

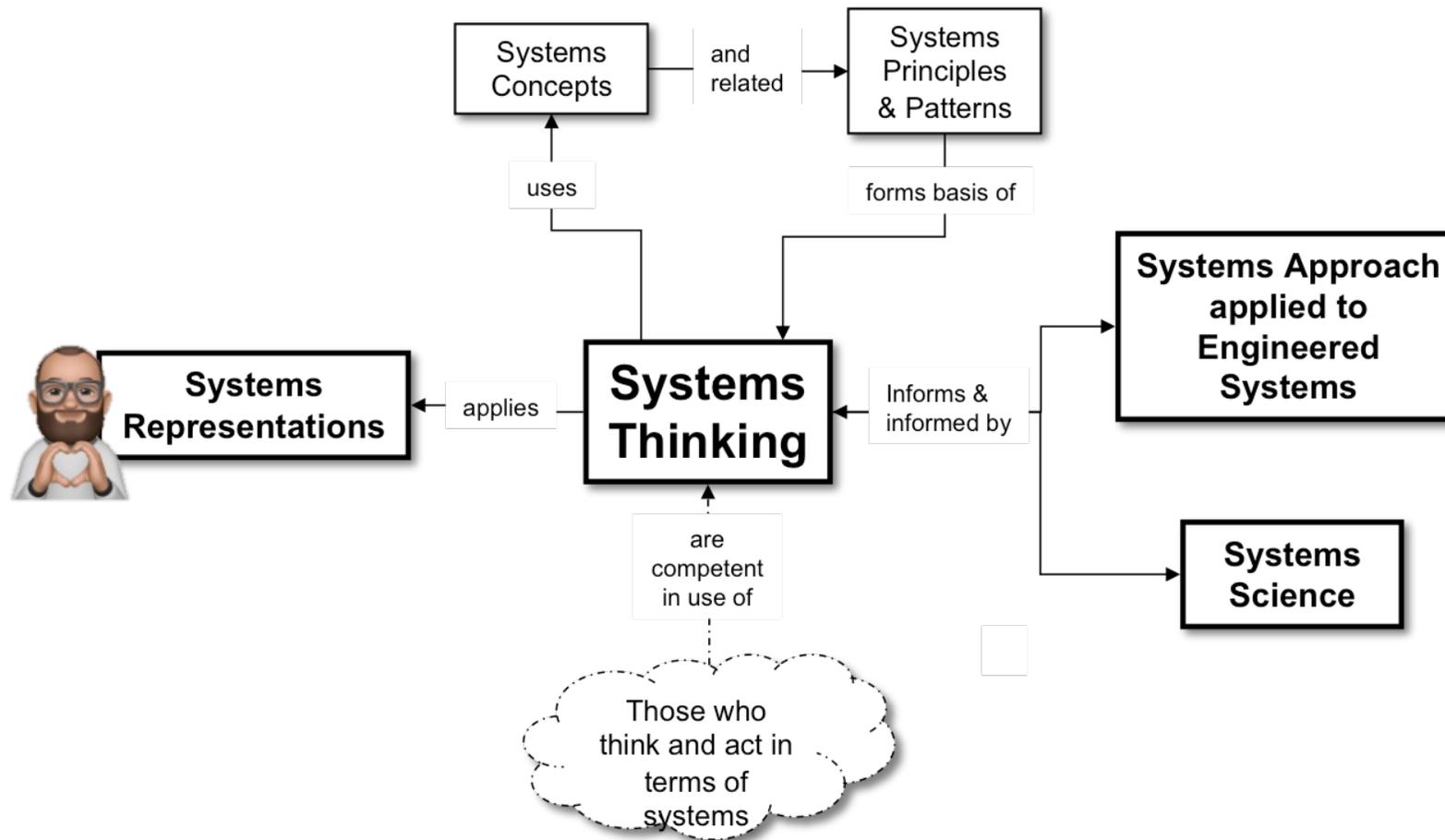


IEA-S – DEPARTAMENTO DE SISTEMAS ESPACIAIS  
(SPACE SYSTEMS DEPARTMENT)

# FRAMEWORKS, ARQUITETURA E DIAGRAMAS CLÁSSICOS



# SYSTEMS THINKING



[Concepts of Systems Thinking - SEBoK \(sebokwiki.org\)](http://sebokwiki.org)



[Principles of Systems Thinking - SEBoK \(sebokwiki.org\)](http://sebokwiki.org)





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

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



## AULA 02 Frameworks e Arquitetura (Rev\_01)

**OBJETIVOS** 02-01 - Clarificar que Engenharia de Sistemas é uma abordagem com diferentes formas  
 02-02 - Apresentar as representações clássicas da Engenharia de Sistemas: árvores, diagrama N2, diagrama de blocos  
 02-03 - Apresentar o Conceito de Arquitetura  
 03-04 - Apresentar a análise estruturada (análise de arquitetura funcional)

**DATA:** 13-Mar

	TÍTULO	#	TÓPICO	ATIVIDADE INDIVIDUAL	ATIVIDADE EM GRUPO
<b>HORA 01</b>	Diferentes abordagens de Engenharia de Sistemas				
		1	Linguagem da Engenharia de Sistemas		
		2	Metodologias de MBSE		
		3	INCOSE	Revisão Metodologias	
		4			
		5			
<b>HORA 02</b>	Arquitetura				
		1	Definição de Arquitetura		
		2	Exemplos de Forma e Funções		
		3	Definição de função	Exercícios de fixação	
		4			
		5			
<b>HORA 03</b>	Diagramas clássicos				
		1	Diagrama de Blocos		
		2	eFFBD		
		3	DFD		
		4	N2		
		5	IDEFO	Exercícios de fixação	
			<b>* Próximas atividades</b>		



# DIFERENTES ABORDAGENS PARA ENGENHARIA DE SISTEMAS



# LINGUAGEM DA ENGENHARIA DE SISTEMAS



# REDUZINDO A FUNÇÃO: NÓS SOMOS RECONHECEDORES DE PADRÕES

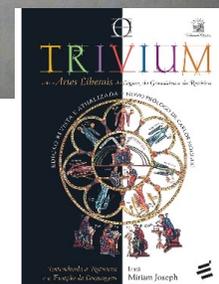
<https://www.psychologytoday.com/blog/the-athletes-way/201311/what-is-the-human-connectome-project-why-should-you-care>



## FUNÇÃO DA GRAMÁTICA

A função fundamental da gramática é estabelecer leis para relacionar símbolos de modo a expressar pensamento. Uma frase expressa um pensamento – uma relação de ideias – numa declaração, numa pergunta, numa ordem, num desejo, numa prece ou numa exclamação. Símbolos categoremáticos são aqueles que são relacionados; símbolos sincategoremáticos são os meios de relacioná-los; a oração é a relação mesma.

As regras para relacionar símbolos regem três operações gramaticais: substituir símbolos equivalentes, combinar símbolos e separar símbolos.



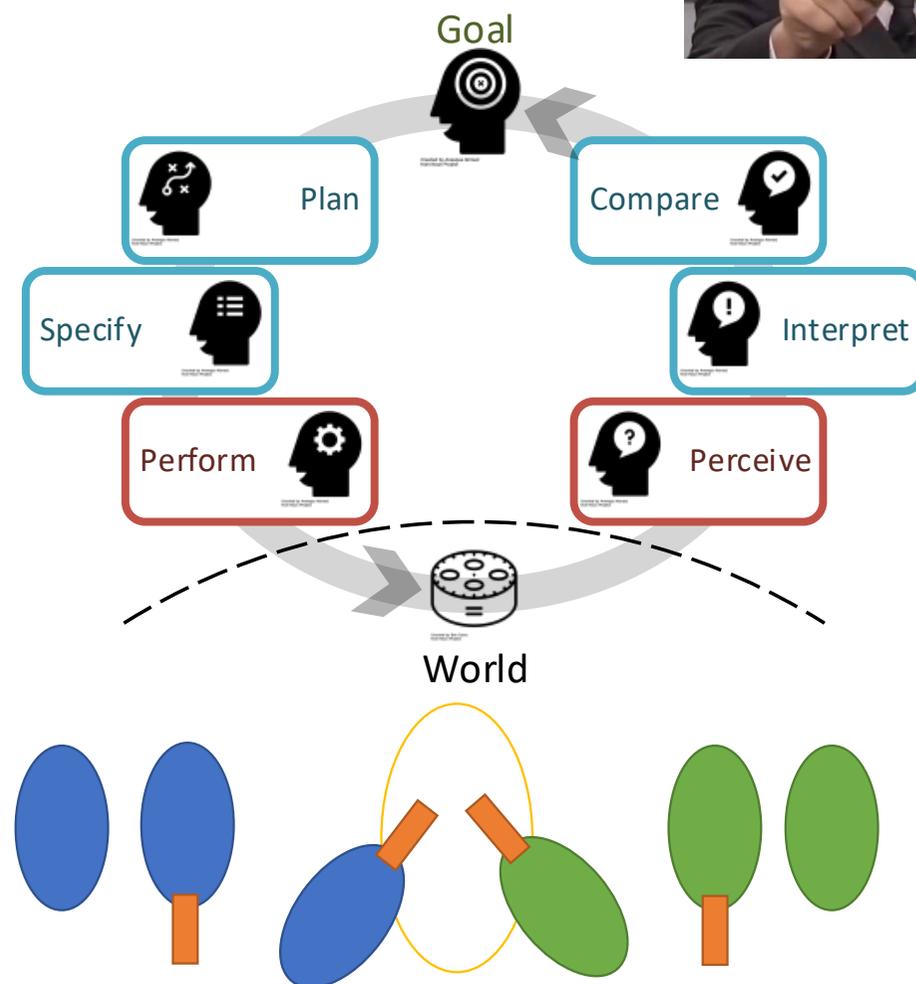


# COGNIÇÃO

Cognição é “o processo mental de adquirir conhecimento e entendimento através do pensamento, inteligência e sentidos”.

Engloba processos como a atenção, a formação do conhecimento, a memória e a memória de trabalho, o julgamento e a avaliação, o raciocínio e a "computação", a resolução de problemas e a tomada de decisões, a compreensão e a produção da linguagem.

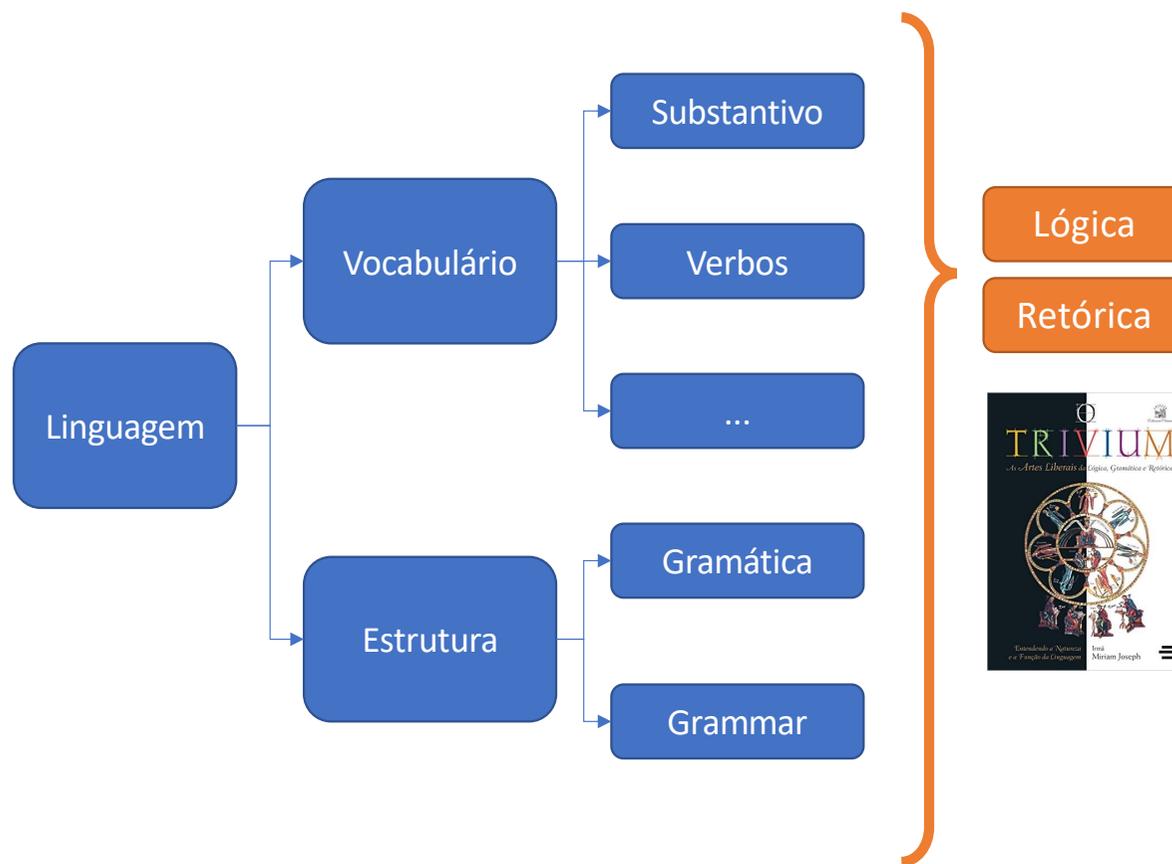
Processos cognitivos usam conhecimento pregresso para gerar novo conhecimento.

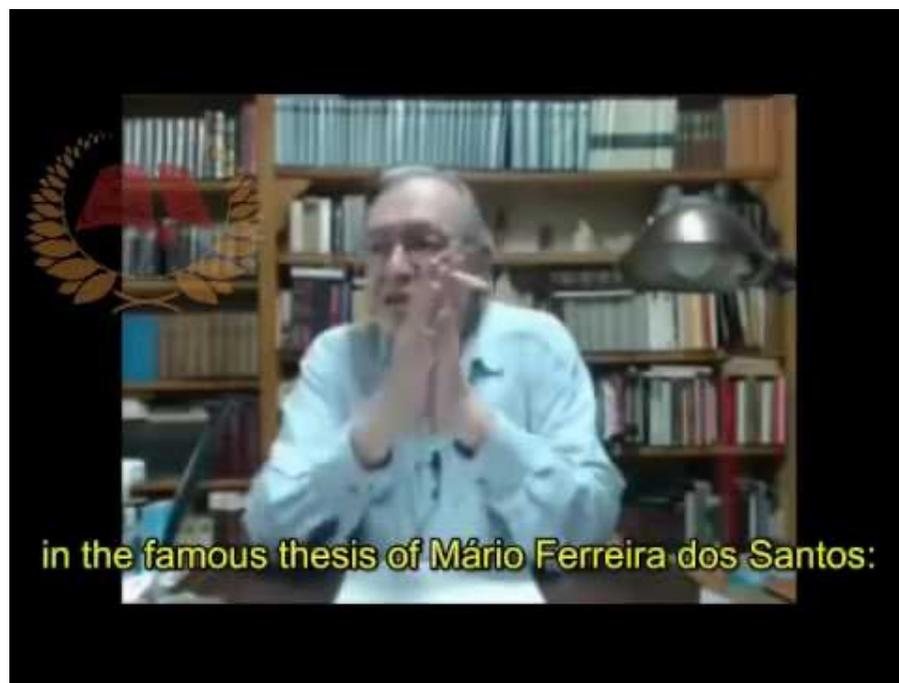




# LINGUAGEM

é um sistema que consiste no desenvolvimento, aquisição, manutenção e utilização de sistemas complexos para comunicação.





<http://www.olavodecarvalho.org/apostilas/presenca.htm>

6:15



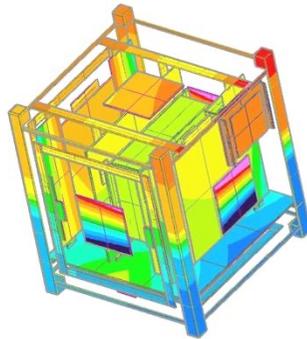
[https://www.ted.com/talks/david\\_mccandless\\_the\\_beauty\\_of\\_data\\_visualization#](https://www.ted.com/talks/david_mccandless_the_beauty_of_data_visualization#)

Linguagem/representações são ferramentas mentais que mapeiam o conhecimento em modelos com a linguagem conhecida. “Usuários” manuseiam “signos” que já são imediatamente reconhecidos (já existindo no seu modelo mental).

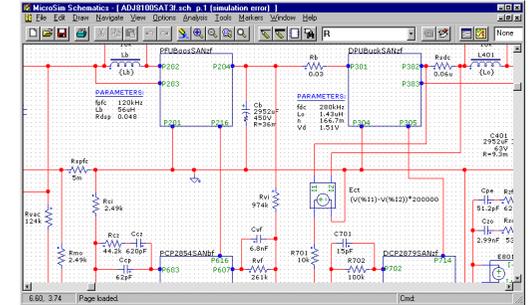


# CADA ENGENHARIA POSSUI SUA LINGUAGEM

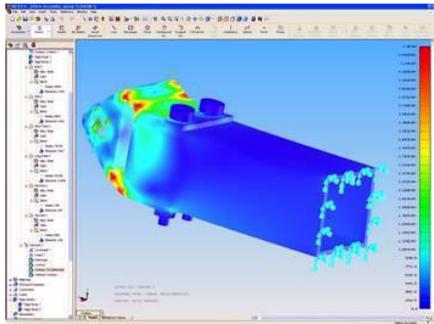
Thermal Eng.



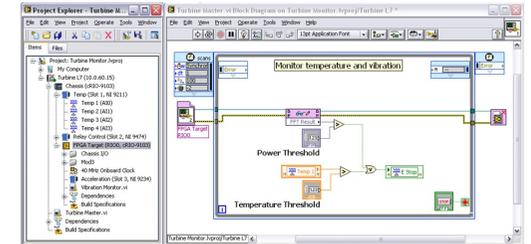
Electrical Eng.



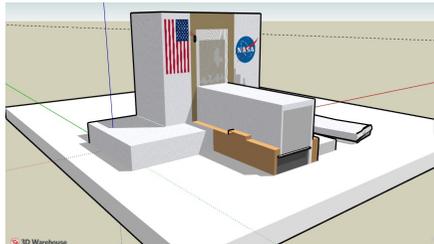
Mechanical Eng.



Control Eng.



Infrastructure Eng.



