



FORÇA AÉREA BRASILEIRA

Asas que protegem o País



1ª SEMANA DE APLICAÇÕES OPERACIONAIS AO PREPARO E EMPREGO



REALIZAÇÃO



APOIO



PARCEIROS



PATROCÍNIO



PROJECT PERFORMANCE
INTERNATIONAL



PROF. DR. CHRISTOPHER SHNEIDER CERQUEIRA



- Engenheiro da Computação pela UNIFEI (2010)
- Mestre e Doutor em Engenharia e Gerenciamento de Sistemas Espaciais pelo INPE (2014 and 2018)
- Atualmente professor do ITA e colaborando no Centro Espacial ITA
- Tópicos de Interesse:
 - Engenharia Simultânea,
 - Engenharia de Sistemas (baseada em Modelos),
 - Interação homem-máquina,
 - Artefatos tangíveis, Realidade Aumentada e Realidade Virtual,
 - Indústria 4.0
- Lattes: <http://lattes.cnpq.br/6516110599667954>
- E-mail: chris@ita.br
- Website: www.cscerqueira.com.br // not well updated







CONCEPTIO

Laboratório de Concepção de Sistemas

LADS-01

Laboratório Aberto de Desenvolvimento de Sistemas 01

LCOS

Laboratório de Controle e Operação de Satélites

LISA

Laboratório de Integração de Sistemas Aeroespaciais
Sala limpa / Bancada electrónica / Bancada de montagem / Simuladores Térmicos

ESTOQUE

SALA DE REUNIÕES

LMOE

Laboratório de Monitoramento de Objetos Espaciais

LADS-03

Laboratório Aberto de Desenvolvimento de Sistemas 03

LADS-02

Laboratório Aberto de Desenvolvimento de Sistemas 02

SALA DE PESQUISA

Sala para Pesquisadores

LADS-04

Laboratório Aberto de Desenvolvimento de Sistemas 04

10870mm





FORÇA AÉREA BRASILEIRA

Asas que protegem o País



SIGE 2022

SIMPÓSIO DE APLICAÇÕES OPERACIONAIS EM ÁREAS DE DEFESA

ENGENHARIA DE SISTEMAS BASEADA EM MODELOS

Prof. Dr. Christopher Shneider Cerqueira

2022-09-28

REALIZAÇÃO



APOIO



PARCEIROS

Spectrum

KEYSIGHT

EDASIM
INTEGRATING IDEAS

PATROCÍNIO



ROHDE & SCHWARZ
Make ideas real

Tektronix
A Tektronix Company

MAK
TECHNOLOGIES



VISIONA
Tecnologia Espacial



OBJETIVO

Introduzir os conceitos norteadores da Engenharia de Sistemas baseada em Modelos e sua relação com o ciclo de vida da DCA-400-6





AGENDA

MOTIVAÇÃO

~30min

INTRODUÇÃO

~30min

ENGENHARIA DE SISTEMAS
BASEADA EM MODELOS

DCA-400-6

Fase de Definição

MMMF

ENGENHARIA SIMULTÂNEA

CONSIDERAÇÕES FINAIS





MOTIVAÇÃO

~10min

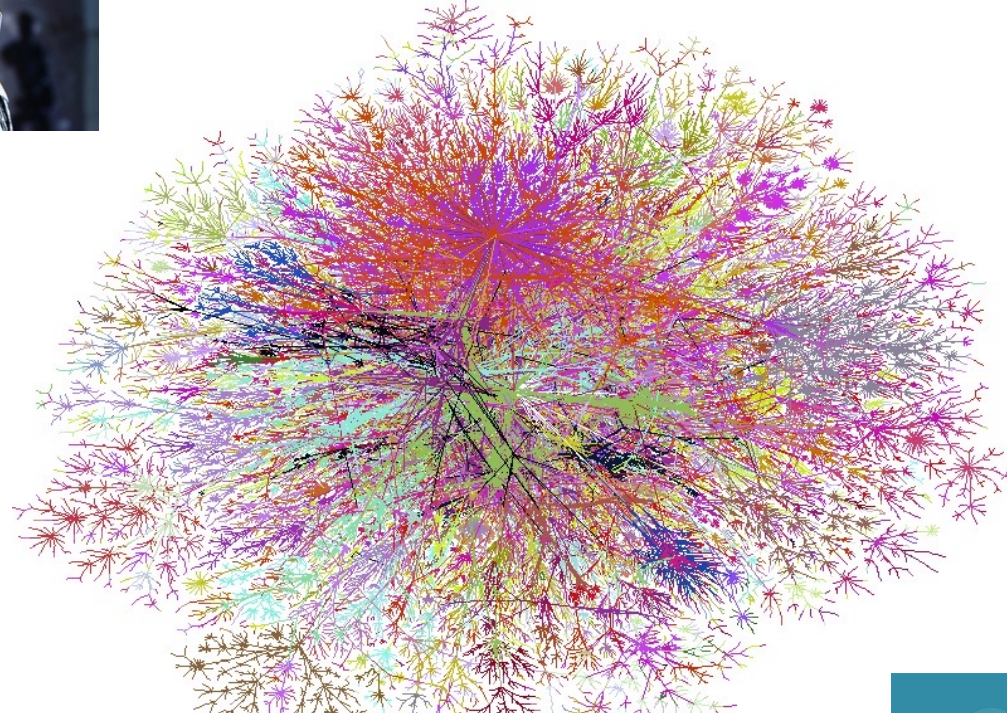
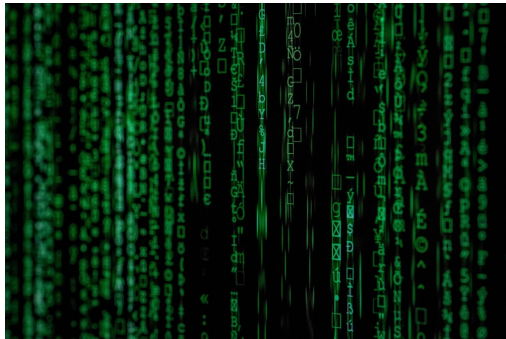




Ending scene from 1971 movie THX 1138



<https://www.youtube.com/watch?v=atMdf0rhbpI>



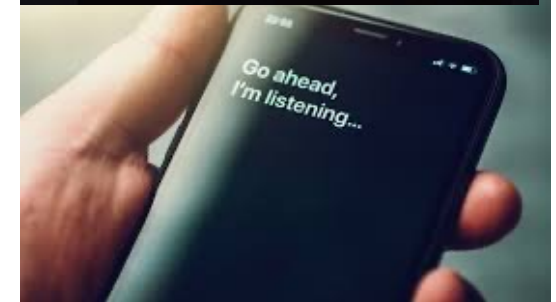
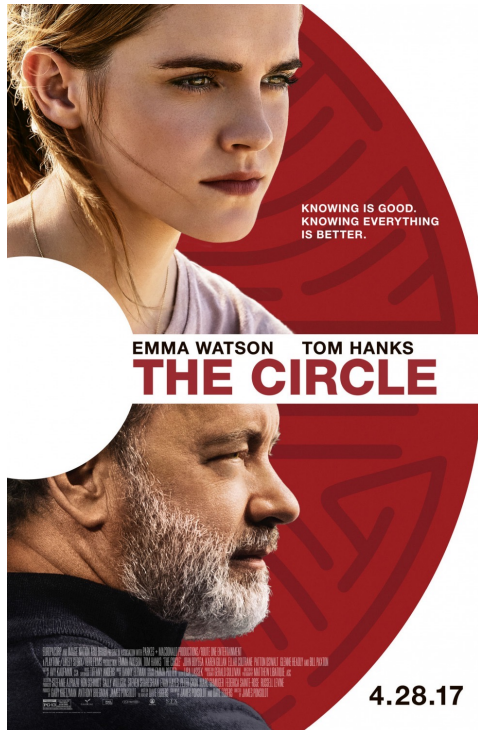
<https://ioannodolga.blog/2017/09/14/complexity-theory-ii-m-woermann/>



HACKERS INVADEM JEEP

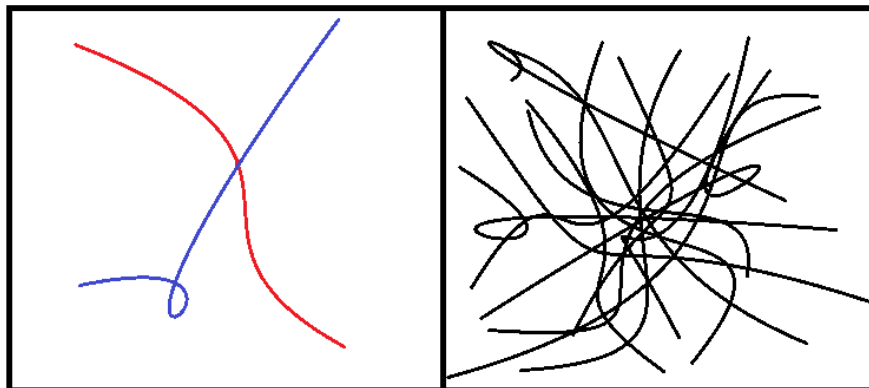


MUNDO INTERCONECTADO



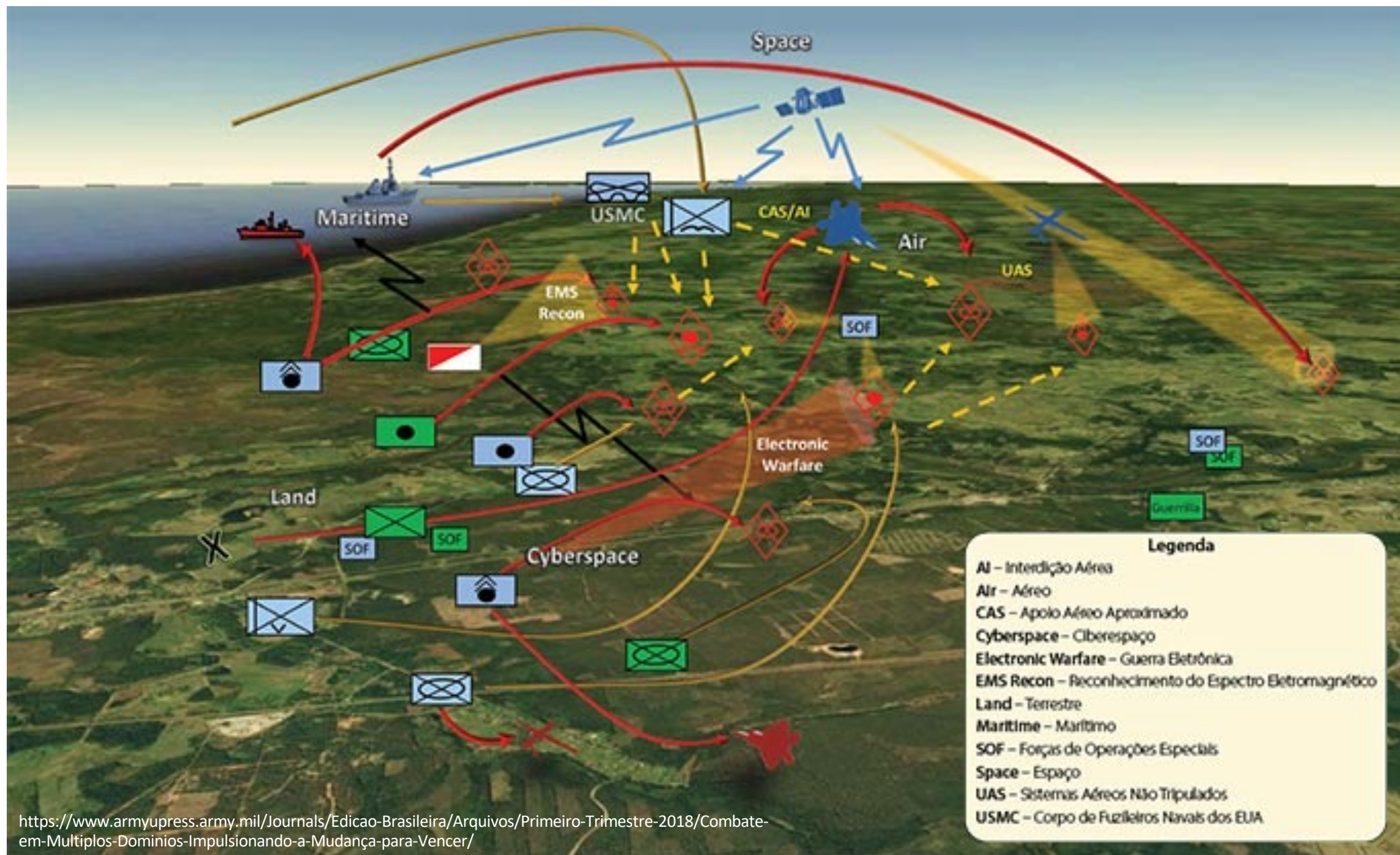
PROBLEMAS COMPLEXOS

- **Problemas complexos** são problemas que são muito difíceis (ou impossíveis) de resolver.
 - **Emergem** de sistemas/situações complexas.
- O termo "wicked" vem da **resistência à uma resolução**.



Traditional Problem

Wicked Problem





INTRODUÇÃO

~30min





Introdução

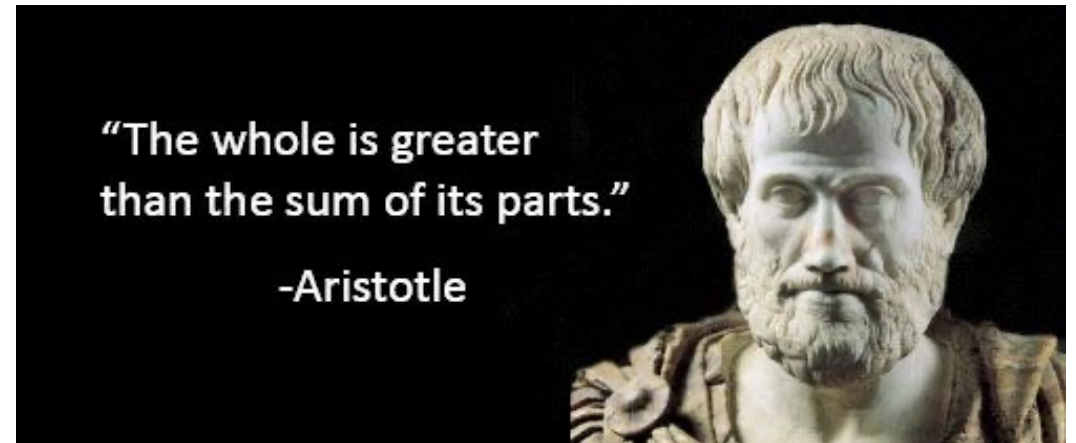
- Sistema
- Engenharia de Sistemas
 - Ciclo de Vida
 - CONOPs e Arquitetura
 - Requisitos e VV
- Engenharia de Sistemas baseada em Modelos
 - Linguagem
 - Modelos
- Engenharia Simultânea





SISTEMA





"... that the whole is not the same as the sum of its parts are useful in meeting the type just described; for a man who defines in this way seems to assert that the parts are the same as the whole. The arguments are particularly appropriate in cases where the process of putting the parts together is obvious, as in a house and other things of that sort: for there, clearly, you may have the parts and yet not have the whole, so that parts and whole cannot be the same." [4]

**"The System is something besides,
and not the same, as its elements."**

[4] Aristotle 100a, Topics, Translated by W. D. Ross

Definição de Sistema

Um sistema é

<um conjunto, uma combinação, um grupo, uma coleção, uma configuração, um arranjo, uma organização, uma montagem, [10]>

de

<partes, componentes, elementos, objetos, subsistemas, entidades [6]>

, que

<combinados, integrados, organizados, configurados, organizados [5]>

de uma forma,

<criam, habilitam, motivam [3]>

<proriedades, funções, processos, capacidades, comportamentos, dimensões [6]>

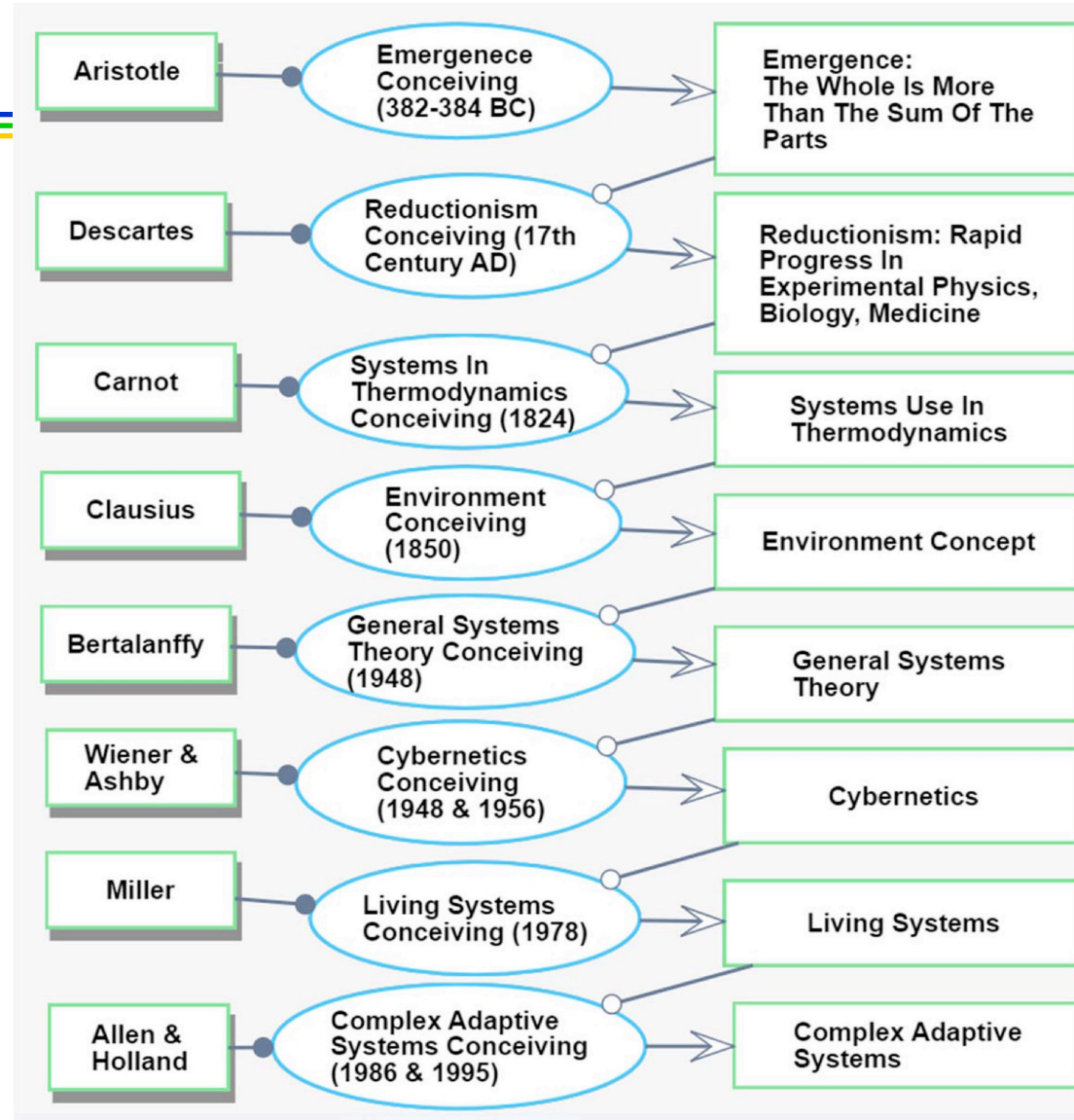
que não

<possuiam, exibiam, apresentados [3]>

por suas

<partes, componentes, elementos, objetos, subsistemas, entidades [6]>

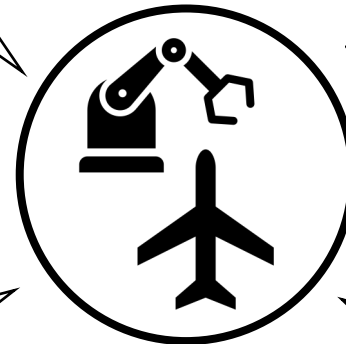
<em separado, individualmente [3]>



Características de um sistema

Funções e outras características emergem com as interações, guiadas pela forma dos relacionamentos. Essa emergencia é que define o sistema

Sistemas possuem forma e função, a forma é um instrumento da função.



Entidades estão ligadas através de relacionamentos que possuem também forma e função. Algumas entidades se relacionam com outras entidades fora da fronteira do sistema.

Sistemas são compostos de entidades, cada uma com sua forma e função. Essas entidades também são sistemas, e o próprio sistema pode ser uma entidade de outro sistema.

Princípio da Emergência

Quando as entidades do sistemas são conectadas, suas interações vão causar funções, comportamentos, performances e outras propriedades que irão emergir.



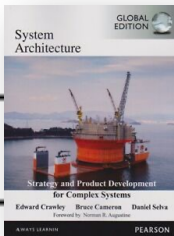
[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)



- **Emergência é o “*poder e a mágica*” dos sistemas.** Ela se refere a que aparece, se materializa, ou transborda quando o sistema existe. **Obter essa emergência é o motivo pela qual fazemos sistemas.**
- **Entender a emergência é o objetivo – e talvez a arte – do pensamento sistêmico.**
- O que emerge geralmente são: **funções/funcionalidades.**
- Funções são as “coisas” que o sistema deve fazer. Nós devemos projetar sistemas, antecipando as funções que irão emergir.

TABLE 2.1 | Types of emergent functions

	Anticipated Emergence	Unanticipated Emergence
Desirable	Cars transport people Cars keep people warm/cool Cars entertain people	Cars create a sense of personal freedom in people
Undesirable	Cars burn hydrocarbons	Cars can kill people



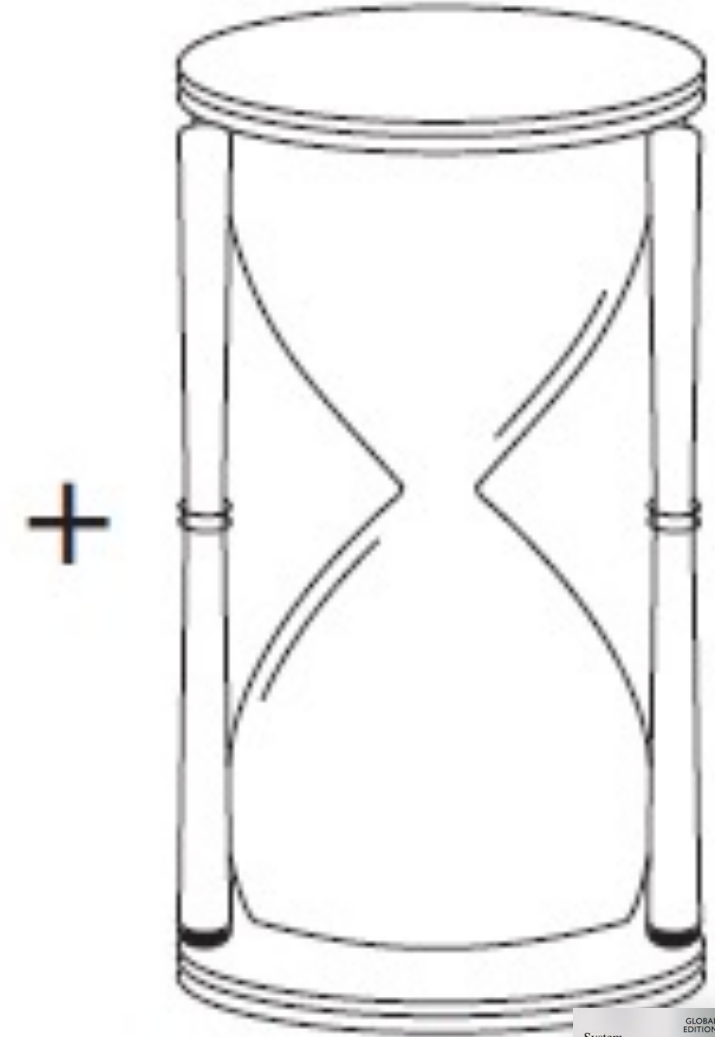


FIGURE 2.1 Emergent function from sand and a funnel: Time keeping. (Source: LOOK D
Bildagentur der Fotografen GmbH/Alamy)

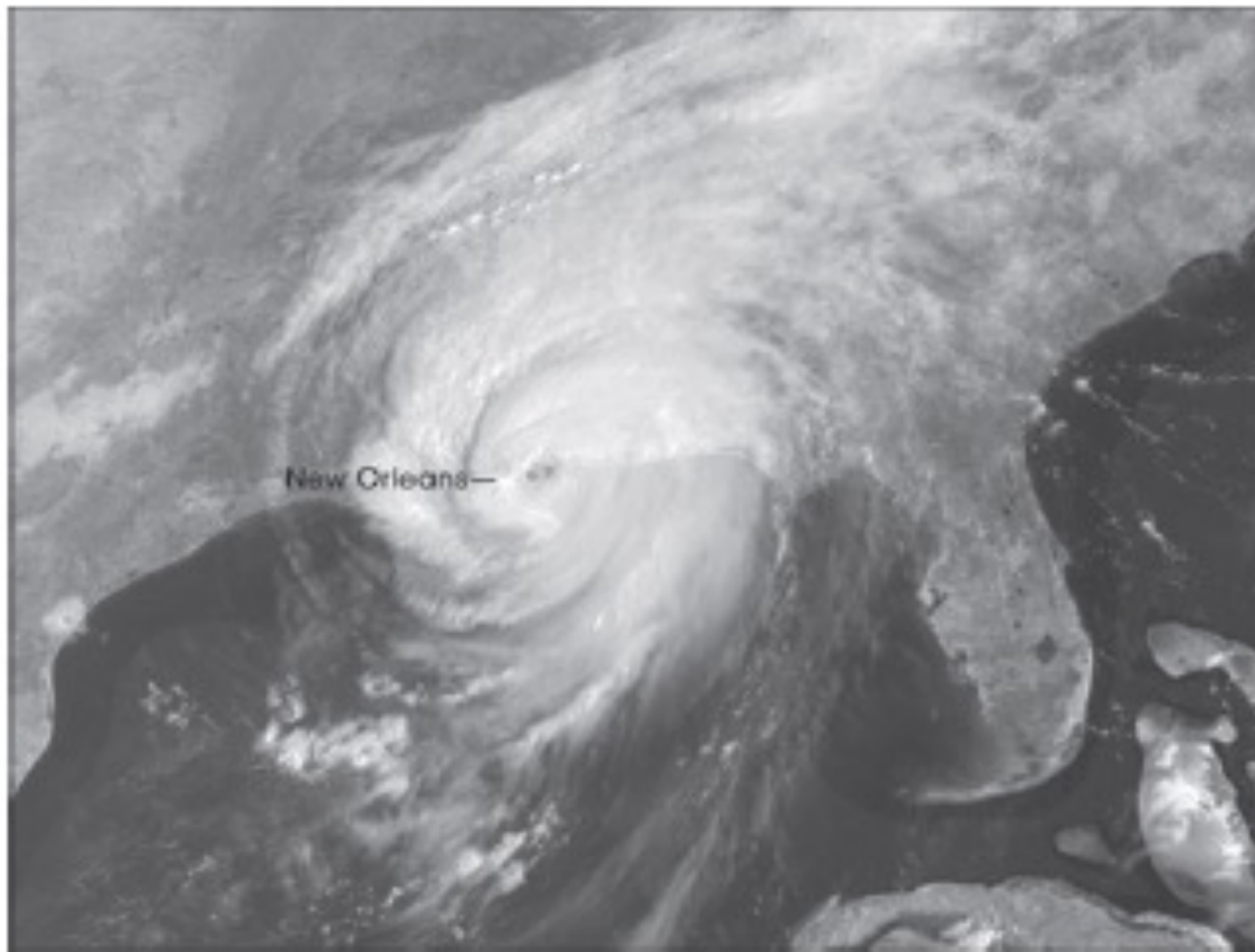


FIGURE 2.3 Emergency as emergence: Hurricane Katrina. (Source: Image courtesy GOES Project Science Office/NASA)



ENGENHARIA DE SISTEMAS



O que é Engenharia de Sistemas?

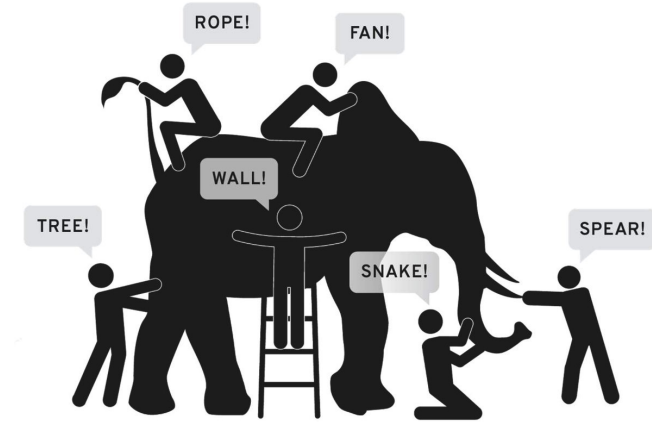
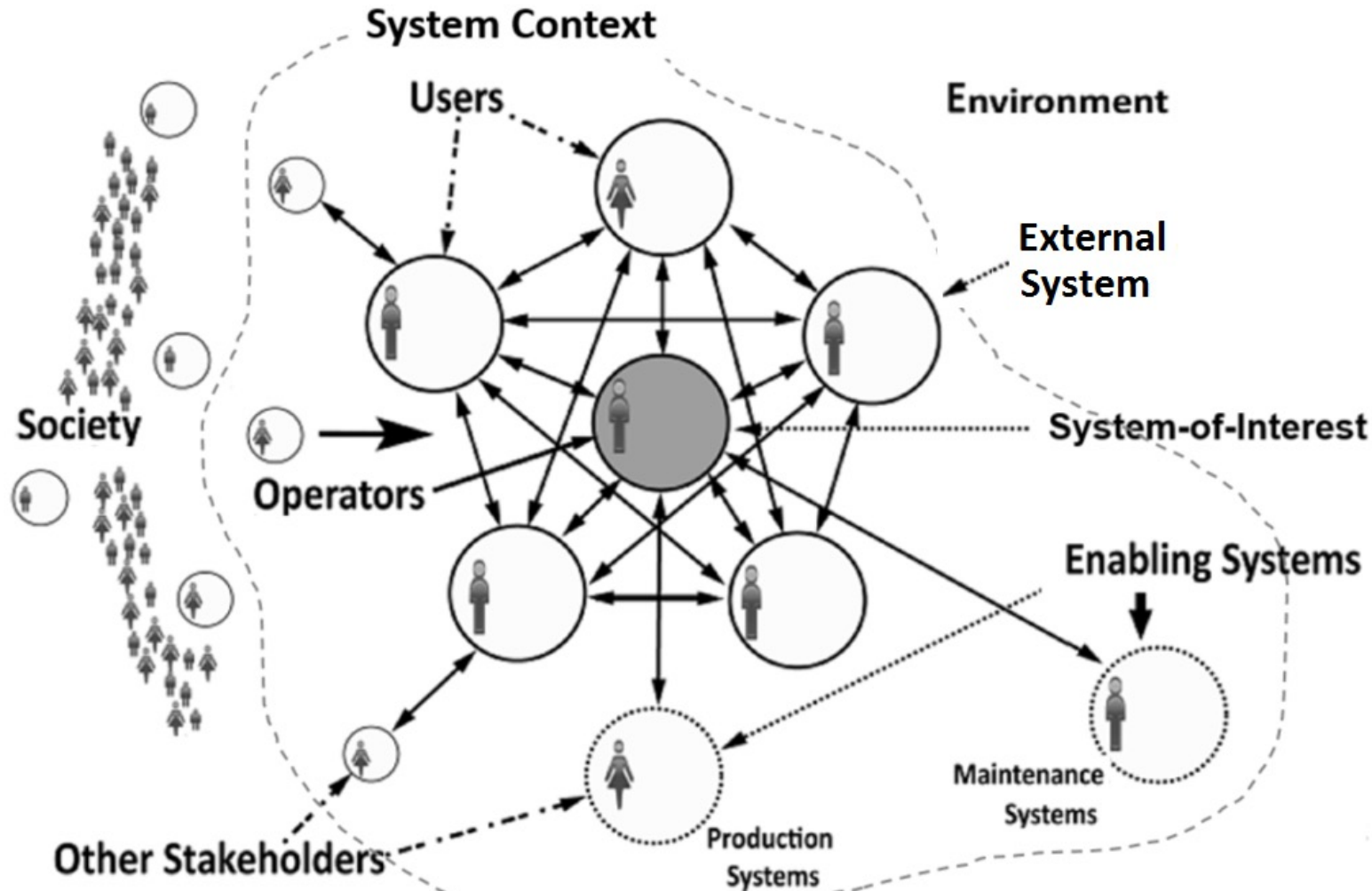
Engenharia de Sistemas são meios e abordagens transdisciplinares, baseados em conceitos e princípios sistêmicos, para permitir uma realização bem sucedida, uso e descarte de sistemas engenheirados.

Focando em:

- estabelecer as **motivações e critérios de sucesso** dos stakeholders (interessados e impactados), ajudando a definir as necessidades atuais e antecipando funcionalidades no início do ciclo de desenvolvimento;
- estabelecer um **modelo de ciclo de vida** apropriado e os processos considerando a complexidade, incertezas e possíveis mudanças;
- **documentar e modelar requisitos e arquiteturas** de cada fase dessa atividade;
- continuar com o **desenvolvimento e a verificação/validação**;
- enquanto considerando **todo o problema** e **todos os sistemas e serviços** que dão suporte ao sistema de interesse.



Stakeholders de um Sistema



ID	Need	Stakeholder
N1	"Need to notify support manager when a 'support request' is initiated."	Support manager
N2	"Need to assign support request to appropriate support engineer."	Support manager
N3	"Need to keep customer informed of the progress of a support request."	Customer (user)



Critérios de sucesso de um sistema

Technical Measurement

INCOSE-TP-2003-020-01

8 years

Sufficient propulsion
for 35 major
corrections

Need to obtain
an allocation of
17.5 liters for
propellant tank
by production

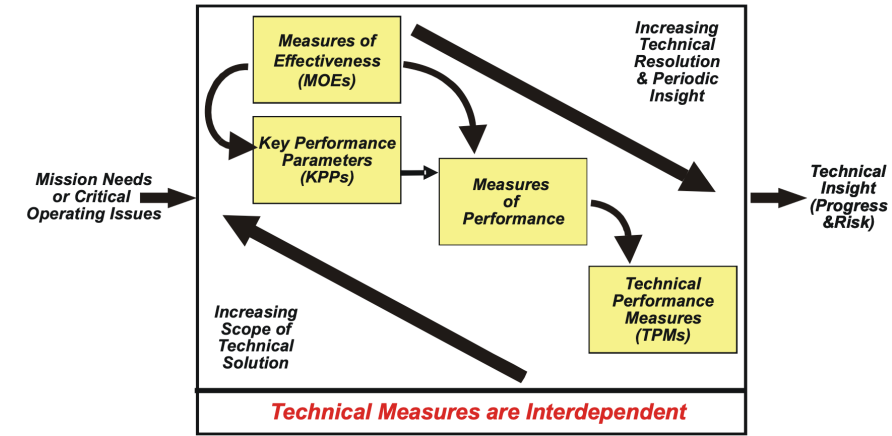
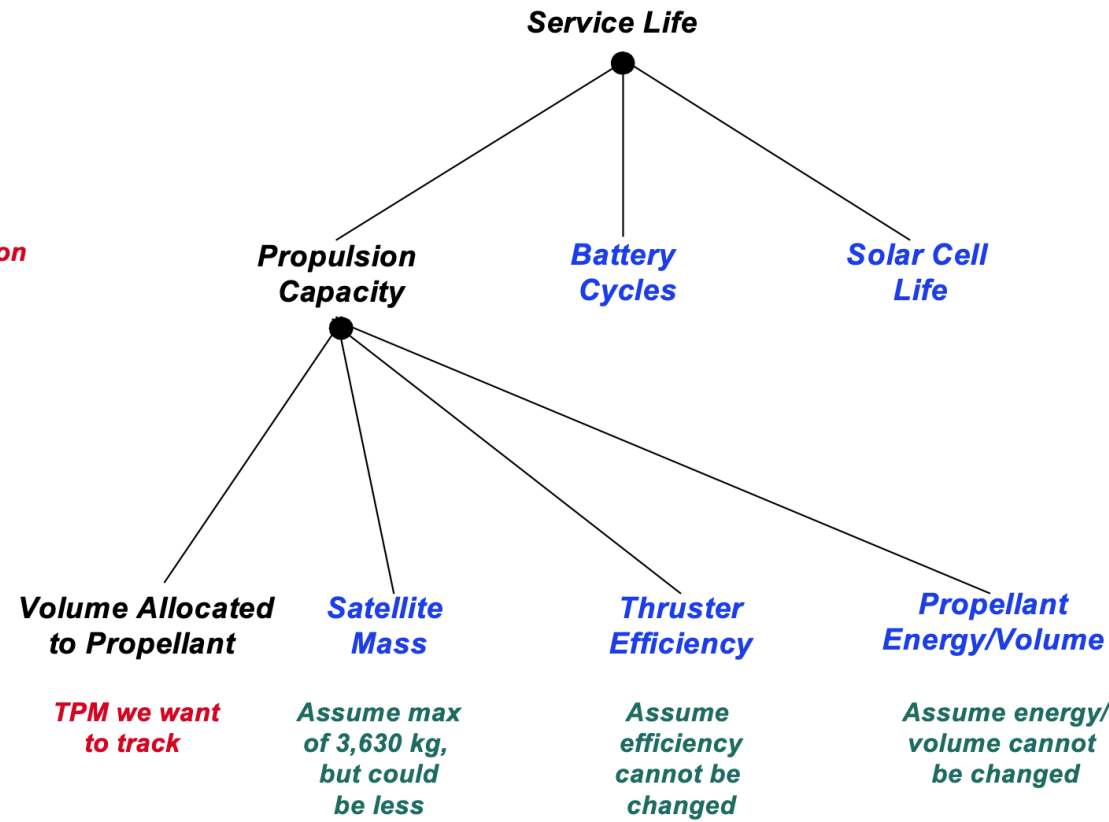


Figure 3-2 Relationship of the Technical Measures

Type	Item	Threshold	Indicator
MOE/KPP	Service Life	At least 8 years	Service Life Expected - trend over time
MOP	Propulsion Capacity	At least 35 major corrections	Orbital Corrections Supported - trend over time
TPM	Volume Allocated to Propellant	At least 17.5 liters	Propellant Tank Capacity - trend over time

Figure 3-3 Notional Example of Selection of MOEs/KPPs, MOPs, and TPMs



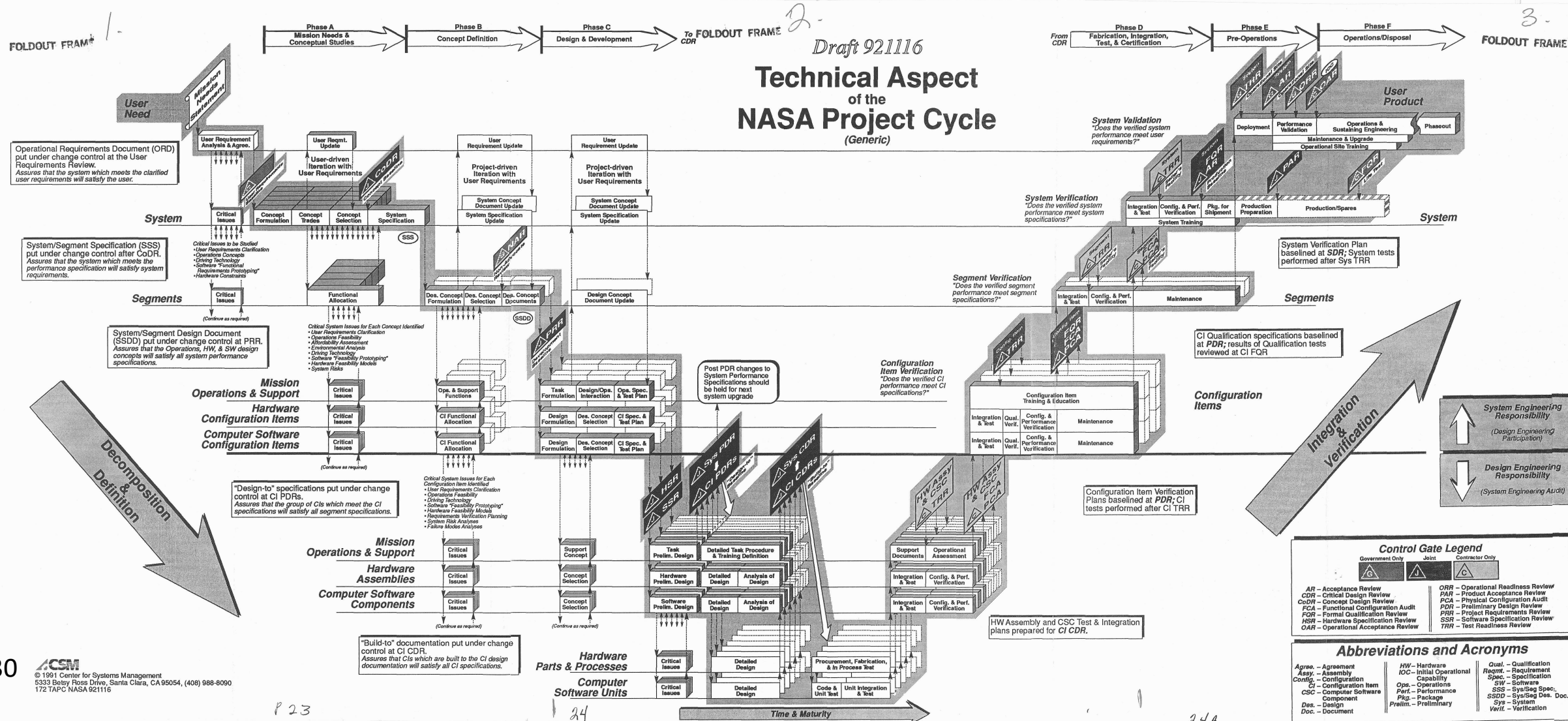
Ciclo de Vida

- **Todos os sistemas** possuem um ciclo de vida
 - Ciclo de vida é uma **série de estágios** que o sistema passa durante seu tempo de vida.
 - O ciclo de vida deve considerar toda a evolução do sistema, da **concepção ao descarte**.



