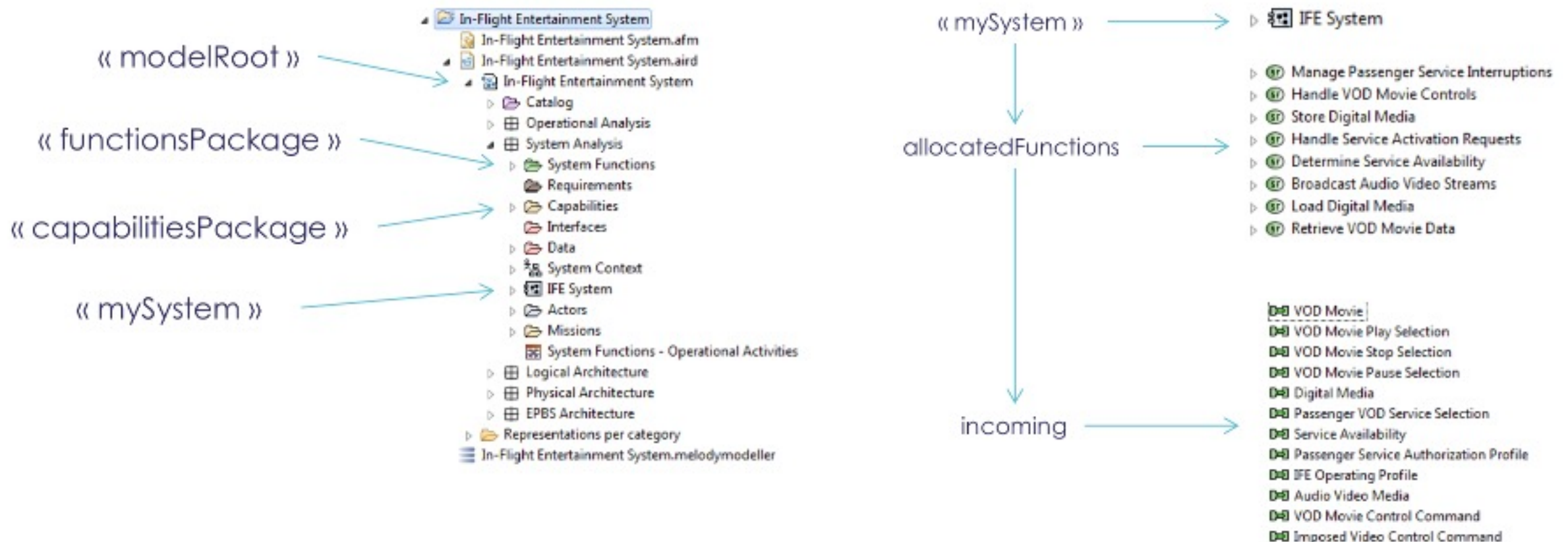


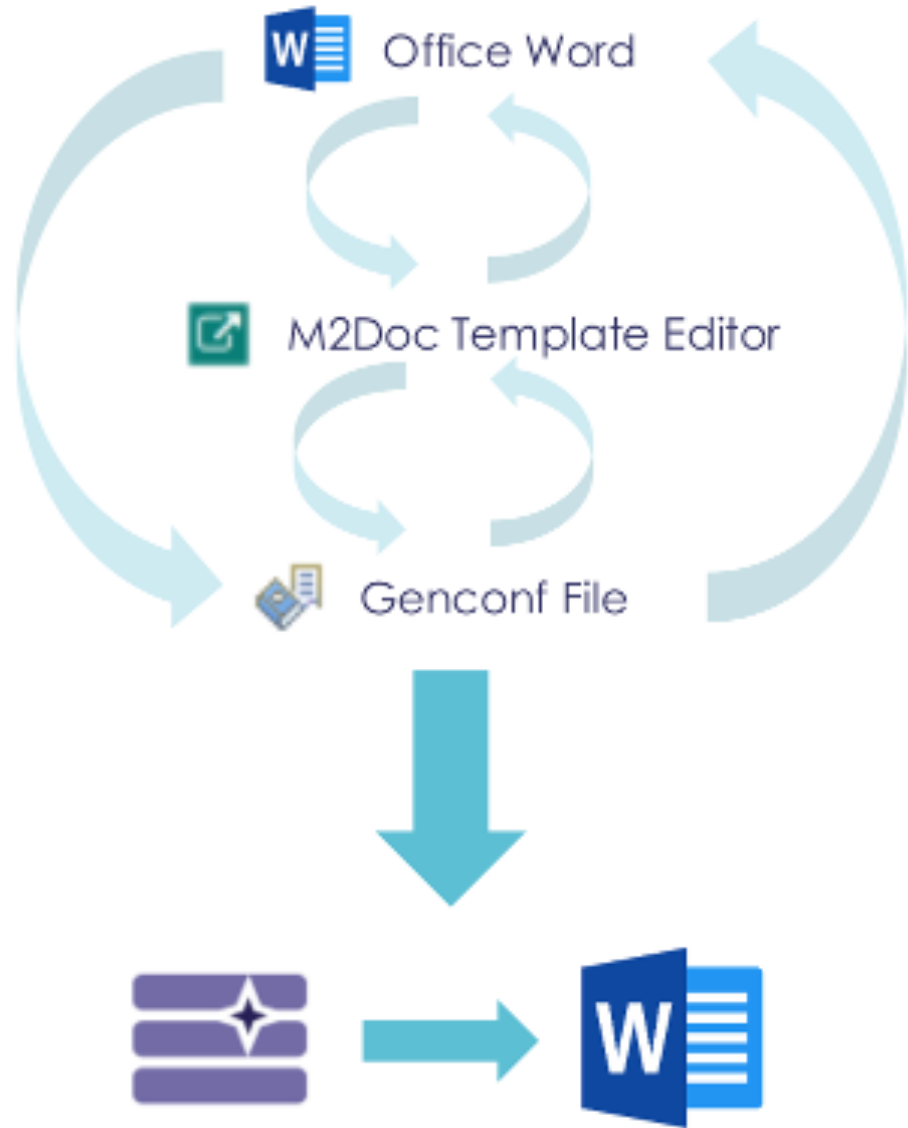
- A linguagem de modelo faz um uso extensivo da **Acceleo Query Language**, que fornece uma linguagem de consulta de modelo.
- Os modelos M2Doc podem ser validados. Se forem encontrados erros, um modelo anotado será produzido descrevendo os problemas encontrados.



PRINCÍPIOS

- Definição dos pontos de entrada do modelo.
- Extração de informações navegando no modelo.





■ Definition of content, navigation, and format

■ Declaration of variables

■ Mapping of variables with model elements, definition of input model and output file

■ Generation of output document



NÃO É FÁCIL E NECESSITA DE PESSOAS ESPECÍFICAS:

- **Template user:** que já possuem o modelo e desejam gerar o documento.
 - Edita modelos e quer produzir documentos
- **Template developer:** que querem criar seu próprio modelo
 - Templates podem ser usados em diferentes documentos.
 - Varios templates podem ser usados em um modelo.
 - Conhece a estrutura de navegação AQL
- **Integrator (Meta-model Expert):** que desejam fornecer geração de documentos em seu próprio projeto usando M2Doc.
 - Cria serviços AQL para criadores de templates.



BÁSICO DE METAMODELAGEM

<https://www.eclipse.org/modeling/emf/>



- EMF : Eclipse Modeling Framework
 - Modelagem de dados Java e framework de integração
- Fornece a infraestrutura para realmente usar os modelos em um aplicativo
- Facilmente acessível
- Open source – EPL licence
 - Necessário conhecimento em Java
 - Aliás também em UML, XML Schema
 - Baseado em modelagem > geração de código
- Possibilita a produção rápida de modeladores
- Utiliza toda a facilidade trazida pelo Eclipse