Systems Eng ???

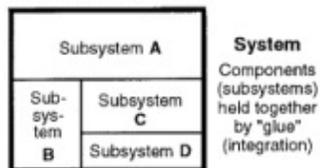




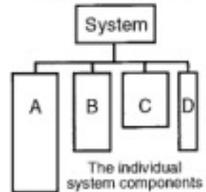
# QUAL A LINGUAGEM DA ENG. SIS.?



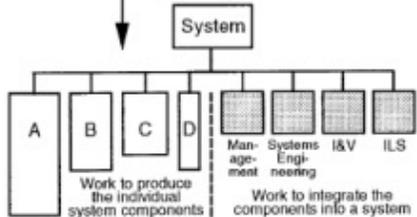
The whole **does more** than the sum of the parts.



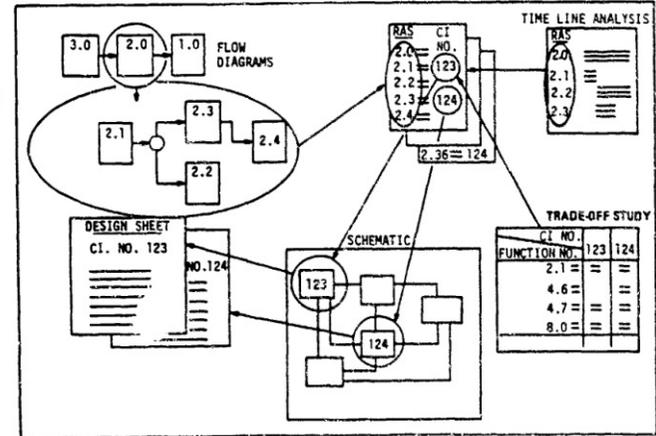
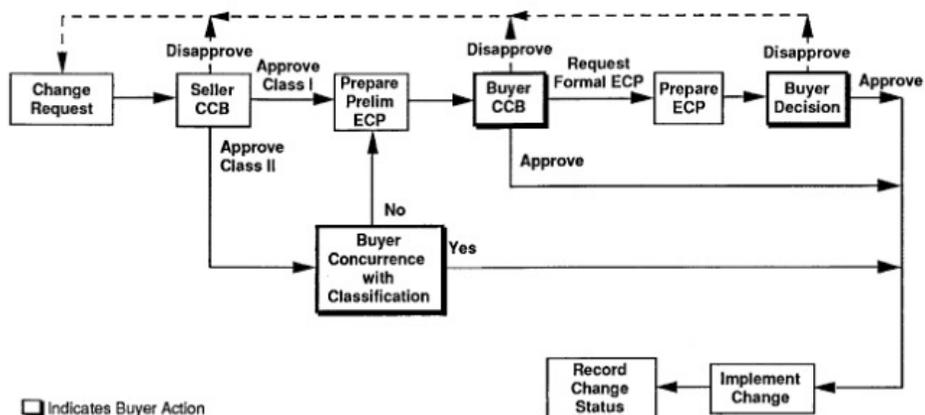
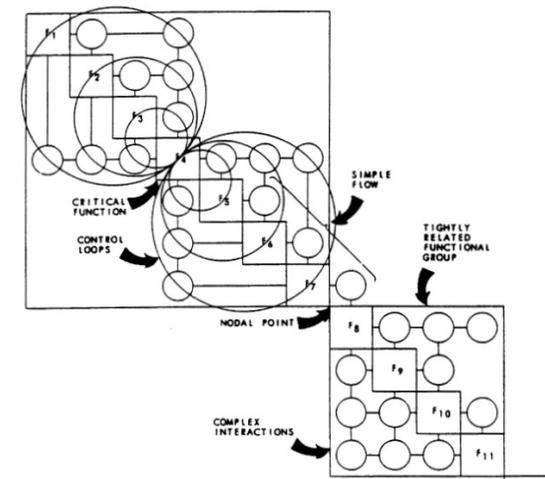
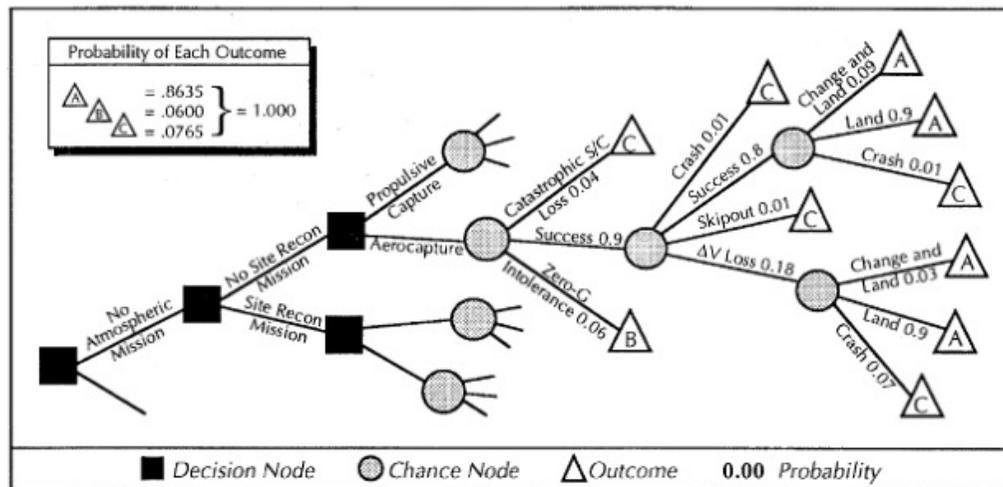
**Product Breakdown Structure (PBS)**  
Shows the components which form the system.



**Work Breakdown Structure (WBS)**  
All work components necessary to produce a complete system

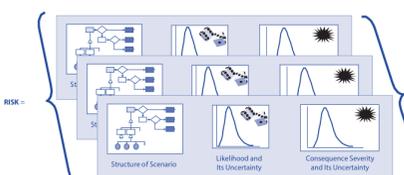
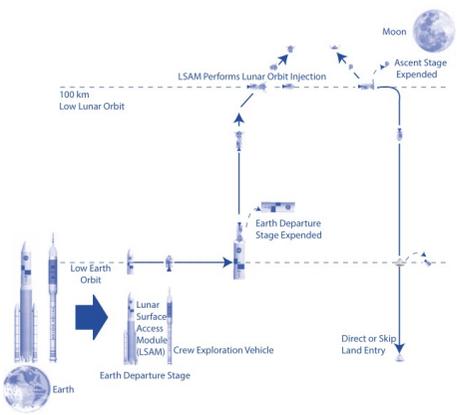
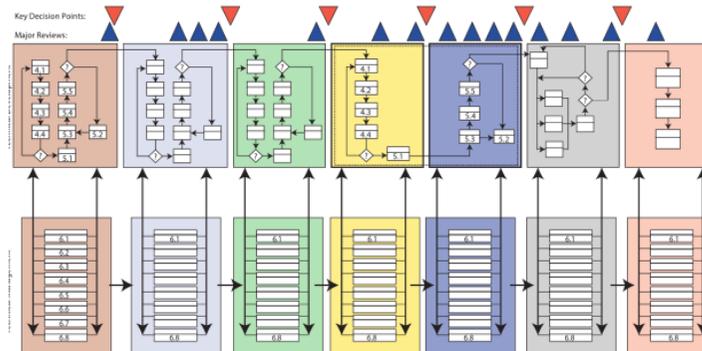
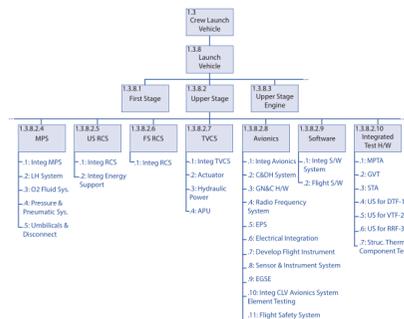
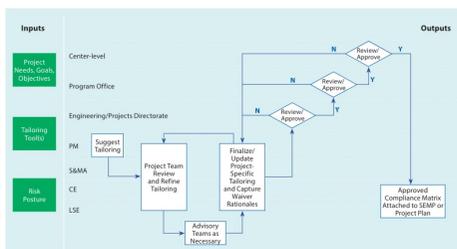


The whole **takes more work** than the sum of the parts.





# COM O TEMPO... FICARAM COLORIDOS



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

TABLE D-1 Requirements Verification Matrix

Requirement No.	Document	Paragraph	Shall Statement	Verification Success Criteria	Verification Method	Facility or Lab	Phase*	Acceptance Requirement?	Pre-flight Acceptance?	Performance Organization?	Results
P-1	xxx	3.2.1.1	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	Test	xxx	5	Yes	No	xxx	TPS xxx
P-1	xxx	Other paragraphs	Other "shall" in PFRS	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

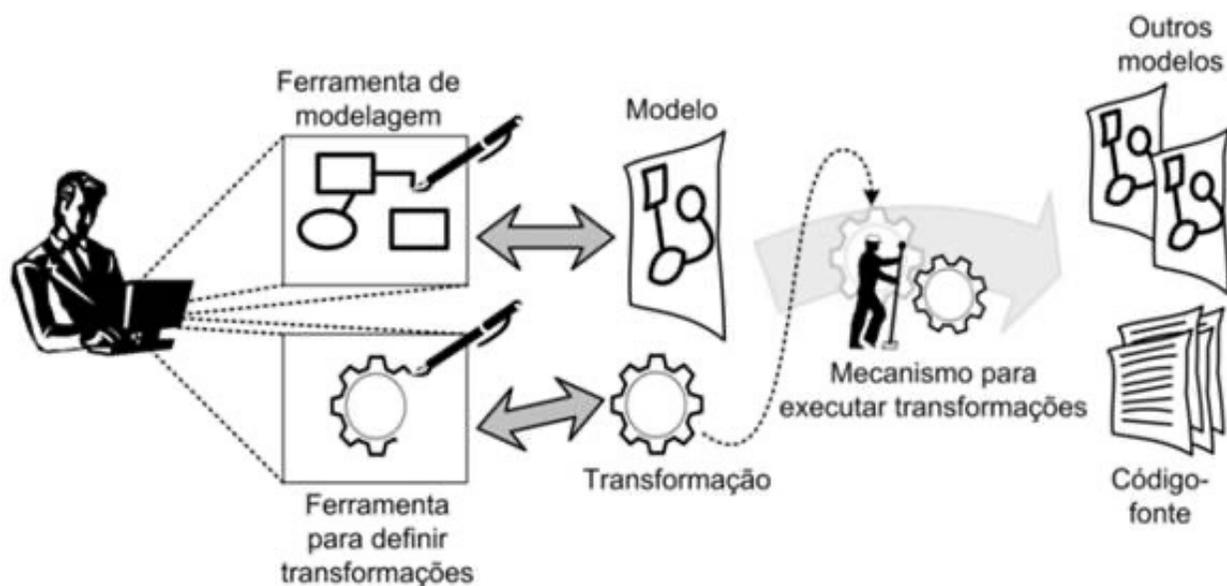
1.0 System	Demonstration Units				Environment		Unit Description		Overall TRL			
	Concept	Brassboard	Developmental Model	Prototypes	Flight Qualified	Relevant Environment	Space Launch Operation	Form		Fit	Function	Appropriate Scale
1.1 Subsystem X												
1.1.1 Mechanical Components												
1.1.2 Mechanical Systems												
1.1.3 Electrical Components				X			X	X	X	X		
1.1.4 Electrical Systems												
1.1.5 Control Systems												
1.1.6 Thermal Systems						X			X	X		
1.1.7 Fluid Systems		X										
1.1.8 Optical Systems												
1.1.9 Electro-Optical Systems												
1.1.10 Software Systems												
1.1.11 Mechanisms	X											
1.1.12 Integration												
1.2 Subsystem Y												
1.2.1 Mechanical Components												



# MODELOS



# MODEL DRIVEN DEVELOPMENT ...



- **MDD** – Model Driven Development
- **MDSD** – Model Driven Software Development
- **MDA** – Model Driven Architect
- **MDSE** – Model Driven Software Engineering
- **MDRE** – Model Driven Reverse Engineering
- **MM** – Model Management
- **ADM** – Architecture Driven Modernization
- **DDD** – Domain Driven Design
- **MBD** – Model Based Development
- ... → infinito

<http://www.devmedia.com.br/model-driven-architecture-com-enterprise-architect/33898>



# MODEL (DRIVEN X BASED X CENTRIC X ORIENTED)

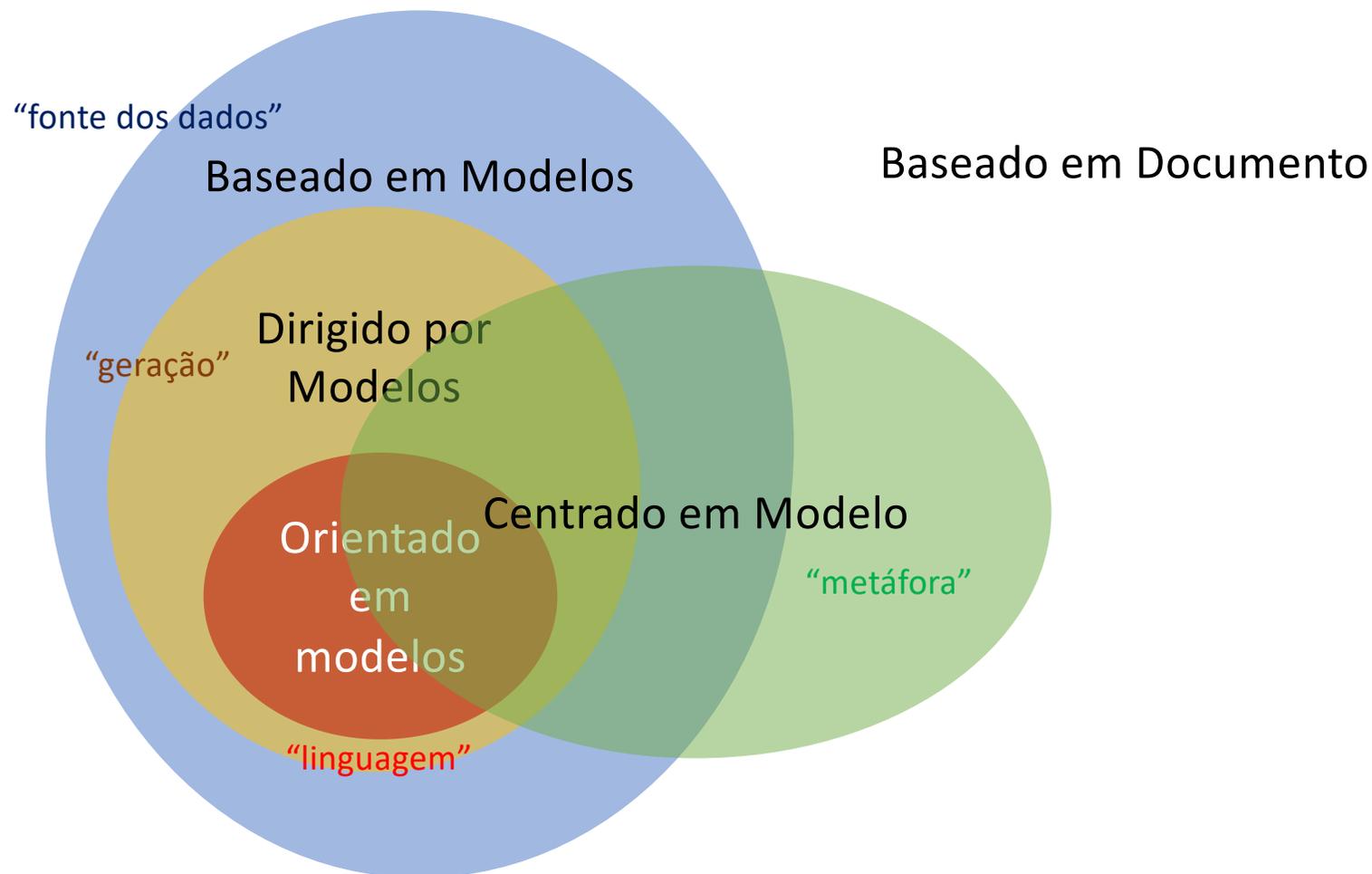
- **Driven (dirigido por):** modelos devem ser usados para **gerar sistemas executáveis**. Geração automática de código.
- **Based (baseado em):** modelos são **a fonte de dados** das atividades do ciclo de vida. Armazenamento de informação.
- **Centric (centrado em):** modelos são a **metáfora** de um sistema. Ferramentas de CAD.
- **Oriented (orientado em):** modelos e o sistema (código) são **indistinguíveis**. Scripts.

<http://onlinelibrary.wiley.com/doi/10.1002/spe.1155/abstract#spe1155-note-0001>

<http://www.3dcadworld.com/why-you-need-to-understand-model-based-engineering/>

<https://www.youtube.com/watch?v=VjGmNjg5cro>

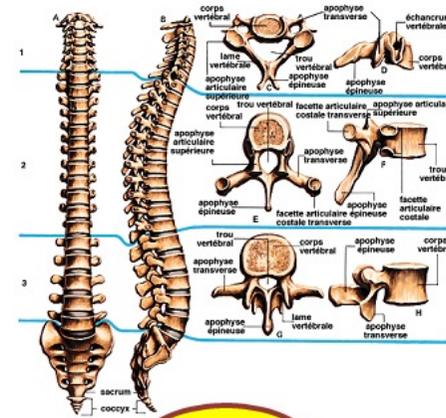
<http://cruise.eecs.uottawa.ca/umple/>



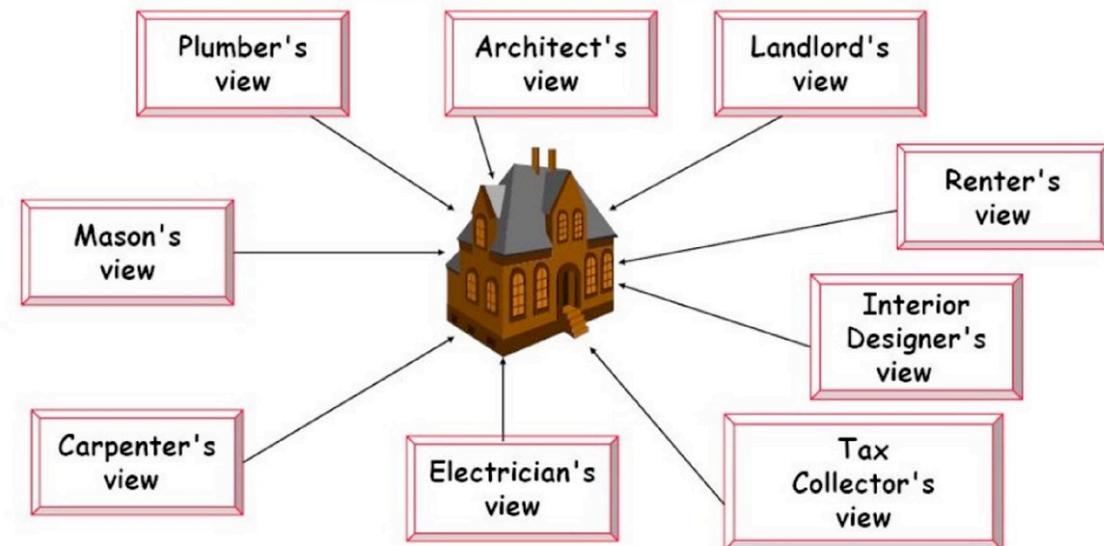


# MODELOS

- Modelos são imagens **simplificadas** da realidade (de um sistema).
- “*modullus*”, “*modus*” (medida)
  - O que? (**mapeamento**)
  - Como? (**redução de contexto**)
  - Pra quem, quem, para o que? (**pragmatismo**)
- Na verdade não é algo moderno...