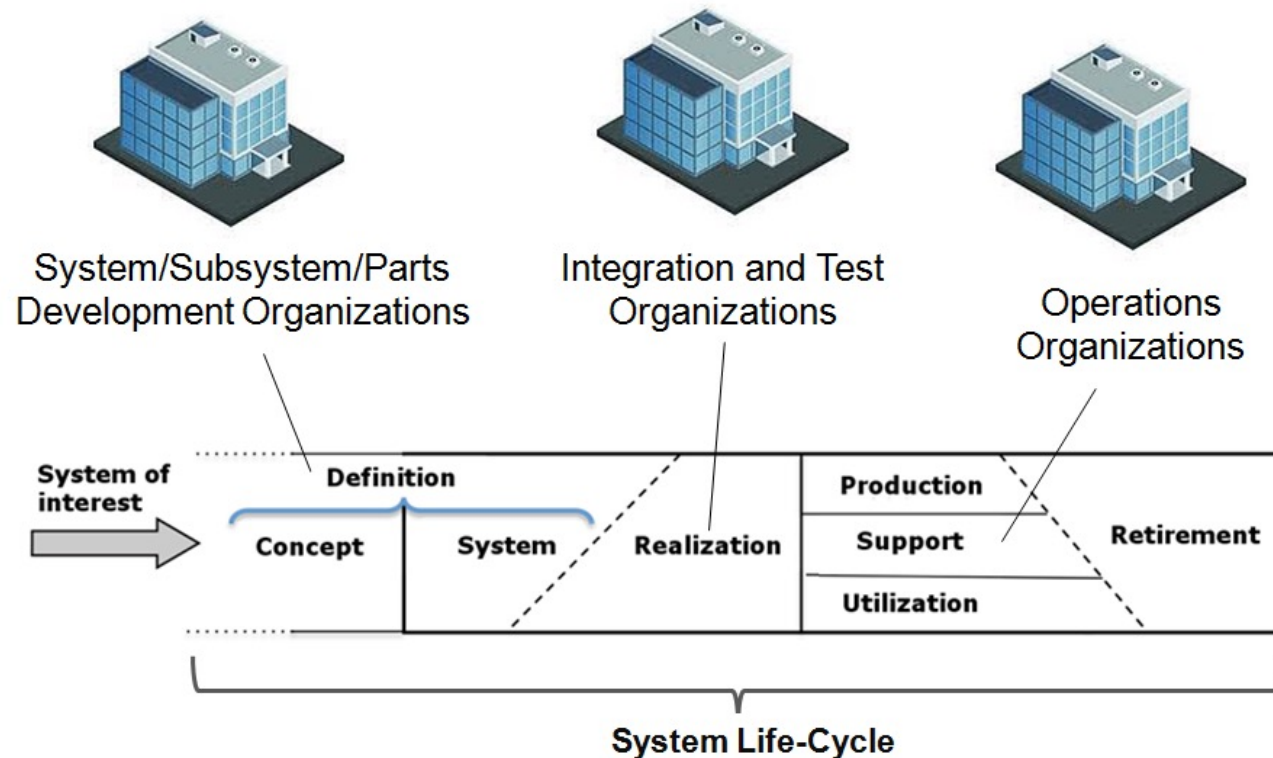


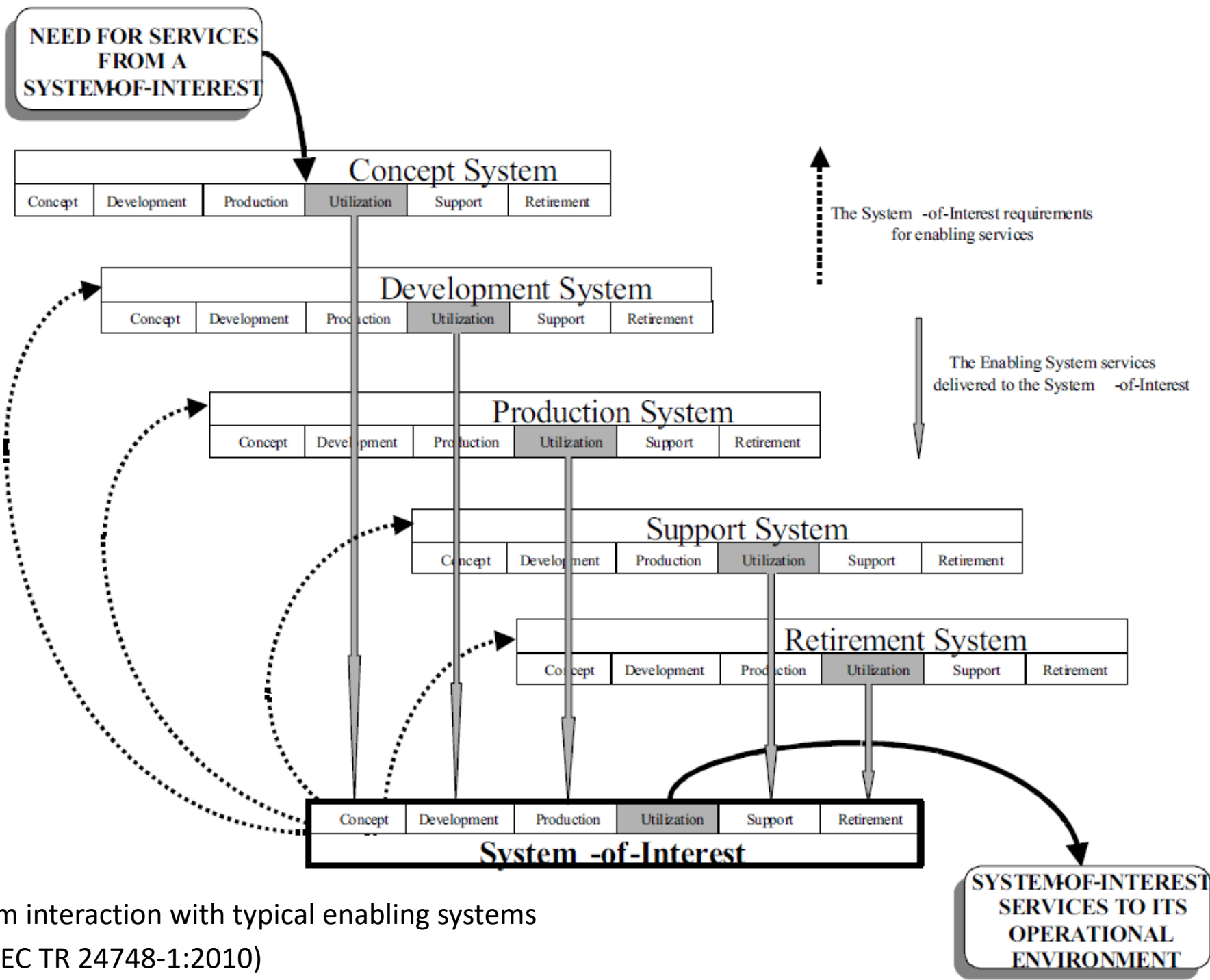
(CLASSICAL) VEE MODEL



Ciclo de vida como um conceito

- O **próprio ciclo de vida é um conceito** que descreve os produtos e organizações que interagem ao longo do processo de desenvolvimento, uso e descarte.

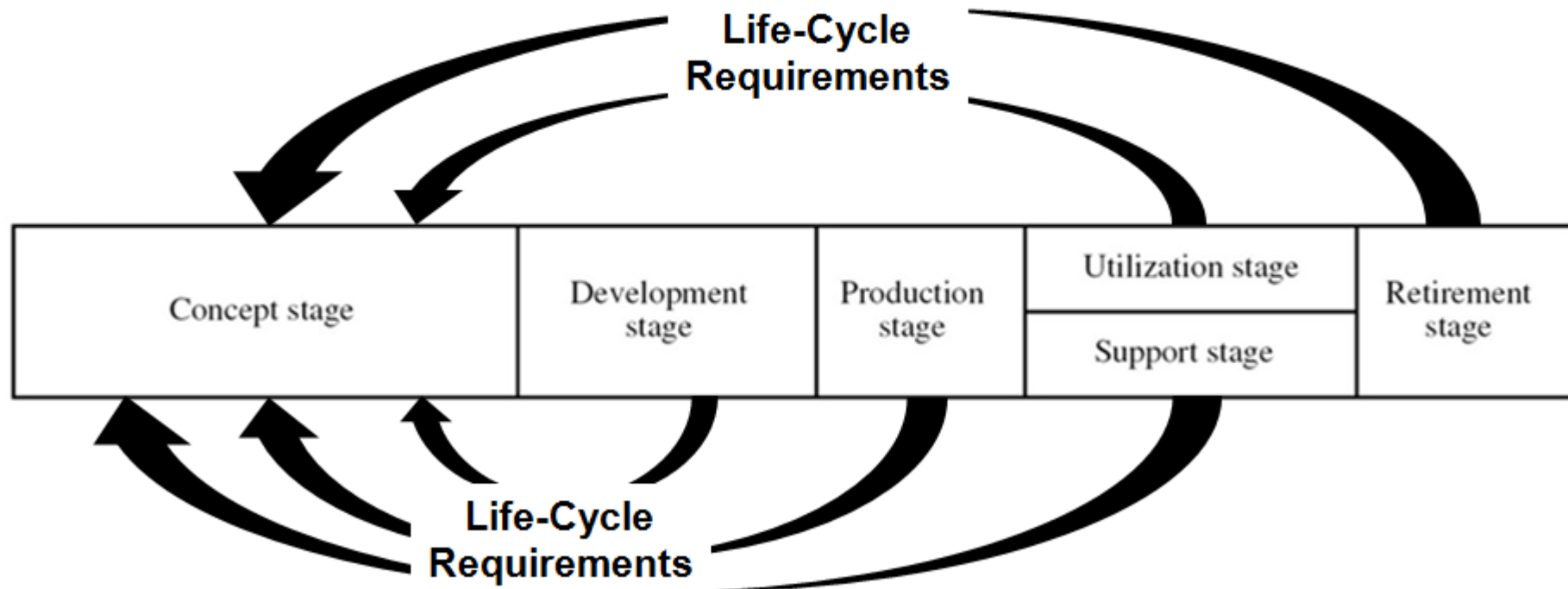




- System interaction with typical enabling systems
- (ISO/IEC TR 24748-1:2010)

Requisitos do ciclo de vida

- O **entendimento do ciclo de vida** permite antecipar funções e atributos.



Requisitos

- Requisitos indicam o que o sistema **deve fazer** (requisitos funcionais).
- **Quão bem** o sistema deve fazer (requisitos de performance)
- **Em qual ambiente** o sistema deve trabalhar (requisitos de ambiente).
- O que o sistema deve fazer para se **conectar com outros sistemas** (requisitos de interface).
- Qual composição de **partes/ subsistemas/ componentes** o sistema precisa para operar (alocação de requisitos).
- O que é preciso fazer **antes de operar** o sistema (verificação/validação).





Alguns pontos sobre requisitos:

- **Rastreabilidade** (traceability) é a relação entre requisitos.
- **Arrazoador** (rationale) são as informações que explicam como/por que/quando o requisito foi escrito/derivado.
- **Método de Verificação** (Verification Method) é a forma como o requisito vai ser comprovado
- **Critério de Sucesso** (Success Criteria) é o critério da verificação que deve ser apresentado para comprovar o requisito.



Escolhendo um método de verificação

Teste é usado quando:

- Técnicas analíticas não produzem resultados adequados.
- Modos de falha que podem comprometer o sistema.
- Elementos associados com sistemas críticos.

Demonstração é usada quando:

- Verificação podem ser realizadas através de observação.
- Resultados tendem a ser: sim ou não
- Requisitos que são mais subjetivos como fatores humanos ou conceitos de operação.
- Sistema real ou representativo

Inspeção é usada quando:

- Diagramas, documentos e dados que podem ser visualmente verificados para comprovar o projeto
- Tipicamente usado para verificar metodos de construção, manufatura, dimensões e registros.

Análise é usada quando:

- Métodos analíticos são possíveis
- Teste não é possível
- Verificação por inspeção não é adequada. Técnicas analíticas e modelos estão validados.

Exemplo de escrita de critério de sucesso

- **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

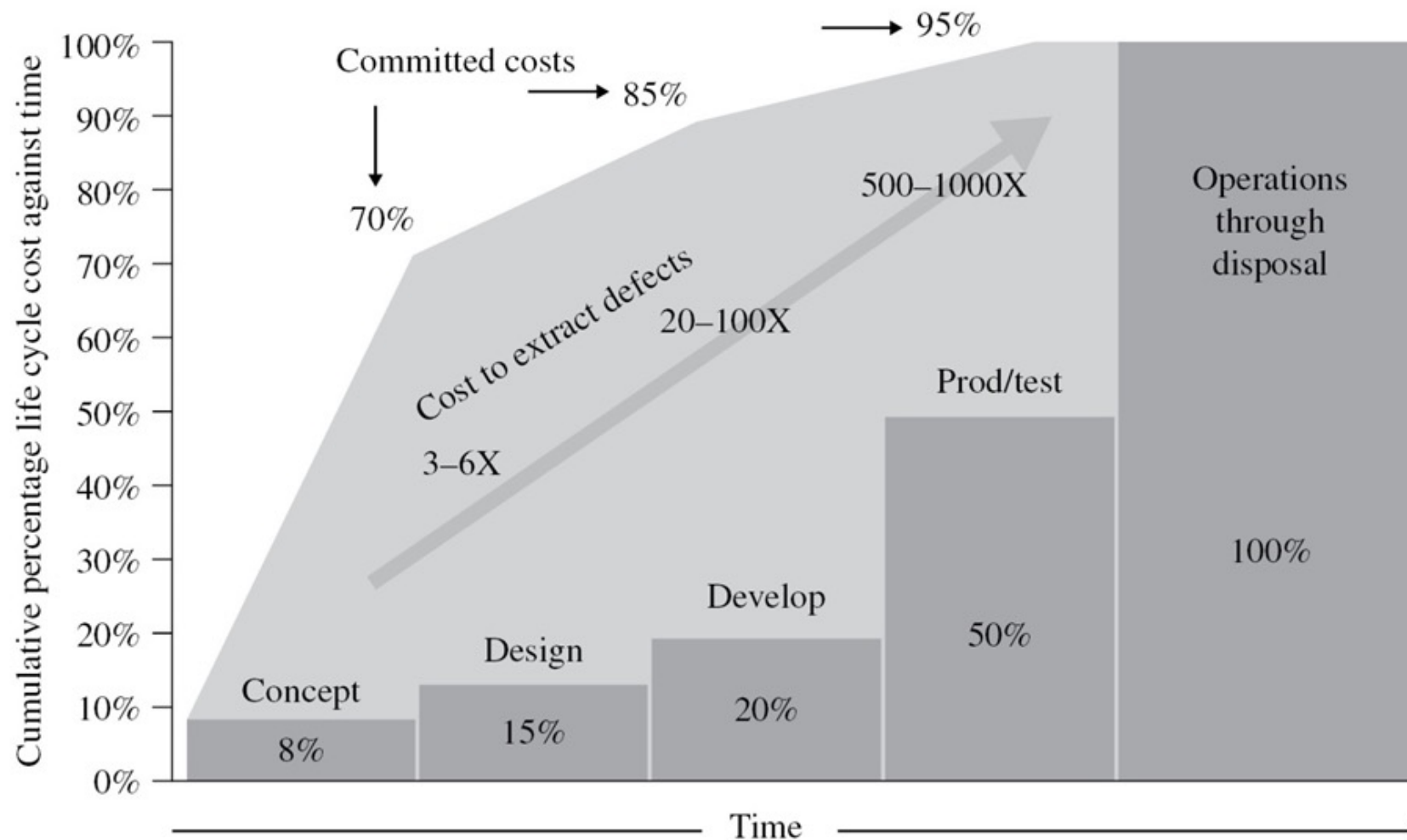
Environment

Note – there are no
Special Conditions

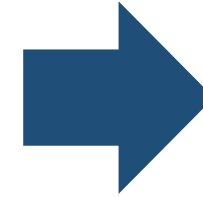
Success Criteria

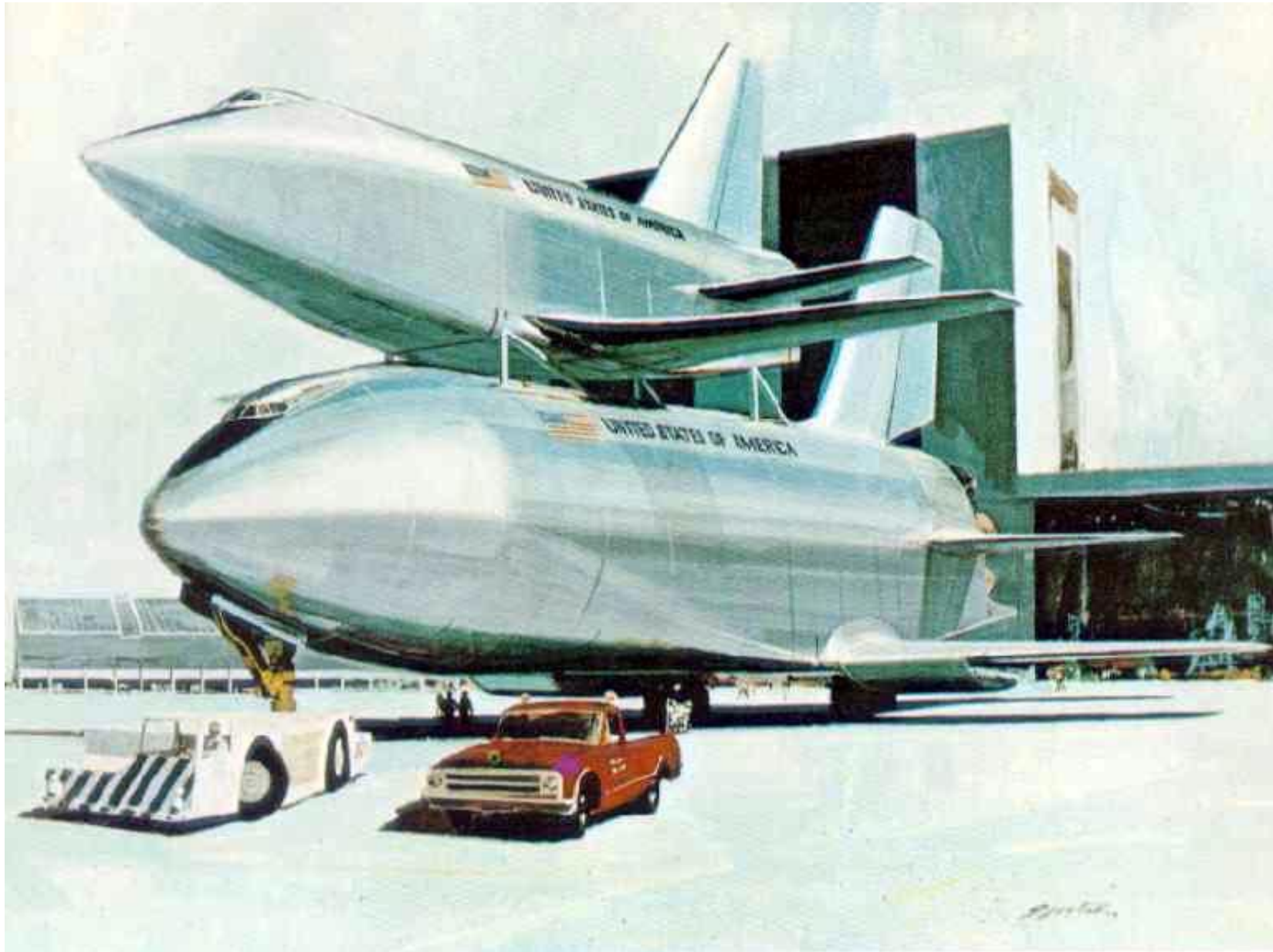


Conceitos impactam nas decisões de todo o ciclo de vida



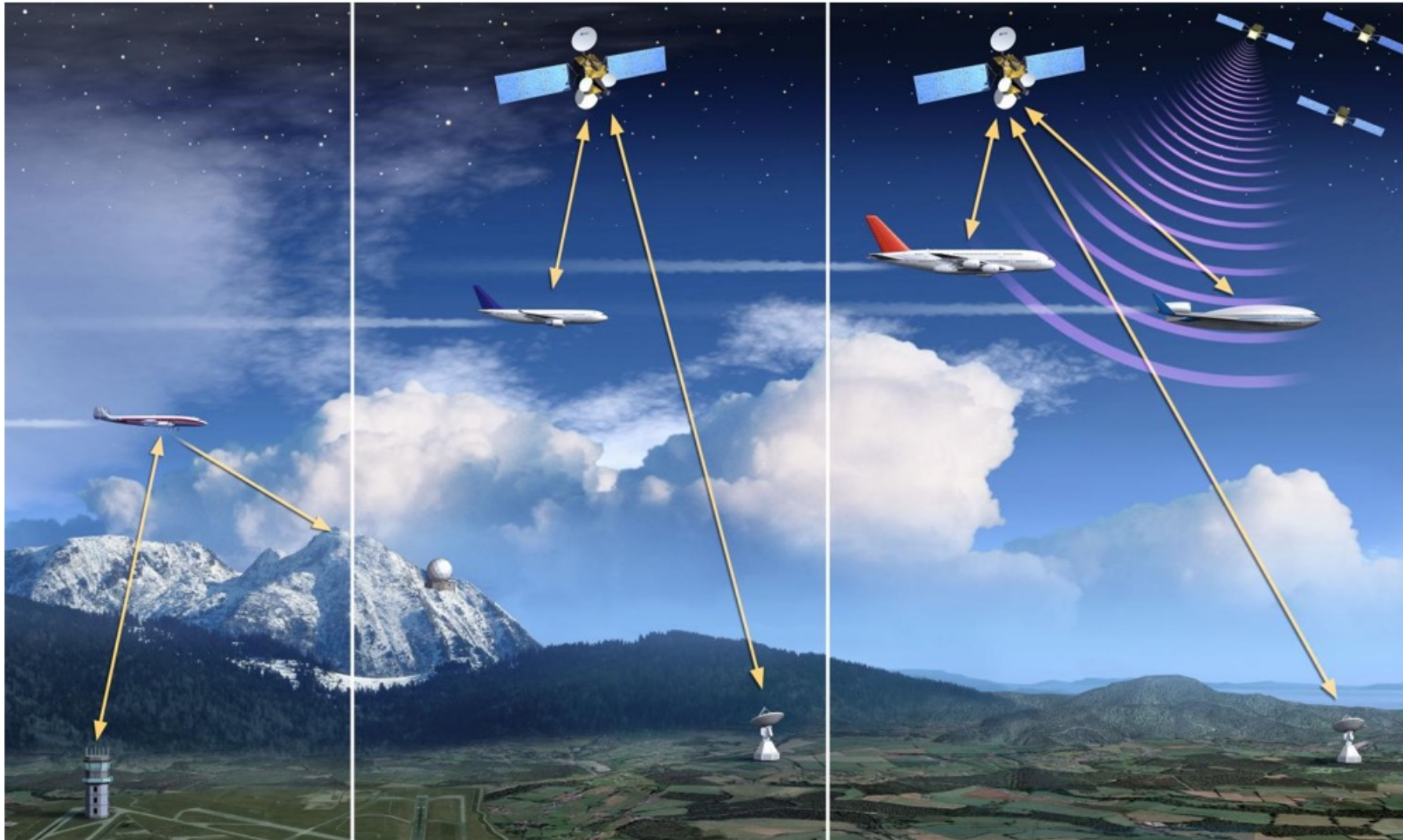
O sistema é um conceito ou o conceito é um sistema?







Conceito de Operações (CONOPs – Concept of Operations)



O que é um CONOPs

- É a descrição de **como** o sistema vai se comportar para atender as expectativas do stakeholders.
- Explica o comportamento do sistema de **uma perspectiva operacional** para facilitar o entendimento das propriedades emergentes do sistema.
- Ilustra o **uso pretendido** do sistema através de recortes de situações de como o sistema deve se comportar.
- **CONOPs** pode ser detalhado em **casos de uso / cenários**



Exemplo de requisitos tirados de um CONOPs

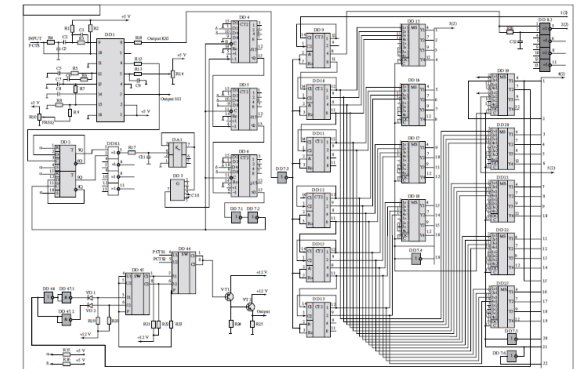
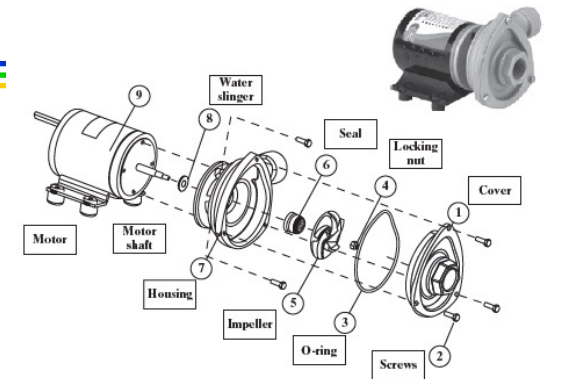
• NEED STATEMENT: PORTABLE AUTOMOTIVE VEHICLE FOR OFF-ROAD

- Use Case: Camping Trip
 - Scn: Load Camping Gear On To Vehicle
 - Function: Provide Space To Store Camping Gear
 - Requirement: The vehicle shall be designed with a minimum cargo storage area of 11 cubic feet.
 - Rationale: Estimated calculation of space needed to accommodate customer's camping equipment
 - Scn: Load Personnel On To Vehicle
 - Function: Provide Capability To Carry Up To 4 People
 - Scn: Drive To Camp Site
 - Function: Provide Capability To Drive 20 Miles Roundtrip
 - Requirement: The vehicle driving range shall be a minimum of 40 miles.
 - Rationale: Customer stated the maximum range (out and back) of use as being 20 miles after getting the vehicle to where it would be used.
 - Function: Provide Capability To Utilize GPS
 - Scn: Get Stuck
 - Function: Provide Vehicle Capability To Get Un-Stuck
 - Scn: Unload Camping Gear/Camp
 - Function: Provide Electrical Interface For DC Powered Camp Eq.
 - Requirement: The vehicle shall provide, 12 VDC +/- 1.5 VDC, 12 Amps maximum, auxiliary utility ports.
 - Rationale: Based on power draw assessment of the connecting equipment
 - Requirement: The vehicle shall provide two utility ports for connecting DC powered camping equipment.
 - Rationale: Customer desired two specific ports to connect equipment simultaneously



Os requisitos e conceitos originam arquiteturas

- Arquitetura é a **materialização de um conceito**.
- É onde ocorre a **alocação das funções e características em uma forma**, e são criadas as **relações entre os elementos e o ambiente de operação**.



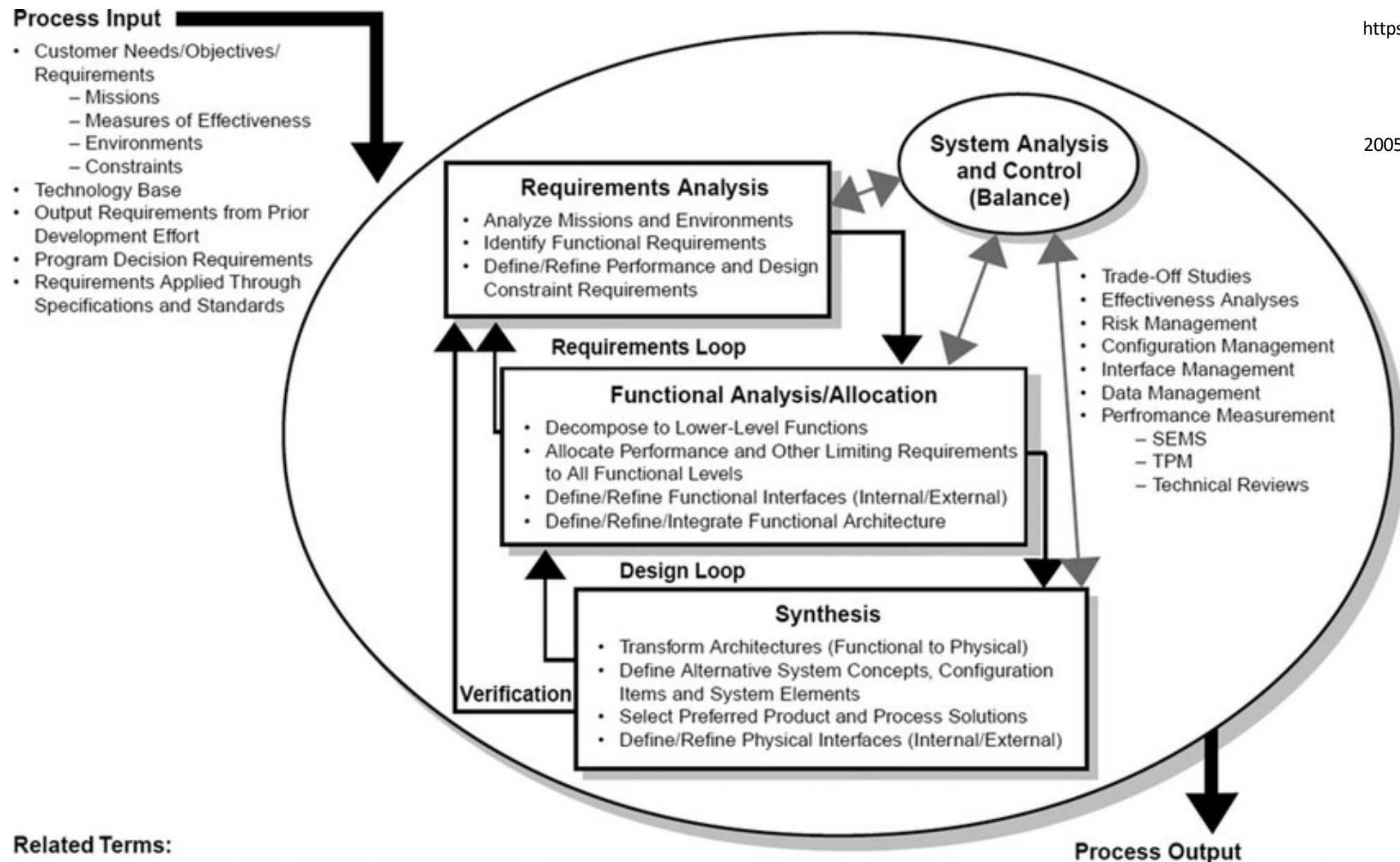
```
1 Procedure bubblesort (List array, number length_of_array)
2   for i=1 to length_of_array - 1;
3     for j=1 to length_of_array - i;
4       if array [j] > array [j+1] then
5         temporary = array [j+1]
6         array [j+1] = array [j]
7         array [j] = temporary
8       end if
9     end of j loop
10  end of i loop
11  return array
12 End of procedure
```

Verificação x Validação

- **Verificando um sistema:**
Construindo o sistema
corretamente: garante
que o sistema atende
aos requisitos.



- **Validando um sistema:**
Construindo o sistema
certo: garante que o
sistema faz o que deveria
fazer no ambiente
desejado.



Related Terms:

- Customer = Organizations responsible for Primary Functions
- Primary Functions = Development, Production/Construction, Verification, Deployment, Operations, Support, Training, Disposal
- Systems Elements = Hardware, Software, Personnel, Facilities, Data, Material, Services, Techniques

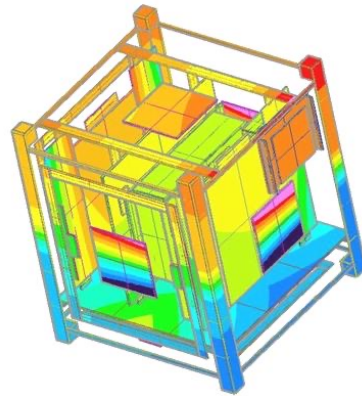
Process Output

- Development Level Dependent
 - Decision Database
 - System/Configuration Item Architecture
 - Specifications and Baselines

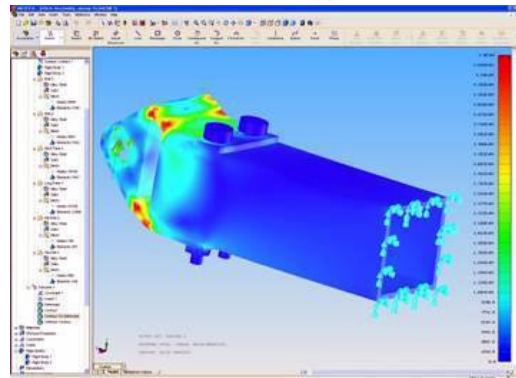


Cada domínio de conhecimento tem uma linguagem

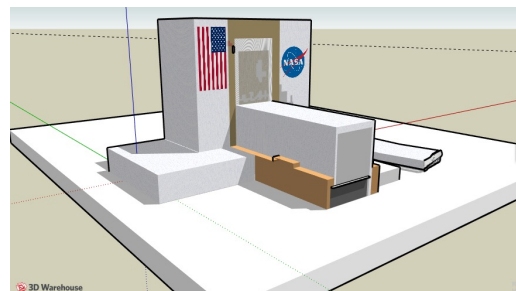
Thermal Eng.



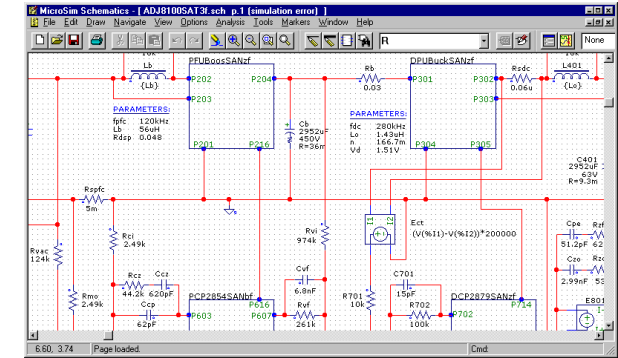
Mechanical Eng.



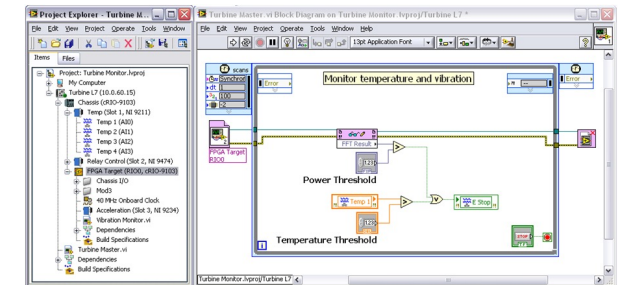
Infrastructure Eng.



Electrical Eng.