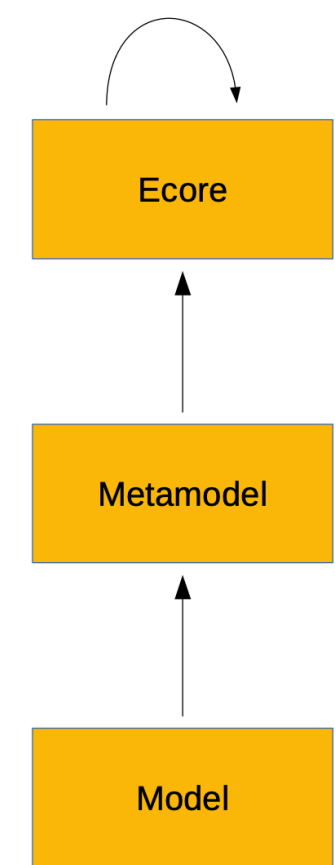
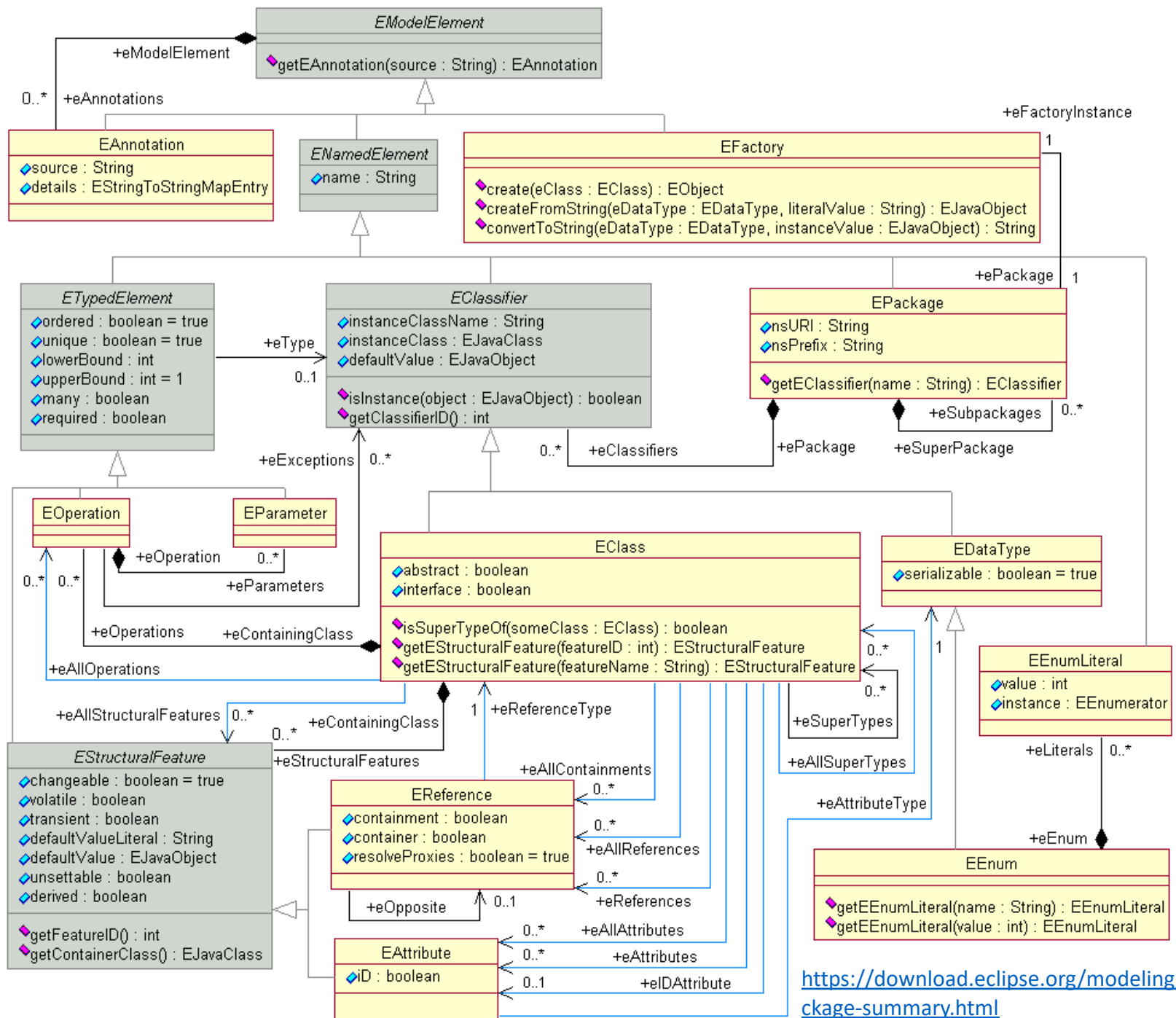
META MODELO