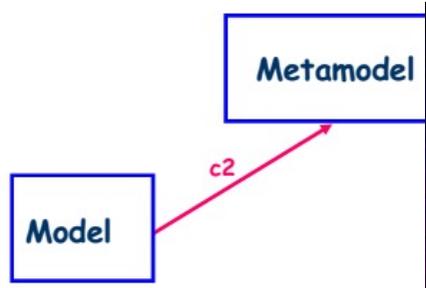
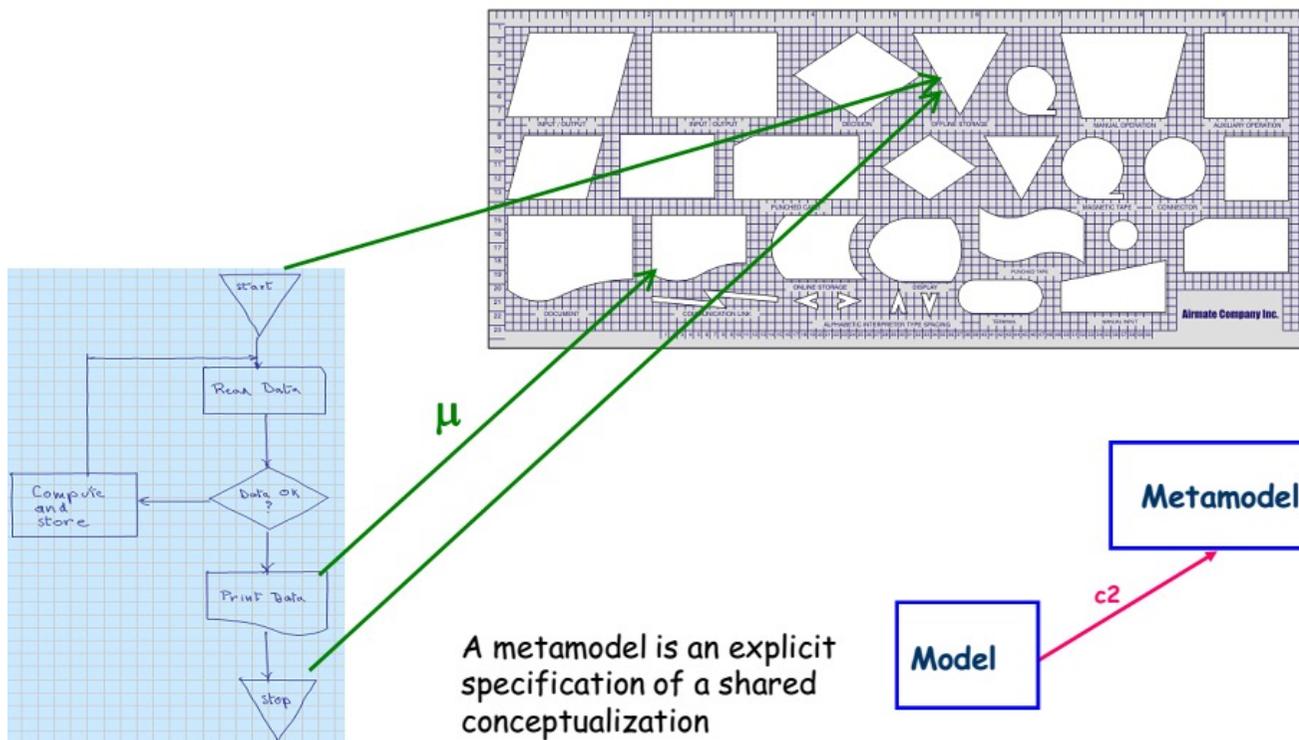


J. Bézivin, *Model Engineering for Software Modernization*, 2004.



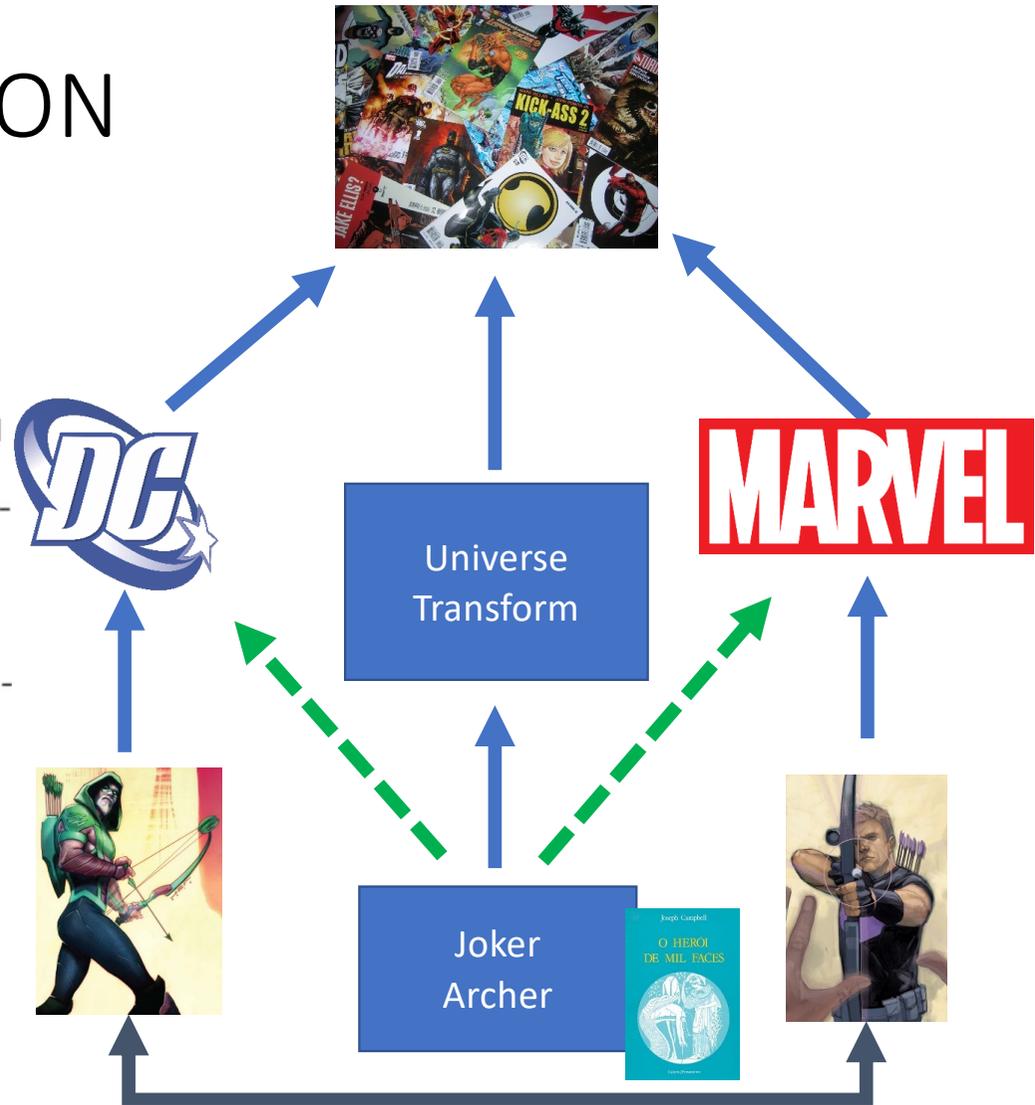
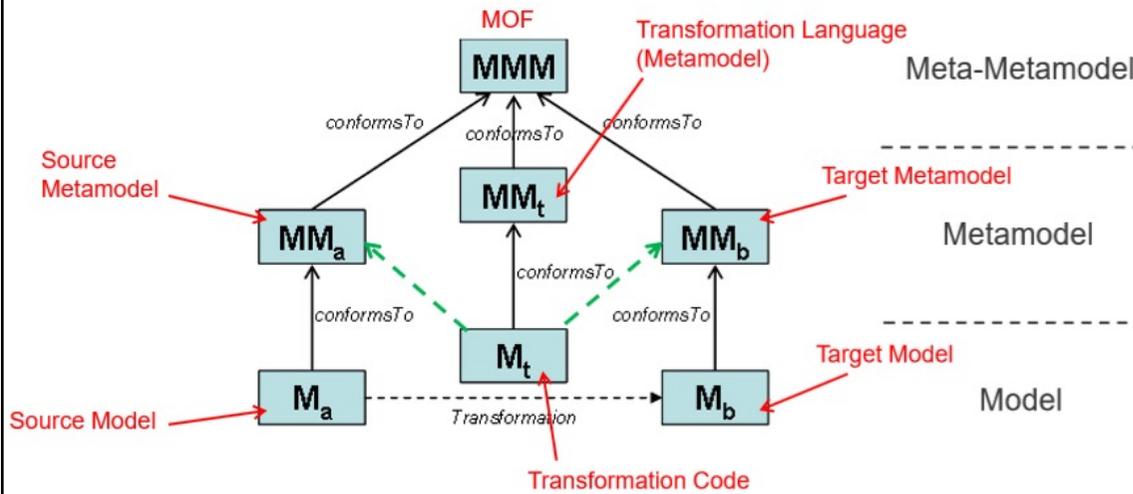


# TODO MODELO TEM UM META-MODELO



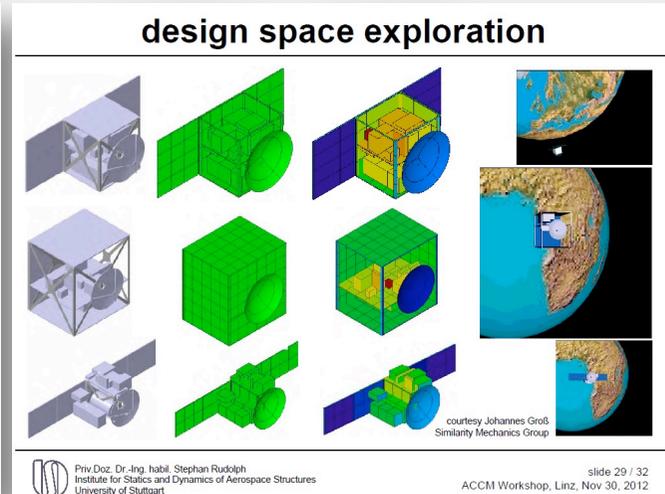
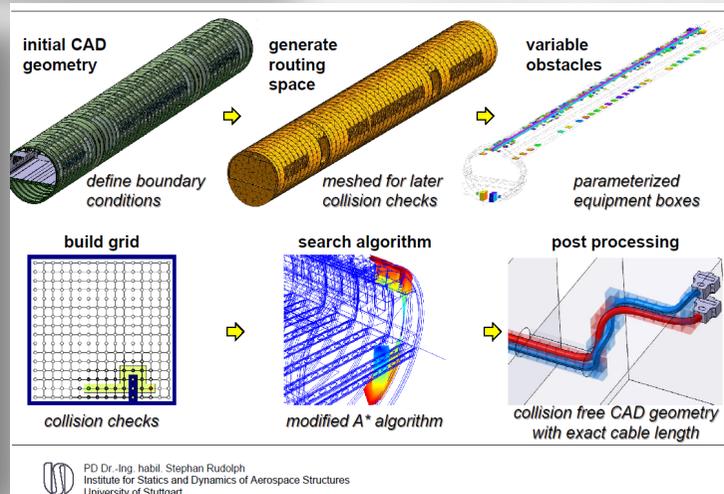
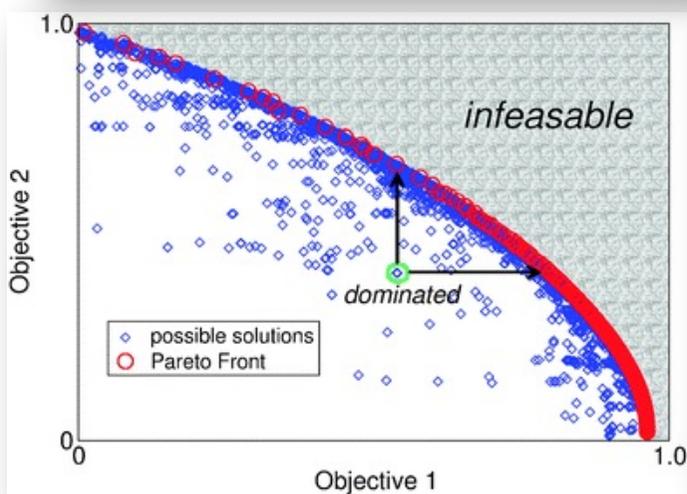
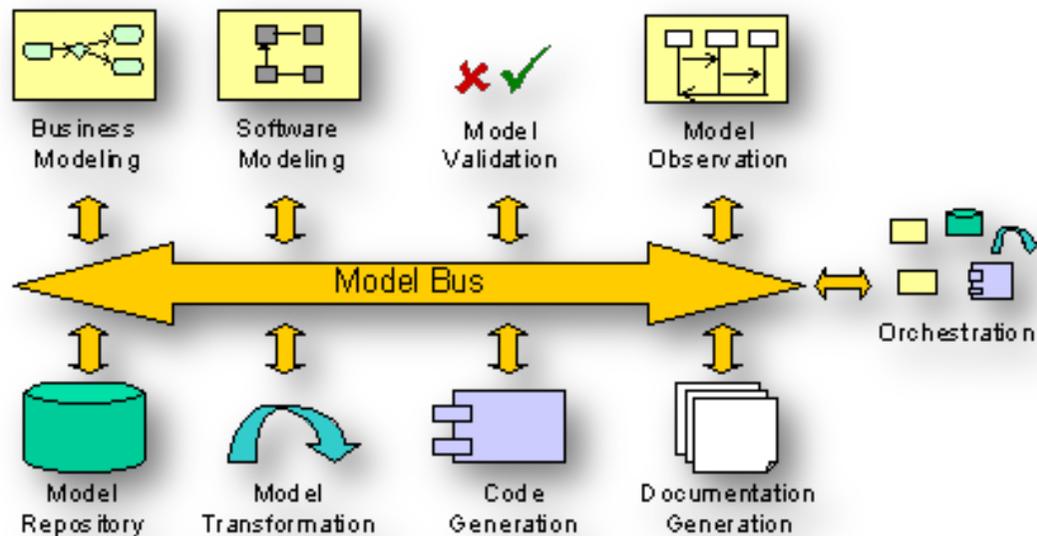
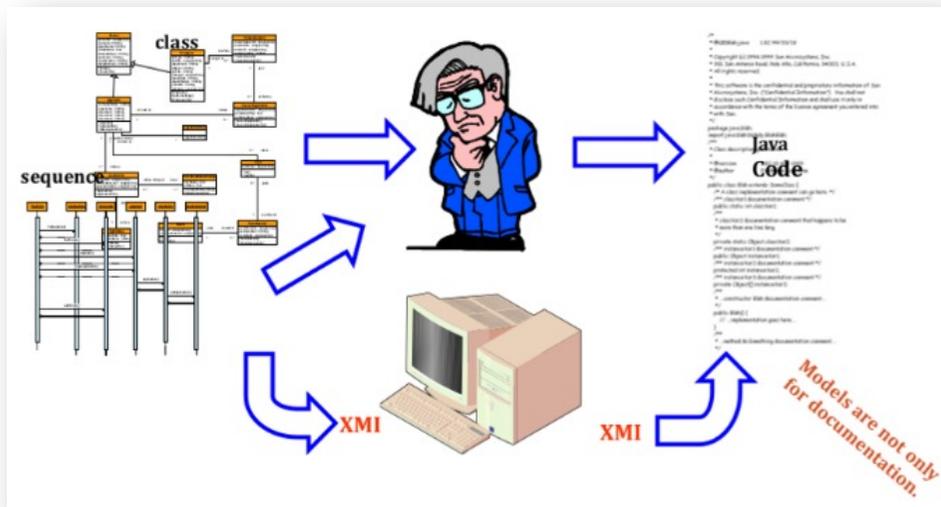


# MODEL TRANSFORMATION





# ISSO TRAZ BENEFÍCIOS





- Isso não é aleatório
- É a consequência da tecnologia computacional estar sendo apropriada pelas outras áreas.
- E não é diferente para a Engenharia de Sistemas.





# APLICANDO MODELOS NA ENGENHARIA DE SISTEMAS



# ENGENHARIA DE SISTEMAS BASEADA EM MODELOS (ESBM)

“**Model-Based Systems Engineering (MBSE)** é a aplicação formal da **modelagem** para suportar as atividades envolvendo **requisitos, projeto, análise, verificação e validação** de sistemas desde a **fase de projeto conceitual, continuando ao longo do desenvolvimento do produto até o descarte.**”



## ENTÃO... ESBM PRECISA DE UMA METODOLOGIA?

- *Apesar do nível de abstração, não há uma linguagem universal (controverso).*
- **Metodologia de ESBM** – Definido como um coleção de processos, métodos e ferramentas.
  - **Processo** – Uma sequencia lógica de atividades que definem “o **que**” precisa ser feito, sem especificar o como cada atividade é feita.
  - **Método** – Técnicas que irão indicar “**como**” cada atividade é feita.
  - **Ferramenta** – Um instrumento que irá realizar um determinado método para aumentar a eficiencia de uma tarefa..