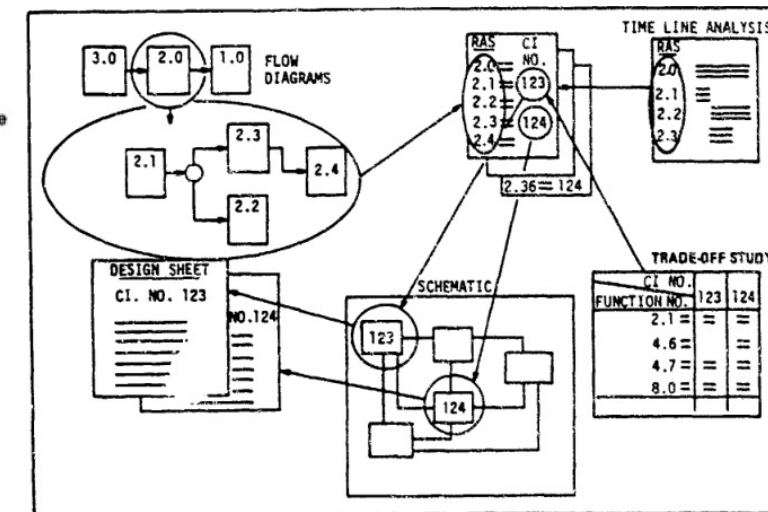
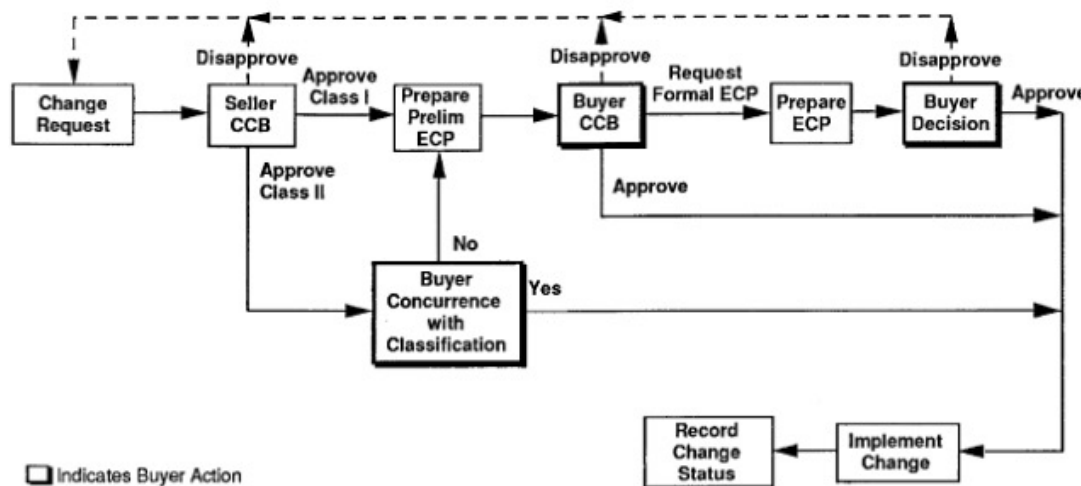
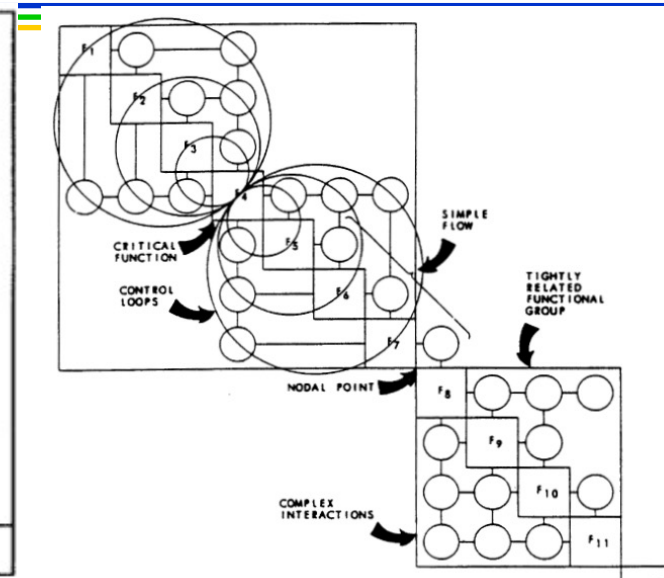
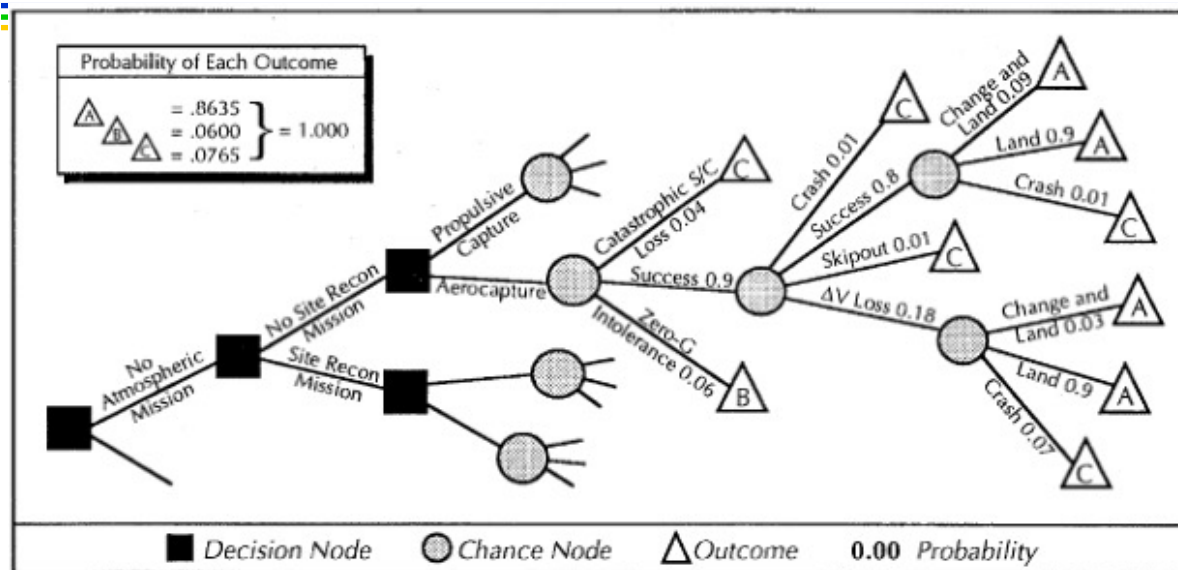
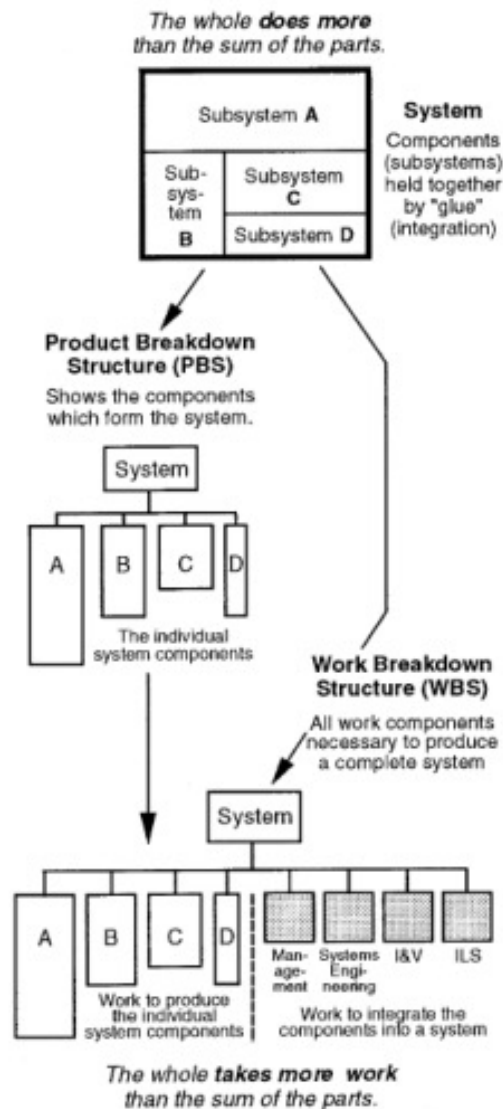
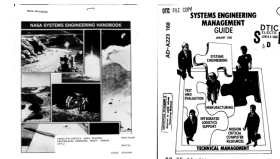
Control Eng.



Systems Eng ???



Qual é a linguagem da Engenharia de Sistemas?



Com o tempo... diagramas coloridos...

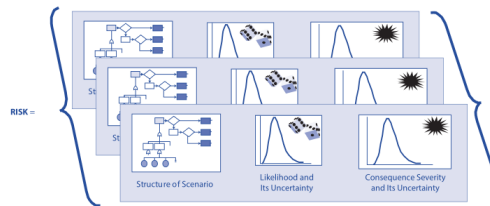
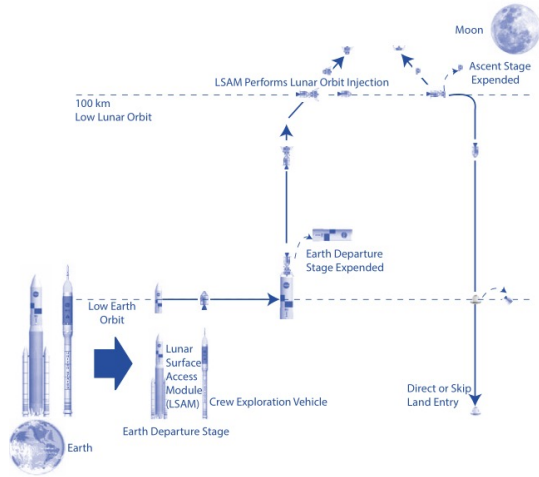
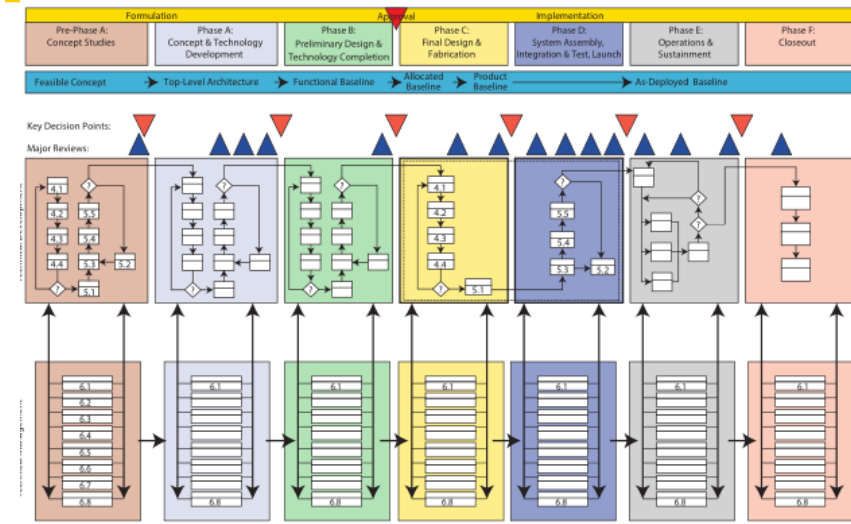
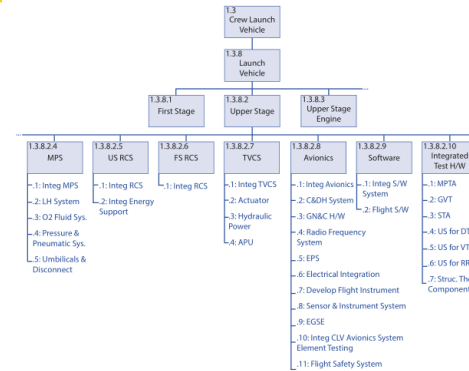
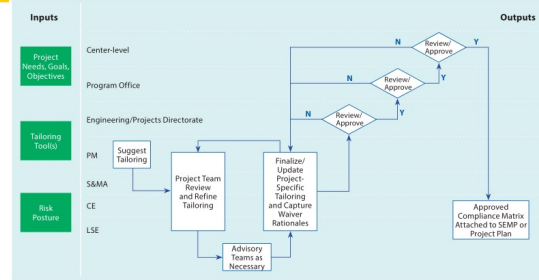


TABLE D-1 Requirements Verification Matrix

Requirement No.	Document	Paragraph	Shall Statement	Verification Success Criteria	Verification Method	Facility or Lab	Phase*	Acceptance Requirement?	Preflight Acceptance?	Performing Organization	Results
Unique identifier or each requirement	Document number the requirement is contained within	Paragraph number of the requirement	Test within reason of the requirement, i.e., the "shall"	Success criteria for the requirement	Verification method for the requirement (analysis, inspection, demonstration, test)	Facility or laboratory used to perform the verification and validation	Phase in which the verification and validation will be performed	Indicate whether this requirement is also verified during any pre-flight or recurring acceptance testing of each unit	Indicate whether this requirement is also verified during initial acceptance testing of each unit	Organization responsible for performing the verification	Indicate documents that contain the objective evidence that requirement was satisfied
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-1	xxx	Other paragraphs	Other "shall" in PTRS	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Memo xxx
S-1 or other unique designator	xxxxx (other specs, ICDs, etc.)	Other paragraphs	Other "shall" in specs, ICDs, etc.	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Report xxx

■ Red = Below TRL 3
■ Yellow = TRL 3, 4 & 5
■ Green = TRL 6 and above
■ White = Unknown
 X = Exists

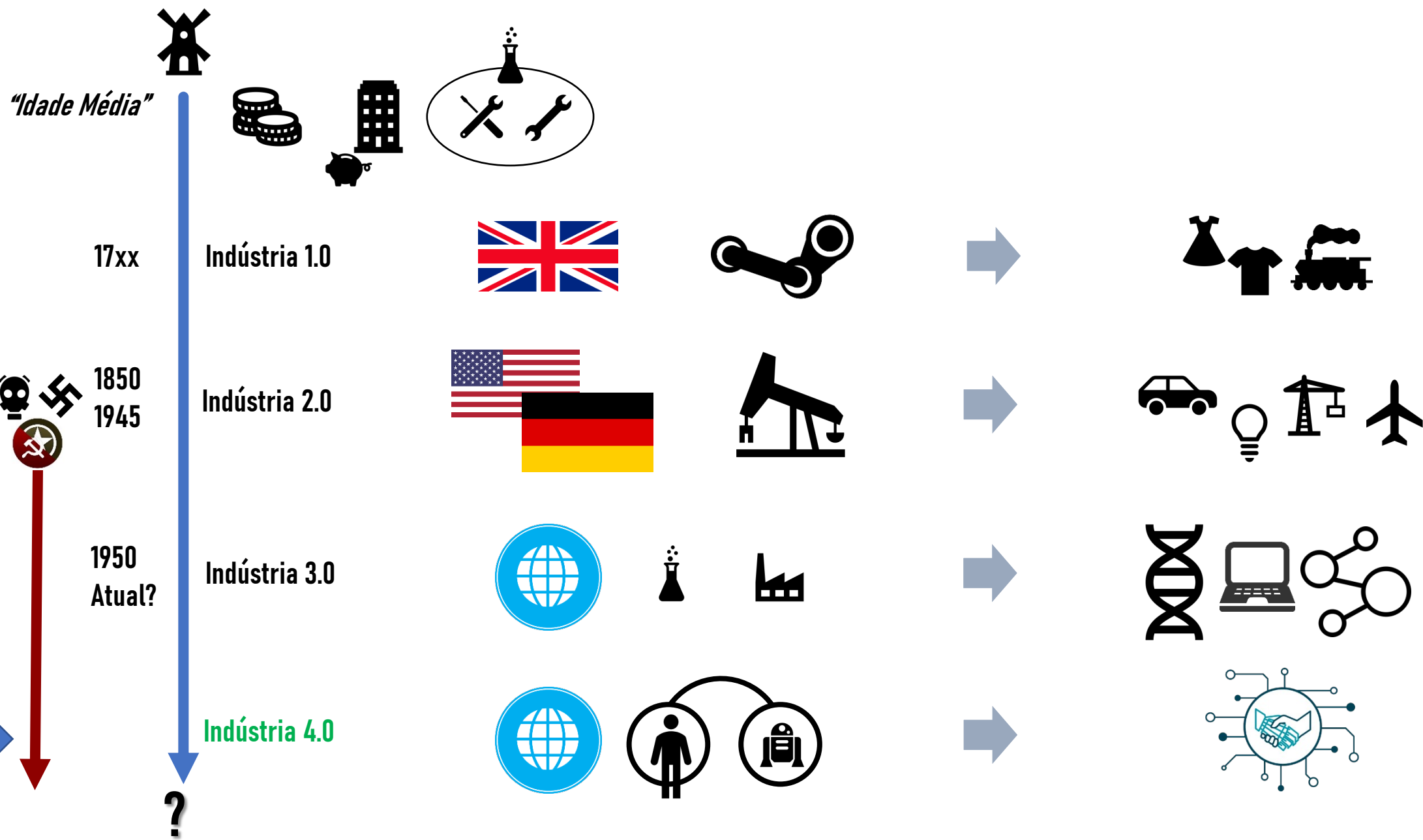
Red = Below TRL 3	Demonstration Units				Environment		Unit Description								
Yellow = TRL 3, 4 & 5	Concept	Breadboard	Brassboard	Developmental Model	Prototype	Flight Qualified	Laboratory Environment	Relevant Environment	Space Environment	Space Launch Operation	Form	Fit	Function	Appropriate Scale	Overall TRL
Green = TRL 6 and above															
White = Unknown															
X = Exists															
1.0 System															
1.1 Subsystem X															
1.1.1 Mechanical Components															
1.1.2 Mechanical Systems															
1.1.3 Electrical Components															
1.1.4 Electrical Systems															
1.1.5 Control Systems															
1.1.6 Thermal Systems															
1.1.7 Fluid Systems															
1.1.8 Optical Systems															
1.1.9 Electro-Optical Systems															
1.1.10 Software Systems															
1.1.11 Mechanisms															
1.1.12 Integration															
1.2 Subsystem Y															
1.2.1 Mechanical Components															



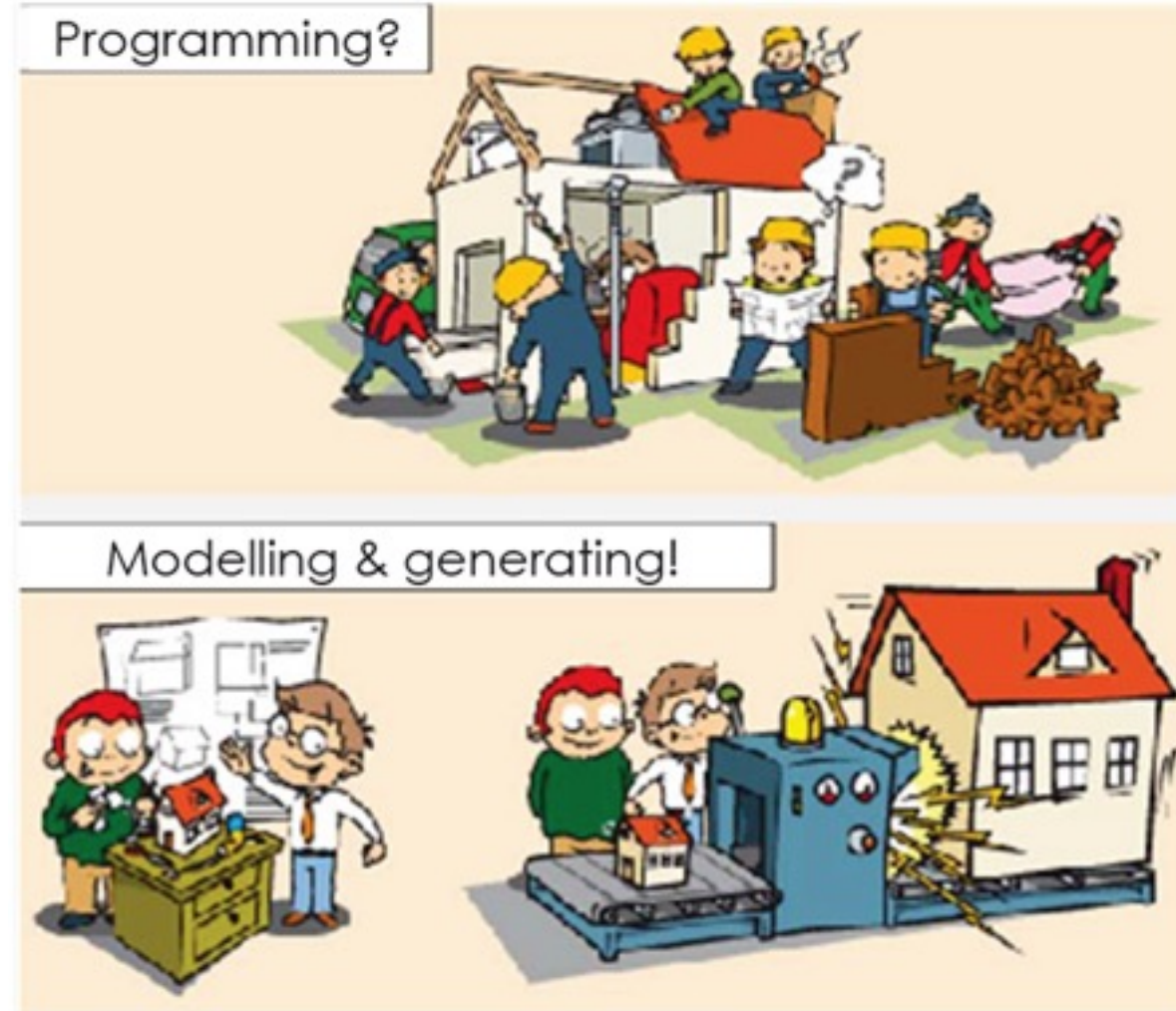


ENGENHARIA DE SISTEMAS BASEADA EM MODELOS

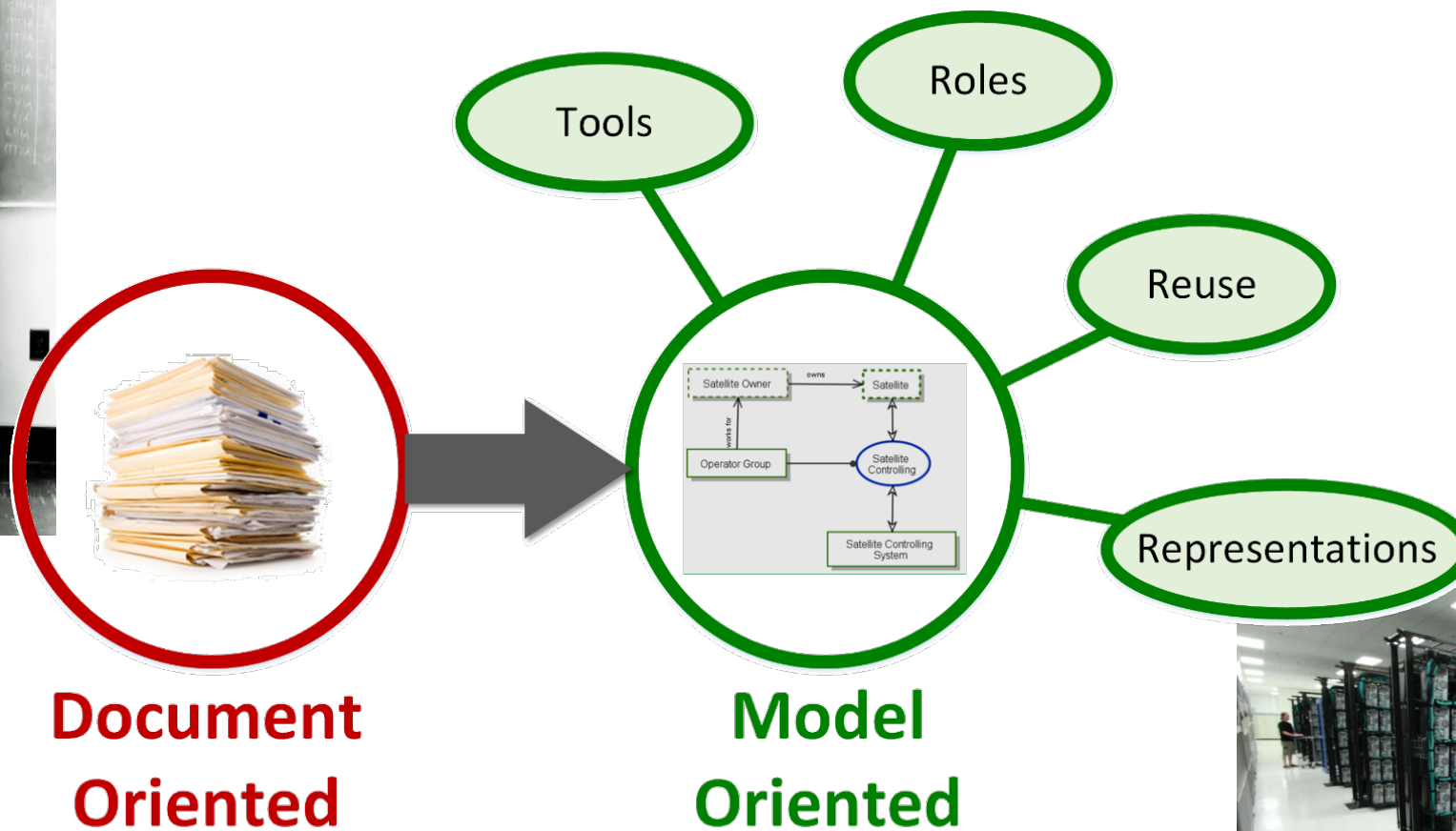




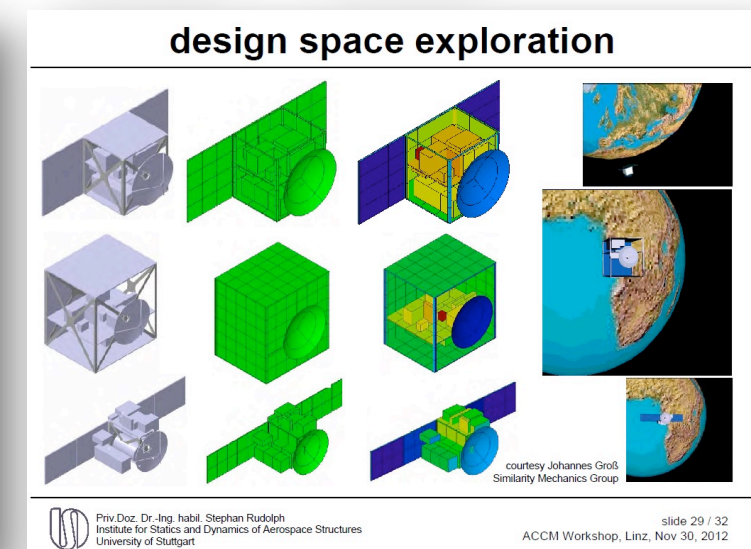
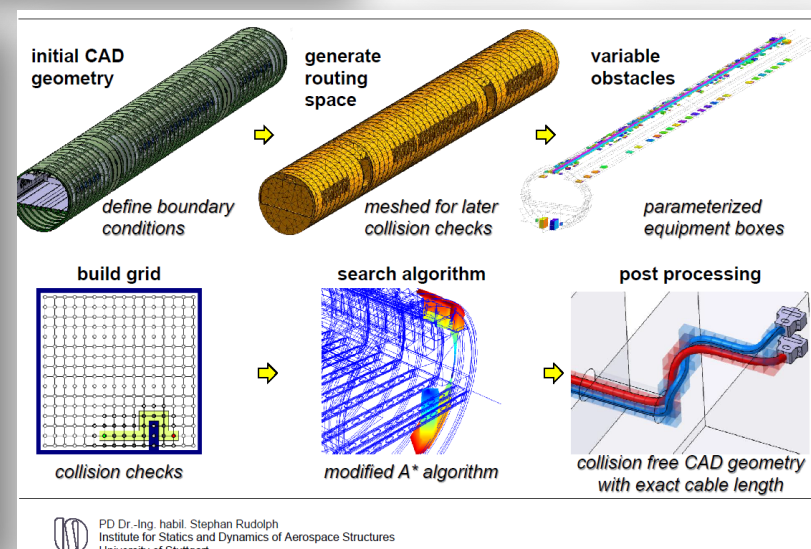
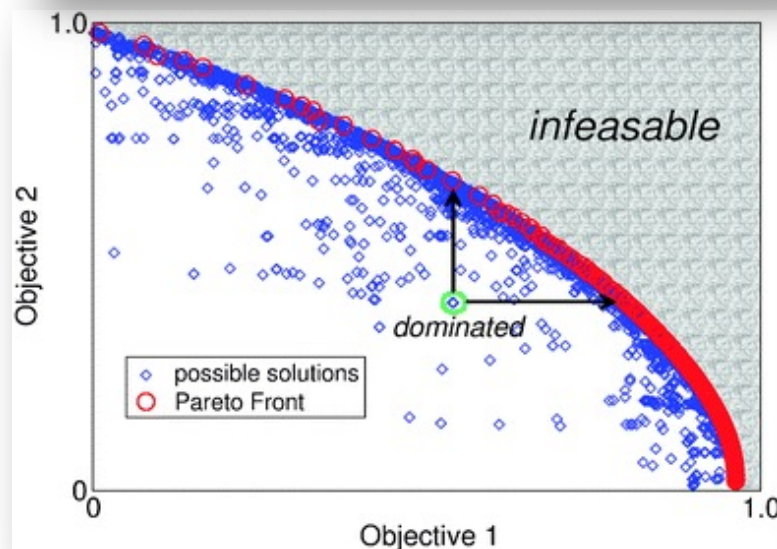
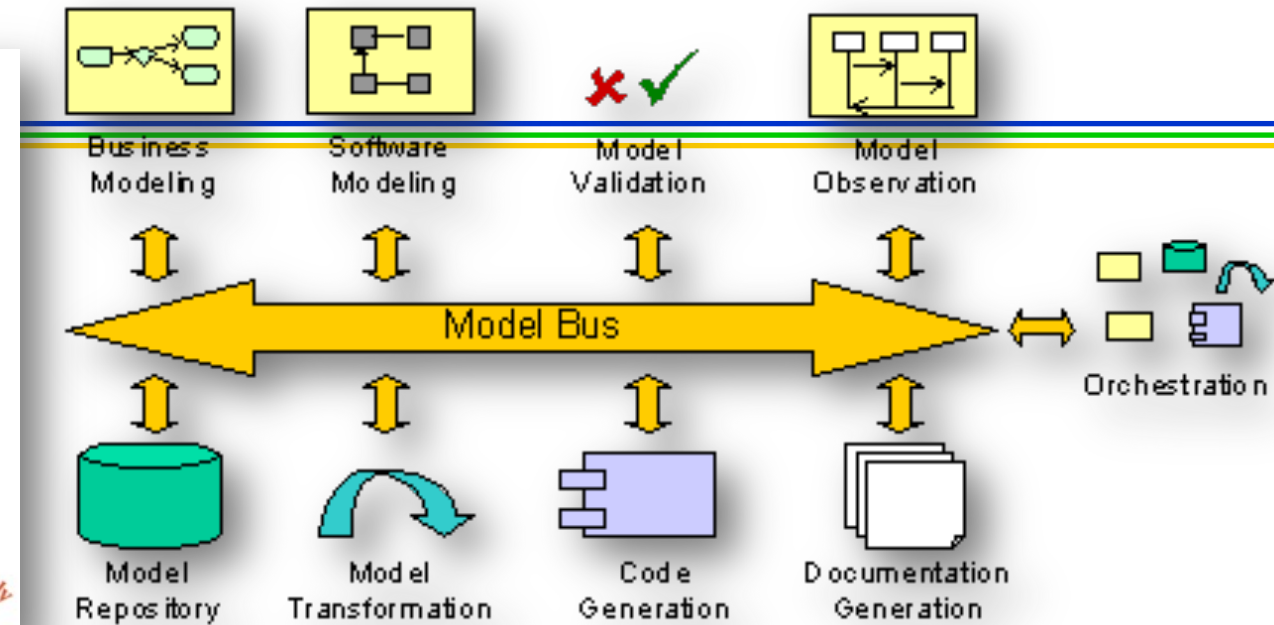
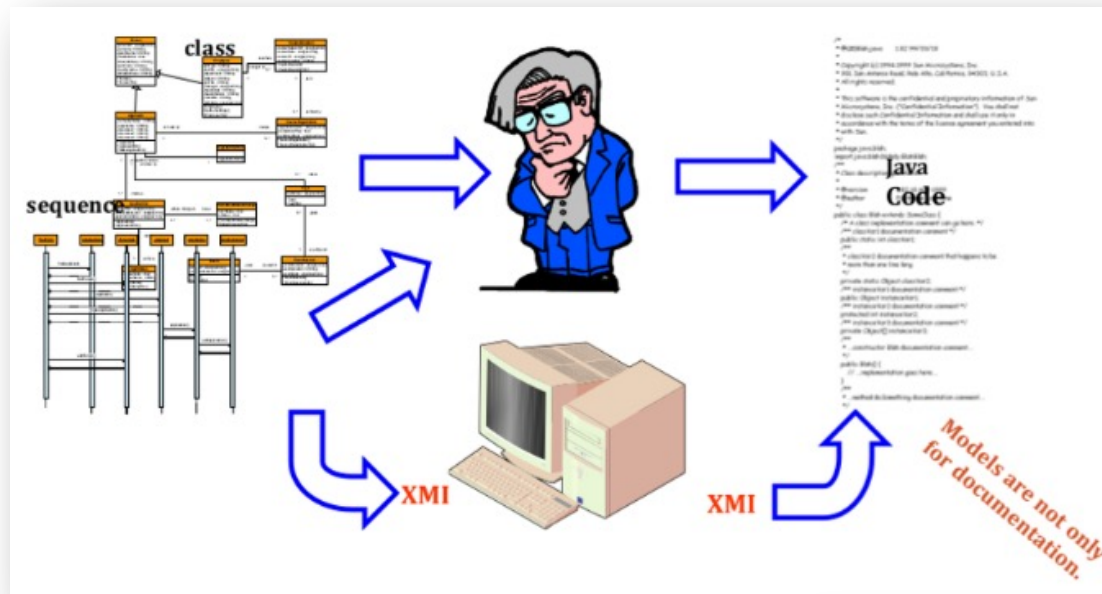
Alteração na forma de fazer engenharia

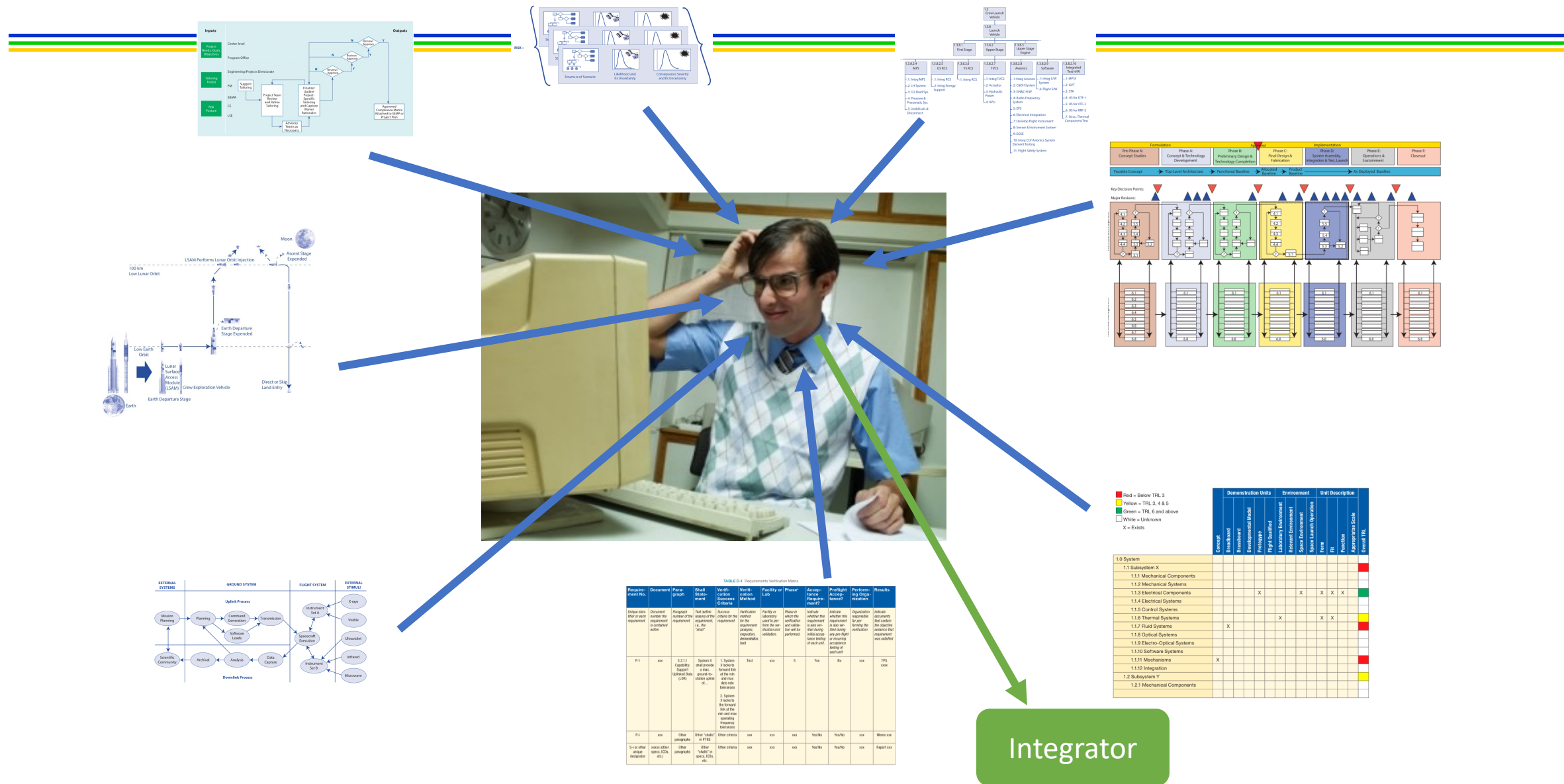


Levou a uma mudança de paradigma



Vários benefícios

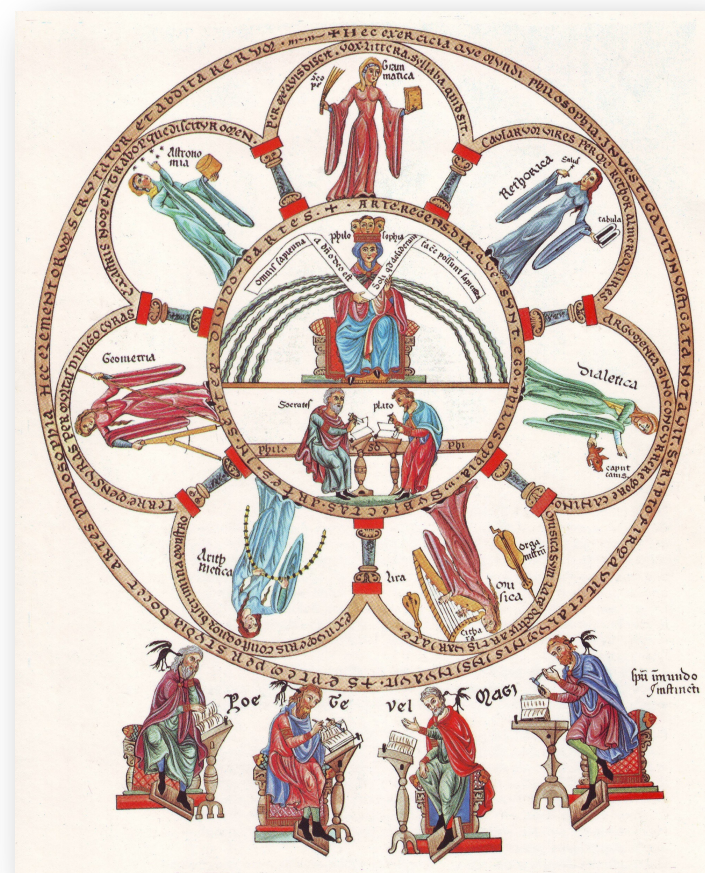




Definição de Engenharia de Sistemas baseada em Modelos (Model based Systems Engineering)

“ESBM é a aplicação de uma metodologia de Engenharia de Sistemas que foca na utilização de modelos na metáfora da Engenharia de Sistemas como o meio primário de troca de informações entre as equipes, ao invés de utilizar a metáfora de documentos.”