- Um modelo EMF é um conjunto de dados
 - Conformação a um metamodelo
 - Composto por objetos com propriedades e relações
- Serializado em um arquivo XMI
- Conceitos próximos aos do diagrama de classes UML
 - Elements ~ Class
 - Attributes
 - References ~ Associations



BENEFÍCIOS

- Um link entre o mundo do desenvolvimento e o mundo da modelagem
 - Converte modelos em código e vice-versa
 - Fornece toda a infraestrutura necessária de M / MM / M2M
- Gratuito
 - Licença EPL
 - Poucos pré-requisitos
 - Pode operar no modo autônomo (sem acesso externo)
- Estabilidade
 - Desenvolvido desde 2002
 - No centro da infraestrutura do Eclipse



<https://download.eclipse.org/modeling/emf/emf/javadoc/2.9.0/org/eclipse/emf/ecore/package-summary.html>



EMF Tutorial

🕒 24 min Read

What every Eclipse developer should know about EMF

This tutorial is an introduction to EMF and explains the basics of EMF. We start by showing you how to build a very simple data-centric application, including the UI, based on EMF. We explain how to define a model in EMF and generate code from it. We explore the API of the generated code, that is, how to create, navigate and modify model instances.

Next we demonstrate how to build a UI based on this model using databinding. For our example, we build an application to manage a bowling league, including matches and players. Later on in the tutorial, we explore the advantages of using AdapterFactories and briefly look at data management in EMF. We also include a few pointers on the most important add-on technologies for EMF. If you are interested in getting fast results building an application based on EMF, maybe [EMF Client Platform](#) is also a good starting point for you, see [this tutorial](#).

PDF Download: This tutorial is also available for download as a PDF [on our website](#).

Installation Requirements: To work through the examples, you'll need to download and install a fresh version of the Eclipse Modeling Tools from the [Eclipse Download Page](#).