# INCOSE LISTA DIVERSAS METHODOLOGIES

## List of Methodologies and Methods

### Methodologies Surveyed in INCOSE 2008 Report

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



### Additional Methodologies Identified as Gaps Since 2008 INCOSE Survey

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





# OMG POSSUI UM WORKGROUP SÓ PARA MBSE

The screenshot shows the MBSE Wiki page for the INCOSE IW 2023 workshop. The page header includes the OMG logo (OMG Standards Development Organization) and the text 'MBSE Wiki'. There is a search bar and navigation links for 'Recent Changes', 'Media Manager', and 'Sitemap'. A breadcrumb trail reads 'Trace: methodology · start · incose\_mbse\_iw\_2023'. The main content area features the title 'Model Based Systems Engineering (MBSE) Workshop at INCOSE IW 2023' and a 'Table of Contents' sidebar. The sidebar lists: 'Model Based Systems Engineering (MBSE) Workshop at INCOSE IW 2023', 'MBSE Workshop Schedule' (with sub-items for Saturday, January 28, 2023 and Sunday, January 29, 2023), and 'Related Sessions at IW 2023 for SE Transformation and MBSE Initiative'. The main text describes MBSE as the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities. It mentions that the MBSE initiative was established in 2007 to realize the Systems Engineering Vision of the 'future of systems engineering is model based' to scale systems engineering by replacing document-oriented practices with models. It also states that since its launch, the MBSE working groups have expanded to become a major part of the INCOSE International Workshop with models of all kinds used in many working groups to push the practice forward. To disseminate this MBSE information, the MBSE Initiative hosts the annual MBSE Workshop. This year the core workshop is being held on one afternoon to avoid the 'fire hose' of the former two-day format. It further notes that as in previous years, the larger MBSE Initiative is an integrated activity occurring over the full duration of the International Workshop. The MBSE Workshop sessions are being held Saturday afternoon according to the schedule below. As in recent years, the workshop is being hosted as a hybrid event for both in-person and online attendance. The page concludes with an invitation: 'We invite you to join us on the cutting-edge of systems engineering for a riveting workshop.'

**Model Based Systems Engineering (MBSE) Workshop at INCOSE IW 2023**

Model-Based Systems Engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout product development to retirement. The MBSE initiative was established in 2007 to realize the Systems Engineering Vision of the “future of systems engineering is model based” to scale systems engineering by replacing document-oriented practices with models.

Since its launch, the MBSE working groups have expanded to become a major part of the INCOSE International Workshop with models of all kinds used in many working groups to push the practice forward. To disseminate this MBSE information, the MBSE Initiative hosts the annual MBSE Workshop. This year the core workshop is being held on one afternoon to avoid the ‘fire hose’ of the former two-day format.

As in previous years, the larger MBSE Initiative is an integrated activity occurring over the full duration of the International Workshop. The MBSE Workshop sessions are being held Saturday afternoon according to the schedule below. As in recent years, the workshop is being hosted as a hybrid event for both in-person and online attendance.

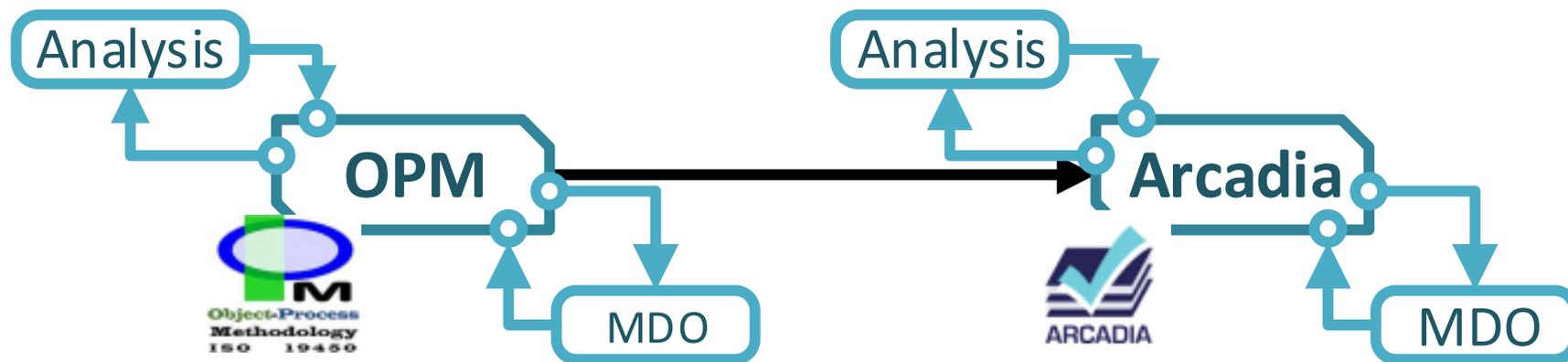
We invite you to join us on the cutting-edge of systems engineering for a riveting workshop.

**MBSE Workshop Schedule**

All times below are in US Pacific Time Zone (US Eastern Time - 3 hours; UTC - 8 hours).



# ABORDAGEM NA DISCIPLINA





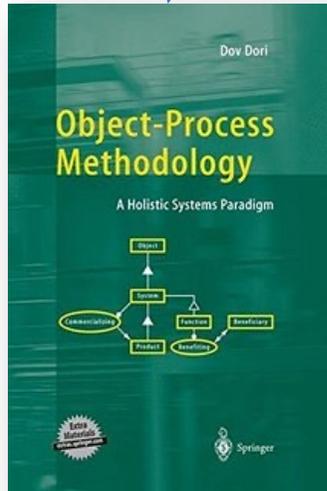
# OPM

Prof. Dov Dori



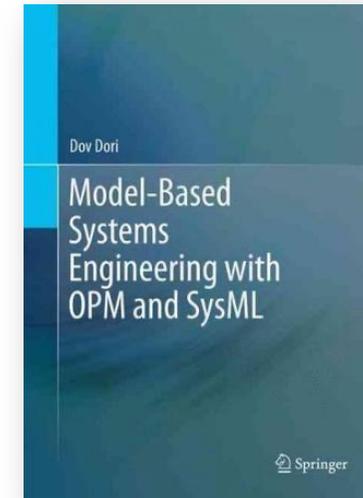
Created in 2002

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



improving and showing it applicability

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



## ISO/PAS 19450:2015<sup>o</sup>

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

### Abstract

Preview ISO/PAS 19450:2015

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.

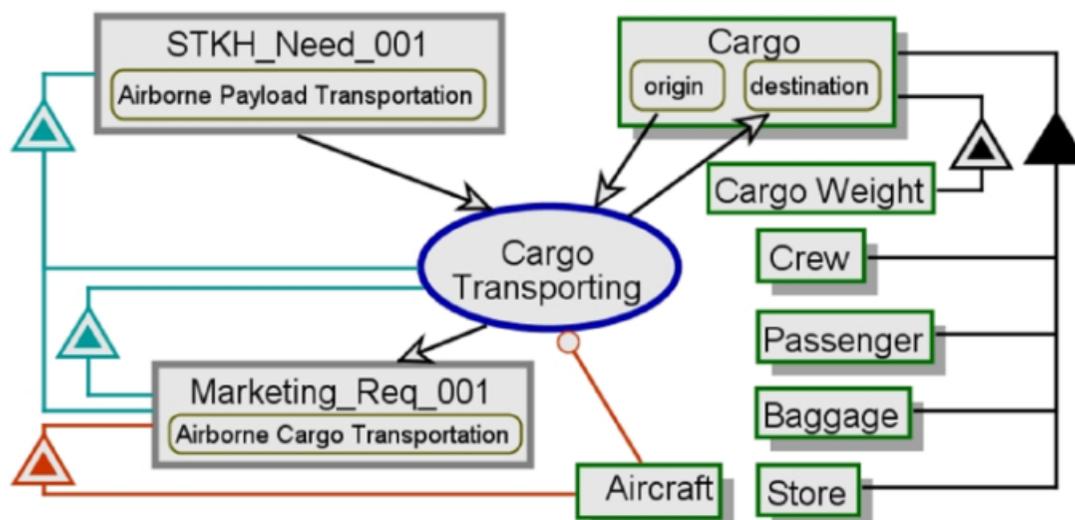
While ISO/PAS 19450:2015 presents some examples for the use of Object-Process Methodology to improve clarity, it does not attempt to provide a complete reference for all the possible applications of Object-Process Methodology.

FORMAT LANGUAGE

PDF English

PAPER English

CHF 198 Add to basket



- STKH\_Need\_001 is Airborne Payload Transportation.
  - Cargo Transporting consumes STKH\_Need\_001.
  - Cargo Transporting yields Marketing\_Req\_001.
  - Marketing\_Req\_001 is Airborne Cargo Transportation.
  - Marketing\_Req\_001 exhibits Aircraft.
  - Cargo Transporting requires Aircraft.
  - Aircraft is physical.
  - Cargo Transporting changes Cargo from origin to destination.
- Function Defining  
Requirements Identifying  
Requirements Allocating
- STKH\_Need\_001 exhibits Marketing\_Req\_001, as well as Cargo Transporting.
  - Cargo Transporting exhibits Marketing\_Req\_001.
- Traceability
- Cargo is physical.
  - Cargo can be origin or destination.
  - Cargo exhibits Cargo Weight.
  - Cargo consists of Passenger, Baggage, Store, and Crew.
  - Passenger is physical.
  - Baggage is physical.
  - Store is physical.
  - Crew is physical.
- Configuration management



# XP Z67-140 - ARCADIA

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

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

Accédez aux tutoriels Identifiez-vous

< Retour SUIVRE

### 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

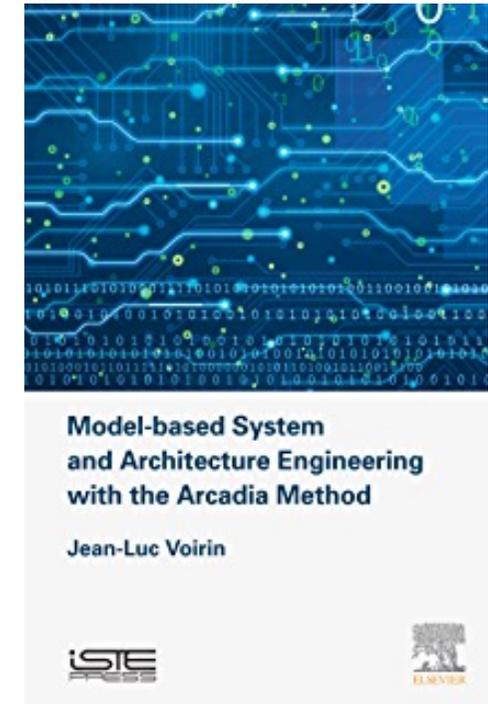
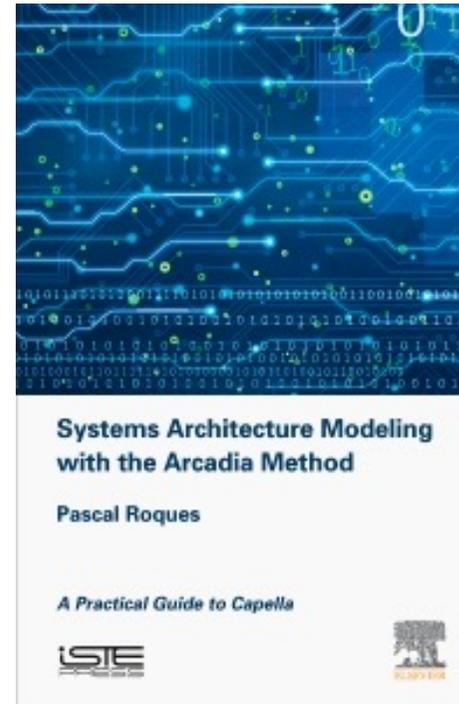
Résumé : La méthode ARCADIA peut être appliquée à la définition de la conception de tout type de système, en se concentrant sur la description et l'évaluation des propriétés de conception (coût, performance, sécurité, réutilisation, consommation, poids ...).

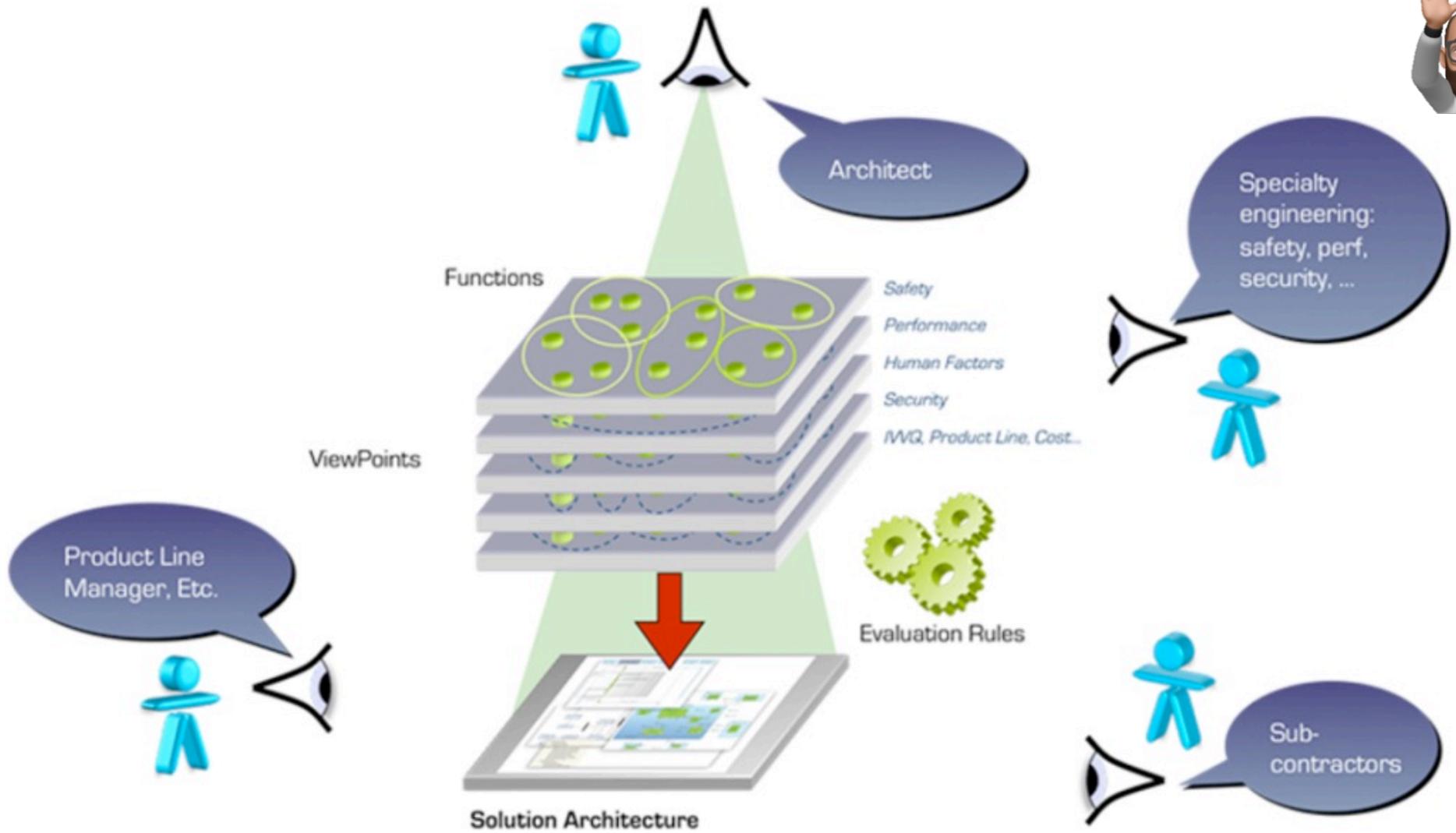
Je veux en savoir plus J'accède à la consultation

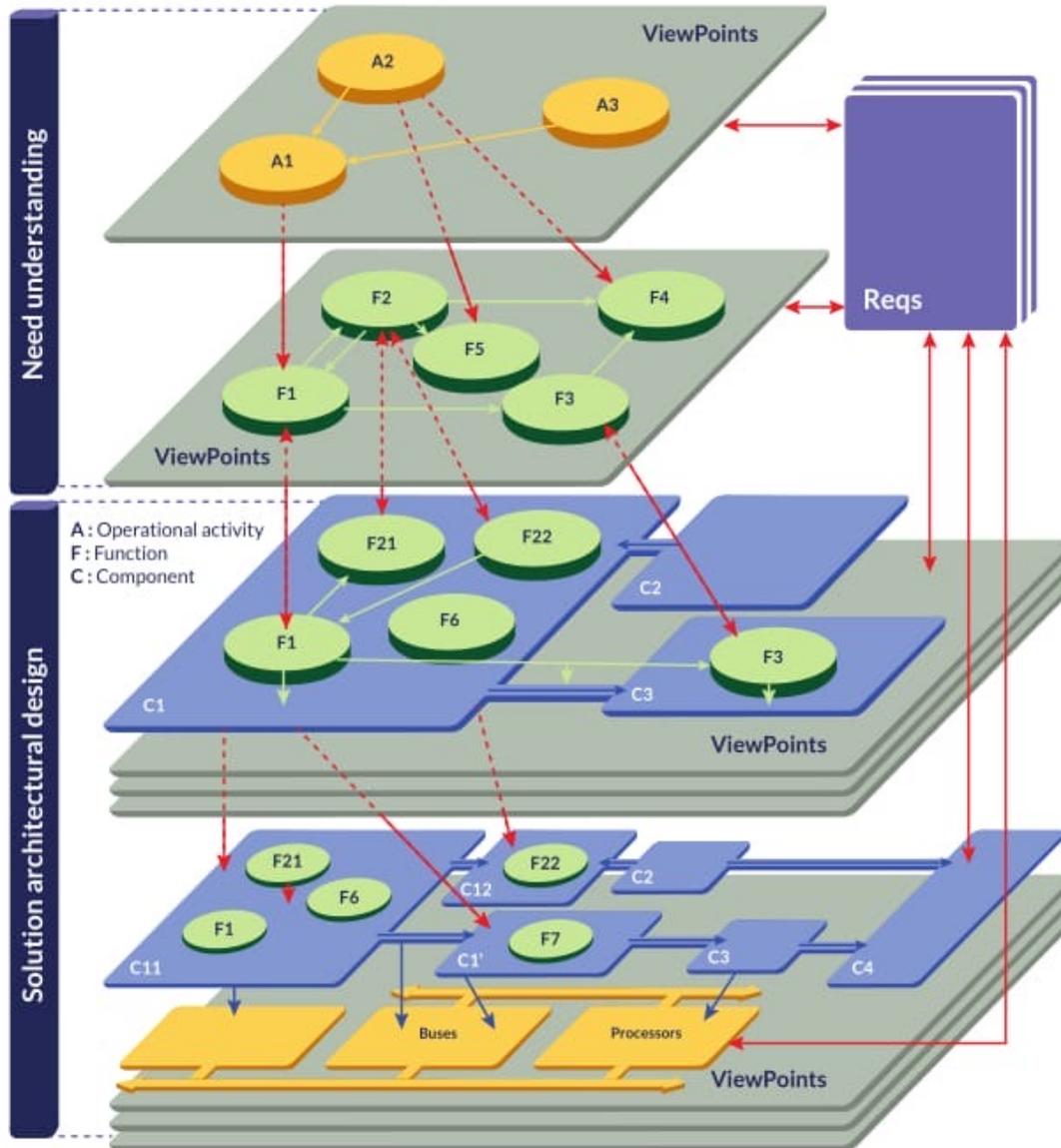
Vie de la norme

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/norme/XP%20Z67-140/tech...)