MBSE **formaliza a linguagem** da Eng. de Sistemas



TRIVIUM AND THIS COURSE

We need to understand the "thing as-it-is-symbolized"

SysML and OPM symbols

We need to understand the "thing as-it-is-known"

SysML and OPM Diagram Interpretation

We need to understand the "thing as-it-is-communicated"

Integrated Framework

The integrated information (on the model) is the single source of truth

<https://augustineacademy.com.au/subjects/trivium/>

INCOSE lista diversas metodologias

List of Methodologies and Methods

Methodologies Surveyed in INCOSE 2008 Report

Name	Primary Point of Contact
INCOSE Object-Oriented Systems Engineering Method (OOSEM)	✉ safriedenthal@gmail.com
IBM Rational Telelogic Harmony-SE	✉ peter.hoffmann@telelogic.com
IBM Rational Unified Process for Systems Engineering (RUP-SE)	✉ mcantor@us.ibm.com
Vitech Model-Based Systems Engineering (MBSE) Methodology Vitech	✉ jlong@vitechcorp.com
JPL State Analysis (SA) Methodology JPL State Analysis (SA)	✉ Robert.D.Rasmussen@jpl.nasa.gov
➡ Dori Object-Process Methodology (OPM)	✉ dori@ie.technion.ac.il

Additional Methodologies Identified as Gaps Since 2008 INCOSE Survey

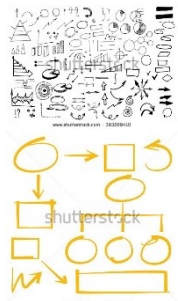
Name	Primary Point of Contact
Weilkiens Systems Modeling Process (SYSMOD)	✉ Tim.Weilkiens@oose.de
Fernandez Process Pipelines in OO Architectures (PPOOA)	✉ joselfernandez@telefonica.net
An Ontology for State Analysis: Formalizing the Mapping to SysML	✉ nicolas.f.rouquette@jpl.nasa.gov
 ISO-15288, OOSEM and Model-Based Submarine Design	✉ Paul.Pearce@deepbluetech.com.au
Alstom ASAP methodology	✉ marco.ferrogallini@transport.alstom.com
Pattern-Based Systems Engineering (PBSE)	✉ schindel@ictt.com
➡  Arcadia, a model-based engineering method	 Polarsys/Capella



OBJECT PROCESS METHODOLOGY

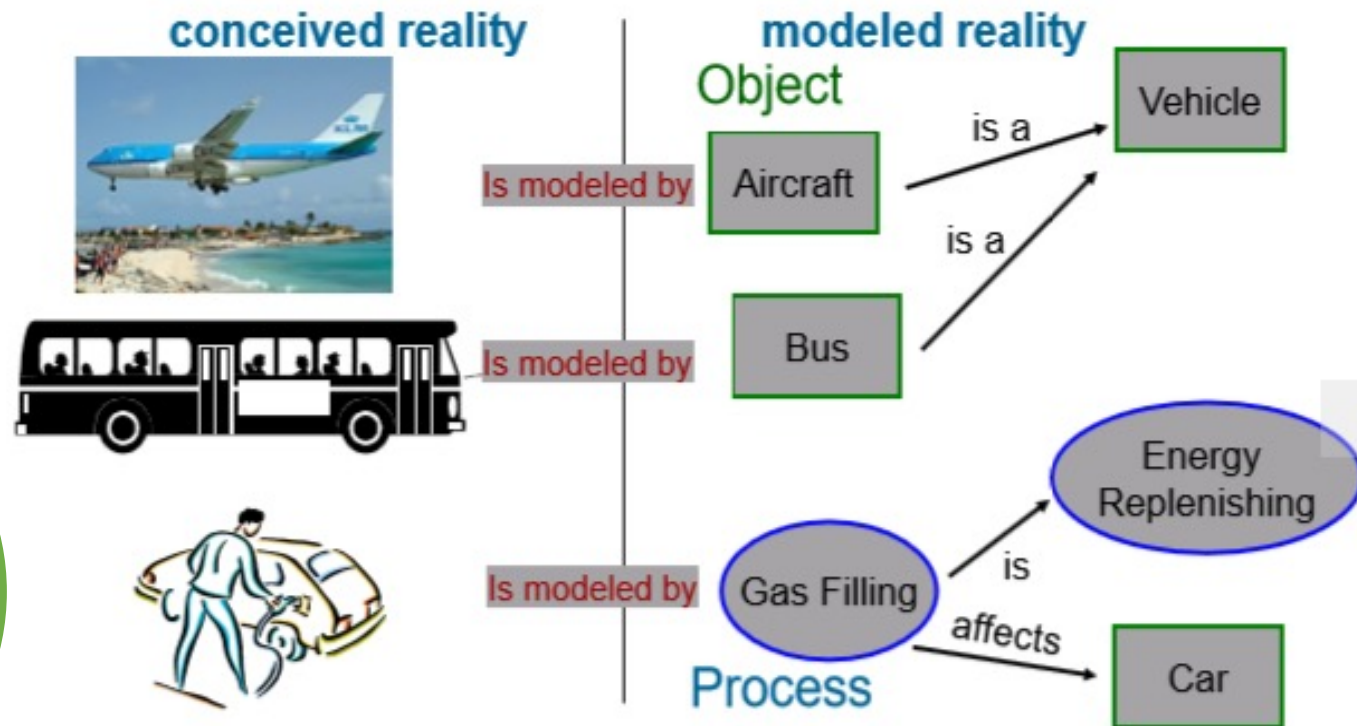
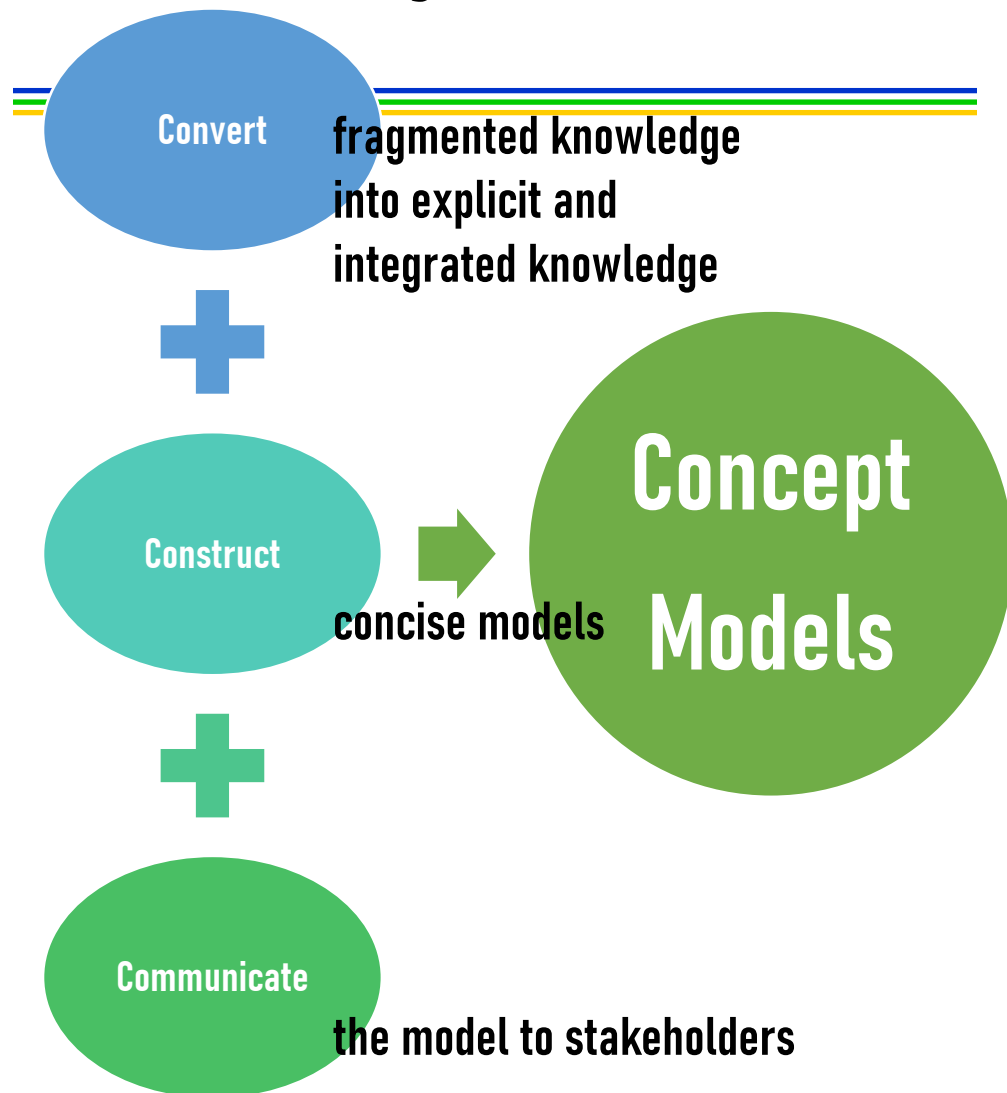


O começo: explicando ideias...



- Pegue uma caneta e um pedaço de papel
- Faça uns rabiscos com uns nomes
- Enquanto conversa, desenhe umas linhas e setas para interligar os rabiscos
- Veja a reação às proposições
- Continue...

O começo



- **simples** porém **expressivo**, e
- **intuitivo** porém **formal**

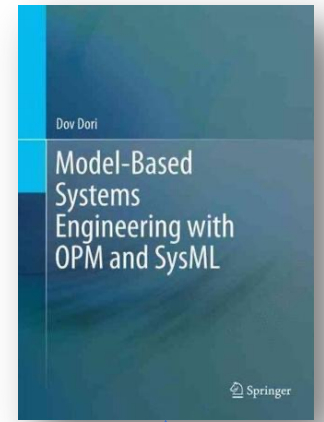
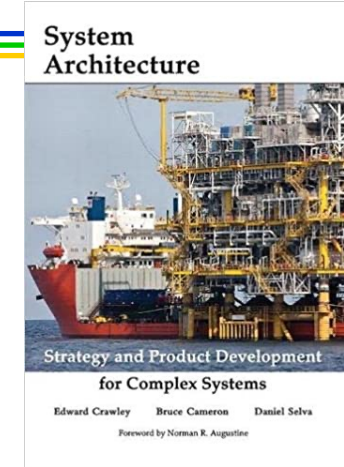
OPM – OBJECT PROCESS METHODOLOGY

Prof. Dov Dori



Created in 2002

- A Single Diagram – Maps Behaviour and Structure
- 2 Building Blocks and 10 basic relations
- Designed to “Systemic View” and “Concept Modelling”
- Simulation Ready



- ~130 Pages standard
- Published in late 2015
- Intended to “Automation Systems and Integration”
- Has the “power” of a ISO seal.

ISO/PAS 19450:2015

Automation systems and integration -- Object-Process Methodology
(Only available in English)

Abstract

ISO/PAS 19450:2015 specifies Object-Process Methodology (OPM) with detail sufficient for enabling practitioners to utilise the concepts, semantics, and syntax of Object-Process Methodology as a modelling paradigm and language for producing conceptual models at various extents of detail, and for enabling tool vendors to provide application modelling products to aid those practitioners.

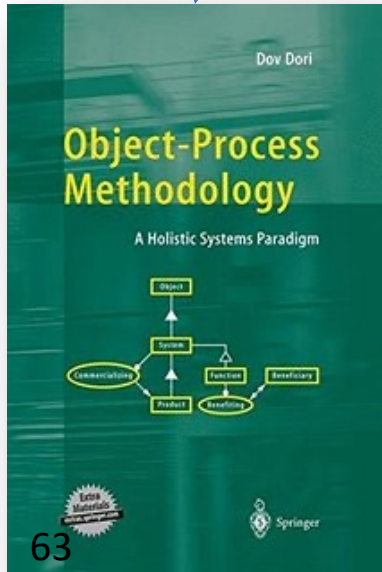
While ISO/PAS 19450:2015 presents some examples for the use of Object-Process Methodology to improve clarity, it does not attempt to provide a complete reference for all the possible applications of Object-Process Methodology.

FORMAT  LANGUAGE

☒ PDF English

☐ HTML English

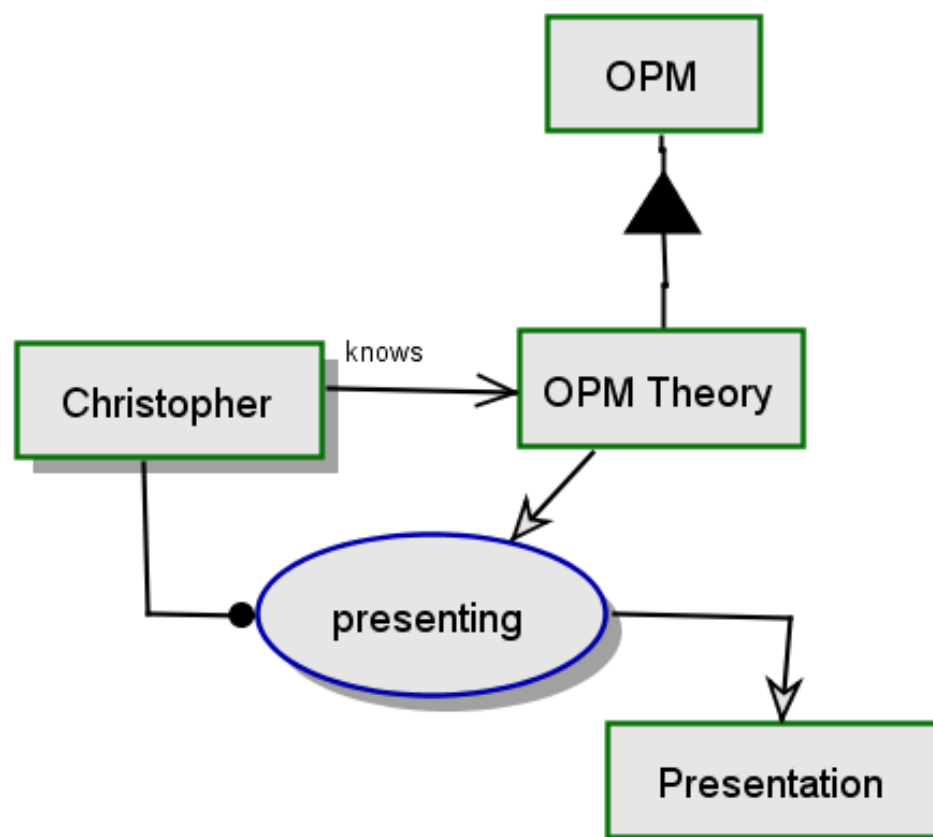
CHF 198 [Add to basket](#)



[Dori] Object-Process Theorem

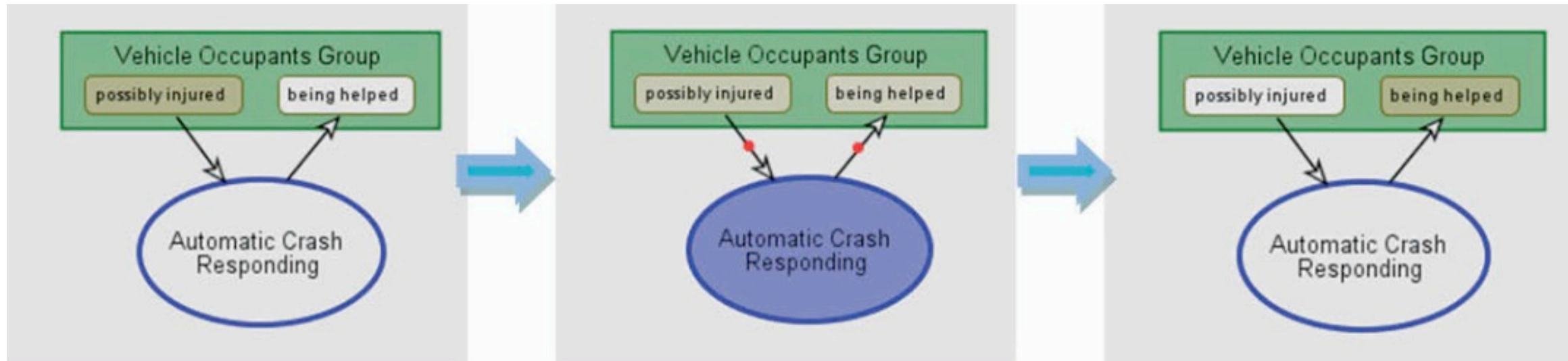
Objects with **states**, **processes**
and their relations among
them constitute a **necessary**
and **sufficient universal**
ontology to describe a **system**.

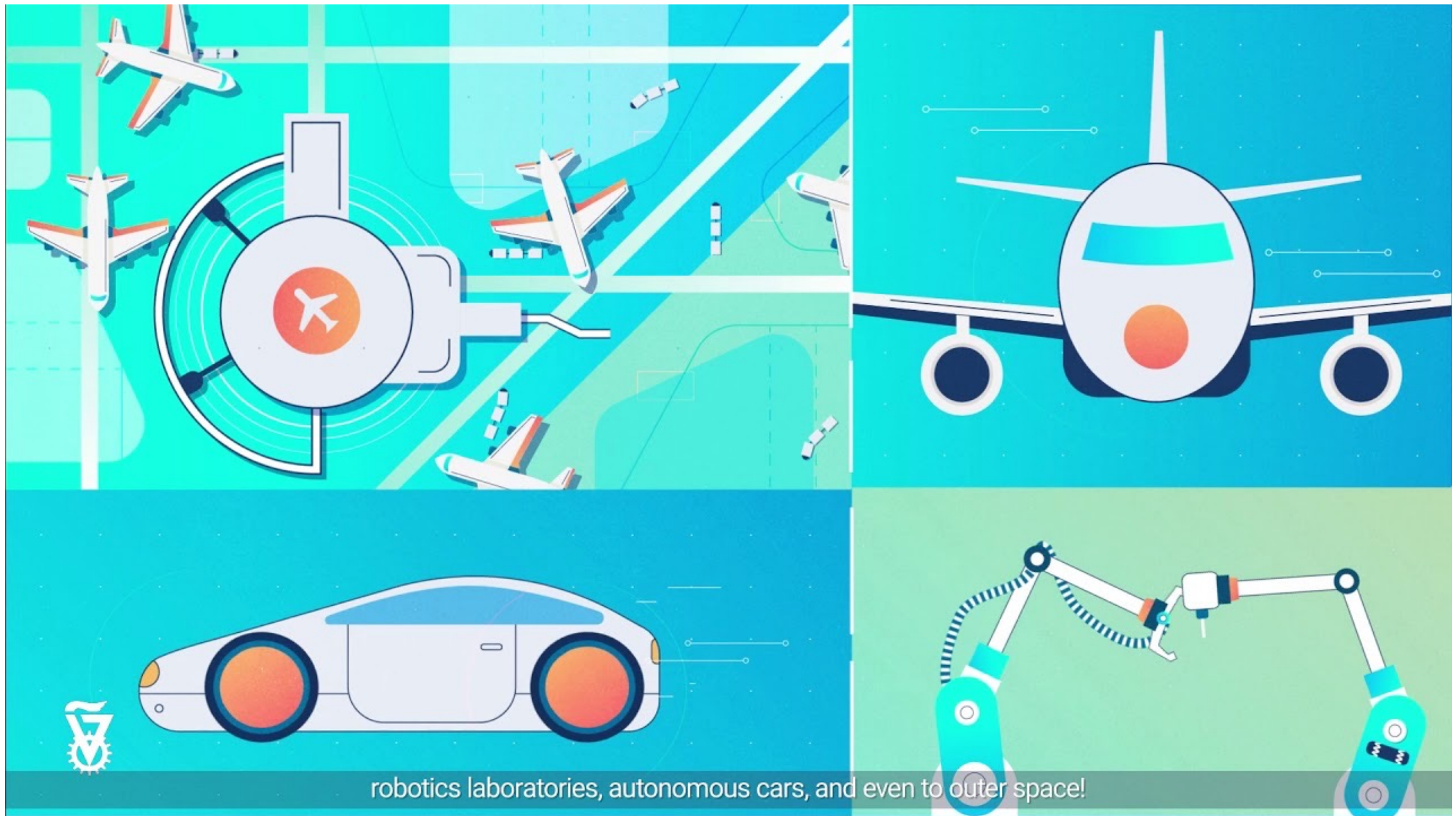
Dois canais cognitivos: visual-OPD e textual-OPL



Christopher is physical.
Christopher knows OPM Theory.
Christopher handles presenting.
OPM consists of OPM Theory.
presenting is physical.
presenting consumes OPM Theory.
presenting yields Presentation.

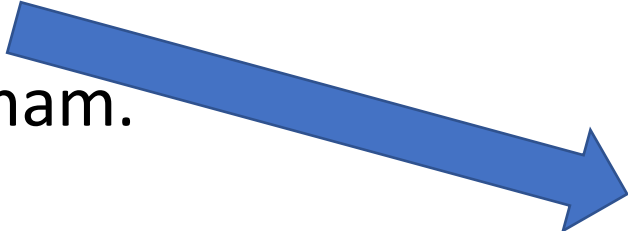
Simulação

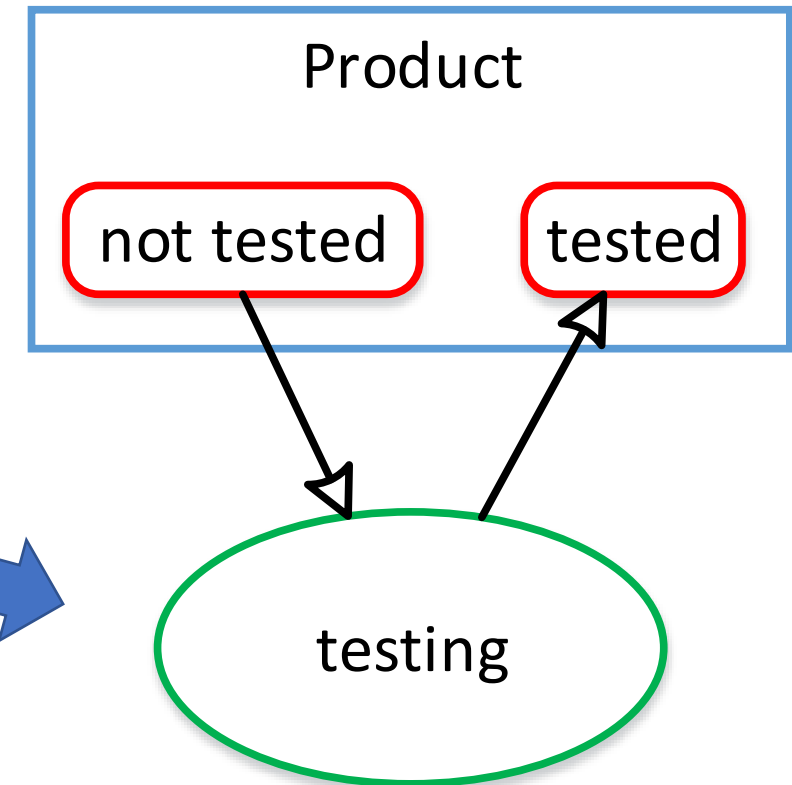




OPM só tem dois blocos

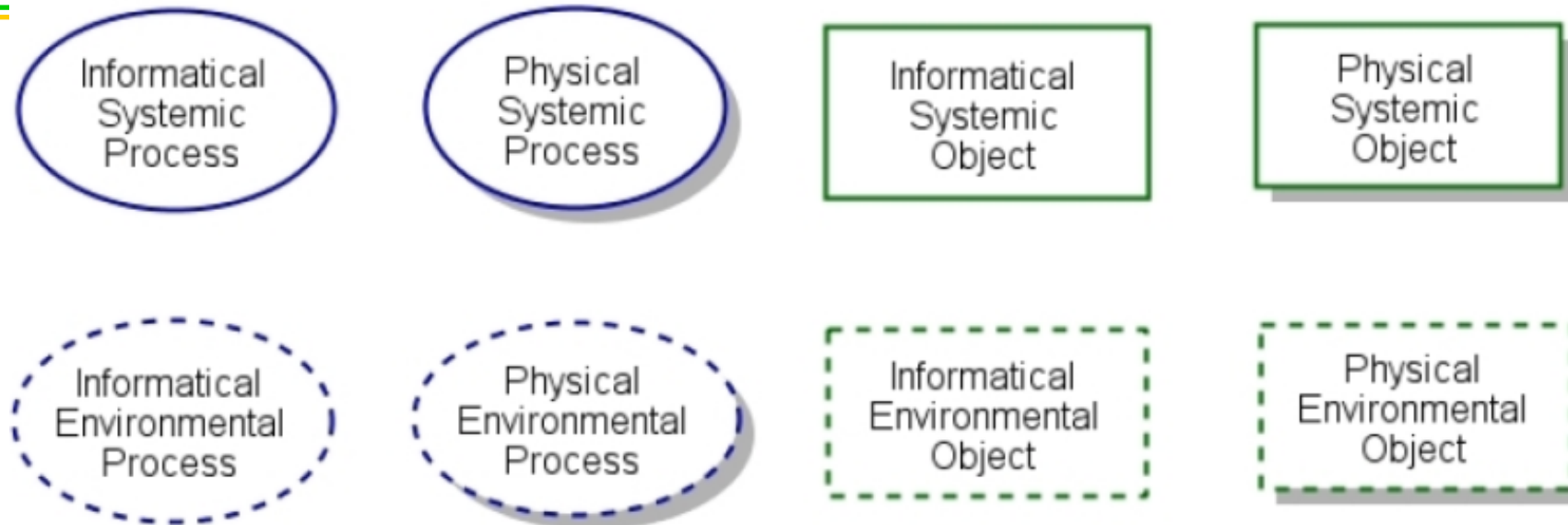
1. **Objetos com estados** 
As coisas que existem ou podem existir.

2. **Processos** 
As coisas que transformam.



Todos os outros elementos são relações entre coisas.

Essência e natureza



Informatical Systemic Process is an informatical and systemic process.

Physical Systemic Process is a physical and systemic process.

Informatical Systemic Object is an informatical and systemic object.

Physical Systemic Object is a physical and systemic object.

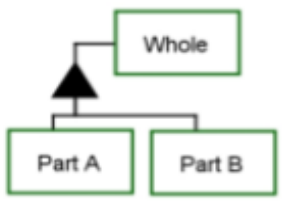
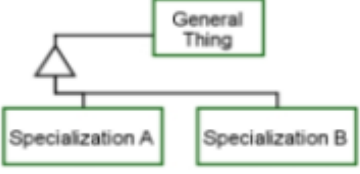
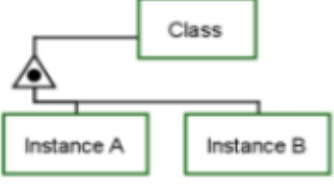
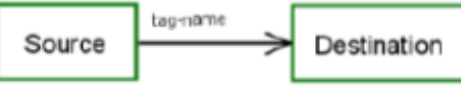
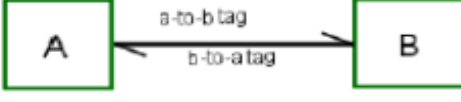
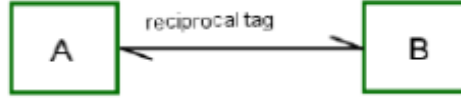
Informatical Environmental Process is an informatical and environmental process.

Physical Environmental Process is a physical and environmental process.

Informatical Environmental Object is an informatical and environmental object.

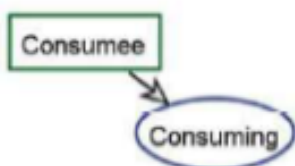
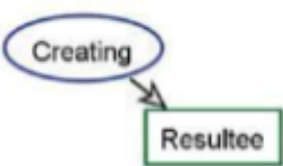
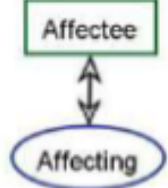
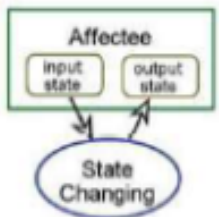
Physical Environmental Object is a physical and environmental object.

Resumo de links estruturais

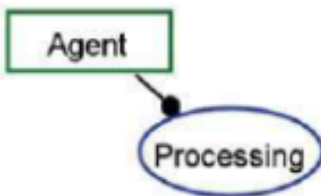
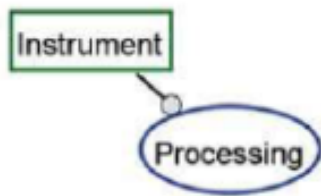
Aggregation-Participation		Whole consists of Part A and Part B.	—
Exhibition-Characterization		Exhibitor exhibits Attribute A as well as Operation B.	—
Generalization-Specialization		—	Specialization A and Specialization B are General Thing.
Classification-Instantiation		—	Instance A and Instance B are instances of Class.
Unidirectional tagged [Unidirectional null tagged]		Source tag-name Destination. [Source relates to Destination.]	
Bidirectional tagged		A a-to-b tag B. B b-to-a tag A.	
Reciprocal tagged [Reciprocal null tagged]		A and B are reciprocal tag. [A and B are related.]	

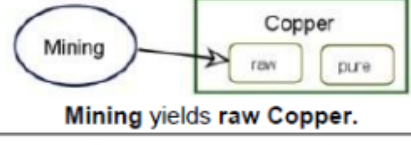
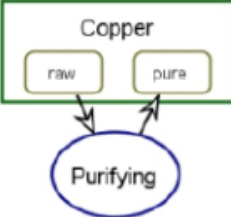
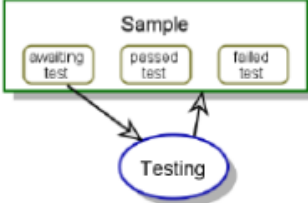
Resumo de links transformacionais

Procedural transforming links

consumption link	result link	effect link	in-out link pair
 Consuming consumes Consume.	 Creating yields Result.	 Affecting affects Affect.	 State Changing changes Affect from input state to output state.

Procedural enabling links

agent link	instrument link
 Agent handles Processing.	 Processing requires Instrument.

Name	Semantics	Sample OPD & OPL	Source	Destination
State-specified consumption link	The process consumes the object if and only if the object is in the specified state.	 Eating consumes edible Food.	consumee state	process
State-specified result link	The process generates the object in the specified state.	 Mining yields raw Copper.	process	resultee state
Input-output-specified effect link pair (consisting of one state-specified <i>input link</i> and one state-specified <i>output link</i>)	The process changes the object from a specified input state via the <i>input link</i> to a specified output state via the <i>output link</i> .	 Purifying changes Copper from raw to pure.	affectee source state	affecting process
			affecting process	affectee destination state
Input-specified effect link pair (consisting of one state-specified <i>input link</i> and one state-unspecified <i>output link</i>)	The process changes the object from a specified input state to any output state.	 Testing changes Sample from awaiting test.	affectee source state	affecting process
			affecting process	affectee
Output-specified effect link pair (consisting of one state-unspecified <i>input link</i> and one state-specified <i>output link</i>)	The process changes the object from any input state to a specified output state.	 Cleaning & Painting changes Engine Hood to painted.	affectee	affecting process
			affecting process	affectee destination state

Name	Semantics	Sample OPD & OPL	Source	Destination
State-specified agent link	The human agent enables the process provided she is at the specified state.	 Healthy Miner handles Copper Mining.	agent state	enabled process
State-specified instrument link	The process requires the instrument at the specified state.	 Copper Mining requires operational Drill.	instrument state	enabled process



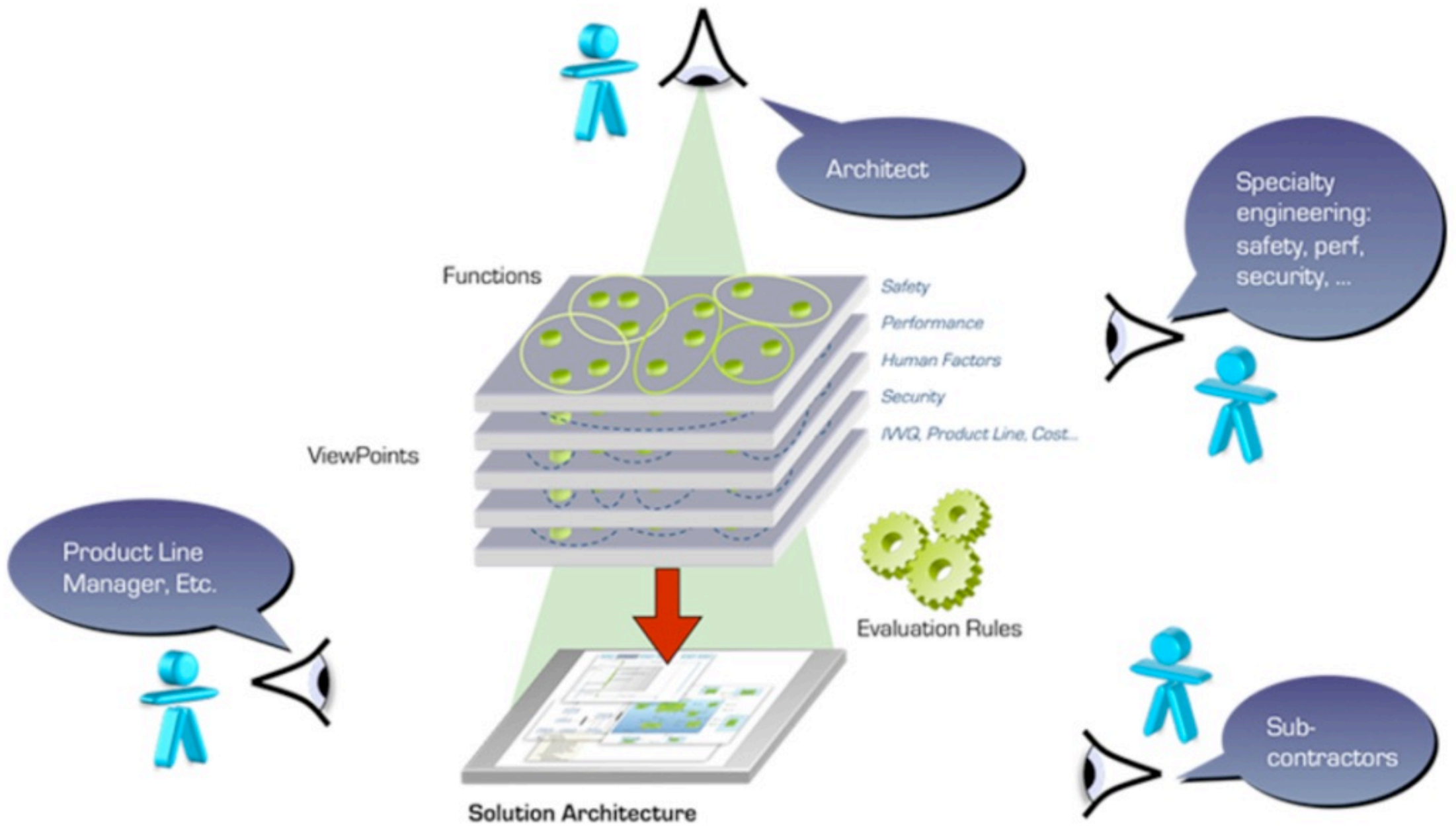


ARCADIA



Evoluindo: como nós podemos descrever arquiteturas?

- Analisando necessidades de stakeholders
- Analisando propriedades emergentes que interagem com as necessidades dos stakeholders
- Criação de arquitetura funcional
- Realização da seleção de fabricantes
- Distribuir compras pela cadeia de suprimento



XP Z67-140 - ARCADIA

https://norminfo.afnor.org/norme/XP%20Z67-140/tech...