Introduction



ACCELEO QUERY LANGUAGE

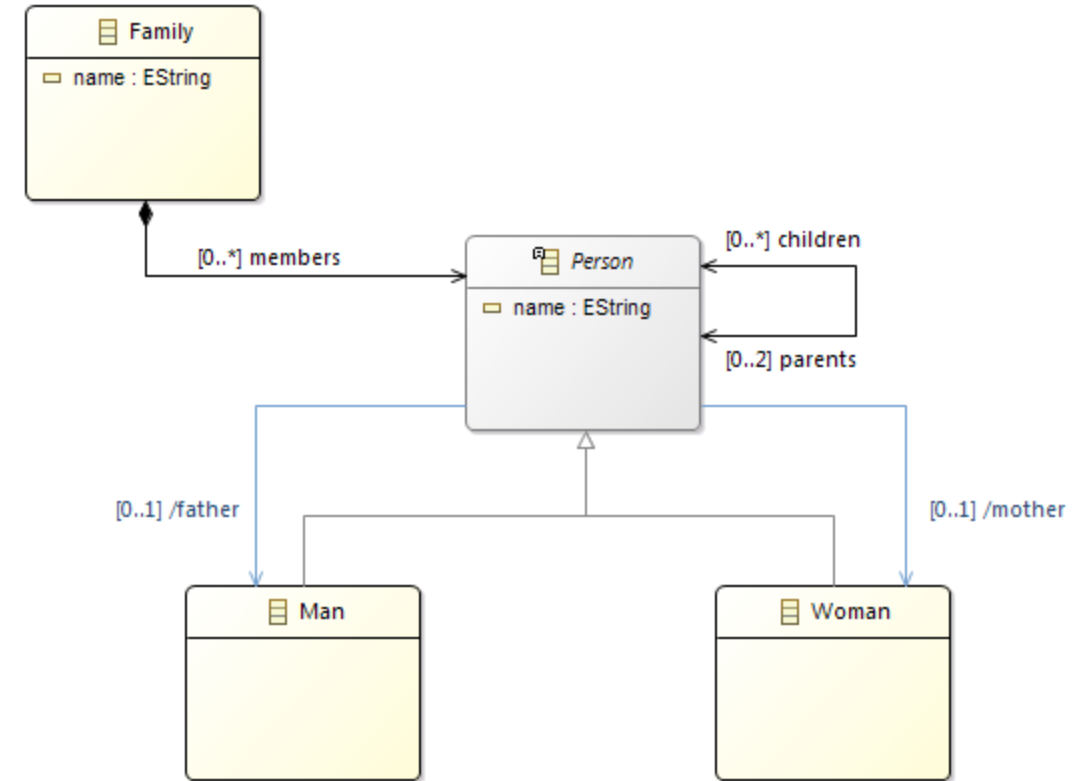


INTRODUÇÃO

- A Acceleo Query Language (AQL) é uma linguagem usada para navegar e consultar um modelo EMF.
- O AQL como mecanismo de consulta é pequeno, simples, rápido, extensível e traz uma validação mais rica do que o interpretador MTL
- O interpretador AQL é usado no Sirius com o prefixo «aql:»



- From a variable one can access field or reference values using the . separator.
- With self being an instance of Person, self.name returns the value of the attribute name and self.father return the father of the person.
- If the attribute or the reference is multi-valued, then self.parents will return a collection.
- Calls can be chained, as such self.parents.name will return a collection containing the names of the parents.
- If one want to access the collection itself, then the separator -> must be used, as such self.parents.name->size() will return the number of elements in the collection whereas self.parents.name.size() will return a collection containing the sizes of each name.





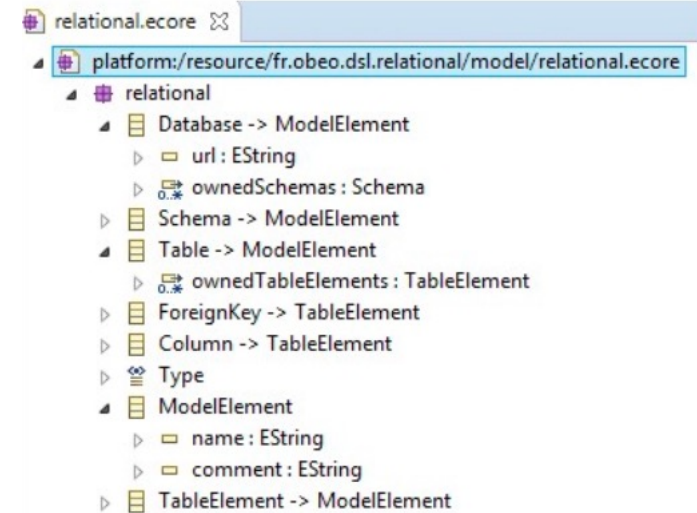
SINTAXE

- A noção de contexto
 - Uma consulta AQL se aplica ao seu contexto (denotado por “self”)
- A navegação está em conformidade com o metamodelo usando a notação de pontos
 - Acesso a referências e atributos
 - Exemplo : `aql:self.ownedSchemas.name`
 - No contexto de um Banco de Dados, recupere os nomes dos esquemas de propriedade do banco de dados atual.



SINTAXE

- Outros exemplos de consulta
 - `aql:self.name` in a Table context → Name of the Table
 - `aql:self.ownedTableElements.name` in the context of Table → Column names list
 - `aql:self.ownedTableElements→size()` in the context of a Table → Number of columns of the Table
- Elementos estáticos vs elementos dinâmicos
 - `aql:'Table_' + self.name` in the context of a Table named 'Vehicle' → "Table_Vehicle"
 - `aql:'Prefixe' + self.name + 'Sufixe'` → "PrefixeVehicleSufixe"





ESTRUTURANDO AS BUSCAS: INTERPRETER

- Disponível em Window > Show View > Interpreter
- Trabalha com representações do Sirius
- Mas também em todos os modelos EMF
- Muito útil para ajustar consultas



Acceleo Query Language

Query and navigate in EMF models

Overview

The Acceleo Query Language (AQL) is a language used to navigate and query an EMF model. In this document, you will find the description of all the services of the standard library of AQL.

Introduction

The Acceleo Query Language (AQL) is a language used to navigate and query an EMF model. In this document, you will find the description of the syntax, all the services and the standard library of AQL.

AQL as a query engine is small, simple, fast, extensible and it brings a richer validation than the MTL interpreter.