### Operational Analysis

What the users of the system need to accomplish

### Functional & Non Functional Need

What the system has to accomplish for the users

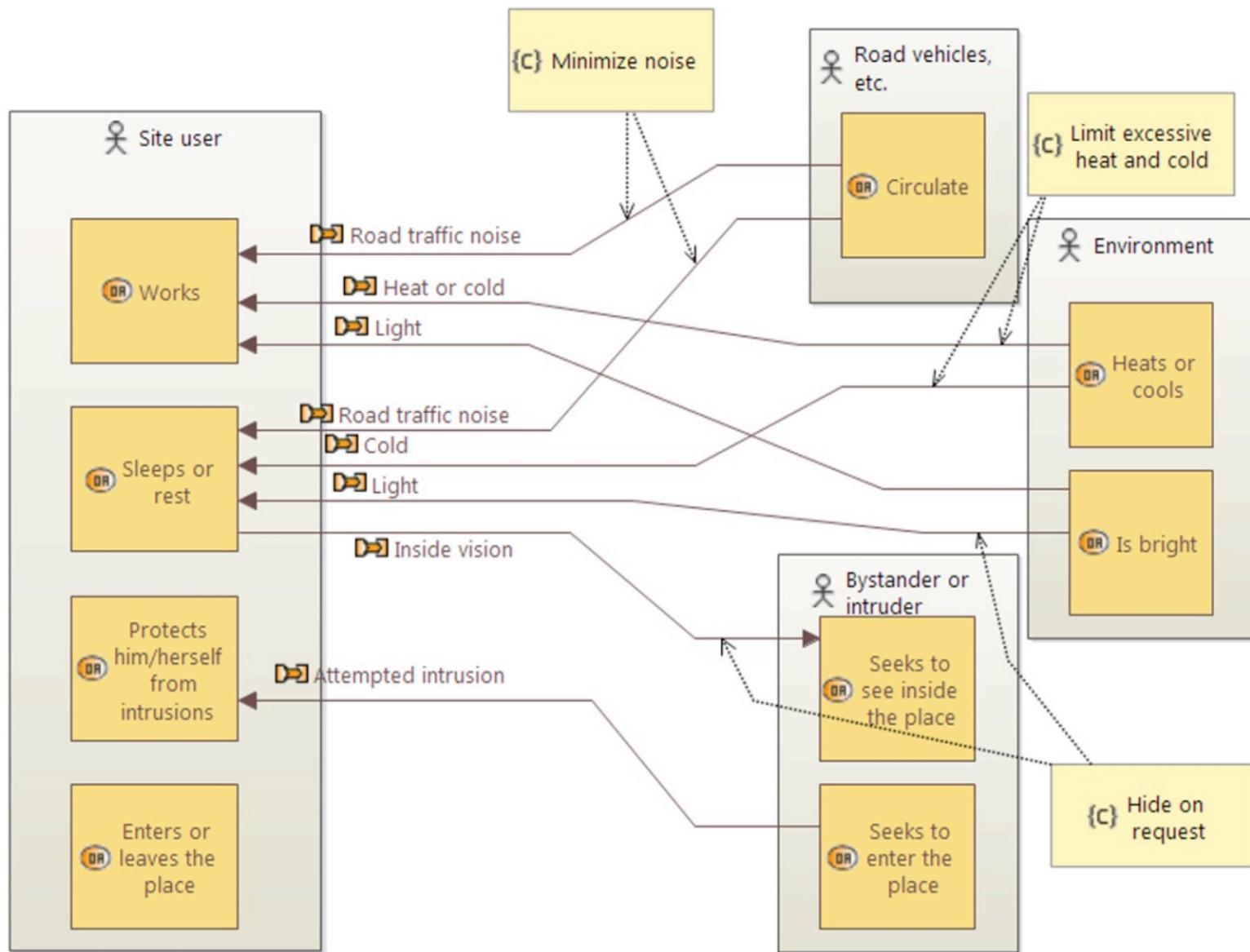
### Logical Architecture

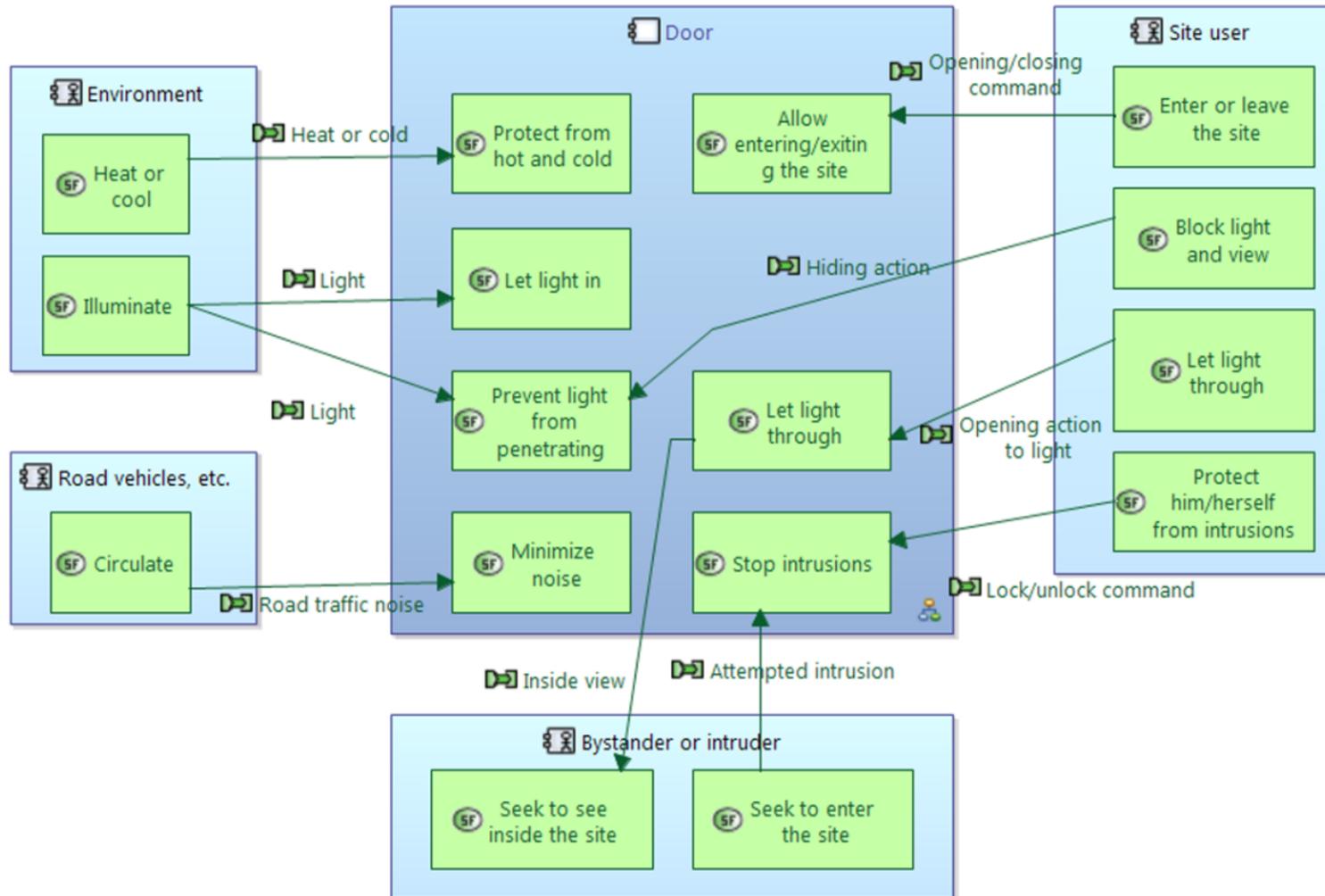
How the system will work to fulfill expectations

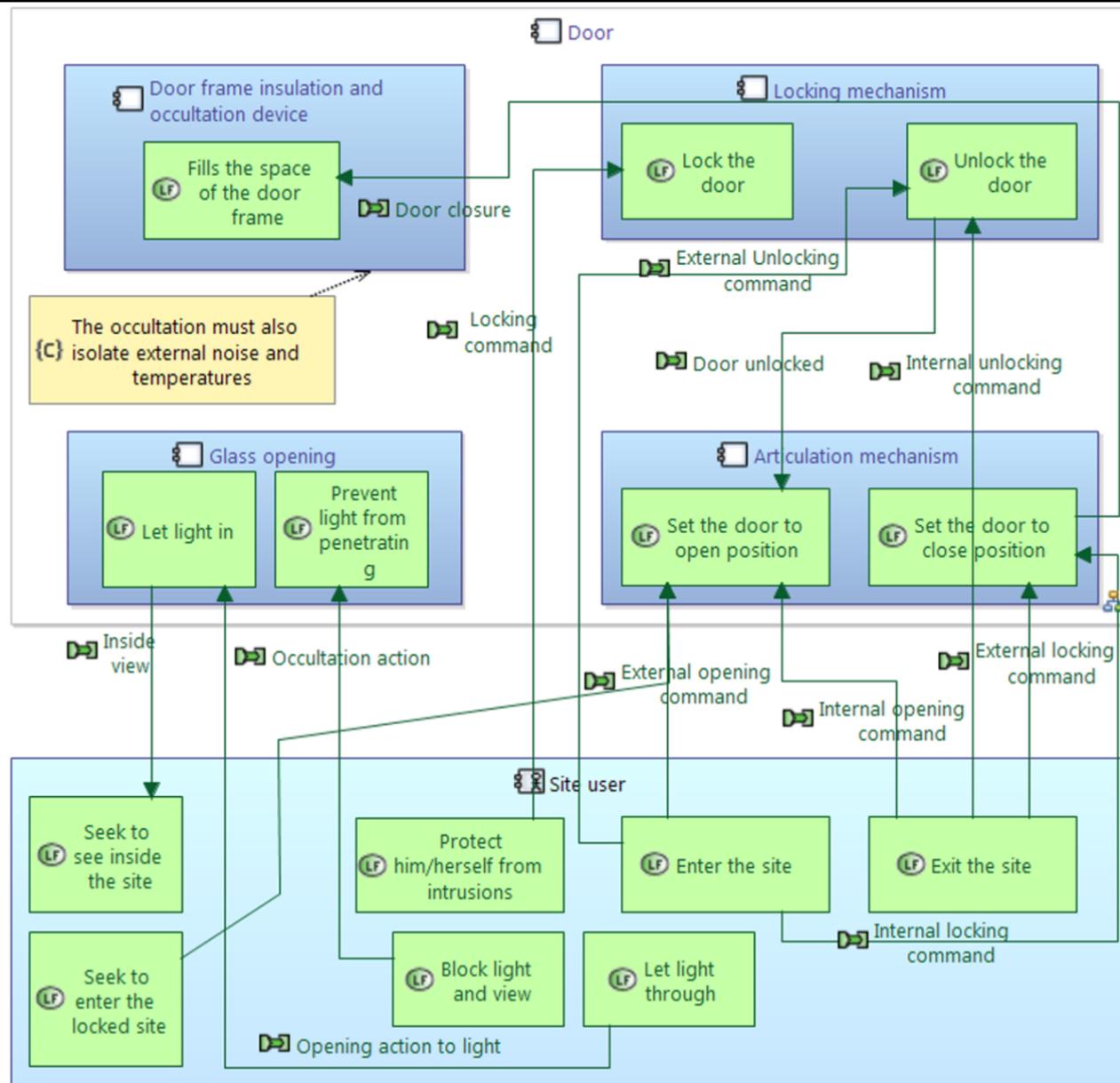
### Physical Architecture

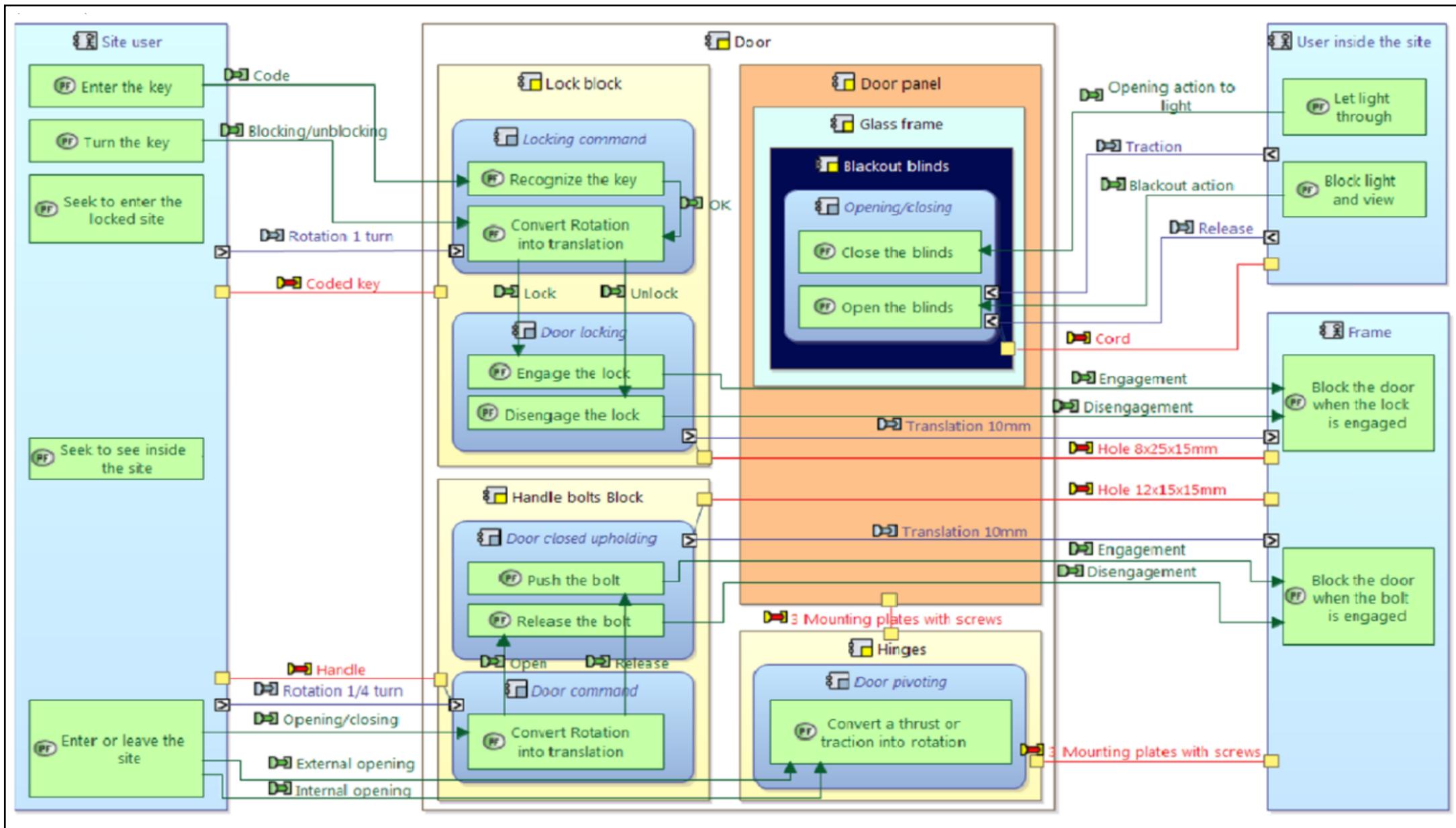
How the system will be developed and built





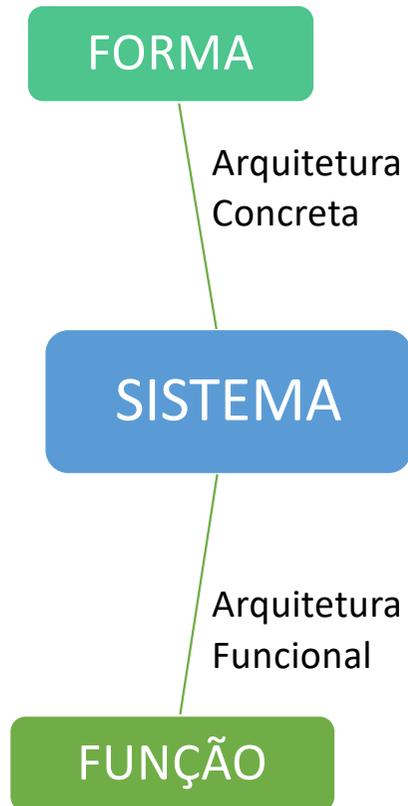








# ARQUITETURA



- Um relacionamento funcional requer um relacionamento entre formas.
- A relação da forma é o instrumento da relação funcional.