AFNOR norm'info Recherche : mot clé, sujet, n° norme

Accédez aux tutoriels Identifiez-vous

< Retour

NORME EN REEXAMEN

Technologies de l'information - ARCADIA - Méthode pour l'ingénierie des systèmes soutenue par son langage de modélisation conceptuel - Description Générale - Spécification de la méthode de définition de l'ingénierie et du langage de modélisation XP Z67-140

Suivi par la commission : [Ingénierie et qualité du logiciel et des systèmes](#)

Origine des travaux : Française

Type : Expérimentale

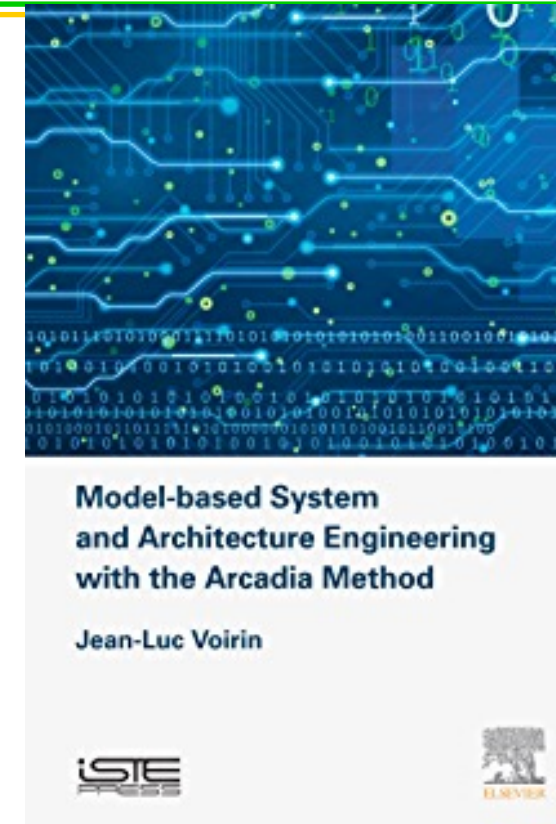
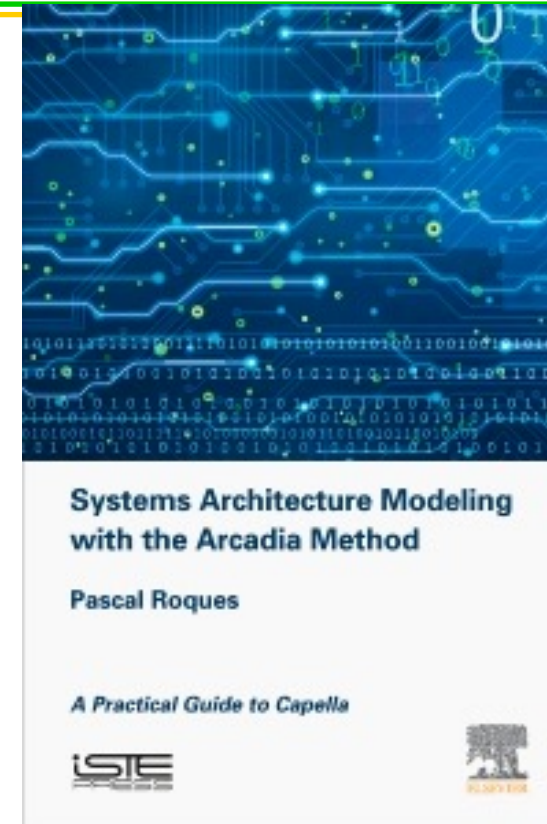
Motif : Nouveau document

Résumé : La méthode ARCADIA peut être appliquée à la définition de la conception de tout type de système, en se concentrant sur la description et l'évaluation des propriétés de conception (coût, performance, sécurité, réutilisation, consommation, poids ...).

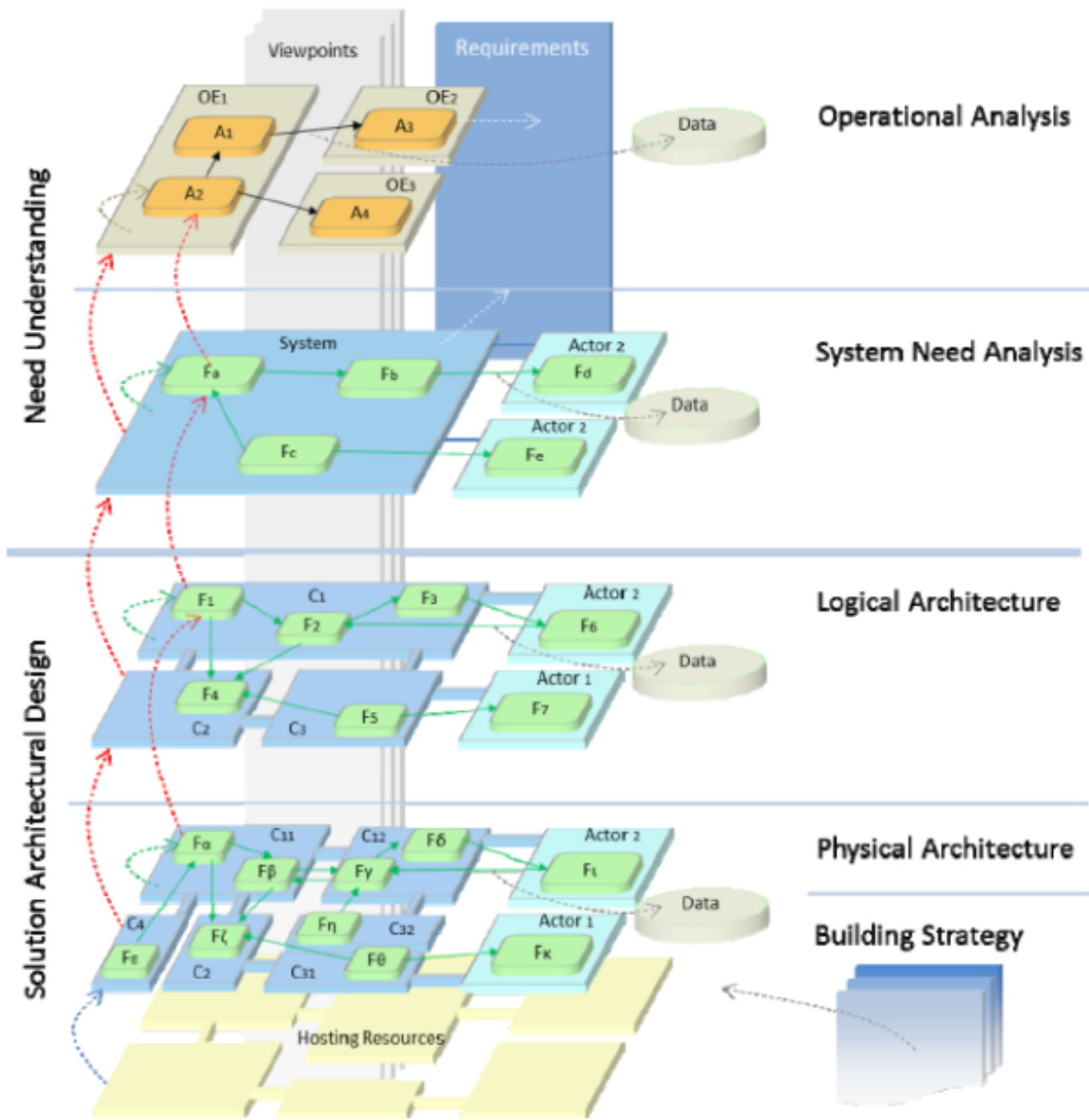
Je veux en savoir plus J'accède à la consultation

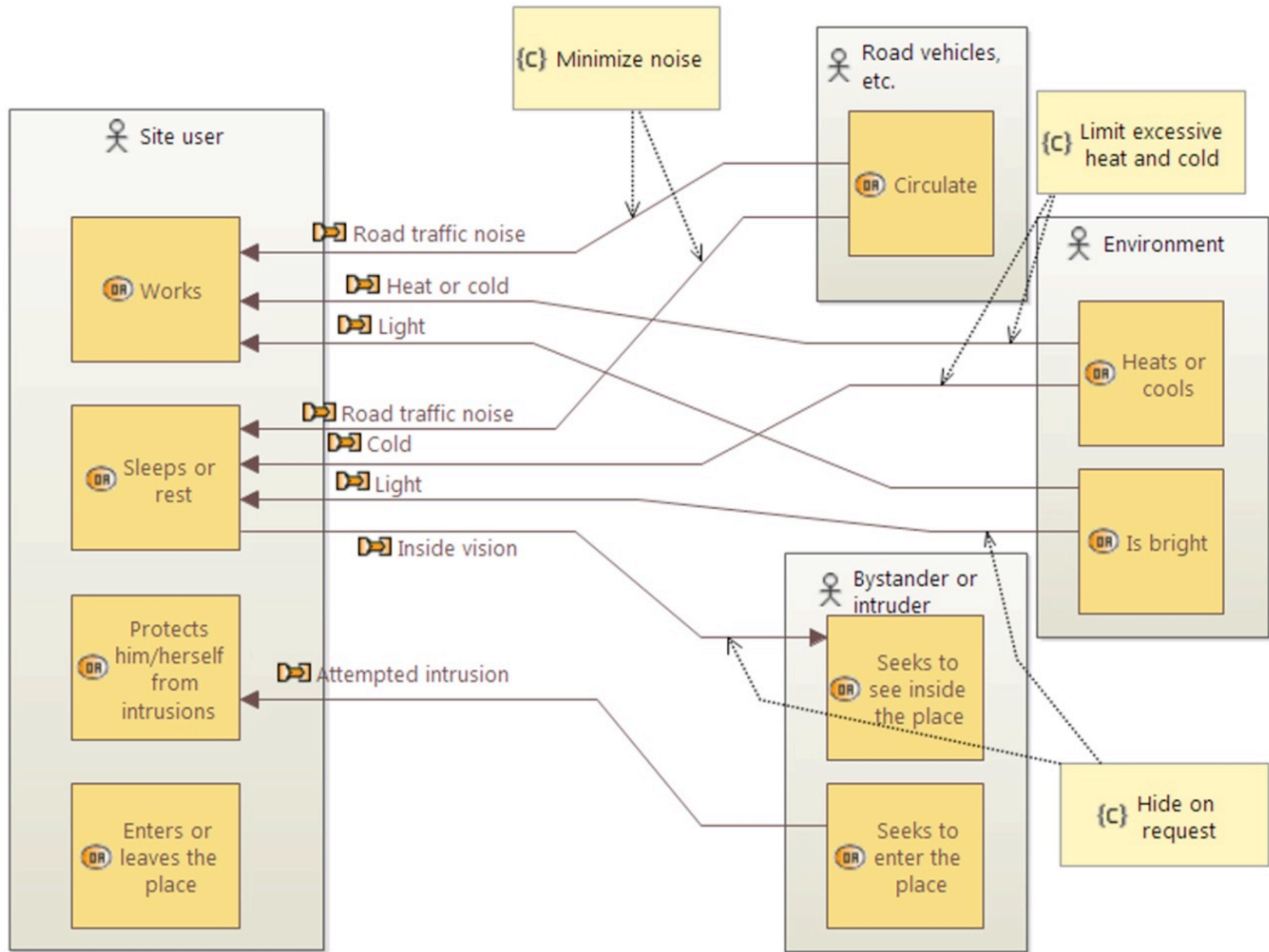
Vie de la norme

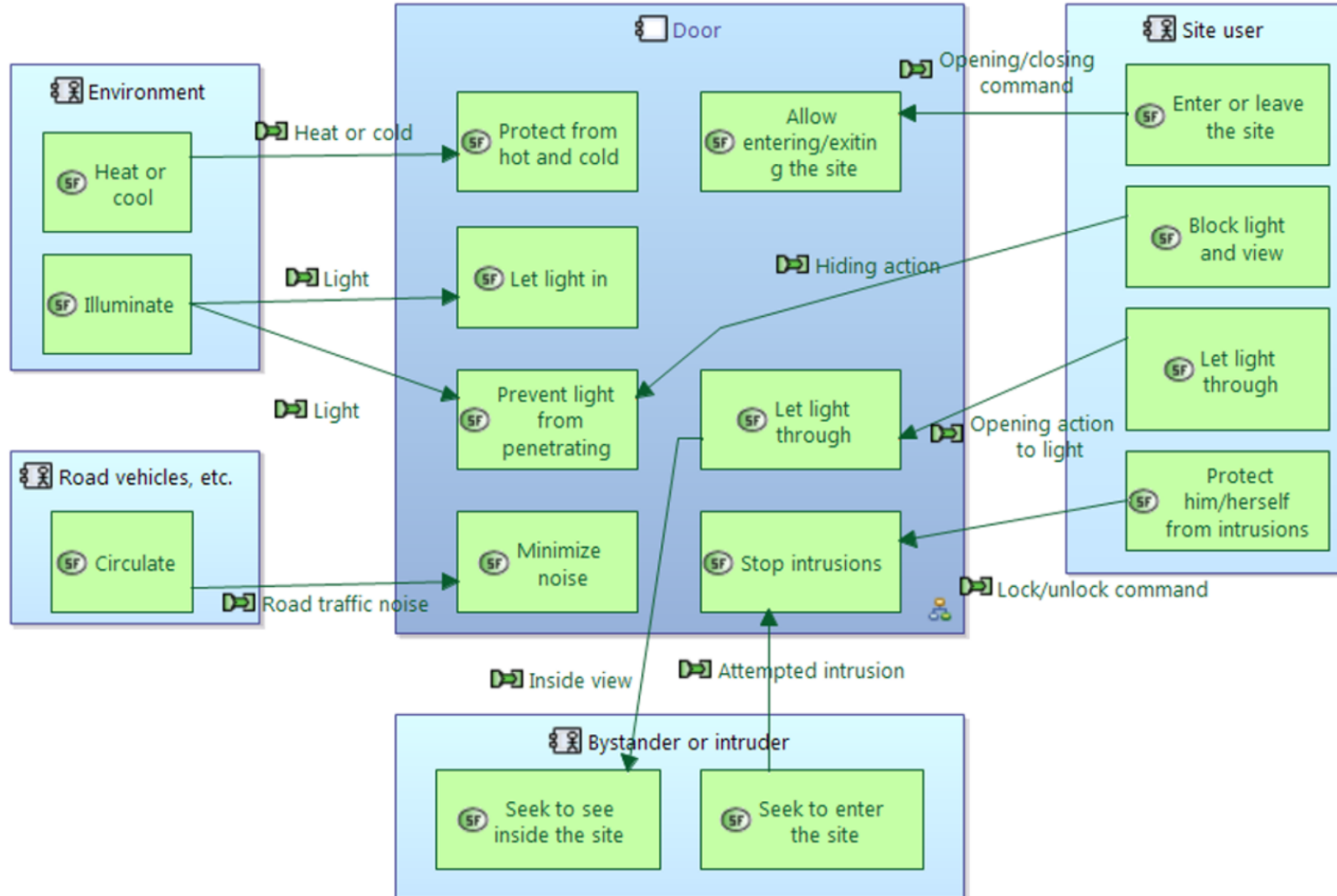
Norme En conception	Norme Enquête publique	Norme Publiée	Norme En réexamen
Inscrite le : 23/11/2017		Publiée le : 07/03/2018	En cours

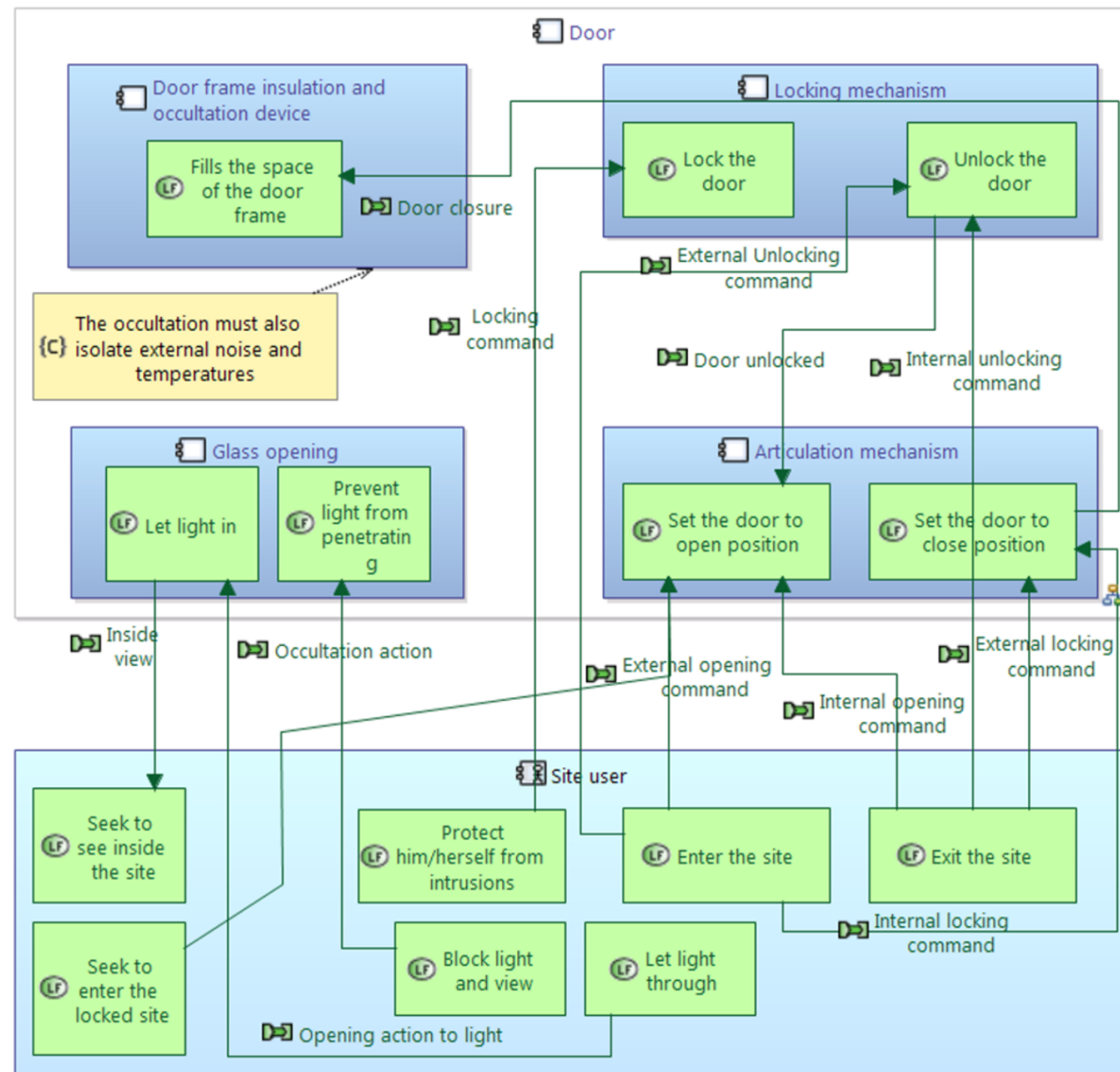


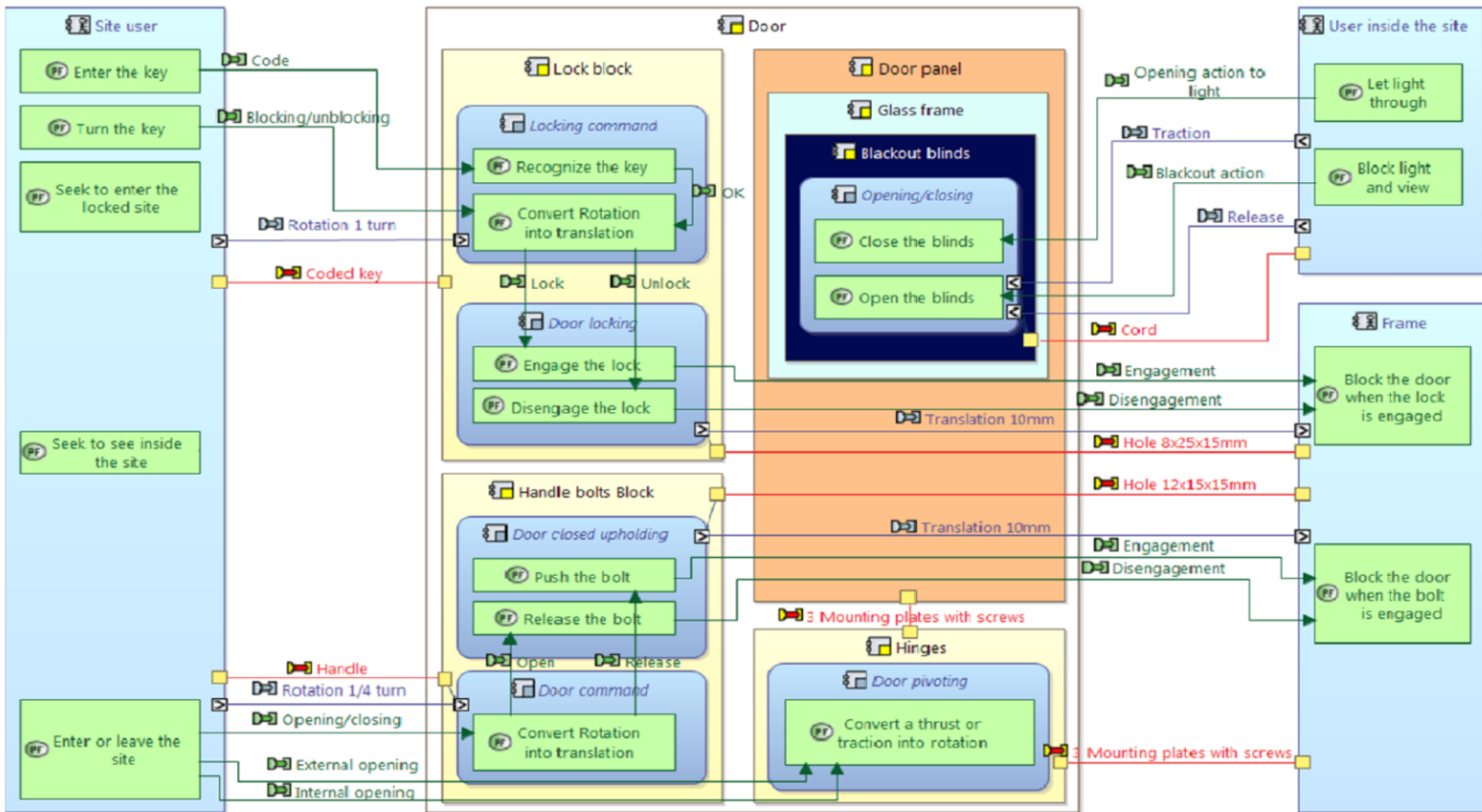
[Norme XP Z67-140 \(afnor.org\)](https://norminfo.afnor.org)

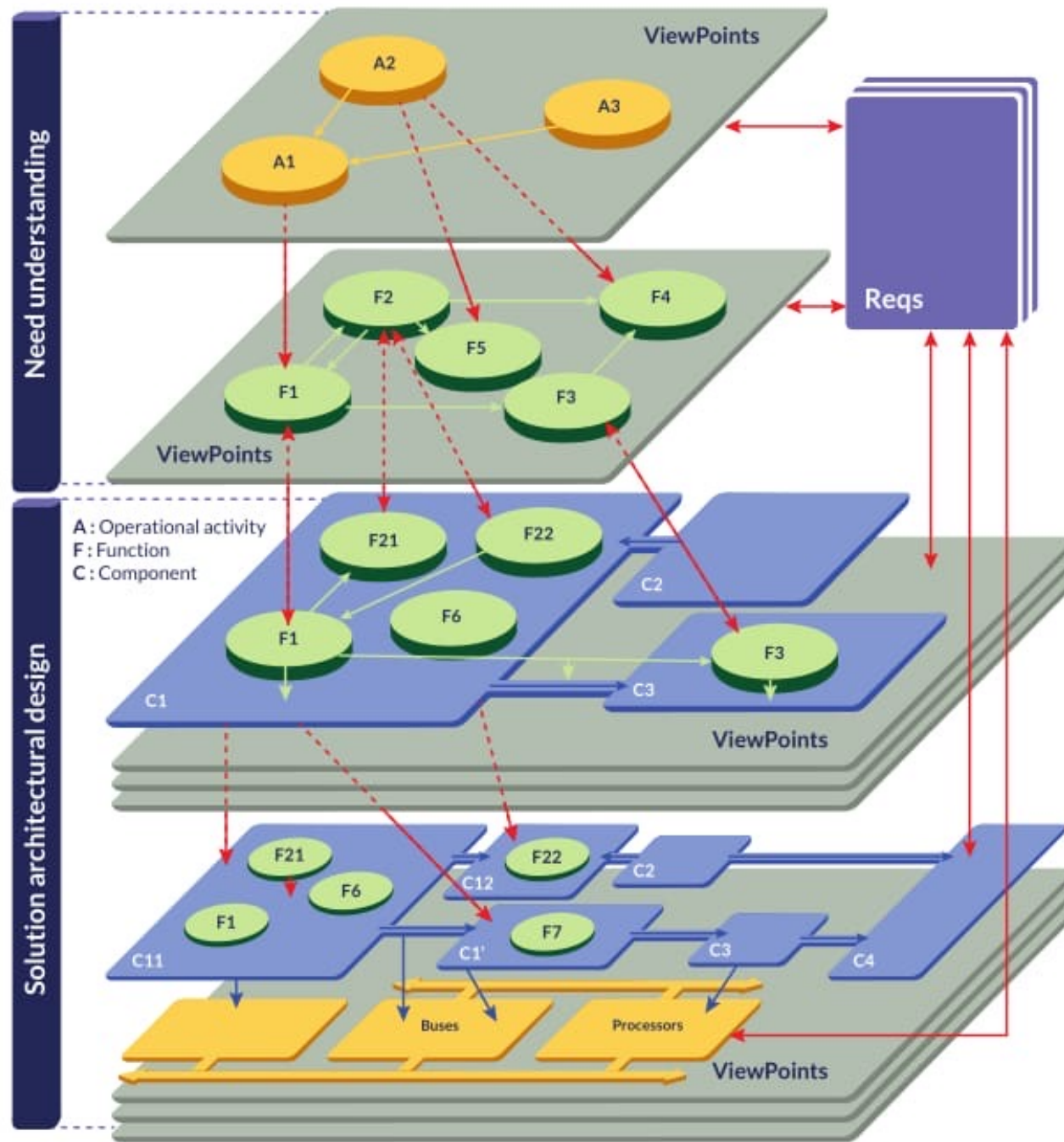










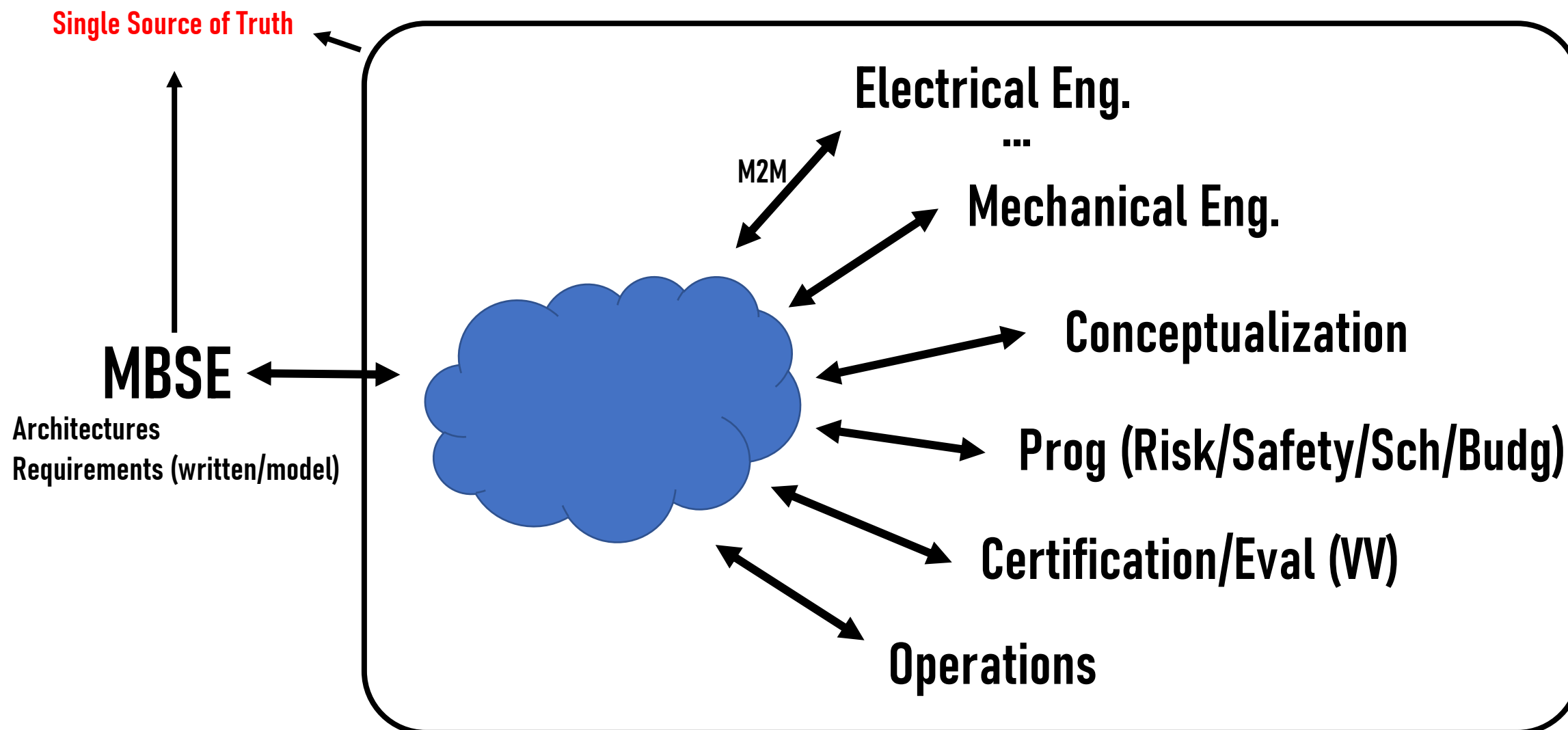




Considerações sobre ESBM



ESBM E AS OUTRAS ENGENHARIAS





DCA-400-6

Fase de Definição



MINISTÉRIO DA DEFESA
COMANDO DA AERONÁUTICA



LOGÍSTICA

DCA 400-6

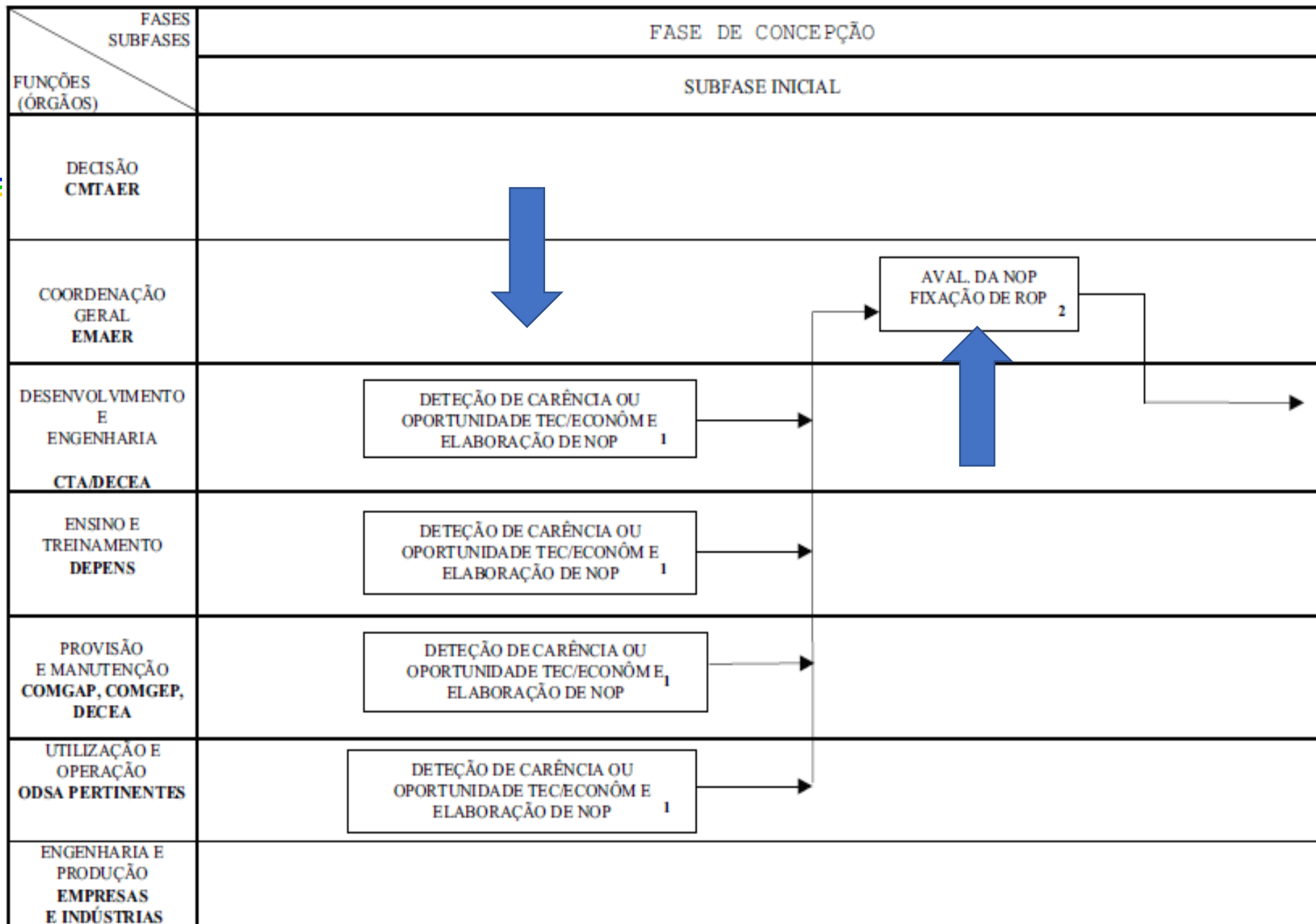
CICLO DE VIDA DE SISTEMAS E MATERIAIS DA
AERONÁUTICA

2007

- A presente Diretriz tem por finalidade ordenar o planejamento e a execução das fases e principais eventos do Ciclo de Vida de Sistemas e Materiais da Aeronáutica, bem como regular tecnicamente a atuação, a interação e a responsabilidade dos Órgãos e Sistemas do COMAER que intervêm no processo.