For those looking for a simple and fast interpreters for your EMF models, AQL can provide you with a lot of features, including:

- Support for static and dynamic Ecore models, no query compilation phase required.
- The least possible overhead at evaluation time. During this phase, the evaluation goes forward and will not even try to validate or compile your expressions. Errors are tracked and captured along the way.
- Strong validation: types are checked at validation time and the metamodels used are analyzed to do some basic type inference.

<https://www.eclipse.org/acceleo/documentation/>



(SUPER) PEQUENO EXEMPLO...

<https://www.m2doc.org/>



INSTALAÇÃO

- Primeiro você precisa baixar Capella.
- Quando o download estiver concluído, extraia o arquivo baixado e execute o executável Eclipse na subpasta Eclipse.
- **INSTALLATION FOR CAPELLA 6.1.X**
 - https://s3-eu-west-1.amazonaws.com/obeo-networkaggregation-releases/capella-extensions/6.1.0_M2Doc3.3.0/full [zip](#) (M2Doc 3.3.0)



INSTALANDO O M2DOC

Workflow of In-Flight Entertainment System

- Operational Analysis**
- System Analysis**
- Logical Architecture**

Available Software

Select a site or enter the location of a site.

Work with: type or select a site

type filter text

| Name | Version |
|----------------------------|---------|
| There is no site selected. | |

Add Repository

Name: Local...

Location: http:// Archive...

| Name | Size | Kind | Date |
|---|----------|-------------|-------|
| capella-6.1.0.202303291413-macosx-cocoa-aarch64.tar | 860,8 MB | tar archive | Today |
| Capella.app | 974,4 MB | Application | Today |
| org.obeonetwork.capella.update.full.zip | 68 MB | ZIP archive | Today |

Available Software

Check the items that you wish to install.

Work with: Capella Extensions 6.1.0.202305021134 - jar:file:/Users/christopherc Pereira/Documents/DEVEL/org.obeonetwork...

type filter text

| Name | Version |
|---|---------|
| <input checked="" type="checkbox"/> > Documentation and Code generators | |
| <input checked="" type="checkbox"/> > Uncategorized | |

Select a wizard

Creates a sample project based on the IFE project , with M2Doc associated Word templates.

Wizards:

type filter text

- Class
- Interface
- Java Project
- Plug-in Project
- General
- Capella
- Capella - M2Doc
- Capella IFE example with M2Doc templates project**
- Eclipse Modeling Framework
- Example EMF Model Creation Wizards
- Git
- Java
- Java Emitter Templates
- Kitalpha
- M2Doc
 - M2Doc project
 - New Generation
 - New template
- MDE Toolkit
- Sirius
- Tasks
- Examples



ABRINDO O EXEMPLO

- ▼ In-Flight Entertainment System With M2Doc
 - ▼ generated
 - LA_Complete.docx
 - SA_Complete.docx
 - In-Flight Entertainment System.afm
 - > In-Flight Entertainment System.aird
 - In-Flight Entertainment System.capella
 - ▼ template
 - Template LA Complete.docx
 - Template SA Complete.docx
 - Template LA Complete.genconf
 - Template SA Complete.genconf

Arquivos que são gerados

Templates de textos

Configuradores



Specification File of `m:self.name`

3 → Description of the system Missions

```
m:if self.containedSystemAnalysis.ownedMissionPkg.eAllContents(ctx::Mission)->size() > 0
m:if self.containedSystemAnalysis.ownedMissionPkg.eAllContents(ctx::MissionPkg)->size() > 0
```

The system Missions are sorted in the following packages:

| Mission package | Missions |
|--|---|
| <code>m:self.containedSystemAnalysis.ownedMissionPkg.name</code> | <code>m:for mission self.containedSystemAnalysis.ownedMissionPkg.ownedMissions</code> <code>m:m:mission.name.asBookmarkRef(mission.id)</code> <code>m:endfor</code> |

```
m:for package | self.containedSystemAnalysis.ownedMissionPkg.eAllContents(ctx::MissionPkg)
```

| <code>m:package.name</code> | <code>m:for mission package.ownedMissions</code> |
|-----------------------------|--|
| | <code>m:m:mission.name.asBookmarkRef(mission.id)</code> <code>m:endfor</code> |

```
m:endfor
m:endif
m:for mission | self.containedSystemAnalysis.ownedMissionPkg.eAllContents(ctx::Mission)
```