*O coração não pode trocar sangue com o pulmão sem uma conexão.*



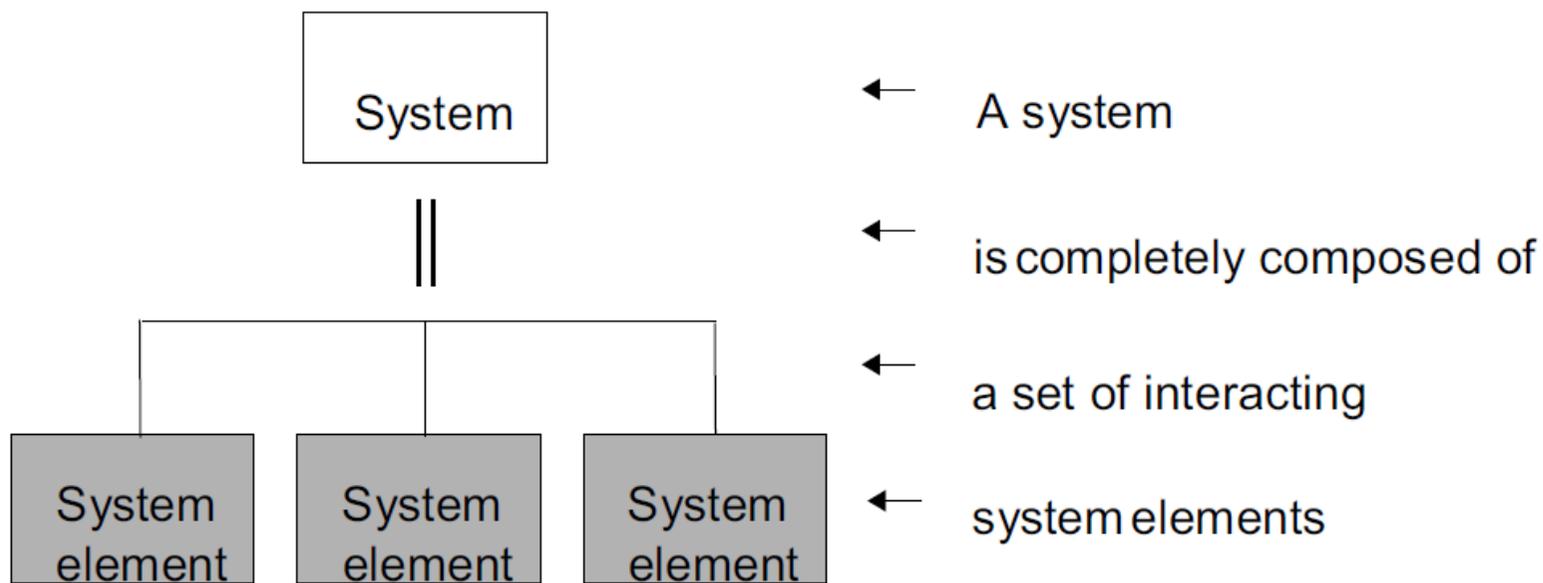
# FILOSOFIA

- **Forma é o que foi ou é eventualmente implementado.**
- **Forma é sobre existência.**
- **Forma é o que o sistema é.** É a manifestação concreta e muitas vezes visível do sistema.
- 
- A forma é a personificação física ou informacional de um sistema que existe ou tem potencial para existir, por algum período, e é instrumental na execução da função. As formas incluem as entidades de forma e as relações entre as entidades. A forma existe antes da execução da função.
- Forma é um atributo produto/sistema.



# HIERARQUIA DO SISTEMA

- Relação entre sistema e elemento do sistema

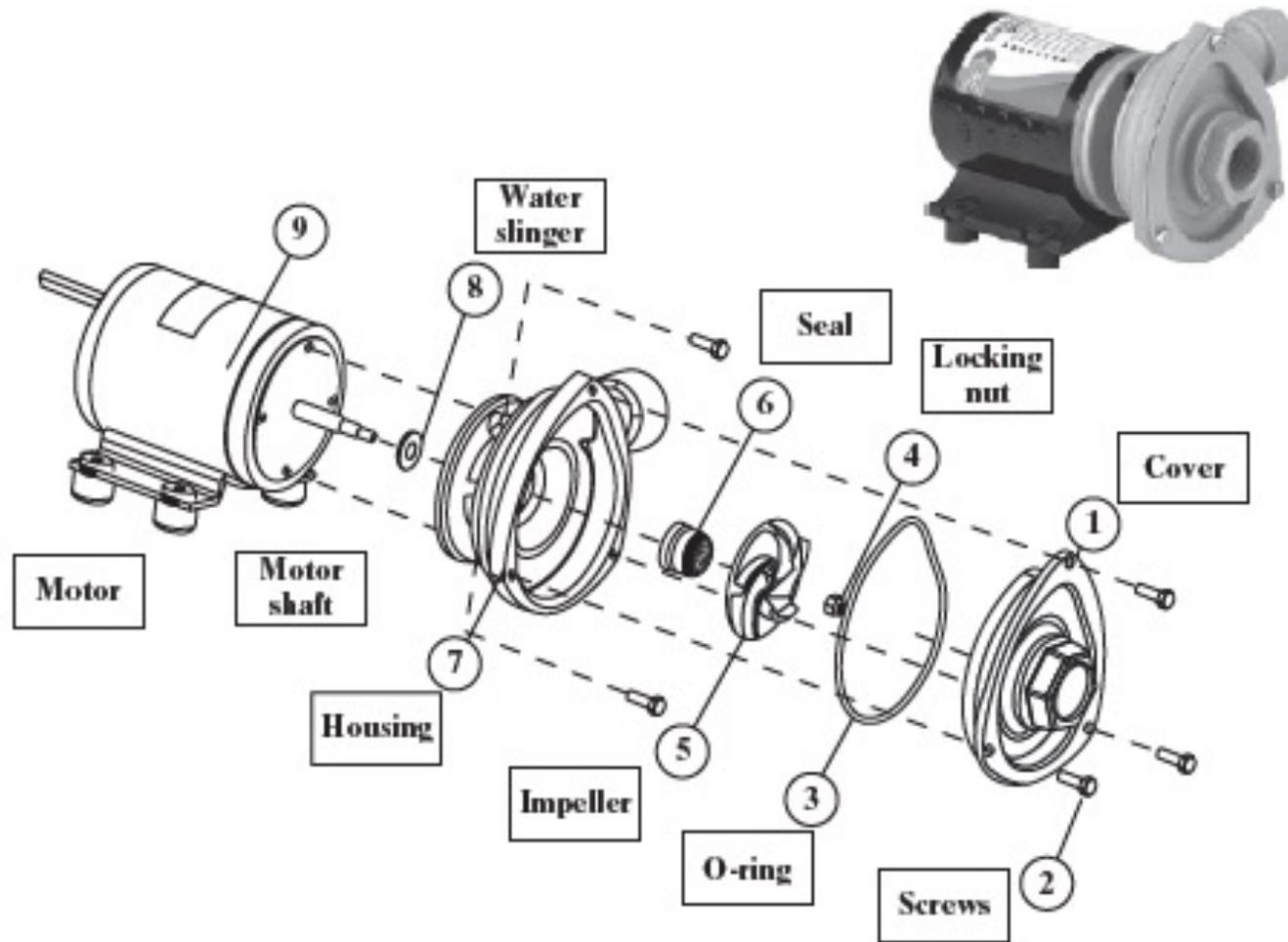


O sistema é decomposto em uma hierarquia de elementos cada vez menores.





# EXEMPLO DE FORMA: BOMBA CENTRÍFUGA







# EXEMPLO DE FORMULÁRIO: CÓDIGO DE SOFTWARE

```
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
```





# ARKHITEKTON



<http://ordinary-citizen.com/2018/05/07/roger-scruton-on-craft-and-beauty/>



A arquitetura de sistemas é a **realização do conceito**, a **alocação da função física/informacional** aos elementos da **forma** e a **definição de relações entre os elementos com o contexto** circundante.



# UM POUCO SOBRE FUNÇÕES



## DEFINIÇÃO DE FUNÇÃO

- **Função** é uma atividade, operação ou transformação que causa ou contribui para o desempenho.
- Em sistemas, **função são as ações para as quais um sistema existe**, o que, em última análise, leva à entrega de valor.
- A **função é executada pela forma**, que é o instrumento da função.
- A **função emerge da interação funcional** entre entidades. Função é um atributo produto/sistema.
- Função é sobre atividade, em contraste com a forma, que é sobre a existência.



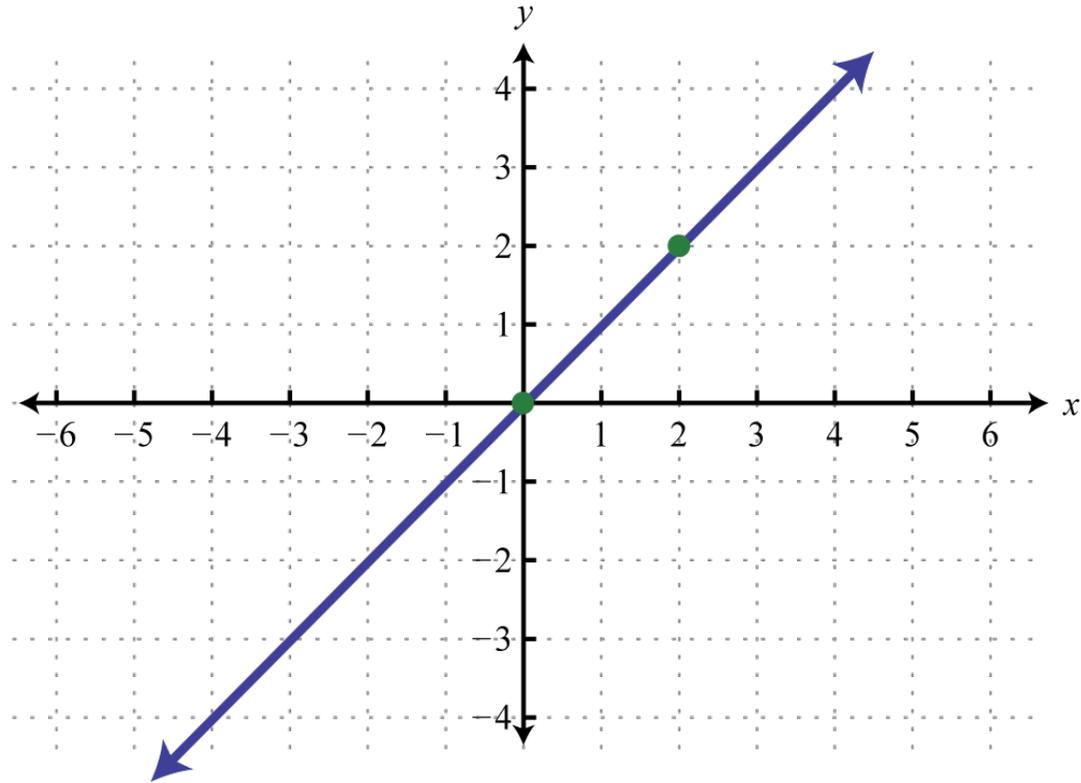
# IMPORTÂNCIA DA FUNÇÃO

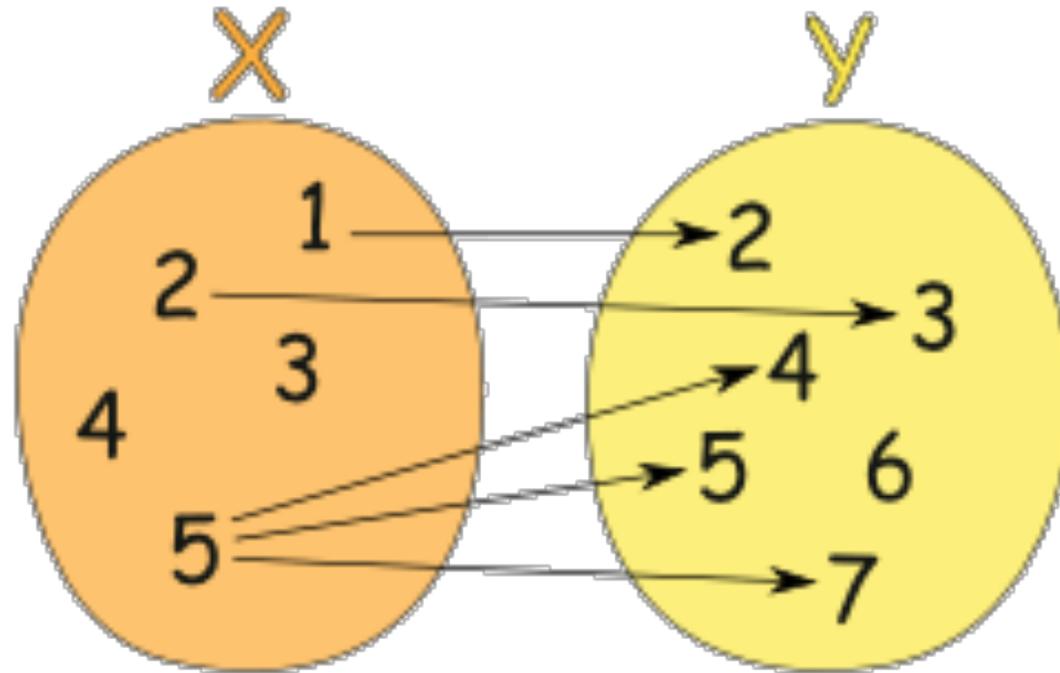
- Assim como a forma tem entidades de forma (objetos) e relações de forma entre as entidades (estrutura), a **função tem entidades de função e relações funcionais** entre as entidades (interações).
- A função que vemos na superfície de um sistema é resultado da emergência que ocorre entre essas entidades de função.
- **Toda a magia dos sistemas e seu surgimento, e quase todo o desafio de projetá-los, é encontrada no domínio funcional.**