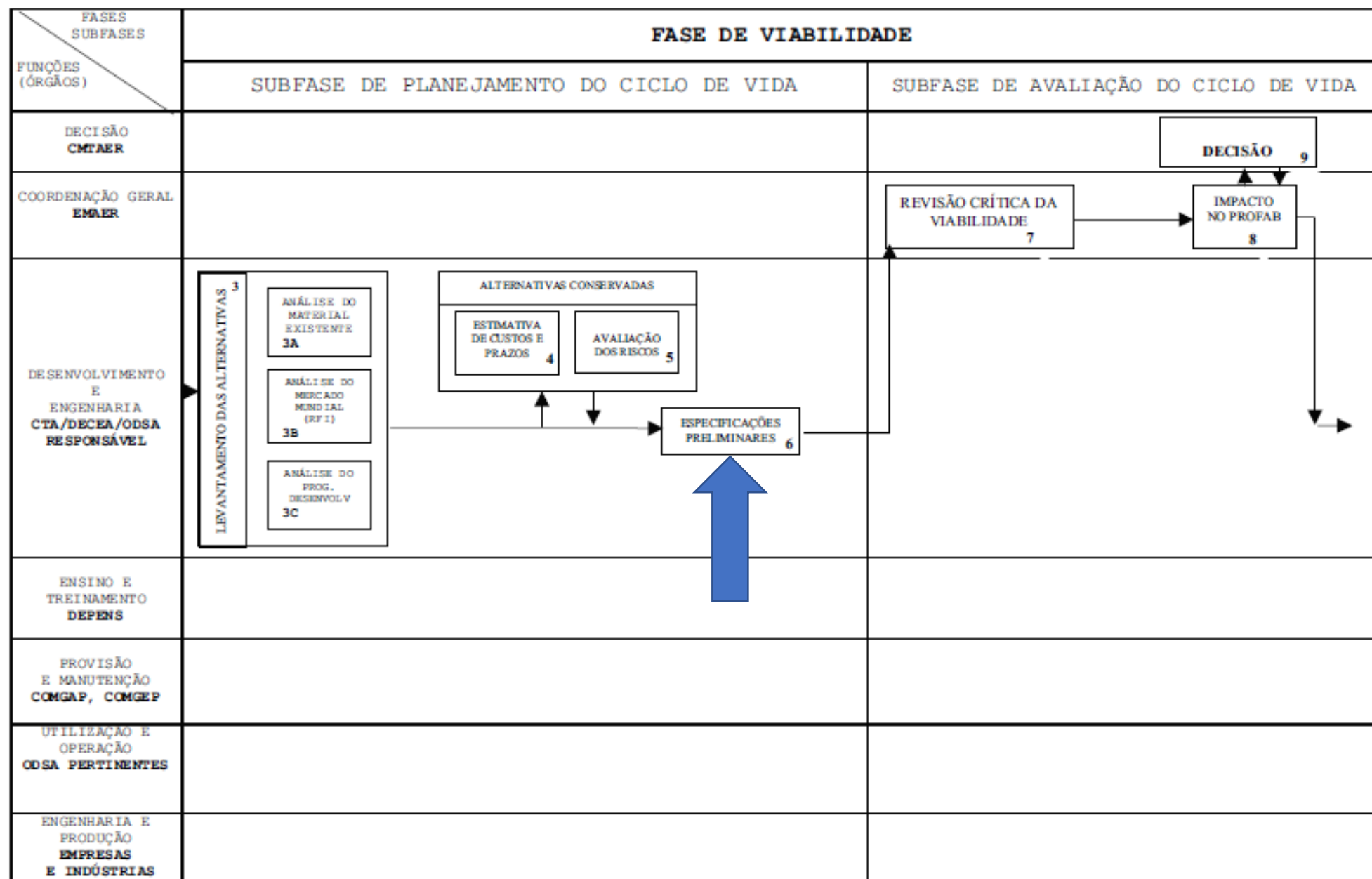
DCA-400-6

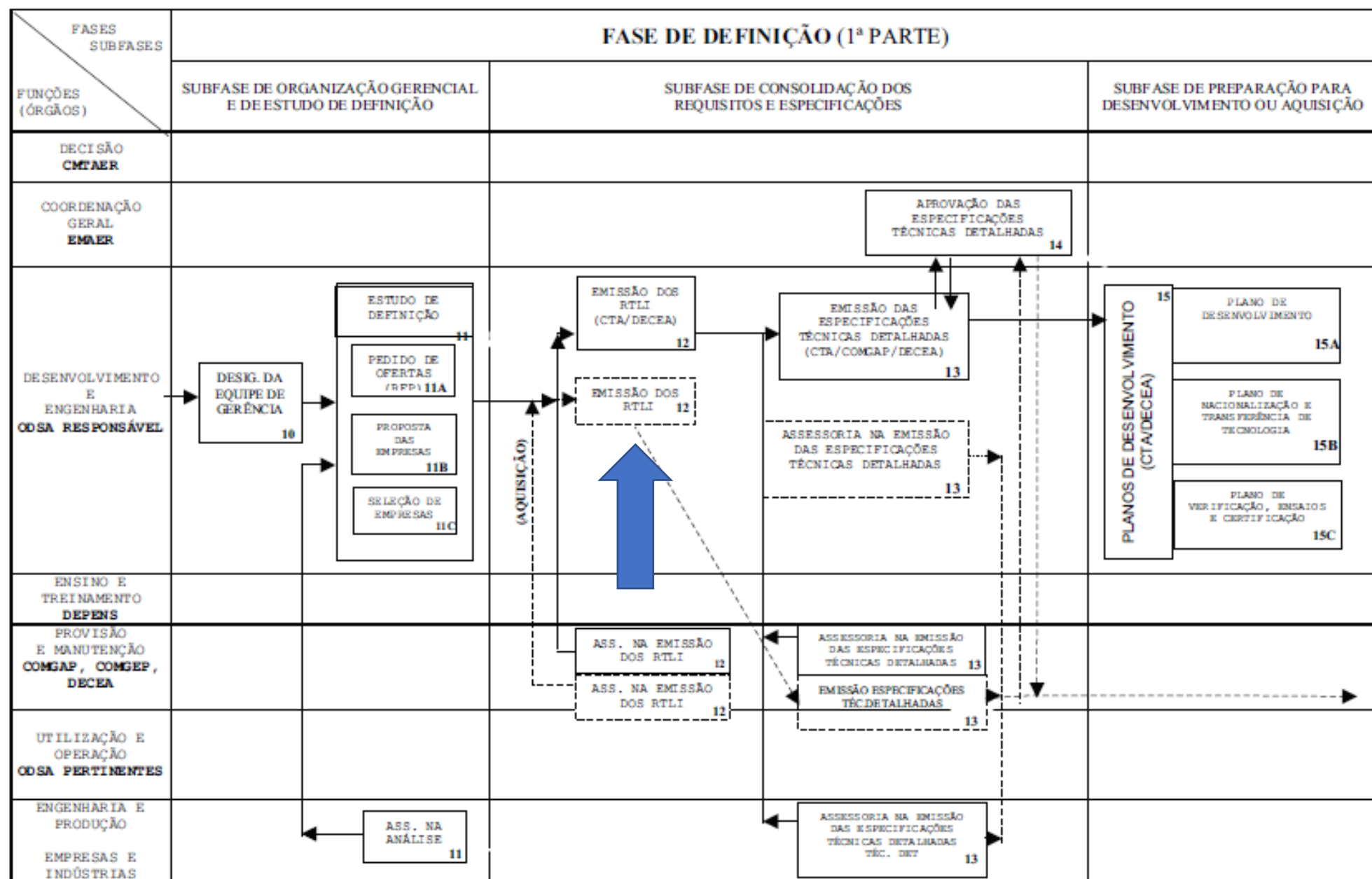




DCA-400-6

88







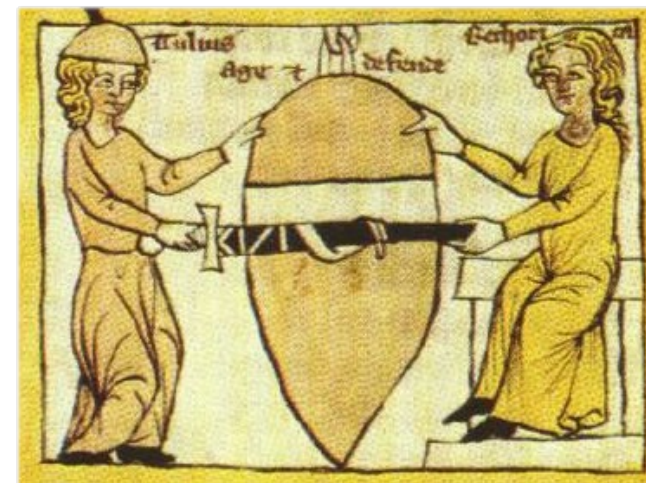
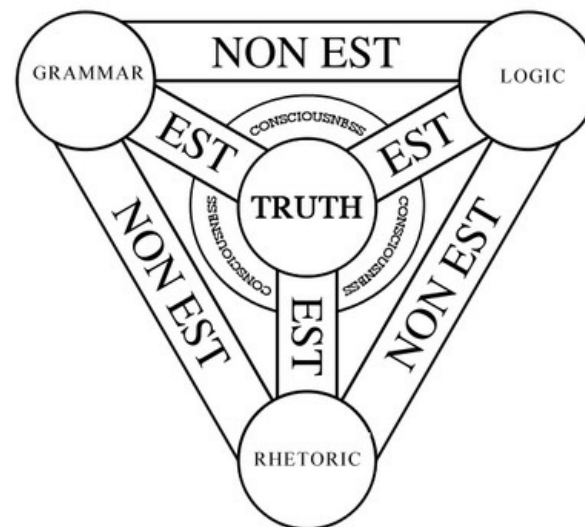
MMMMF



Antes de começar

- ESBM é a **formalização de ES** para que ferramentas “falem” ES.
 - (PPoint / Excel / Visio não representam ES)
- Não existe **bala de prata**
- A própria Engenharia é um wicked problem
- Indústrias usam adaptações da ES.
 - Colocando restrições da cultura organizacional
- Academia pesquisa **ES pela ES**
 - Olhando a filosofia da ES, metáforas, transformações e generalizações
- Então, talvez, uma boa ideia seja hibridizar alguns métodos para “ver o que sai”

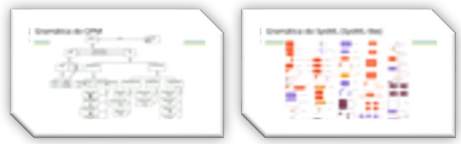
Começando do início --- Trivium



Gramática - Lógica - Retórica

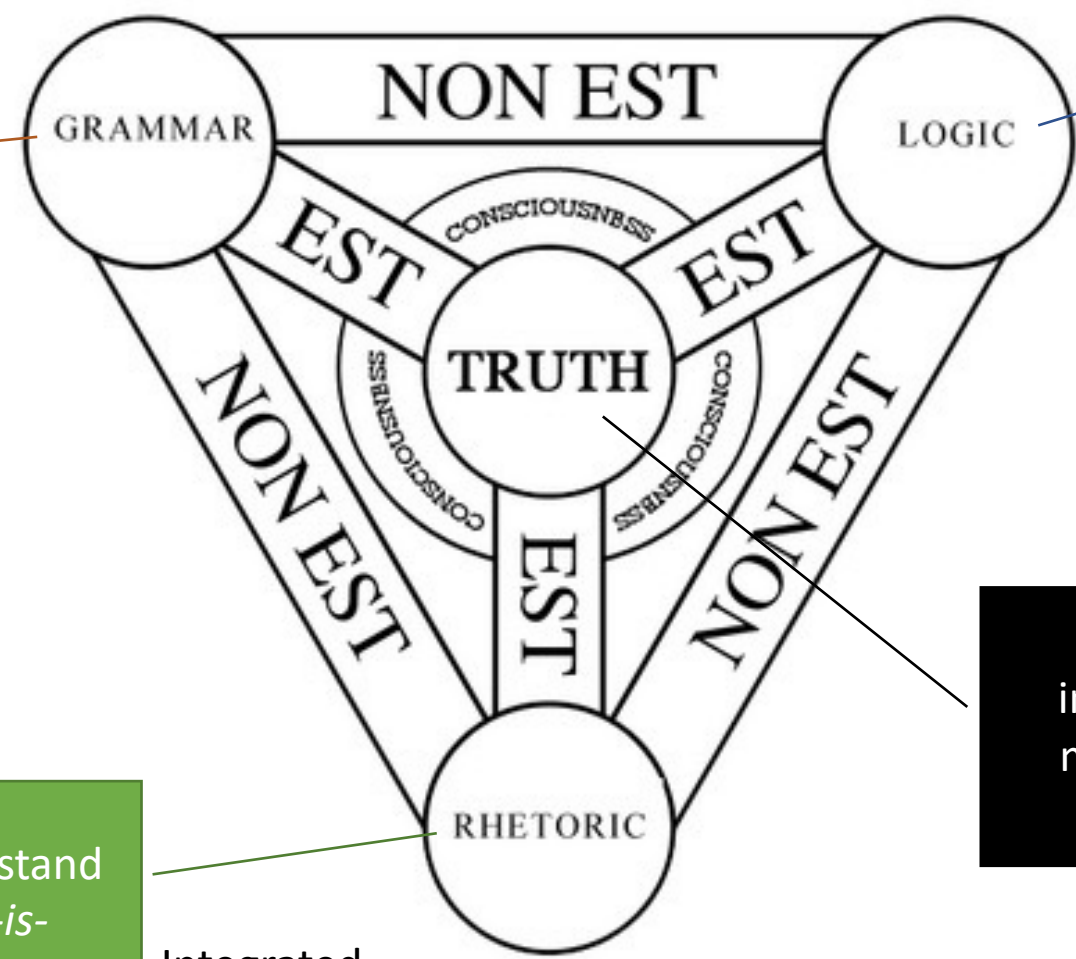
We need to understand the *"thing as-it-is-symbolized"*

SysML and OPM symbols



We need to understand the *"thing as-it-is-communicated"*

Integrated Framework



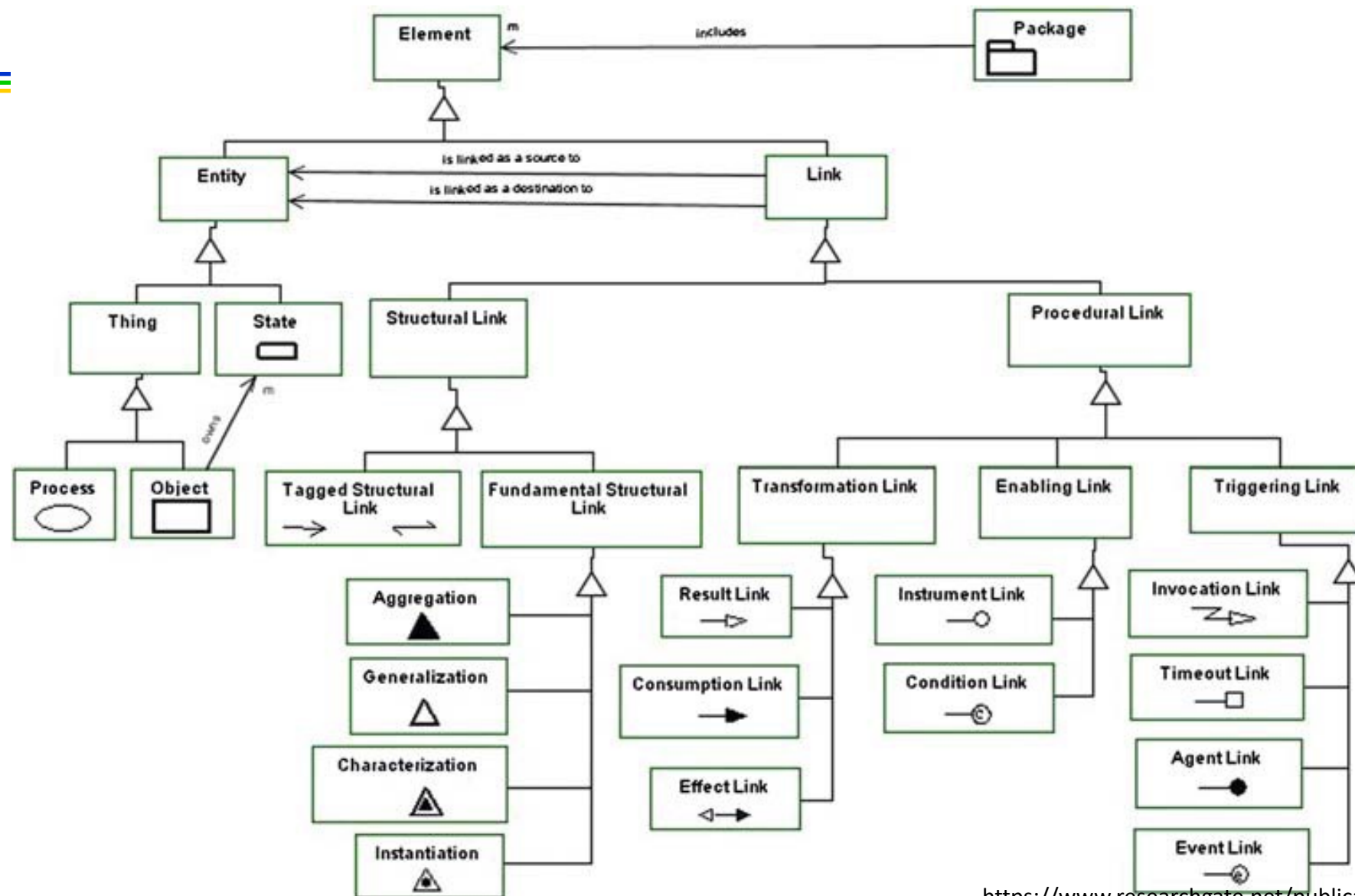
We need to understand the *"thing as-it-is-known"*

SysML and OPM Diagram Interpretation

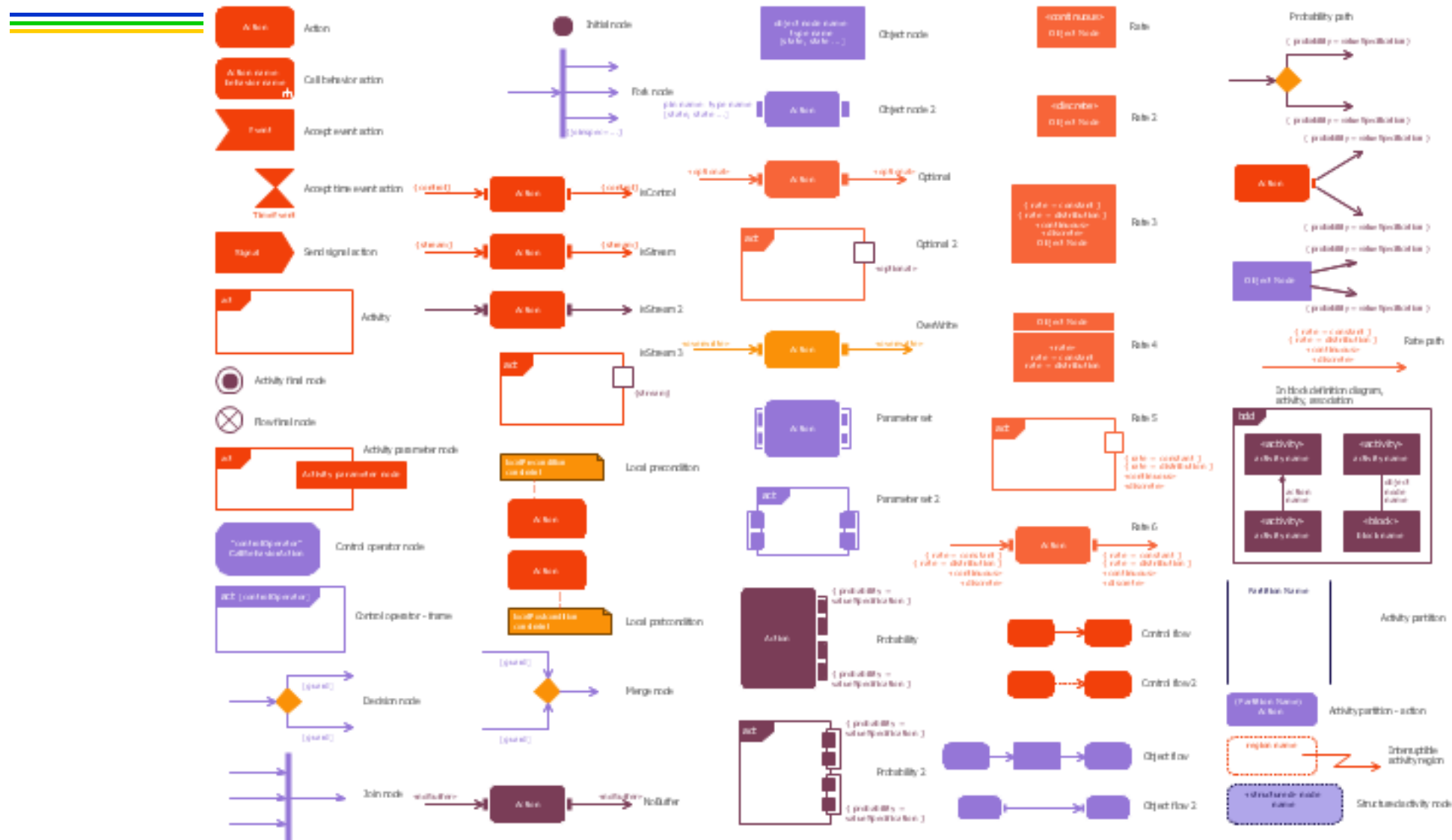


The integrated information (on the model) is the single source of truth

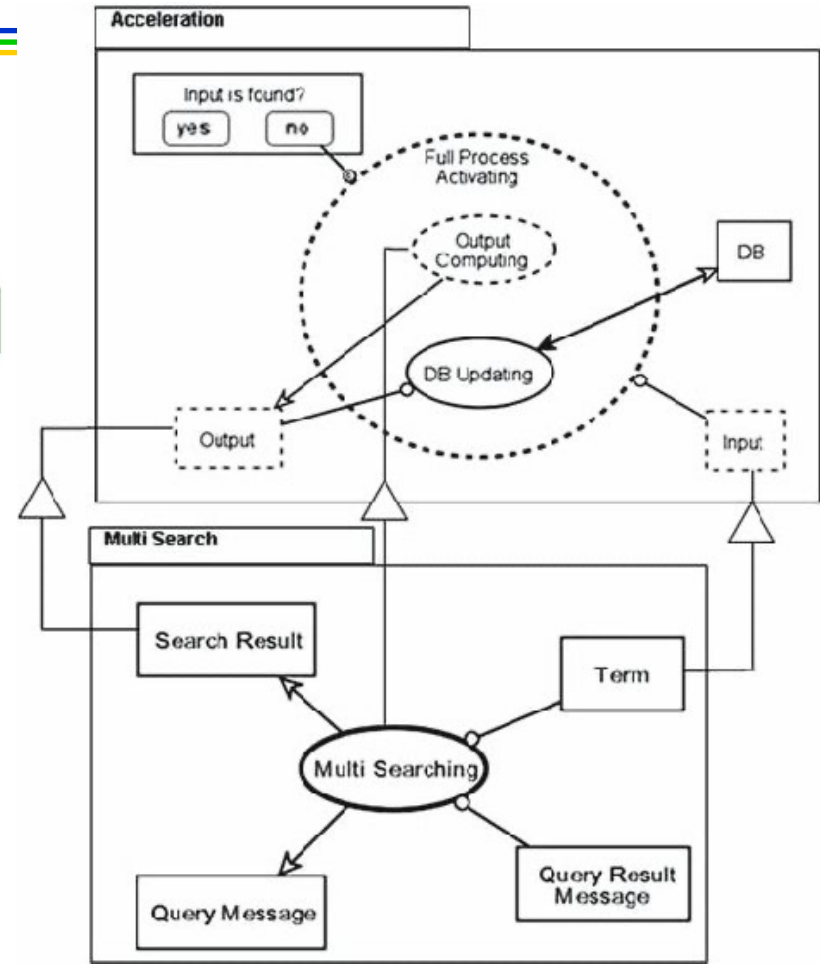
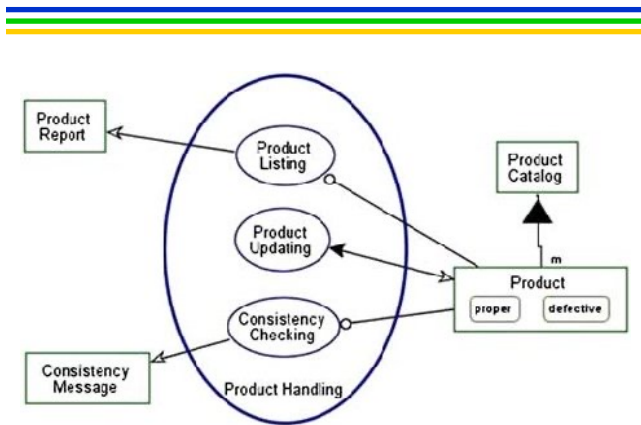
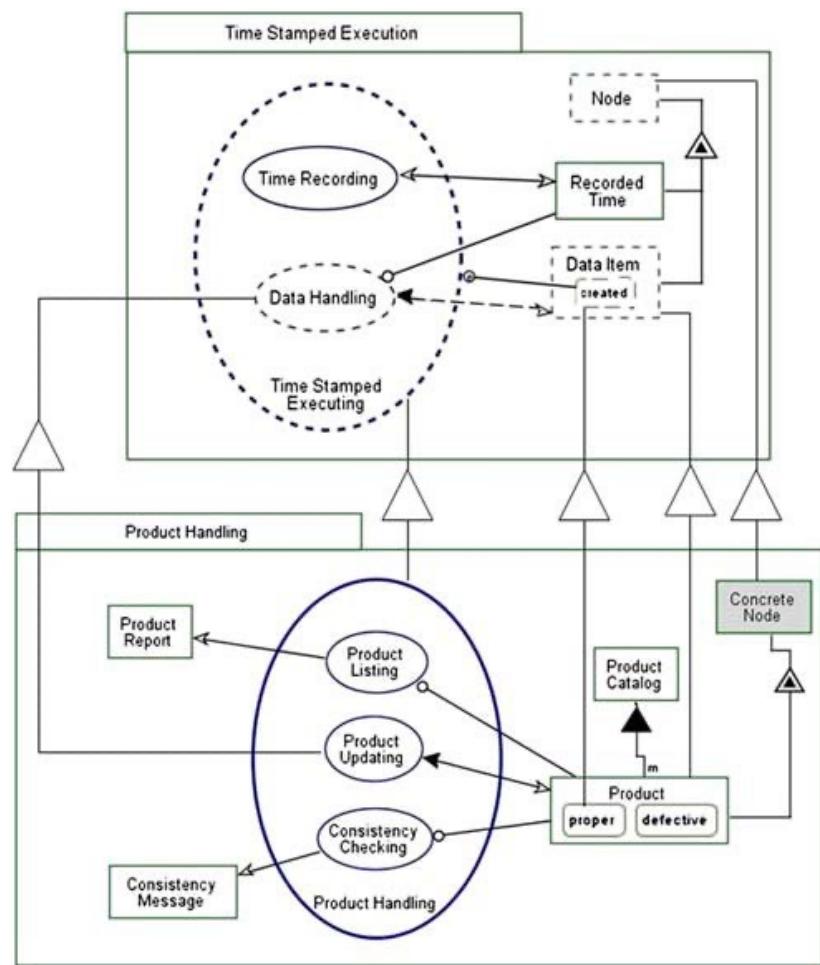
Gramática do OPM



Gramática do SysML (SysML-like)

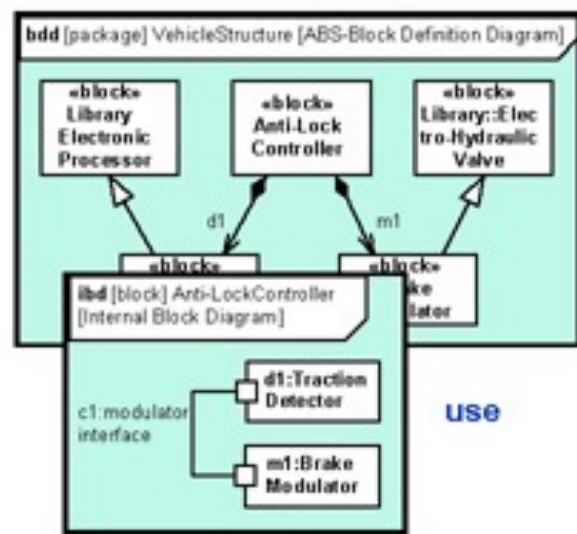


Lógica do OPM

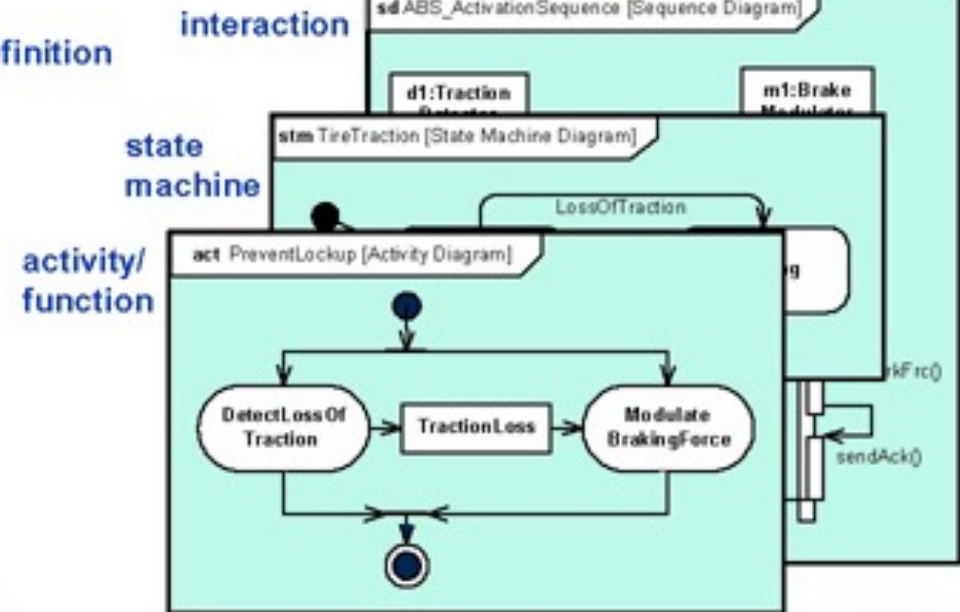


Lógica do SysML (SysML-like)

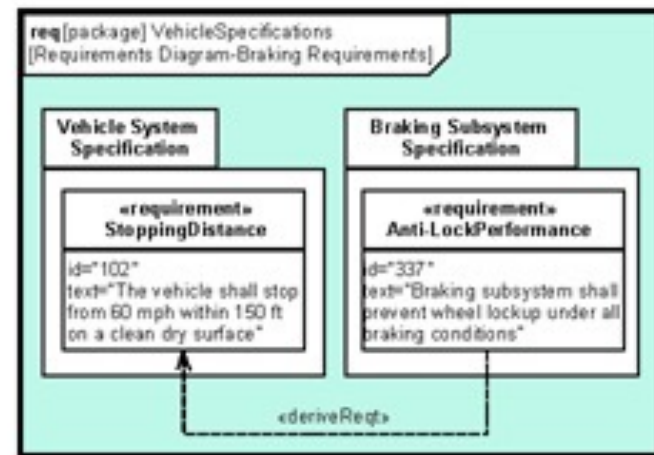
1. Structure



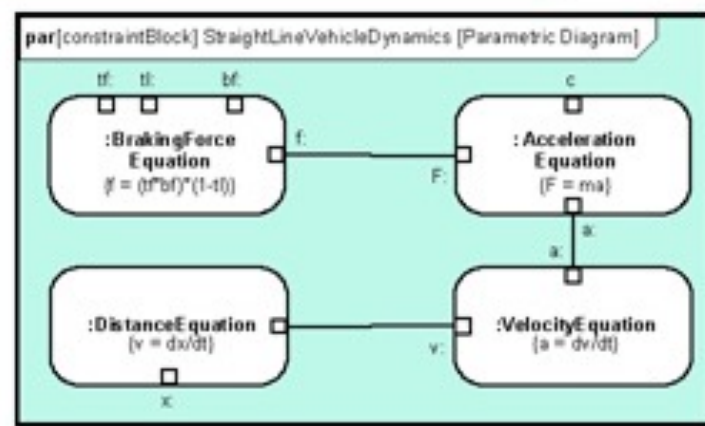
2. Behavior



3. Requirements

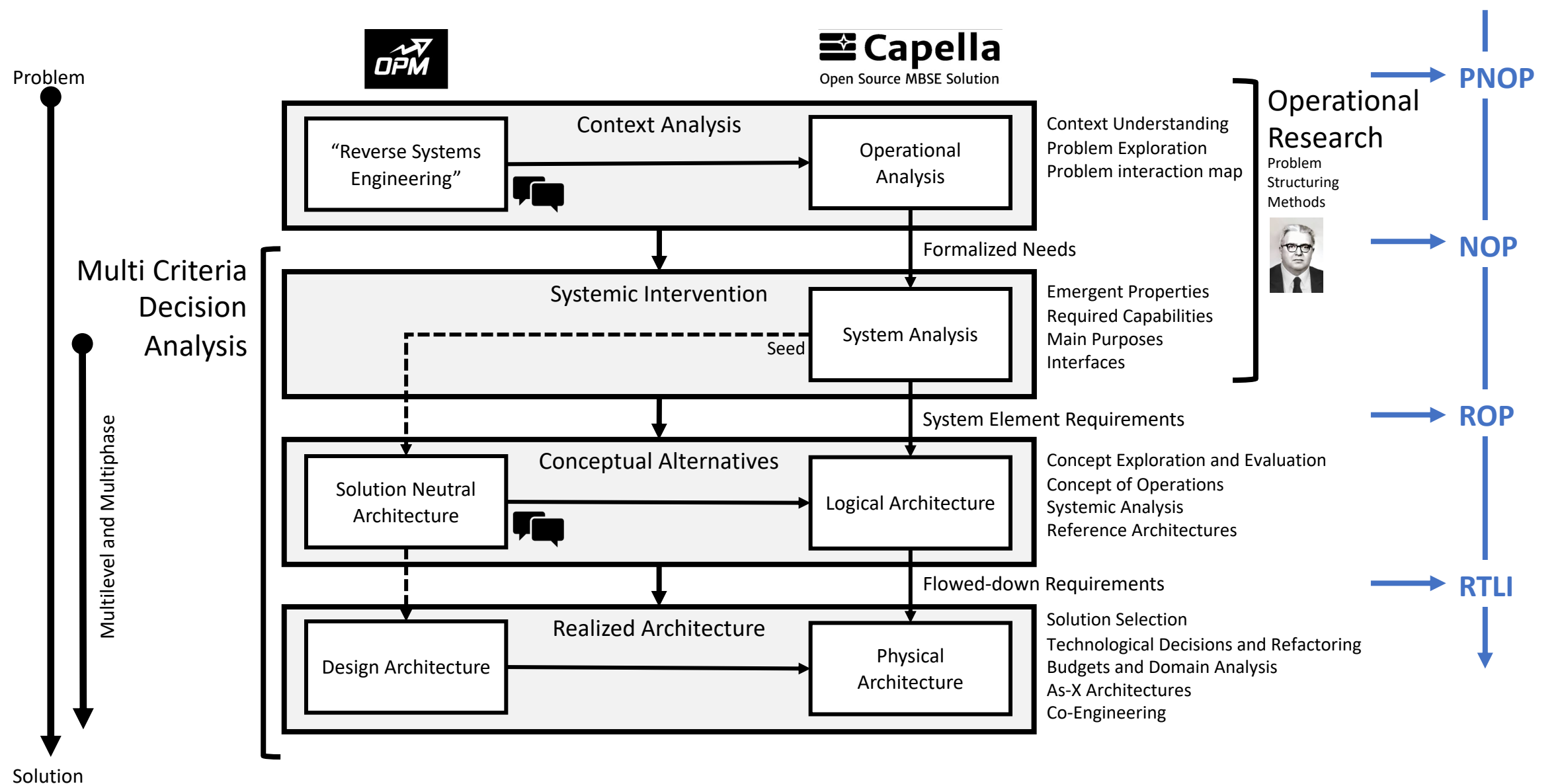


4. Parametrics





MMMF (Multi Methodology MBSE Framework)



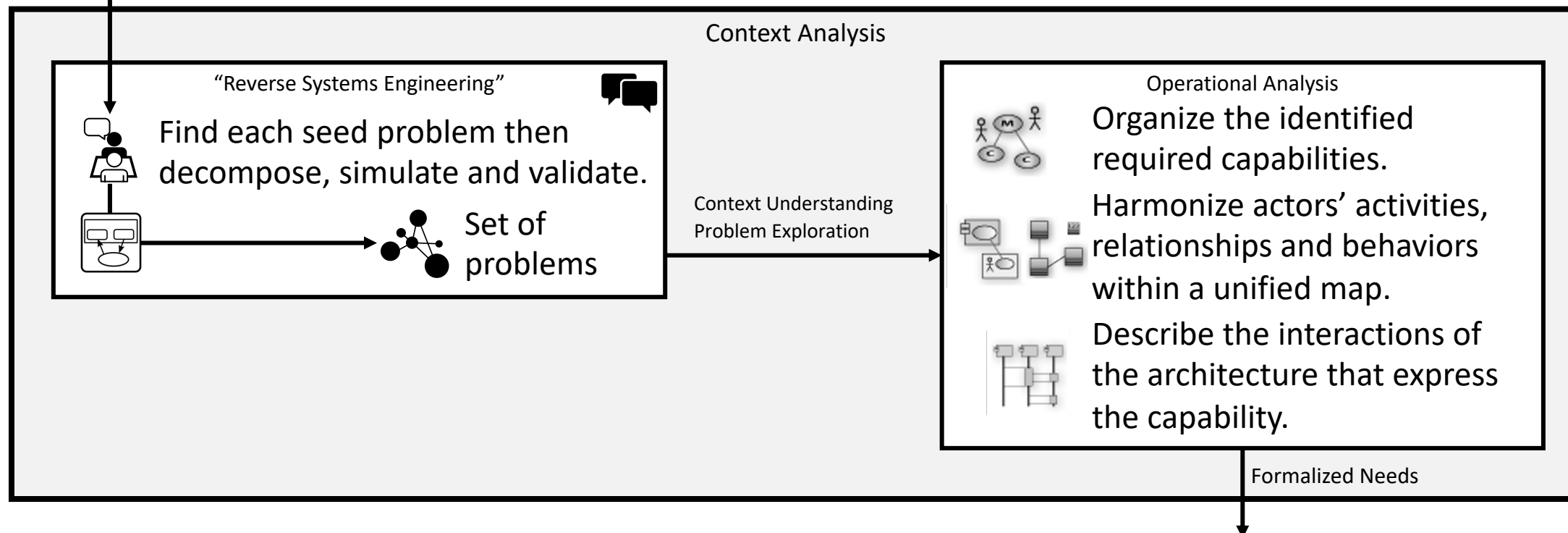
Análise do Contexto



The initial perception: Free interpretation of the problem.



Learning the domain to frame your knowledge



Usando ES para entender o problema



The initial perception: Free interpretation of the problem.



Learning the domain to frame your knowledge

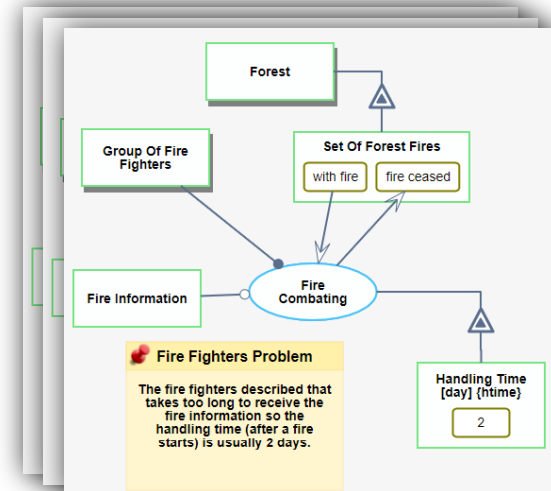
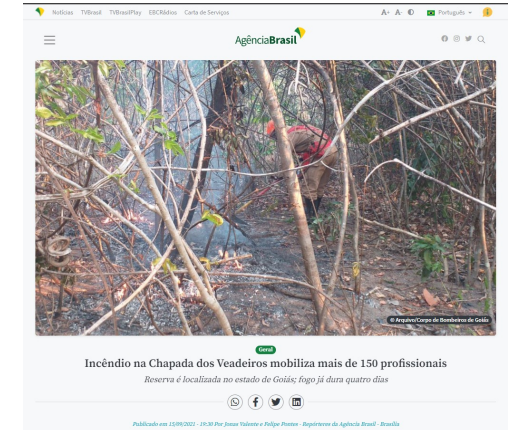
Find stakeholders!



Find each seed problem then decompose, simulate and validate.



Set of problems



Estamos pesquisando a união com métodos de Pesquisa Operacional

Formalização do Problema

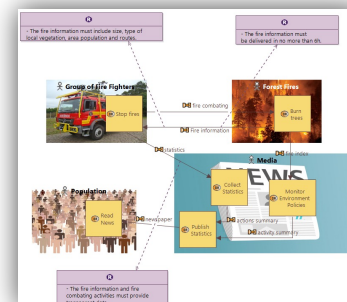
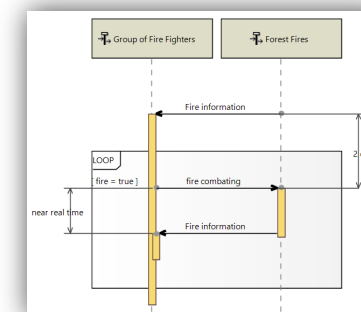
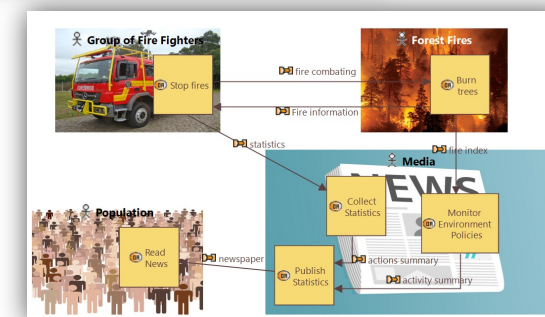
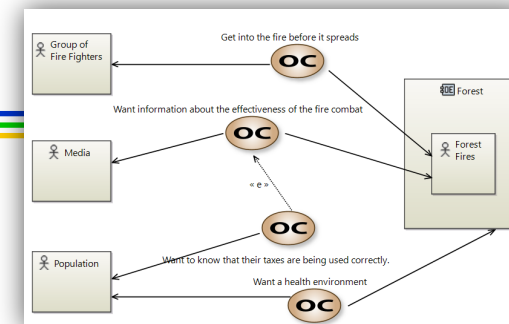
Context Understanding
Problem Exploration



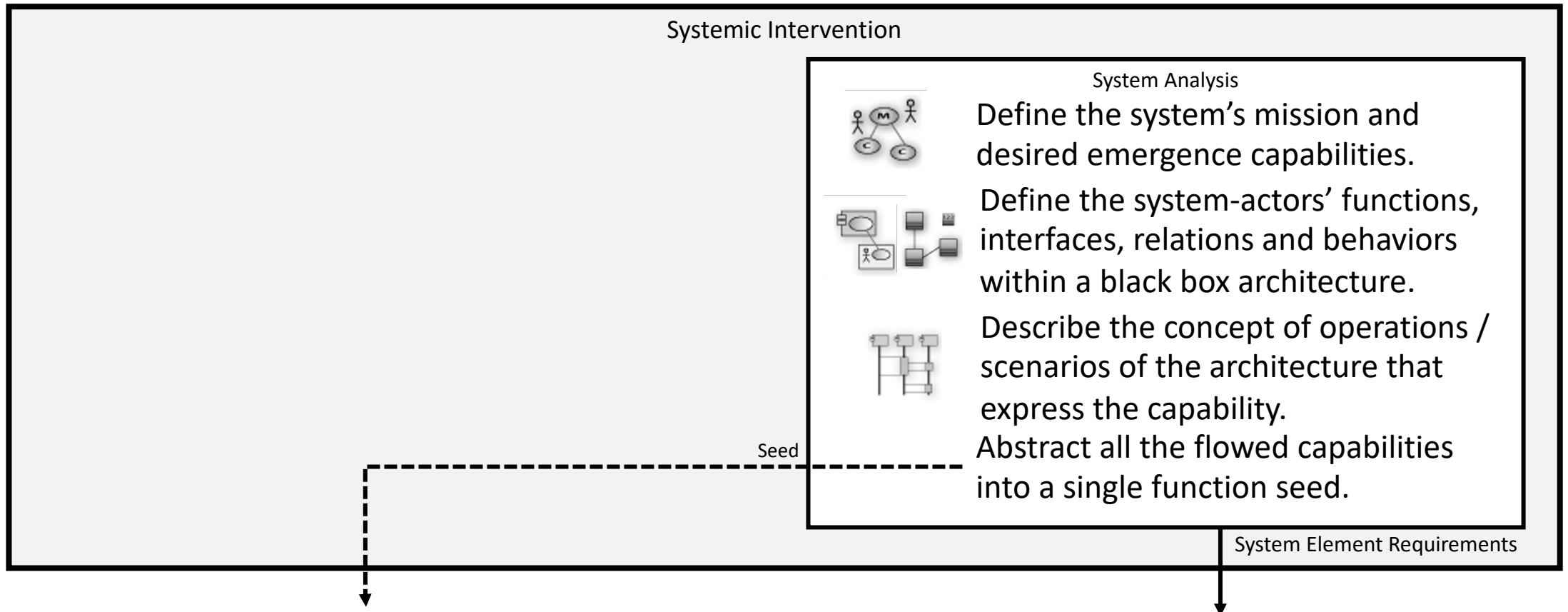
Organize the identified required capabilities.
Harmonize actors' activities, relationships and behaviors within a unified map.
Describe the interactions of the architecture that express the capability.