3.1 → Mission: `m:m:mission.name.asBookmark(mission.id)`

```
Description:
m:if mission.description.trim().size() <= 0
m:m:mission.description.trim().fromHTMLBodyString().replaceLink(mission)
m:else
No description
m:endif
```

```
Involvement actors:
m:if mission.involvedSystemComponents->select(sc | sc.actor)->size() > 0
m:for actor | mission.involvedSystemComponents->select(sc | sc.actor)
```

Specification File of In-Flight Entertainment System

3 → Description of the system Missions

3.1 → Mission: Provide Entertainment Solutions

```
Description:
No description
Involved actors:
REF c8b78c78-5b11-4fc0-87b7-3ca84622efea \h
REF 181a678c-dca9-46c1-9d18-b5a0c457c0de \h
Exploited capabilities:
3.1.1 → Capability: Provide Moving-Map Services
Description:
```

[Wikipedia]
A moving-map system is a real-time flight information video channel broadcast through to cabin project/video screens and personal televisions (PTVs). In addition to displaying a map that illustrates the position and direction of the plane, the system gives the altitude, airspeed, outside air temperature, distance to the destination, distance from the origination point, and local time. The moving-map system information is derived in real time from the aircraft's flight computer systems.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|-------------------------|-----------------------------|------------------------|
| No including capability | Provide Moving-Map Services | No included capability |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|------------------------|-----------------------------|-------------------------|
| No extended capability | Provide Moving-Map Services | No extending capability |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|--------------------|--------------------|------------------|
| | | |



Specification File of In-Flight Entertainment System

3 → Description of the system Missions

3.1 → Mission: Provide Entertainment Solutions

```
Description:
No description
Involved actors:
Aircraft
Passenger
Exploited capabilities:
3.1.1 → Capability: Provide Moving-Map Services
Description:
```

[Wikipedia]
A moving-map system is a real-time flight information video channel broadcast through to cabin project/video screens and personal televisions (PTVs). In addition to displaying a map that illustrates the position and direction of the plane, the system gives the altitude, airspeed, outside air temperature, distance to the destination, distance from the origination point, and local time. The moving-map system information is derived in real time from the aircraft's flight computer systems.

Capability inclusion relations:

| Including capabilities | Current capability | Included capabilities |
|-------------------------|-----------------------------|------------------------|
| No including capability | Provide Moving-Map Services | No included capability |

Capability extension relations:

| Extended capabilities | Current capability | Extending capabilities |
|------------------------|-----------------------------|-------------------------|
| No extended capability | Provide Moving-Map Services | No extending capability |

Capability generalization relations:

| Super capabilities | Current capability | Sub capabilities |
|--------------------|--------------------|------------------|
| | | |



CONSIDERAÇÕES FINAIS



ATIVIDADES PARA A PRÓXIMA AULA

- Voltamos para o nosso exercício:



PROPOSTA DE MISSÃO

1. *Os viajantes do tempo Doc Emmett Brown e Marty McFly não podem revelar a existência de máquinas do tempo pois isso geraria um conflito pela posse da tecnologia.*
2. *Ações específicas na linha do tempo podem ocasionar infinitas linhas temporais acarretando no fim da estrutura do universo.*
3. *Desta forma, eles precisam de meios para estruturar um **sistema logístico seguro** de peças para uma montagem/manutenção ágil das máquinas do tempo, em cada uma das épocas, apoiado por ações de **inteligência, vigilância e reconhecimento**.*
4. *É preciso que se **monitorem os acontecimentos do entorno** do local de desenvolvimento, movimentação de pessoas, eventos, e outros fluxos para criar uma previsão de acontecimentos.*
5. *Deve ser **feito com os recursos disponíveis** em cada uma das épocas, de forma a não levantar suspeitas e colaborar com a ocultação do transito de peças.*





ATIVIDADES PARA A PRÓXIMA AULA

- Adicionar (i) uma proposta de caso de teste no logical architecture e (ii) como o equipamento de fato foi instrumentado no physical architecture.
- Gerar um doc com no mínimo uma arquitetura de cada camada, trazer o título do diagrama e o diagrama automaticamente.