$$f(x) = x$$

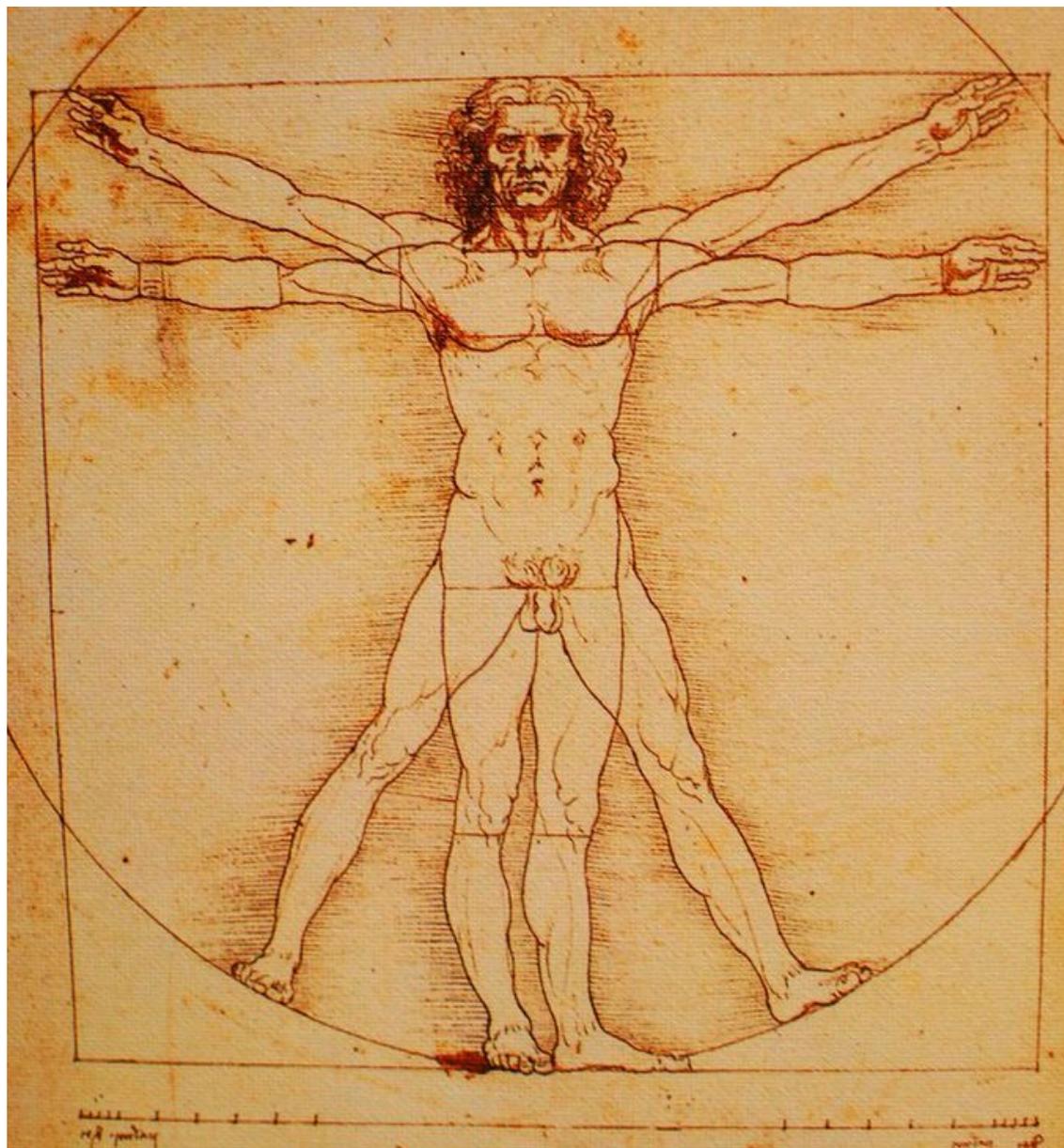
$x$	$f(x)$
0	0
2	2













**off the mark**.com

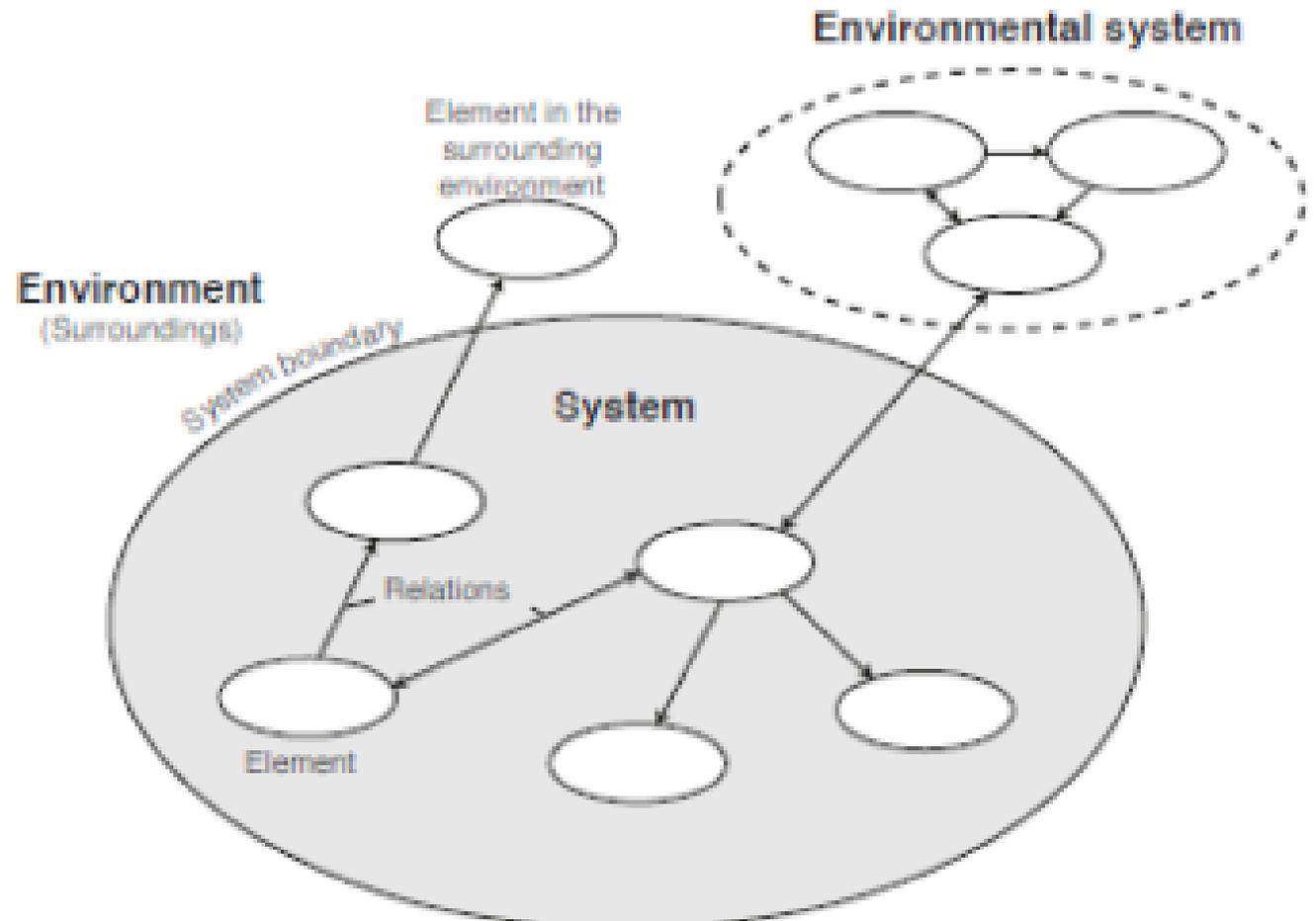
by Mark Parisi





# FRONTEIRA

- Entende-se por fronteira uma **borda entre o sistema e seus arredores** ou o ambiente em que está inserido.





# FUNÇÕES INTERNAS

- Dentro de um sistema existem funções internas e relações entre essas funções internas.
- Juntos, eles definem a arquitetura funcional do sistema.
- A função, o desempenho e as "ilidades" emergem dessas funções e relacionamentos internos.
- A principal tarefa é identificar as entidades de função interna.
-



# IDENTIFICANDO A FUNÇÃO INTERNA

- Como identificamos as funções internas?
  - Existem várias abordagens, incluindo engenharia reversa da forma, projetos e metáforas.
- Nós raciocinamos, “O que esses elementos fazem?”
  - Muitas vezes, apenas observar as operações também é valioso.. Ao aplicar essas abordagens, devemos primeiro nos concentrar nos processos internos que levam à criação de valor.
  - Também devemos aplicar o conhecimento e a experiência do domínio.

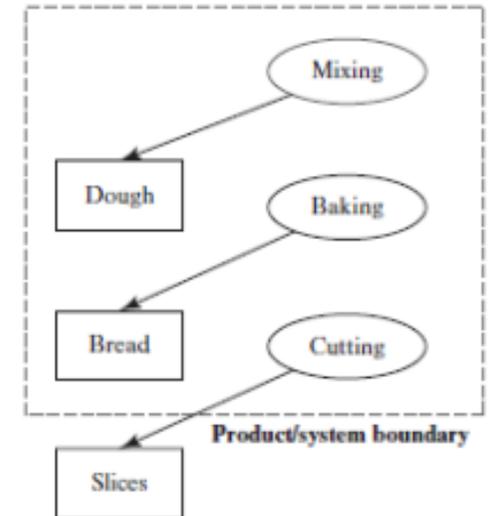


FIGURE 5.10 Internal functions of bread slice making.



# FUNÇÕES EXTERNAS

- Entrega de valor
  - Primeiro, a função deve ser entregue externamente; deve cruzar a fronteira do sistema e influenciar algo no contexto.
  - **A função agrega valor quando atua externamente ao sistema.** É importante que a função seja entregue externamente, que não seja apenas interna ao sistema..
- Restrição de ambiente
  - **Elementos externos também podem restringir/conduzir o sistema.**
  - Essas funções são (geralmente) incontrolláveis e são fonte de ESTADOS/MODOS para o sistema.



# MODELAGEM FUNCIONAL

- A modelagem funcional em Engenharia de Sistemas é uma **representação estruturada de funções** (ou seja, atividades, ações, processos, operações) dentro do sistema modelado.
- O objetivo do modelo funcional é descrever as funções e processos, auxiliar na descoberta das necessidades de informação, ajudar a identificar oportunidades e estabelecer uma base para determinar os custos de produtos e serviços.



## PARTICIONAMENTO FUNCIONAL

- O particionamento funcional é o processo de agrupamento de funções que se ajustam logicamente aos componentes que provavelmente serão usados e para minimizar as interfaces funcionais.
- **O particionamento é realizado como parte da decomposição funcional. Ele identifica agrupamentos lógicos de funções que facilitam o uso de componentes modulares e projetos de sistema aberto.**
- O particionamento funcional também é útil para entender como os equipamentos ou componentes existentes (incluindo comerciais) funcionarão com ou dentro do sistema..



## "RECONSTRUTIBILIDADE"

- A decomposição funcional refere-se ao processo de resolução de uma relação funcional em suas partes constituintes de tal forma que a função original possa ser reconstruída a partir dessas partes.





# ENTENDIMENTO

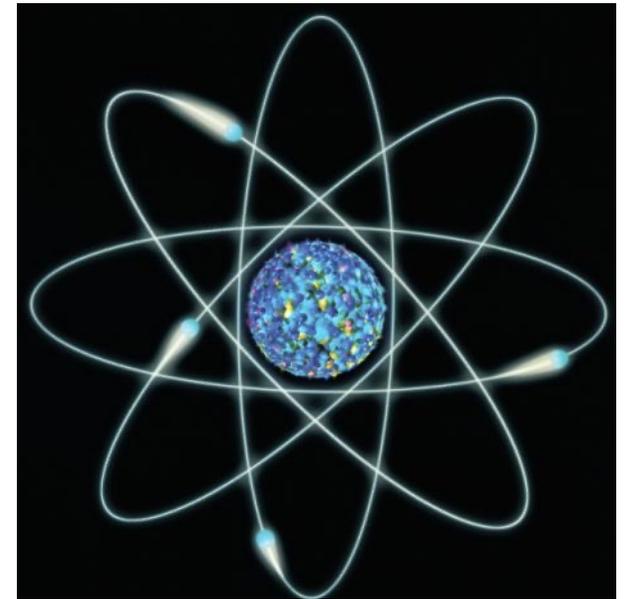
- Em geral, esse processo de decomposição é realizado com o propósito de obter informações sobre a identidade dos componentes constituintes ou com o propósito de obter uma representação comprimida da função global – uma tarefa que só é viável quando os processos constituintes possuem um certo nível de modularidade.





## FUNÇÃO ATÔMICA

- Na decomposição funcional de sistemas, que é um método para analisar sistemas, a ideia básica é tentar dividir um sistema de tal forma que cada bloco do diagrama de blocos possa ser descrito sem usar as palavras 'e' ou 'ou'.
- Este exercício força cada parte do sistema a receber uma função pura.



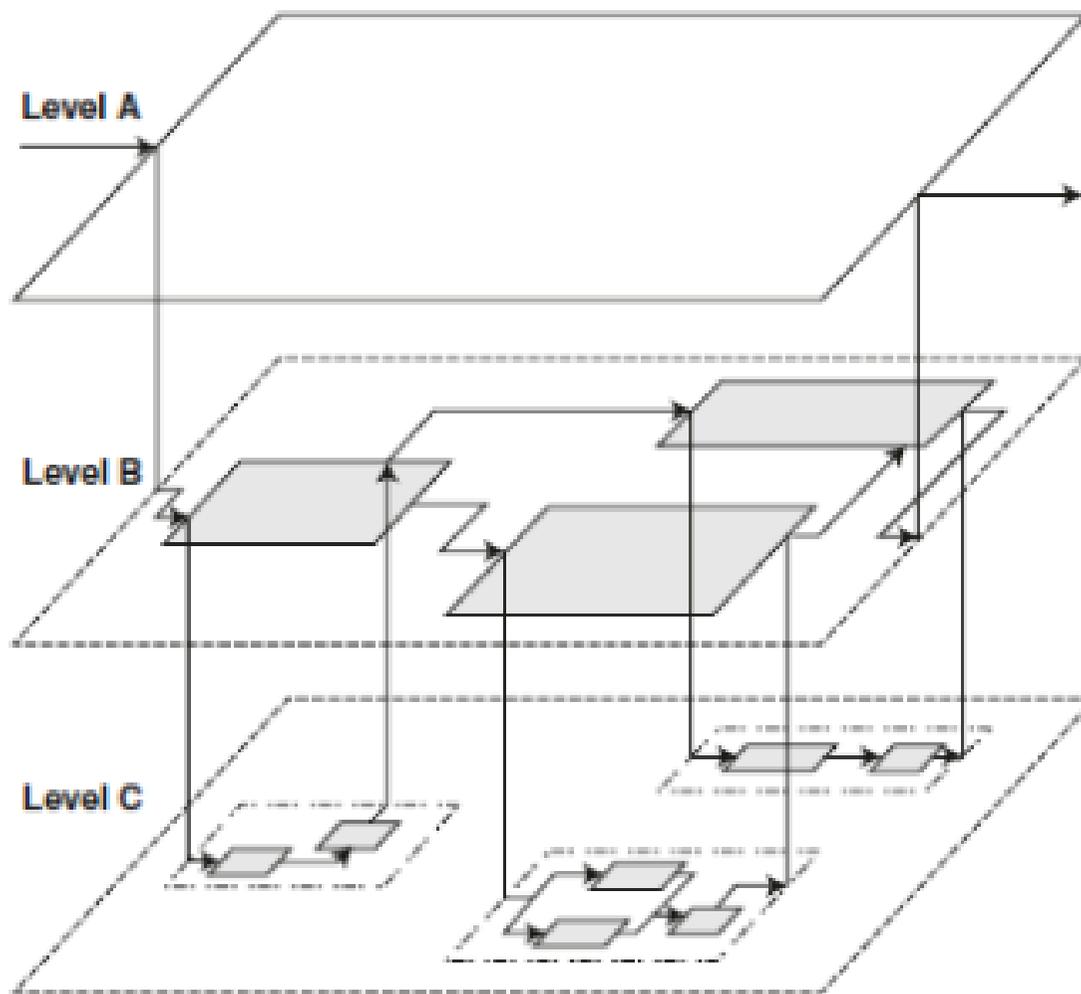


## SIMPLICIDADE

- Quando um sistema é composto de funções puras, elas podem ser reutilizadas ou substituídas.
- Um efeito colateral usual de tal composição é que as interfaces entre blocos se tornam simples e genéricas. Como as interfaces geralmente se tornam simples, é mais fácil substituir uma função pura por uma função relacionada e semelhante.



# DECOMPOSIÇÃO FUNCIONAL



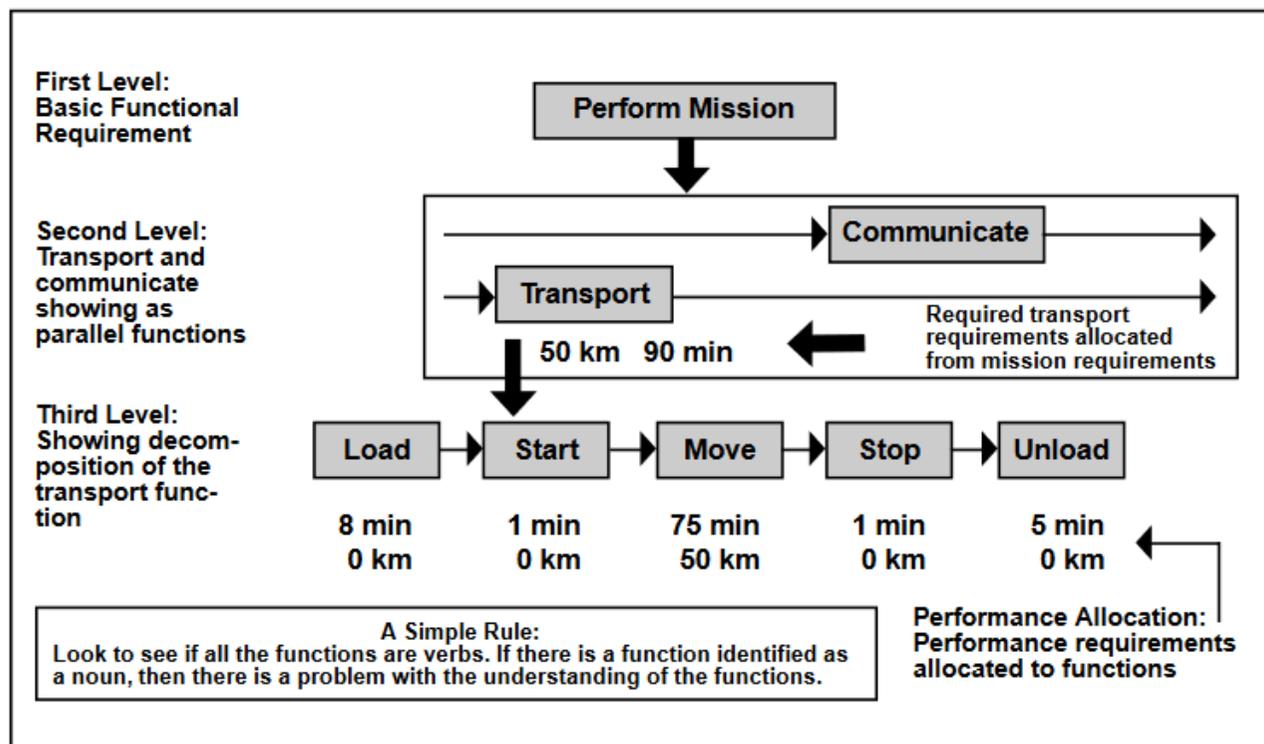


# ARQUITETURA FUNCIONAL

- A arquitetura funcional é (geralmente) uma **decomposição top-down** dos requisitos funcionais e de desempenho do sistema.
- A arquitetura mostra não apenas as funções que devem ser executadas, mas também o sequenciamento lógico das funções e os requisitos de desempenho associados às funções.



# EXEMPLO DE ARQUITETURA FUNCIONAL



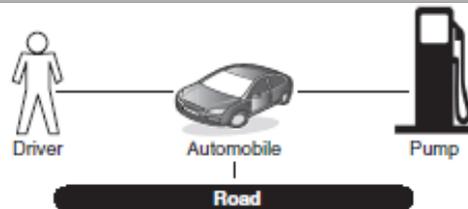
O Corpo de Fuzileiros Navais tem a necessidade de transportar tropas em unidades de nível de esquadrão a uma distância de 50 quilômetros. As tropas devem ser transportadas no prazo de 90 minutos a contar da data de chegada do sistema de transporte. A comunicação constante é necessária durante o transporte de tropas.



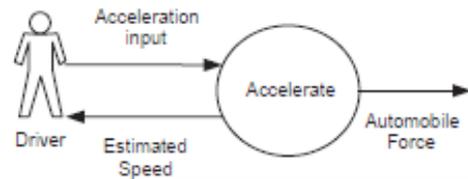
# MODELOS (DIAGRAMAS) CLÁSICOS



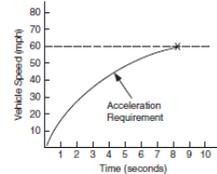
# APLICAÇÃO TÍPICA DO PROCESSO DE ENGENHARIA DE SISTEMAS



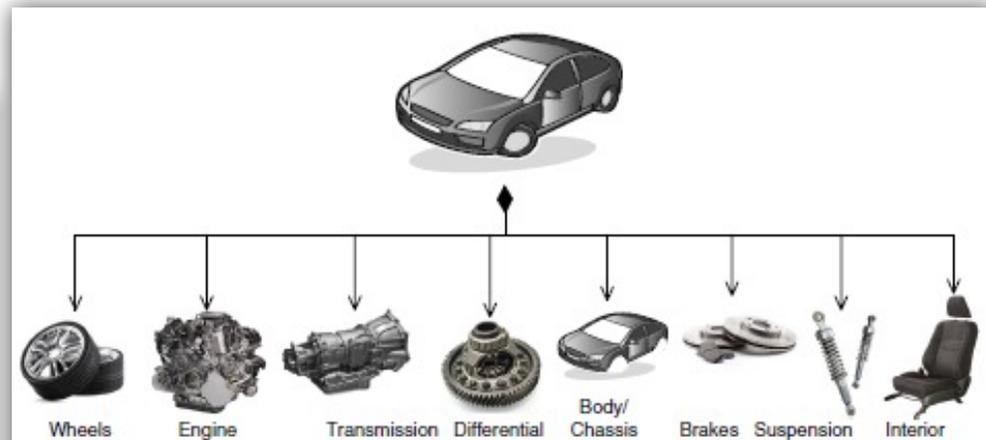
**FIGURE 1.2**  
Defining the system boundary.