Formalized Needs

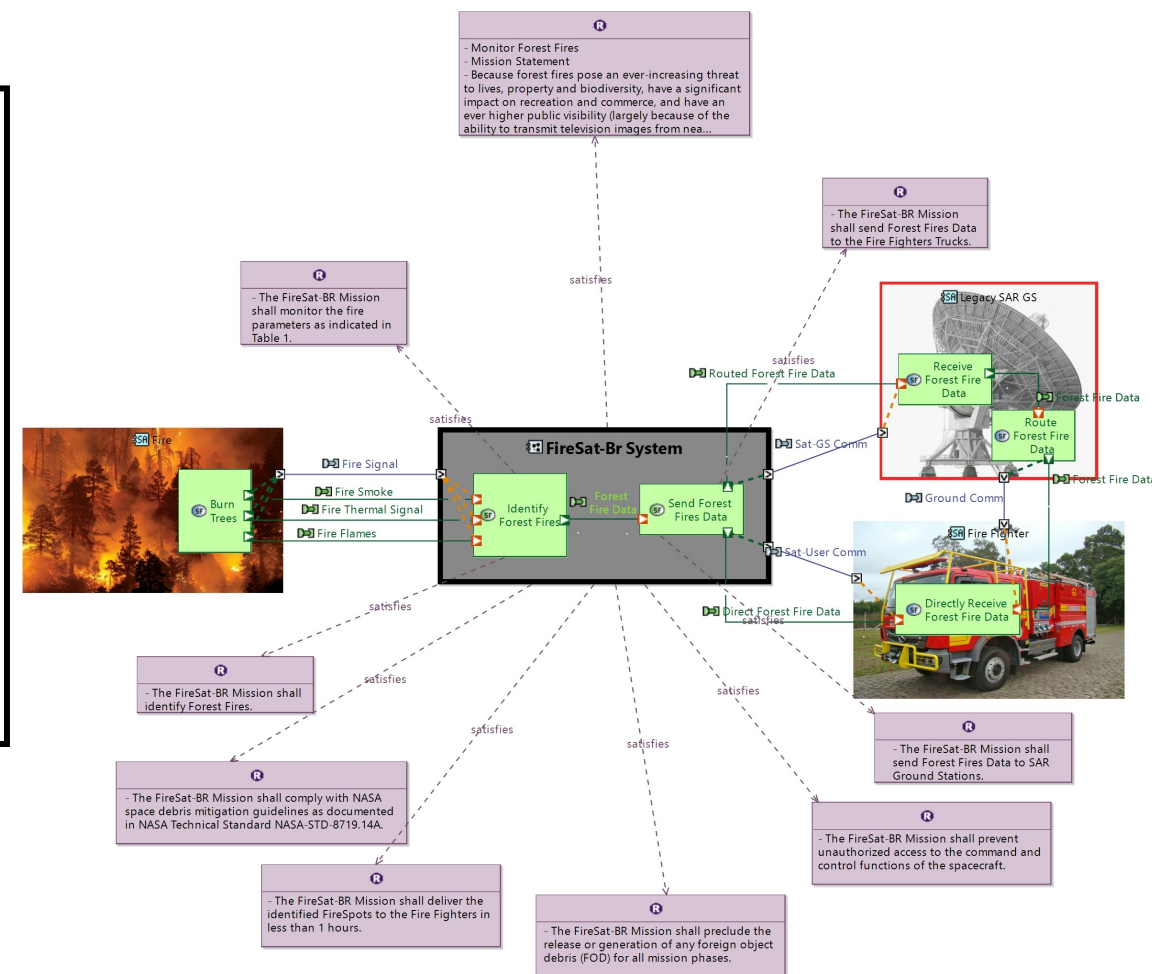
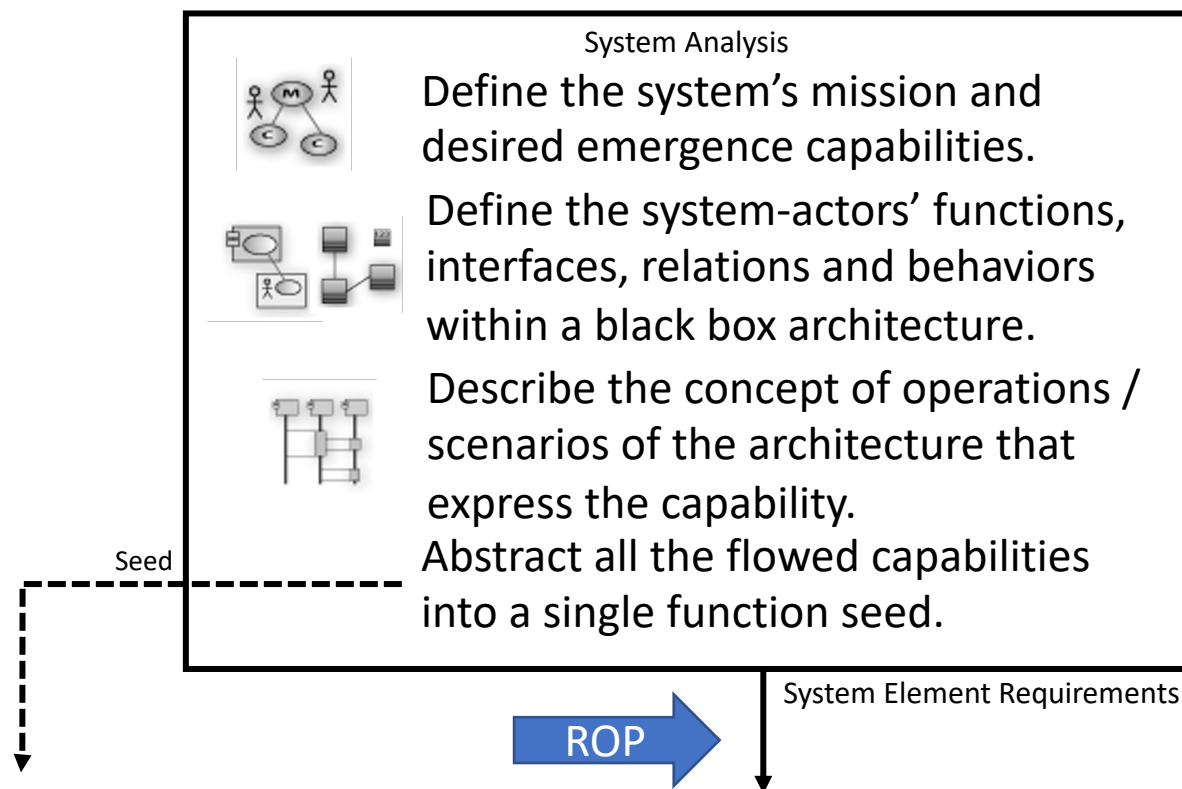
NOP



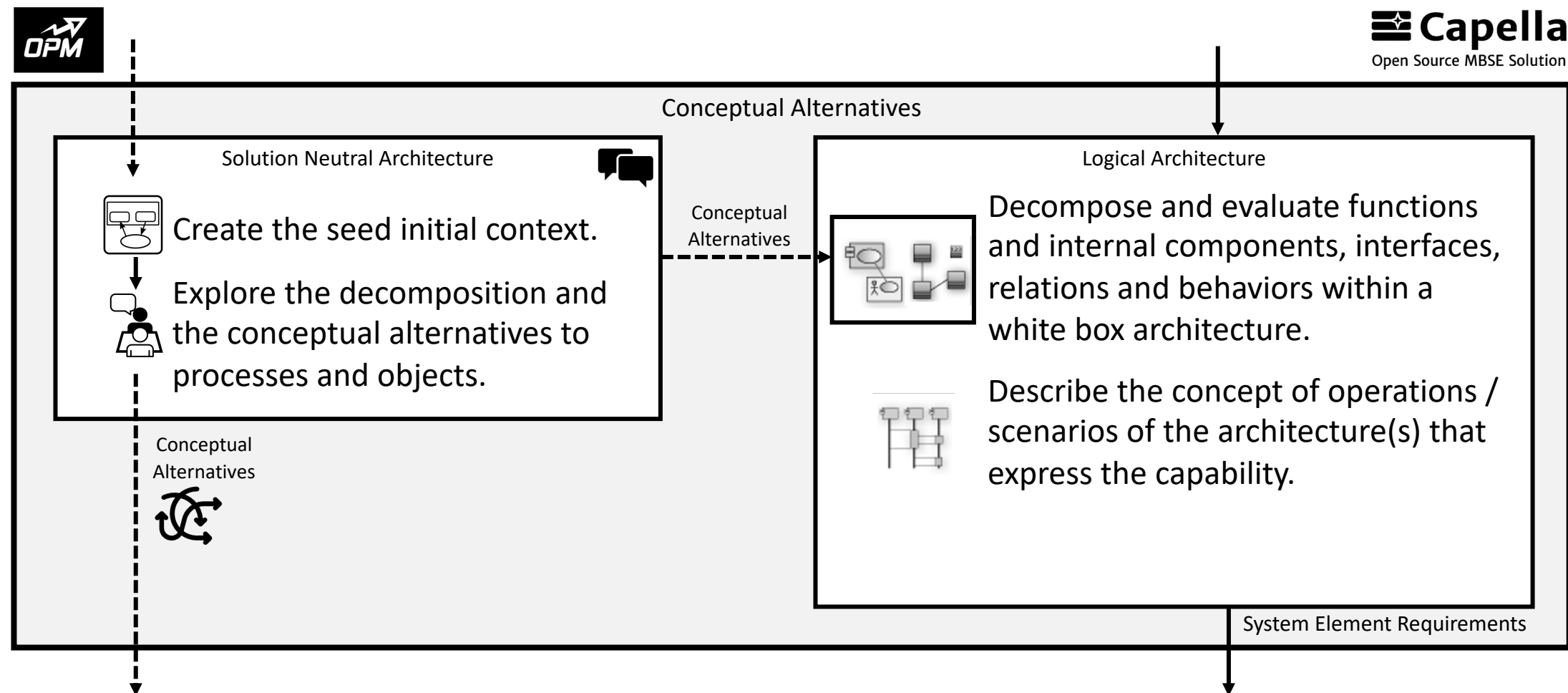
Proposição de um sistema solução



Sistema Solução



Exploração Conceitual



Brainstorm de Soluções



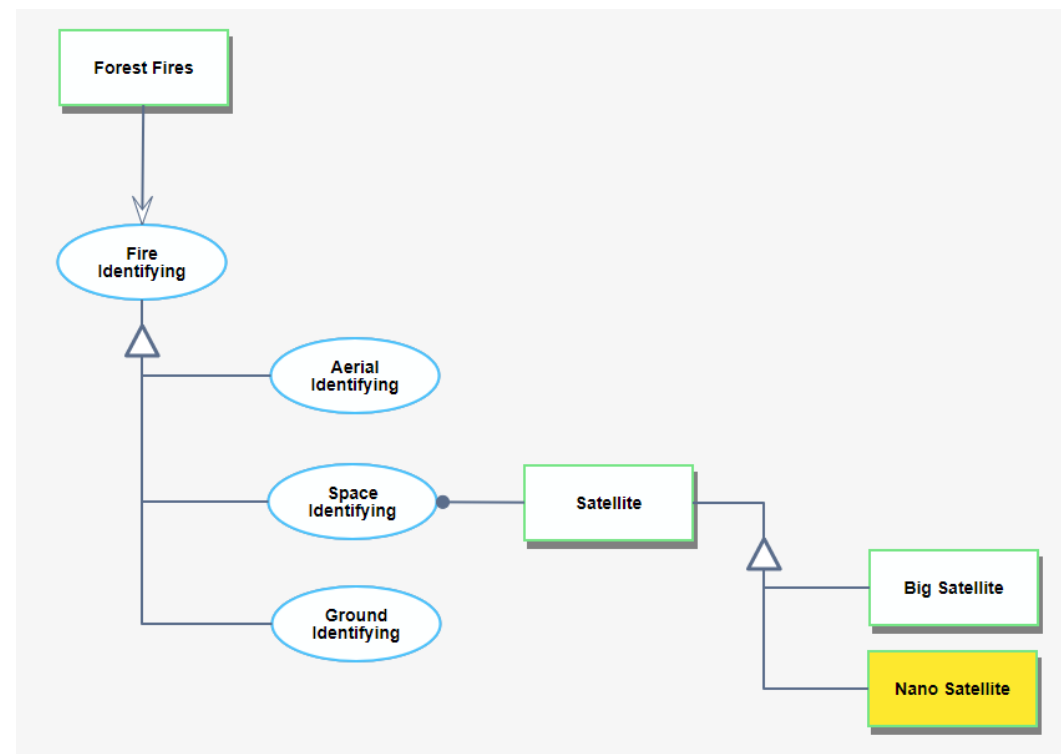
Create the seed initial context.



Explore the decomposition and the conceptual alternatives to processes and objects.



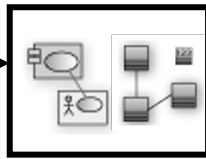
Conceptual
Alternatives





Formalização de conceitos

Conceptual Alternatives

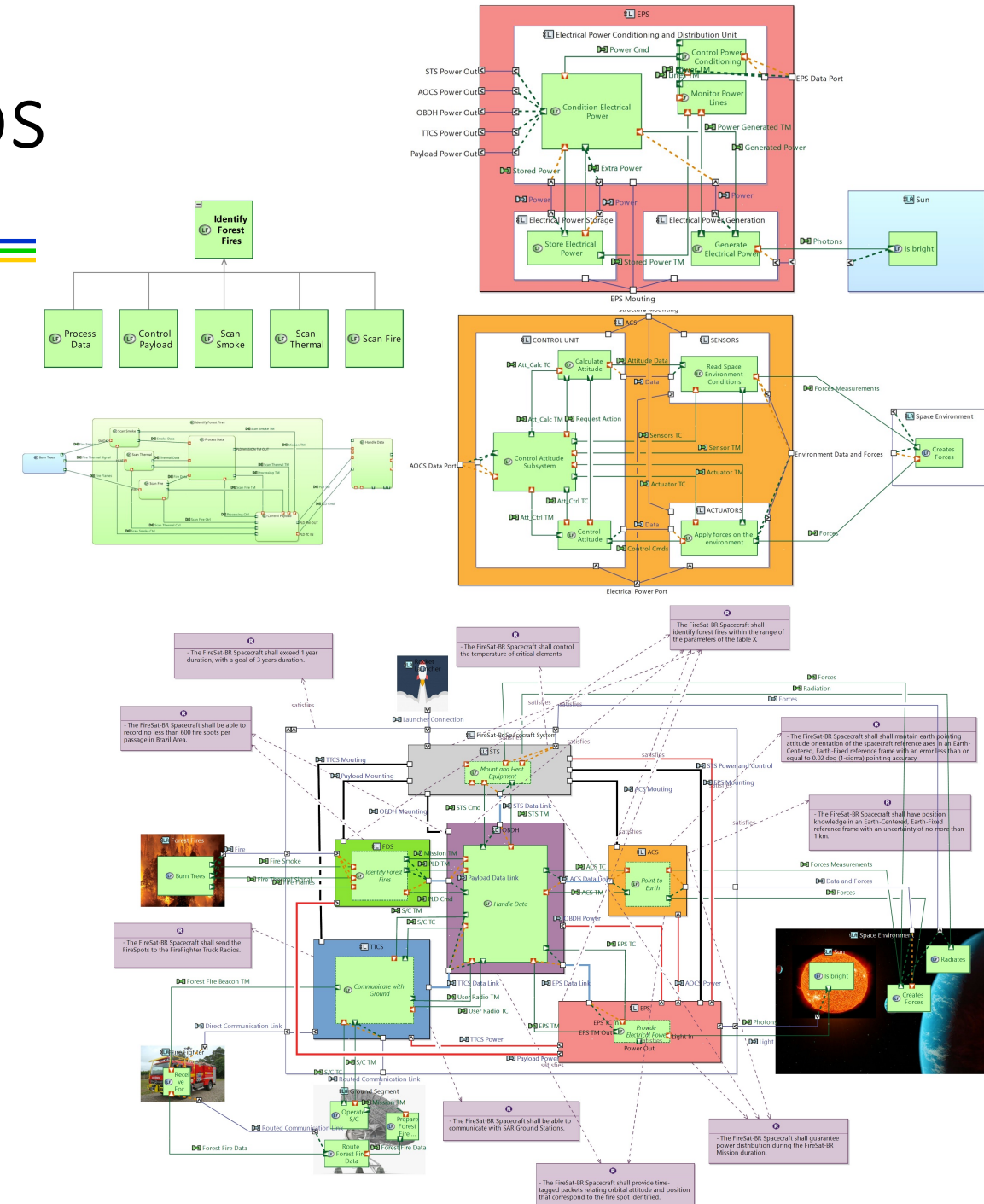


Decompose and evaluate functions and internal components, interfaces, relations and behaviors within a white box architecture.

Describe the concept of operations / scenarios of the architecture(s) that express the capability.



System Element Requirements





Conceito de operação modelado

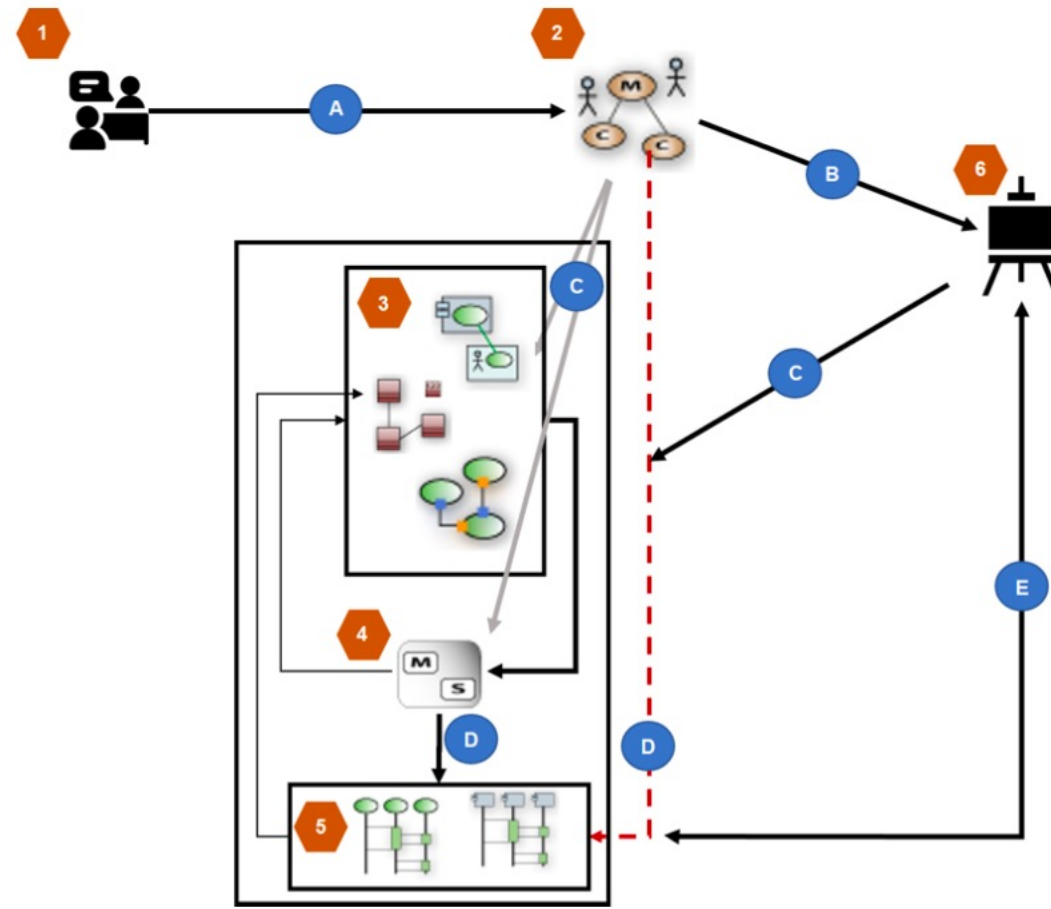


Fig. 1. Methodology Diagram.

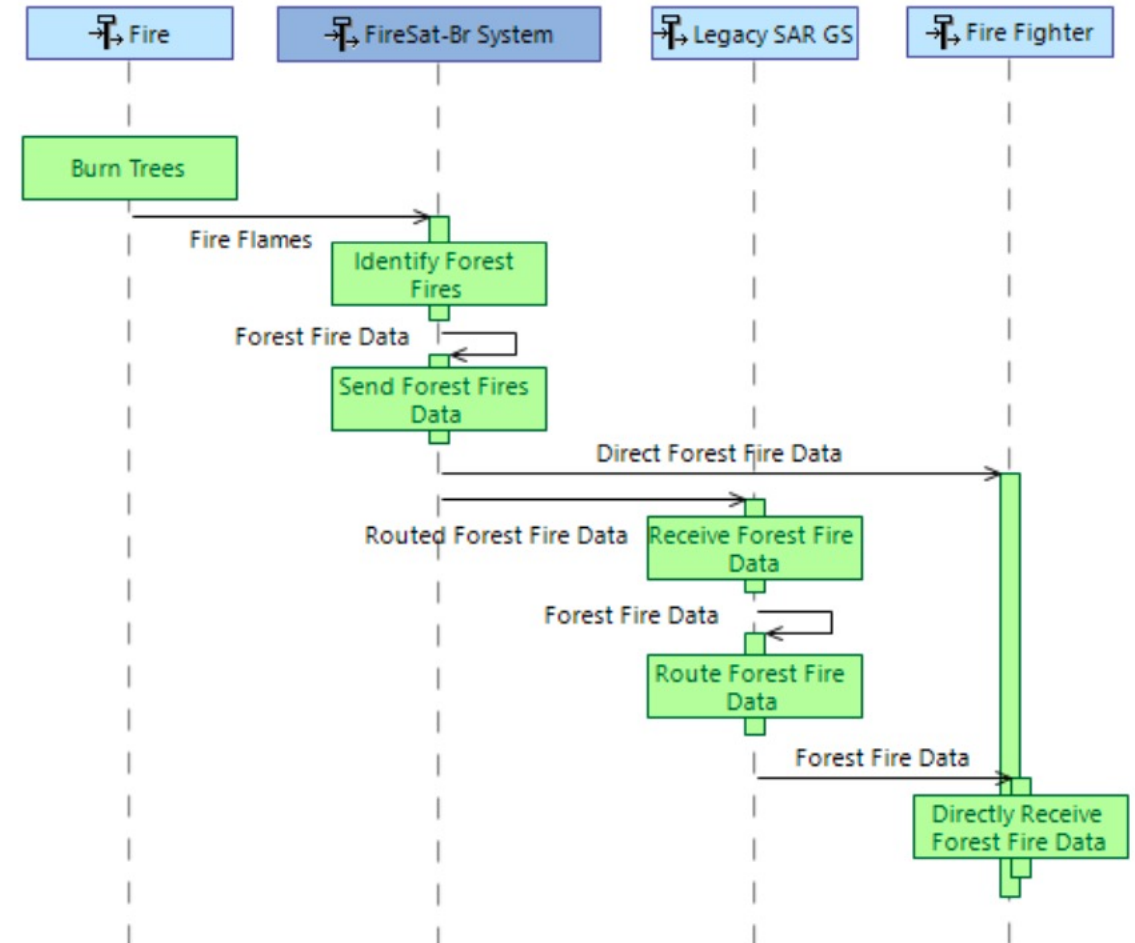
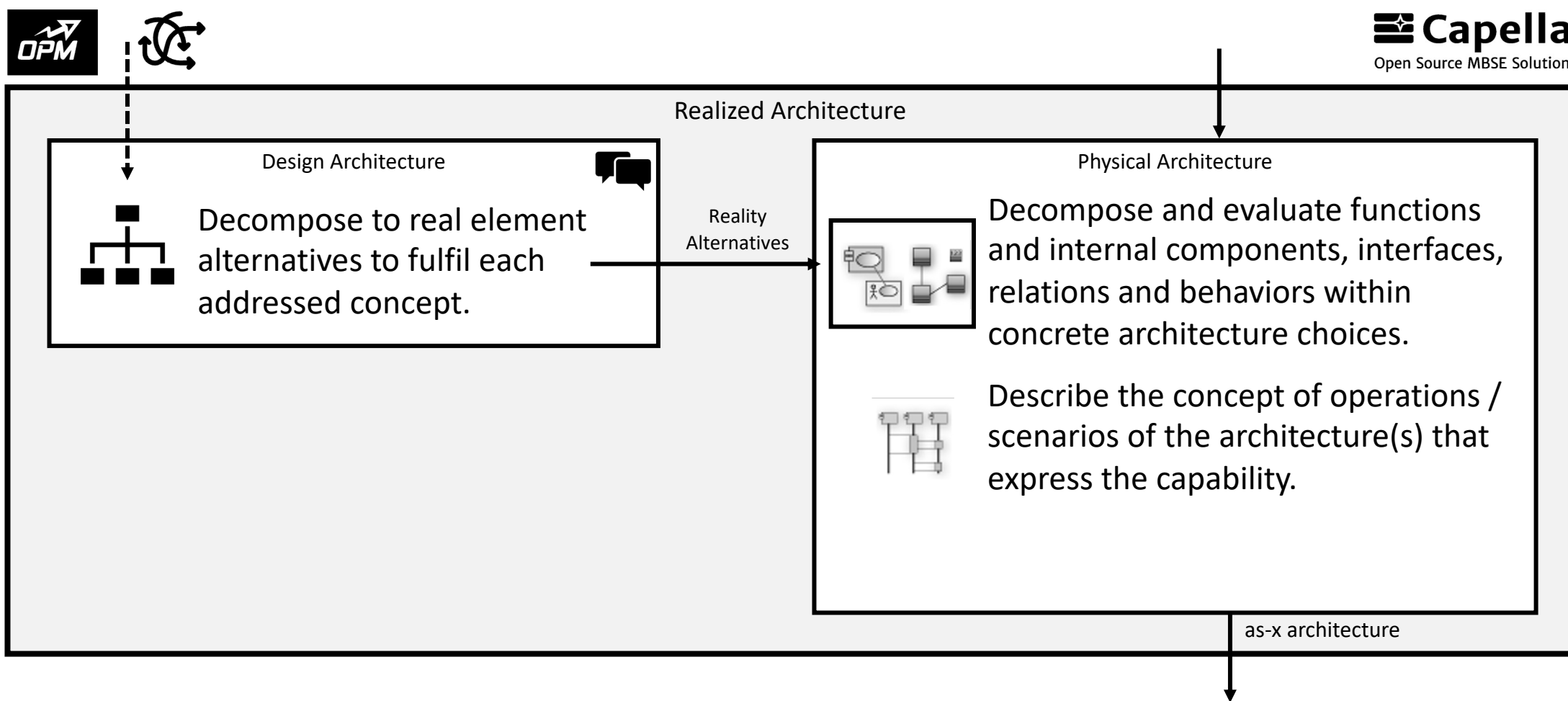


Fig. 6. Model Scenario.



Arquitetura “a ser” concretizada





Reality Alternatives

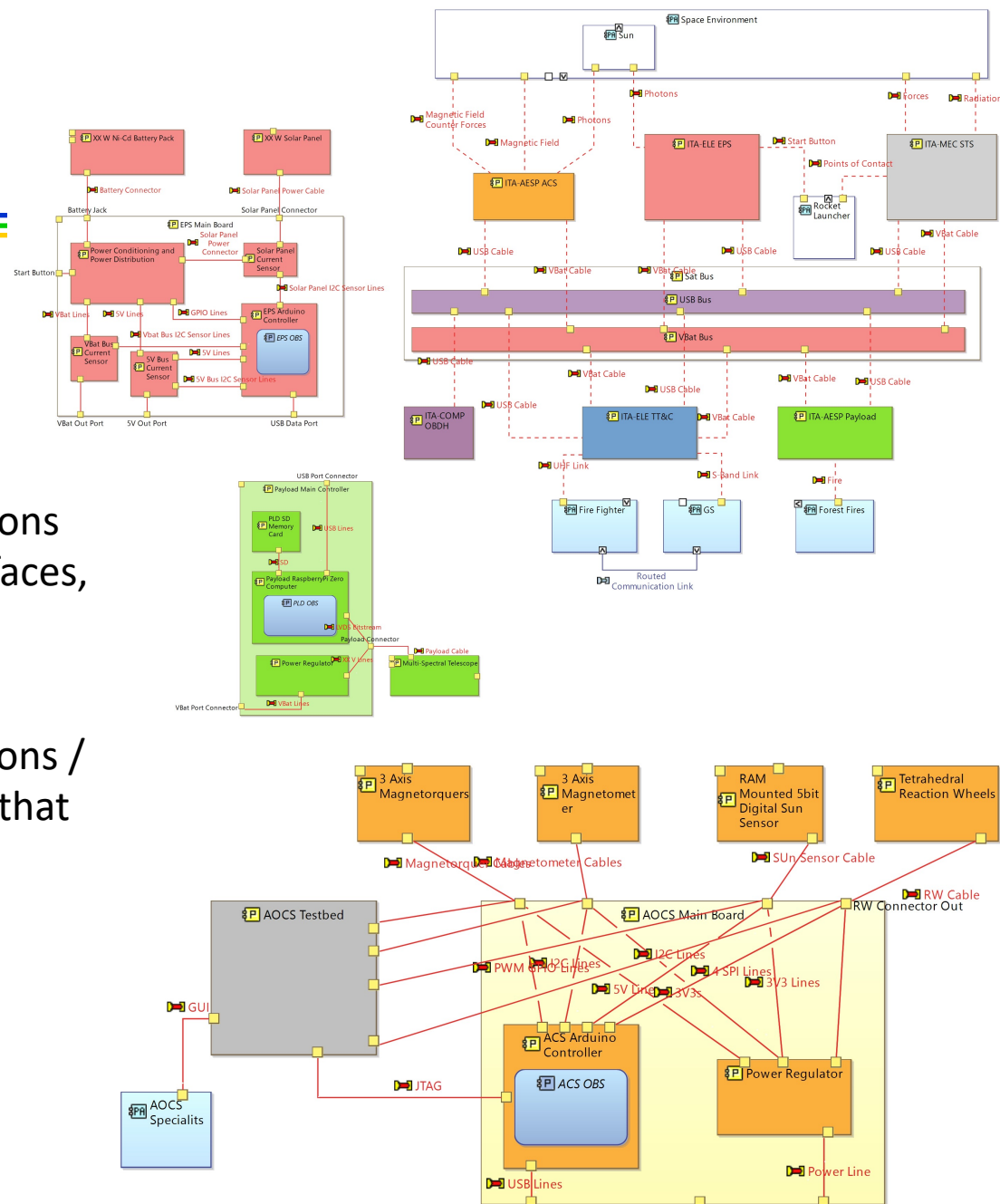


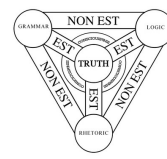
Decompose and evaluate functions and internal components, interfaces, relations and behaviors within concrete architecture choices.



Describe the concept of operations / scenarios of the architecture(s) that express the capability.

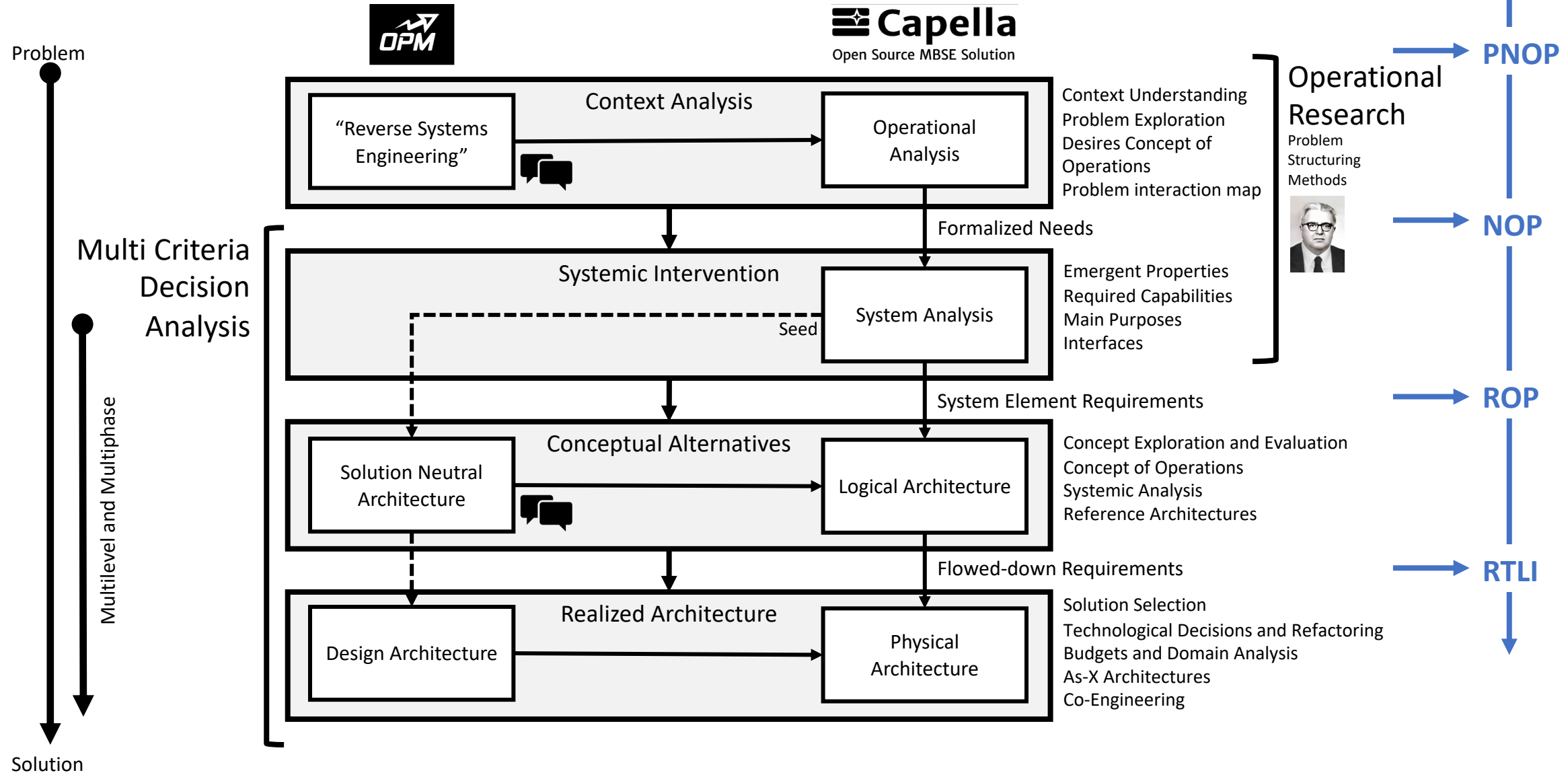
as-x architecture





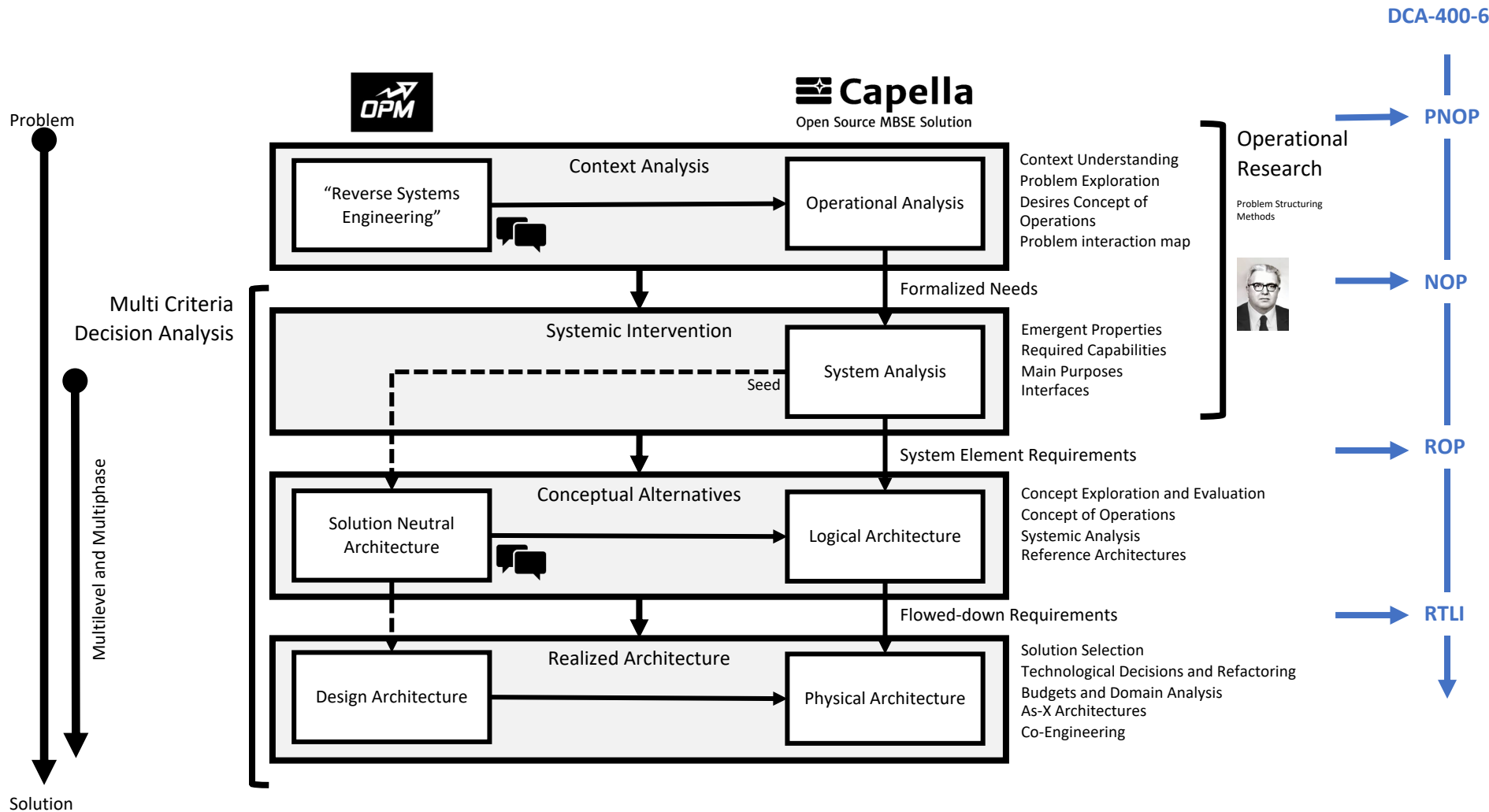
MMMF (Multi Methodology MBSE Framework)

DCA-400-6



MMMF (Multi Methodology MBSE Framework)

C L A S S S

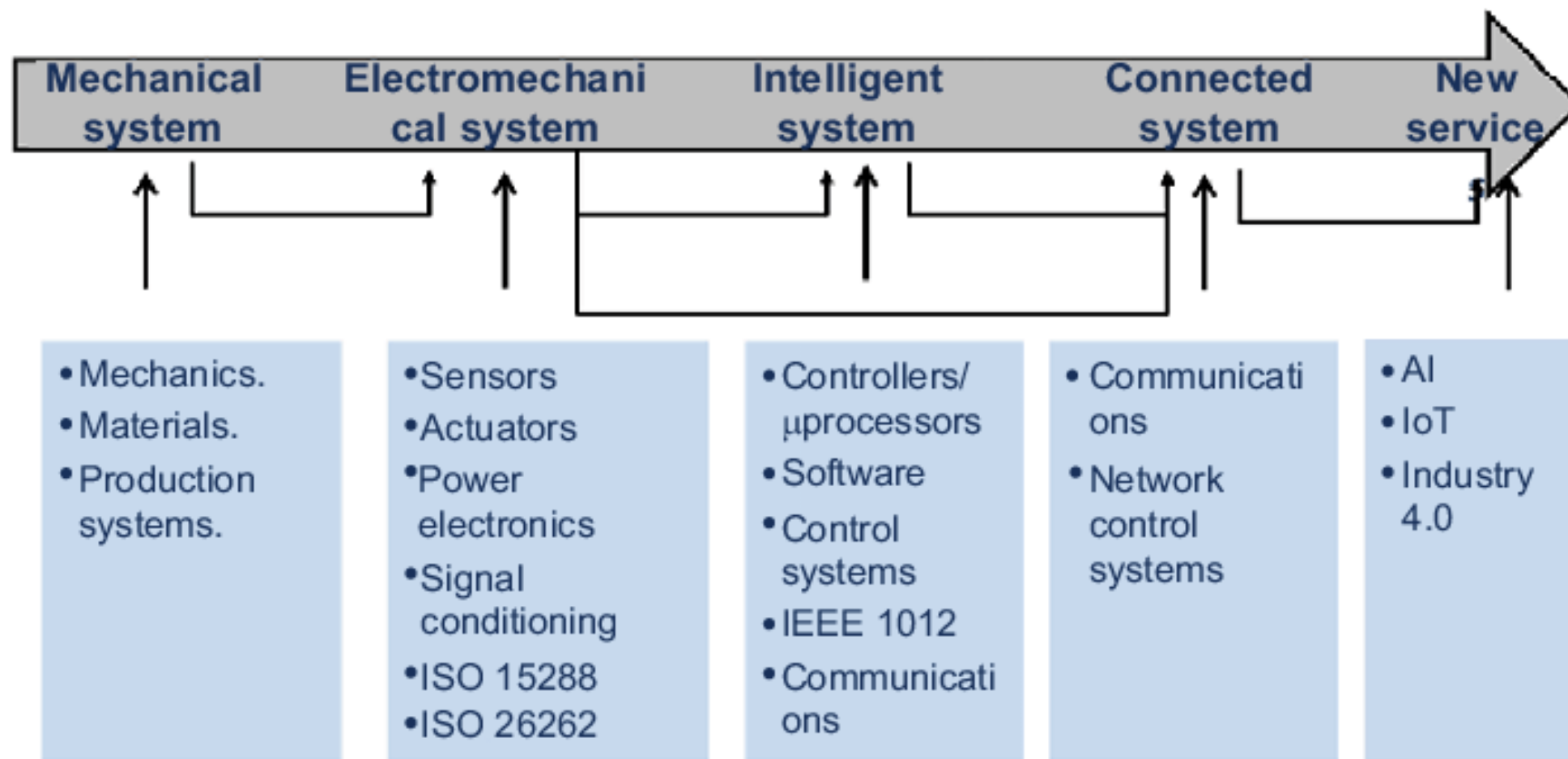




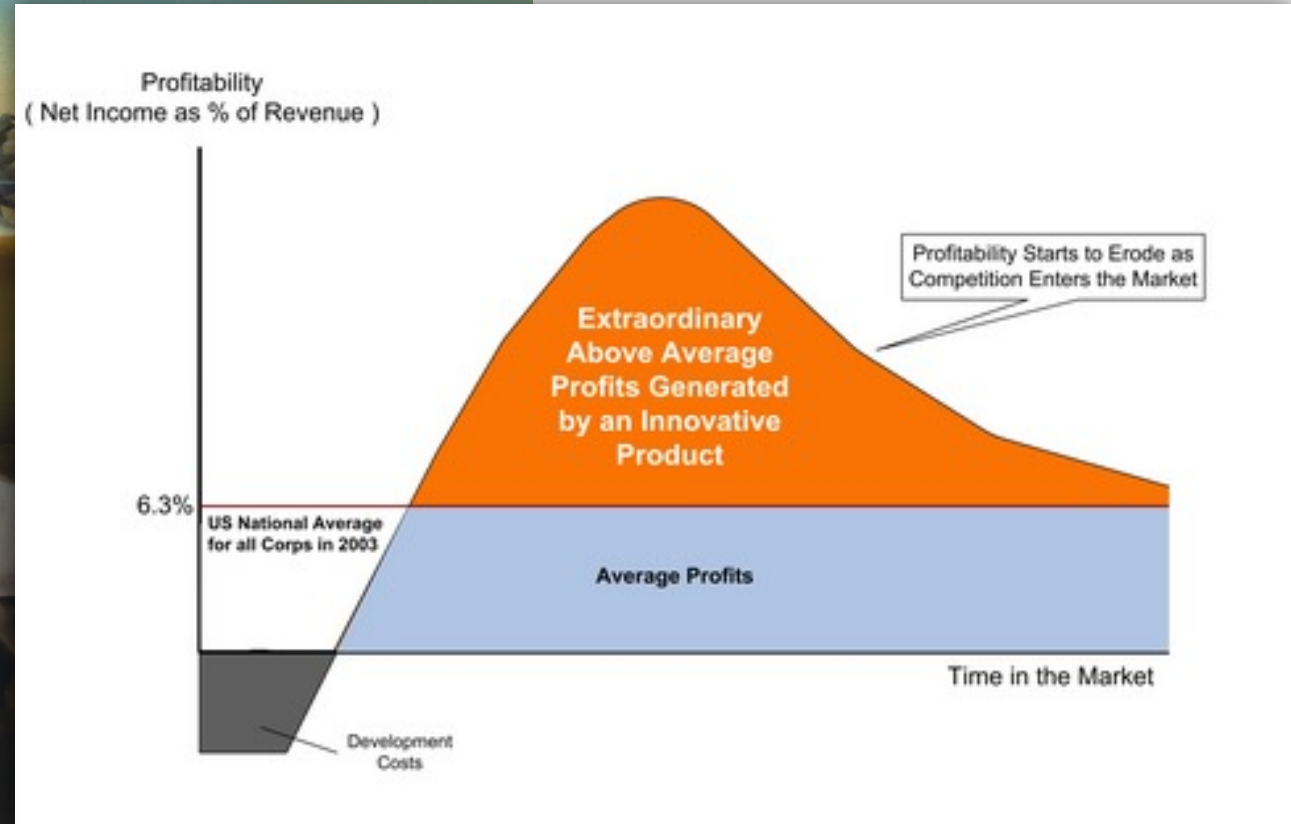
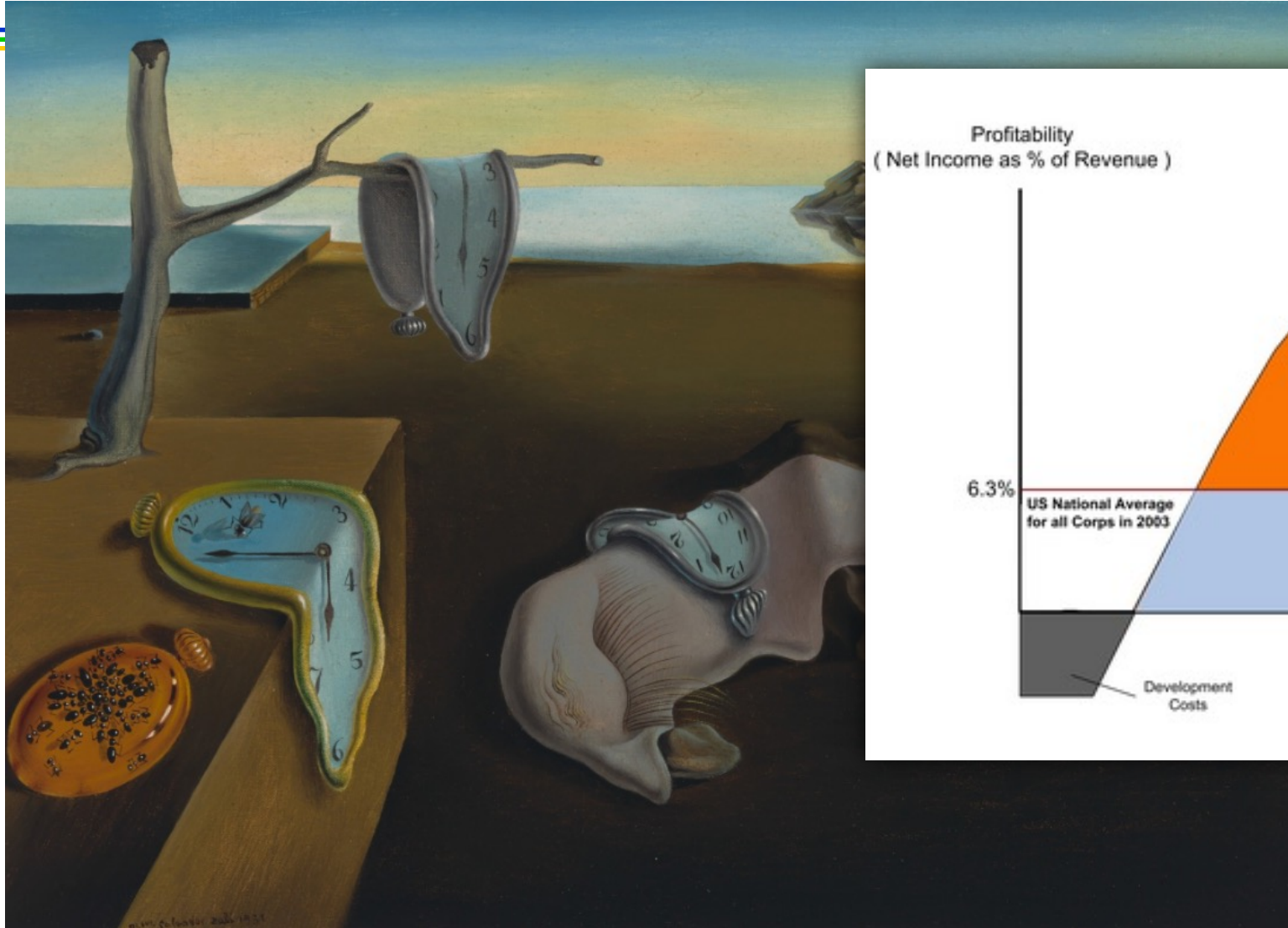
ENGENHARIA SIMULTÂNEA



Crescimento da complexidade dos sistemas

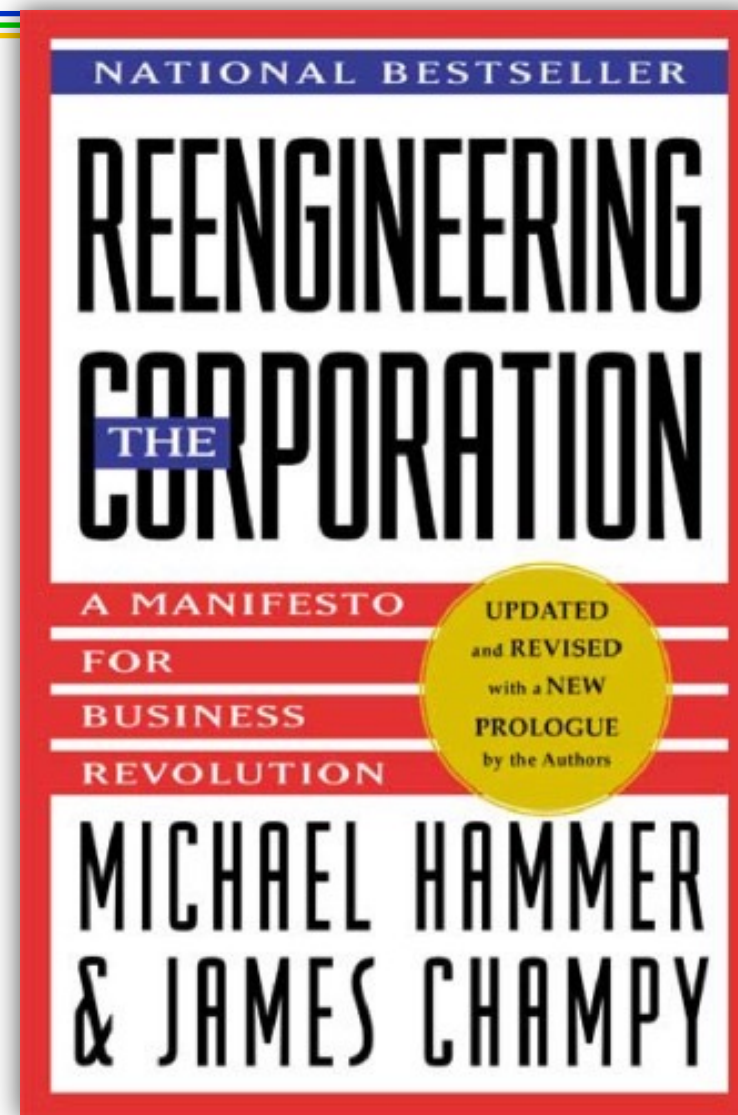
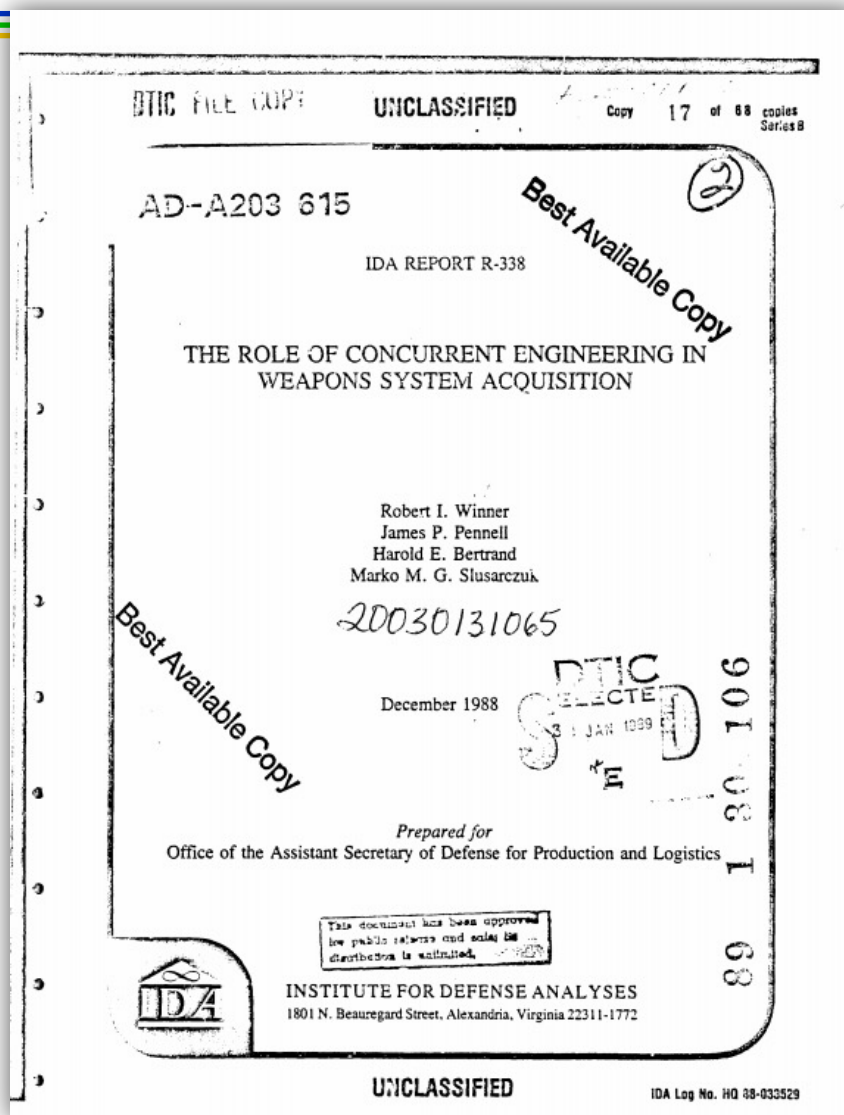


Velocidade do mercado



[Time to Market Knowledge Center and Forum - 12manage](#)

Organizações começaram a propor novas formas de acelerar o time-to-market





Participants at the first IDA concurrent engineering workshop discussed concurrent engineering practices in several U.S. companies. They described the use of methods that included traditional system engineering practices and new engineering and management approaches. DoD and Air Force initiatives to improve the acquisition process were also presented. Based on the discussion at that workshop, on further contributions from participants, and consultation with various reviewers, the following definition was developed:

Concurrent engineering is a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements.



2001 – Manifesto Ágil

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

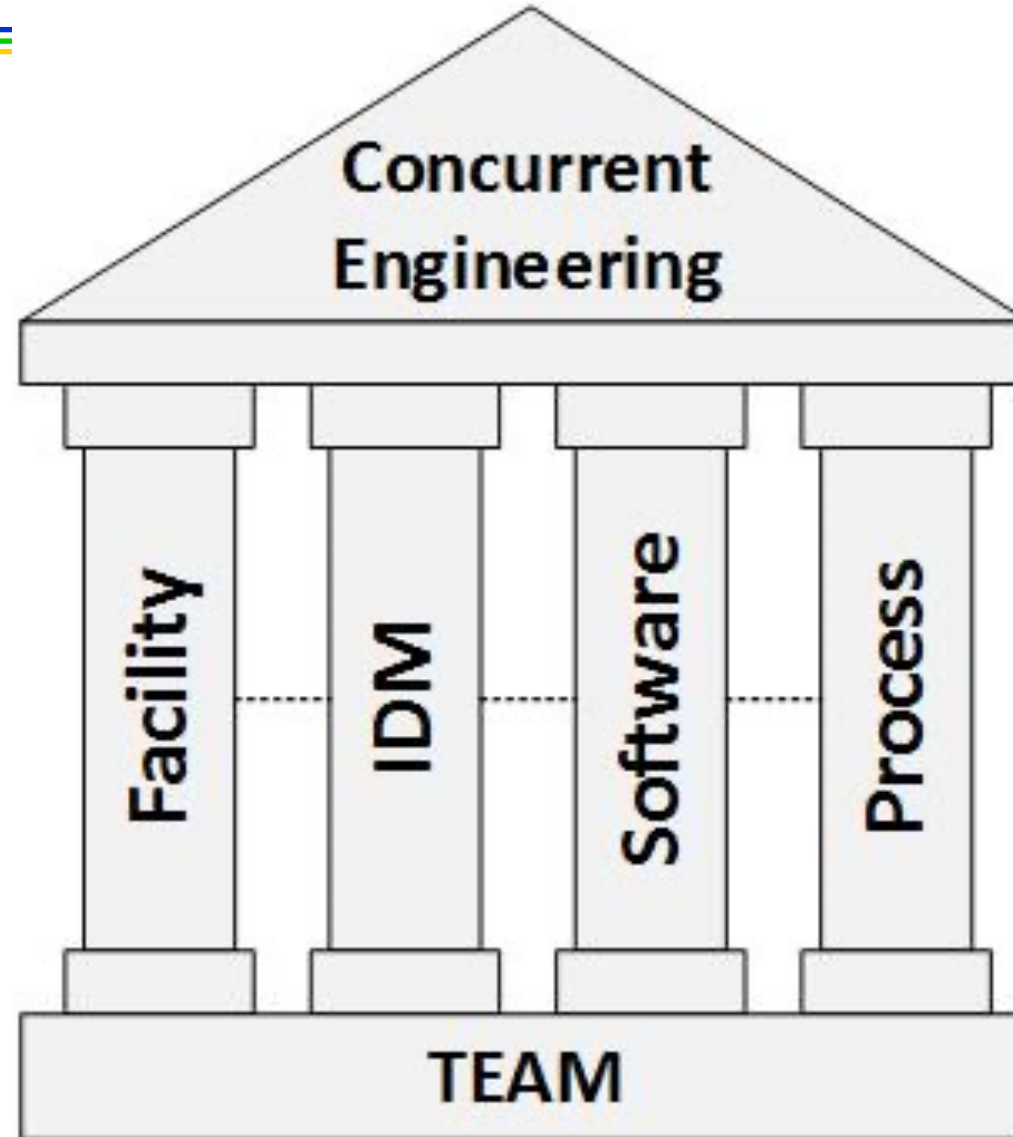
Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

© 2001, the above authors
this declaration may be freely copied in any form,
but only in its entirety through this notice.

- We follow these principles:
 - Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
 - Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
 - Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
 - Businesspeople and developers must work together daily throughout the project.
 - Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done.
 - The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
 - Working software is the primary measure of progress.
 - Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
 - Continuous attention to technical excellence and good design enhances agility.
 - Simplicity--the art of maximizing the amount of work not done--is essential.
 - The best architectures, requirements, and designs emerge from self-organizing teams.
 - At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



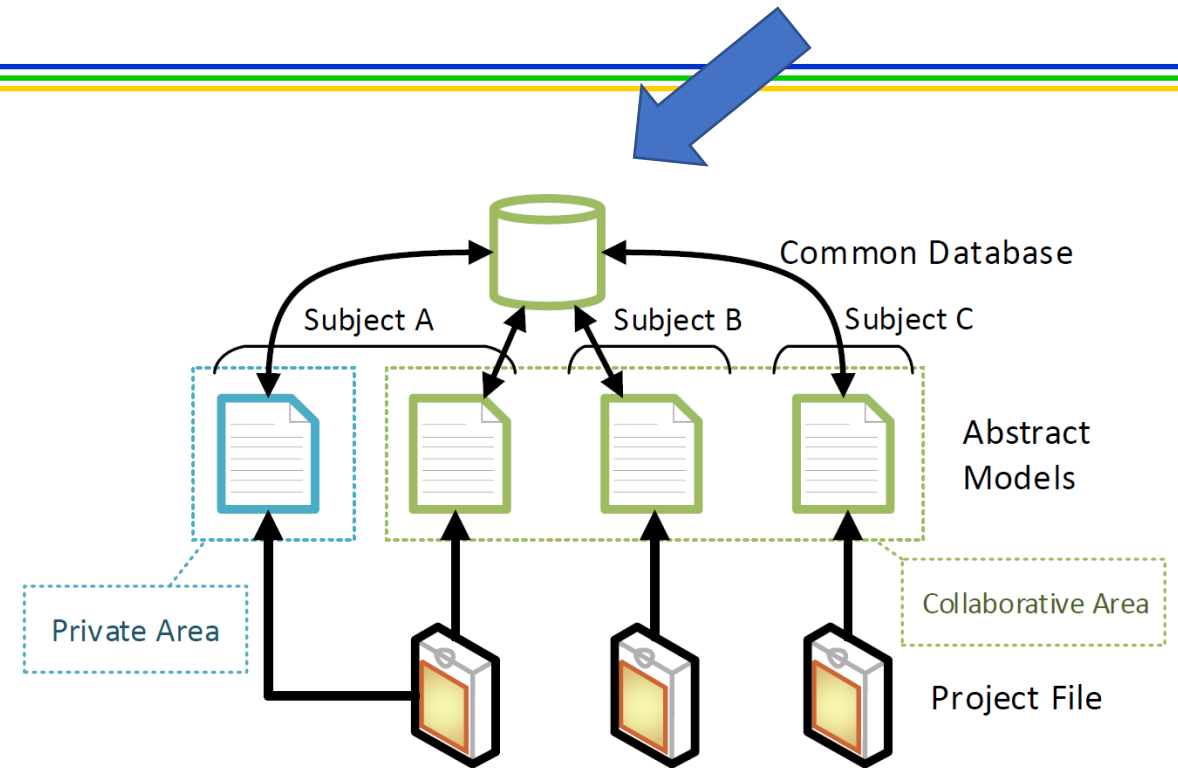
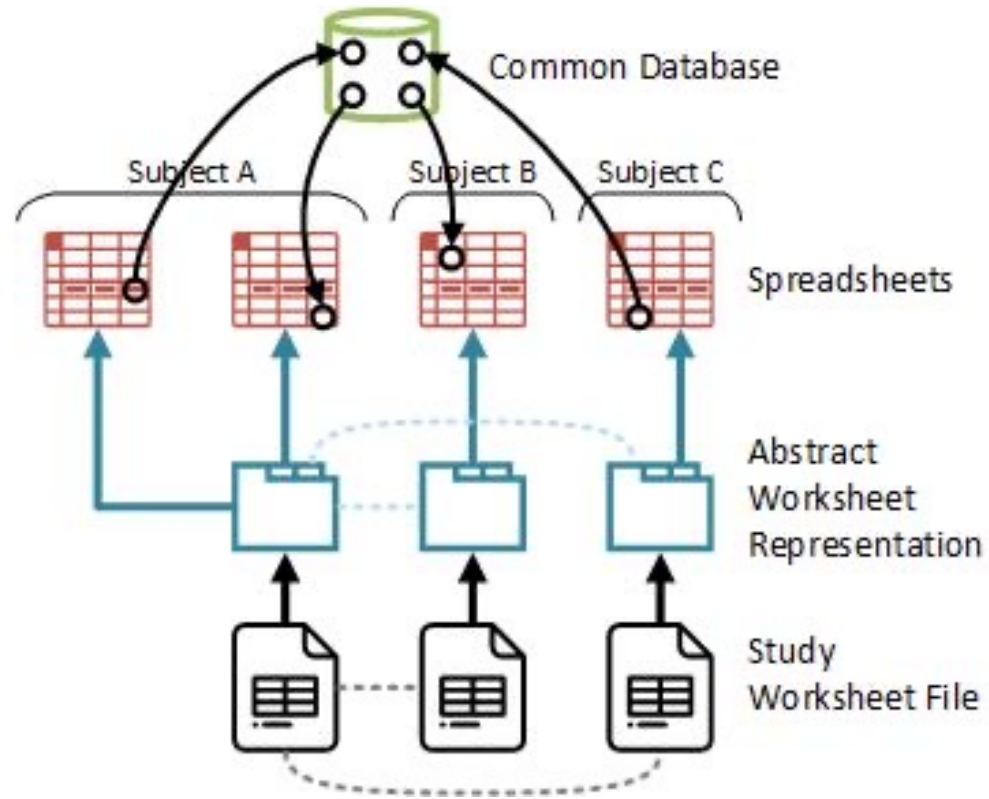
Engenharia Simultânea







De documentos para modelos





CONSIDERAÇÕES FINAIS



DEMANDA DO PESE

3.5 Processos de Engenharia do PESE

3.5.1 Os ciclos de vida dos SE são permeados por um processo de engenharia de sistemas para desenvolvimento, fabricação, lançamento, fiscalização, acompanhamento, aceitação, operacionalização e descarte desses SE.

30/174

MD20-S-01

3.5.2 Os processos do ciclo de vida dos SE devem seguir a DCA 400-6 [5] e, no que couber, outros processos utilizados pelos demais países na definição e desenvolvimento de missões de sistemas espaciais.

3.5.3 Nesse processo de ciclo de vida de SE, será empregado o conceito de Salas Integradas de Gestão de Projetos (SIGP), de forma a permitir o envolvimento e o acompanhamento das etapas do projeto a custos menores, sem perda de qualidade. As SIGP serão equipadas com recursos de multimídia e informática, além de ferramentas de software para trabalhos em grupo, que permitirão a interação em tempo real de equipes técnicas e gerenciais geograficamente afastadas. Essa infraestrutura minimizará os custos e o tempo no desenvolvimento dos requisitos, dos sistemas, na implantação propriamente dita e nos testes de aceitação de sistemas, também permitindo um acompanhamento adequado dos processos. As SIGP são instaladas nos pontos de interesse a definição das missões dos SE.



CONCEPTIO

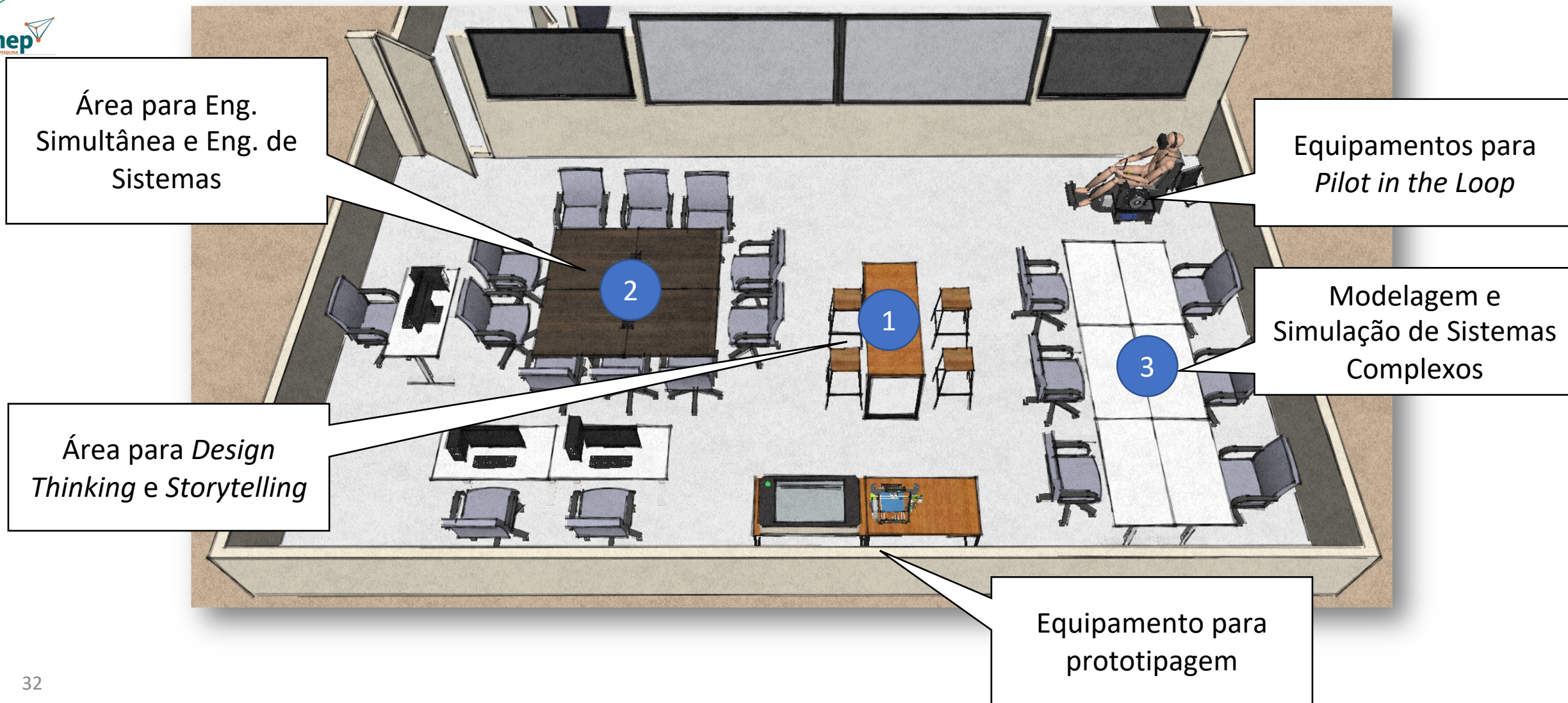
SALA DE CONCEPÇÃO

ENGENHARIA DE SISTEMAS

ENGENHARIA DE MISSÕES

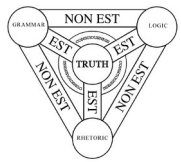
ENGENHARIA SIMULTÂNEA

INFRAESTRUTURA DE DEV - CEI

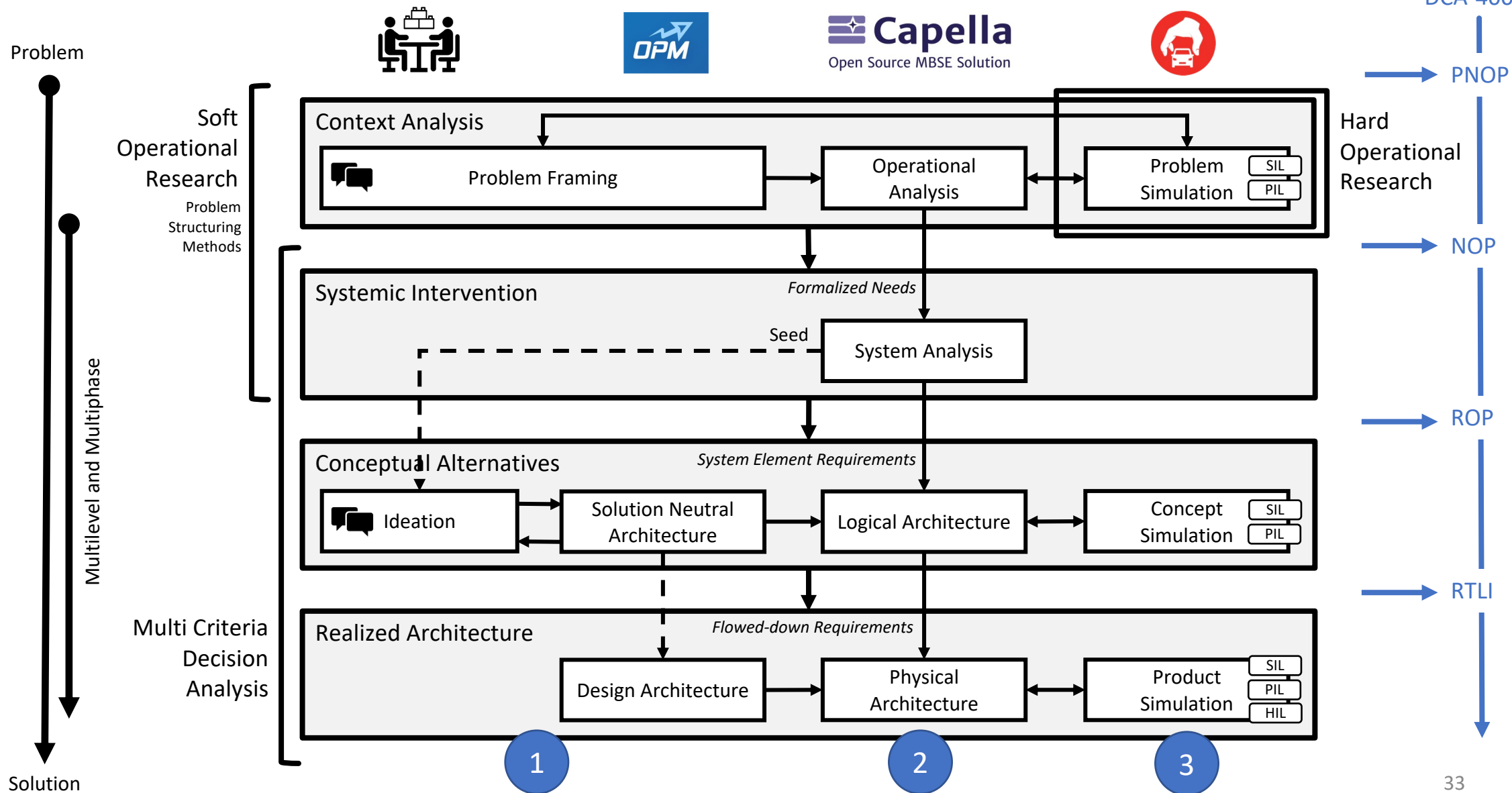




FRAMEWORK DO LABORATÓRIO



DCA-400-6



Algumas considerações...

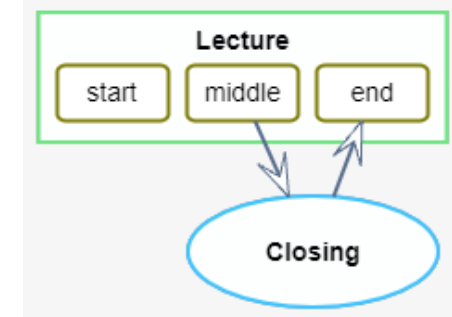
- ES é sobre a entrega do melhor valor para o stk através de um sistema.
- ESBM é apenas ES usando ferramentas
 - Faça nosso curso: TE-265 – Engenharia de Sistemas baseada em Modelos 😊
- Em algum momento ESBM vai dominar ES, e se chamar apenas ES.
- ES é a base para interconectar disciplinas, é o LEGO, ou a lingua franca da engenharia.
 - Perfeita para ambientes de desenvolvimento colaborativo.
 - Façam nosso curso: TE-219 – Engenharia Simultânea 😊



Venha nos visitar

www.cei.ita.br





Perguntas & (talvez) Respostas



Prof. Dr. **Christopher**
Shneider Cerqueira
chris@ita.br



<http://www.cei.ita.br/>