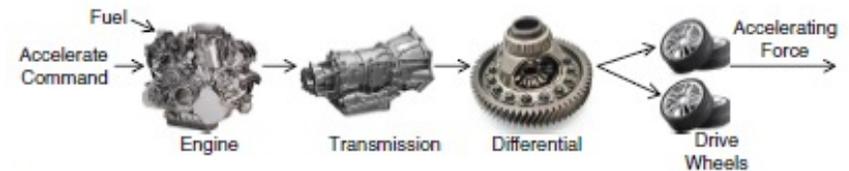
**FIGURE 1.3**  
Specifying the functional requirements.



**FIGURE 1.4**  
Automobile performance requirements.



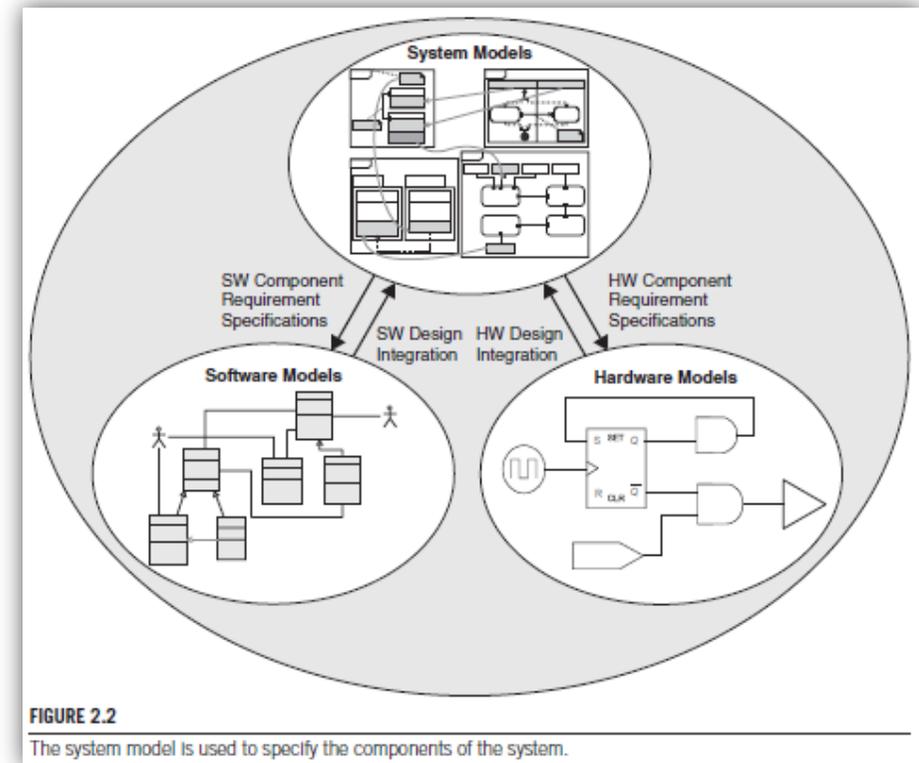
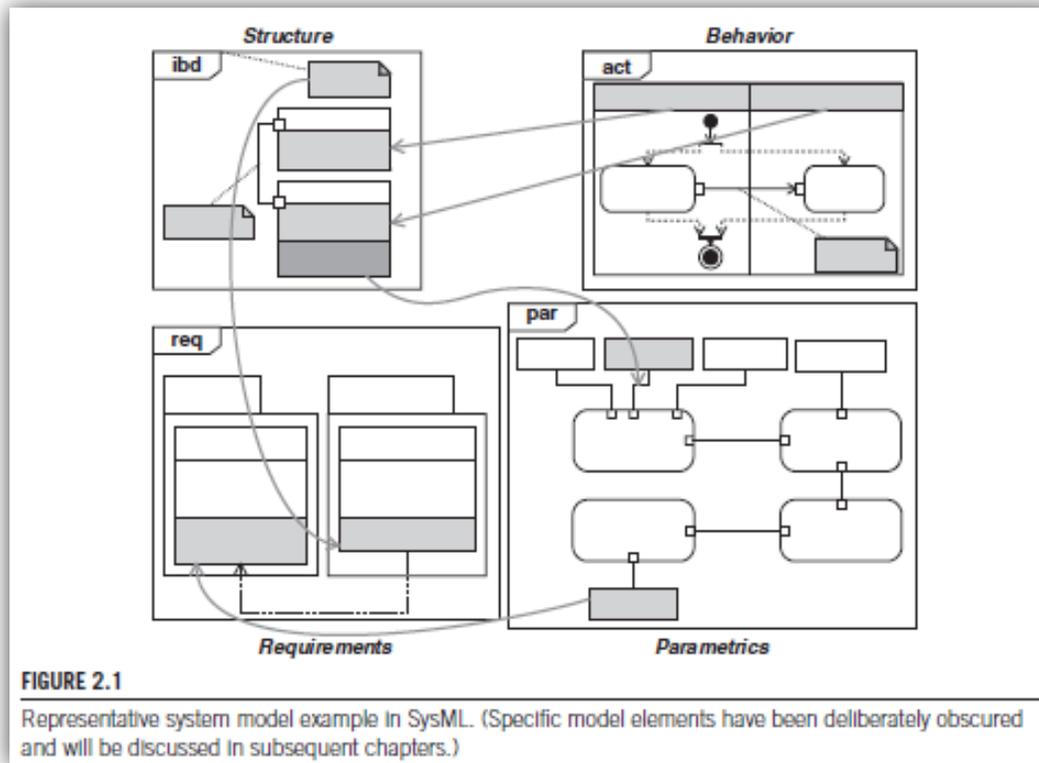
**FIGURE 1.5**  
Automobile system decomposition into its components.



**FIGURE 1.6**  
Interaction among components to achieve the system functional and performance requirements.



# MODELO DE UM SISTEMA





## PROPÓSITO (PORQUÊ – PRAGMÁTICA)

- O uso pretendido para modelar um sistema está associado às **atividades de engenharia de sistemas que o modelo se destina a suportar** em todo o ciclo de vida do sistema e pode incluir os seguintes usos:
  - Caracterizar e avaliar um sistema existente
  - Especificar e projetar um sistema novo ou modificado
  - Avalie o sistema
  - Treinar os usuários sobre como operar ou manter um sistema
  - Suporte à manutenção e/ou diagnóstico do sistema

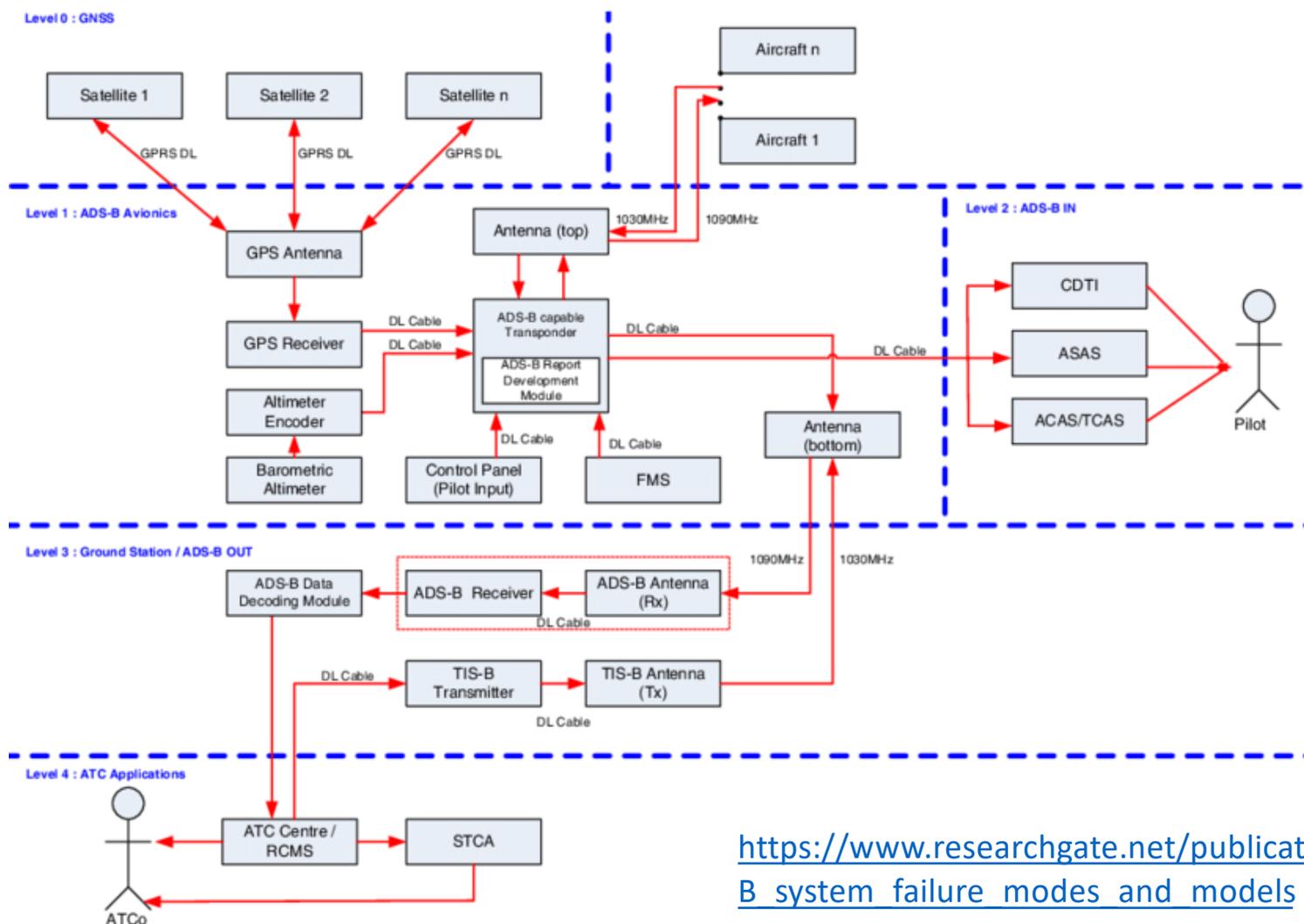


# DIAGRAMAS CLÁSSICOS

- Nos últimos anos, os engenheiros de sistemas **evoluíram para algumas representações gráficas** para apresentar as características funcionais e de fluxo de dados de seu projeto de sistema.
- Os mais comuns são:
  - *Block Diagram,*
  - *(enhanced) Function Flow Block Diagram (FFBD),*
  - *Data Flow Diagram (DFD),*
  - *N2 (N-Squared) Chart,*
  - *IDEF0 Diagram, e*
  - *Behavior Diagram (BD)*



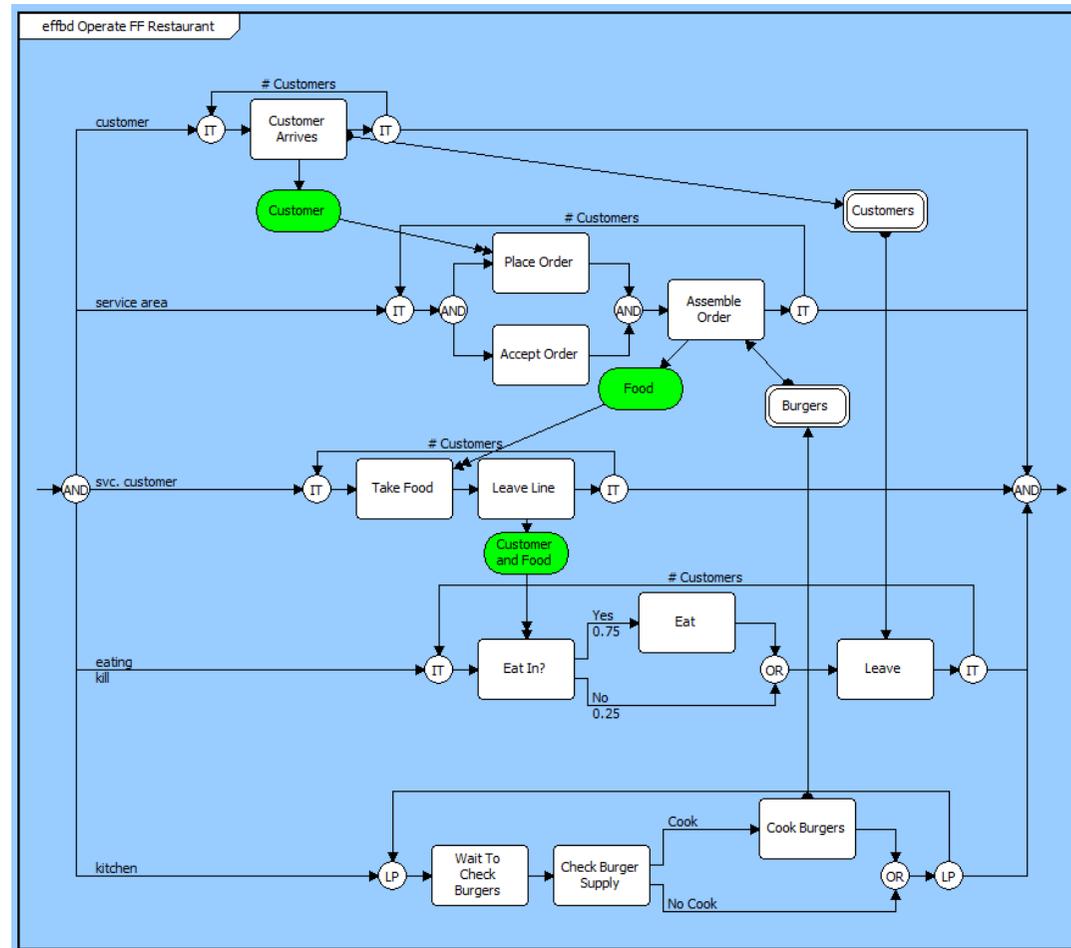
# DIAGRAMAS DE BLOCO



<https://www.researchgate.net/publication/272037785> ADS-B system failure modes and models



# ENHANCED FUNCTIONAL FLOW BLOCK DIAGRAM







# N2 DIAGRAMS

[https://www.nasa.gov/sites/default/files/atoms/files/nasa\\_systems\\_engineering\\_handbook.pdf](https://www.nasa.gov/sites/default/files/atoms/files/nasa_systems_engineering_handbook.pdf)

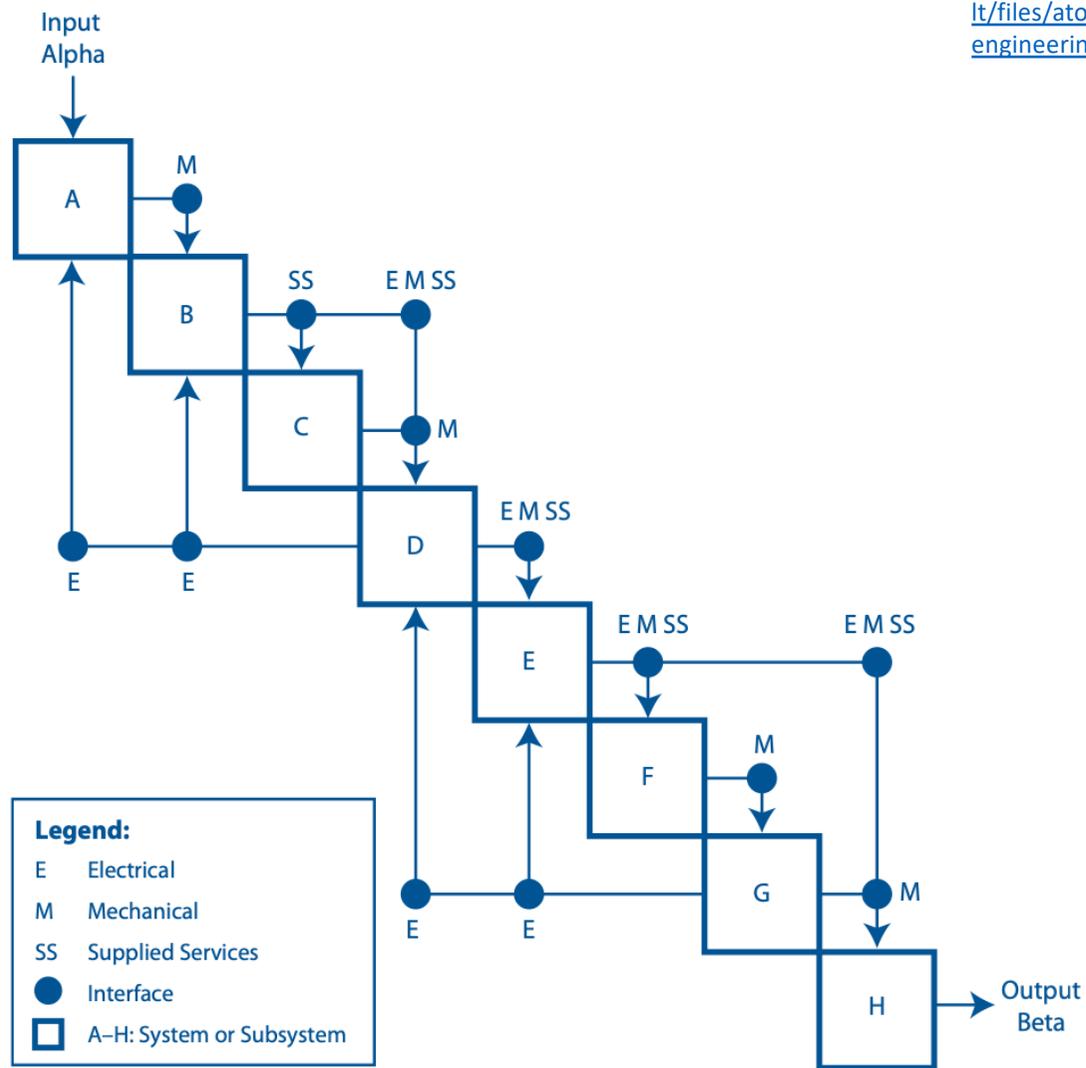
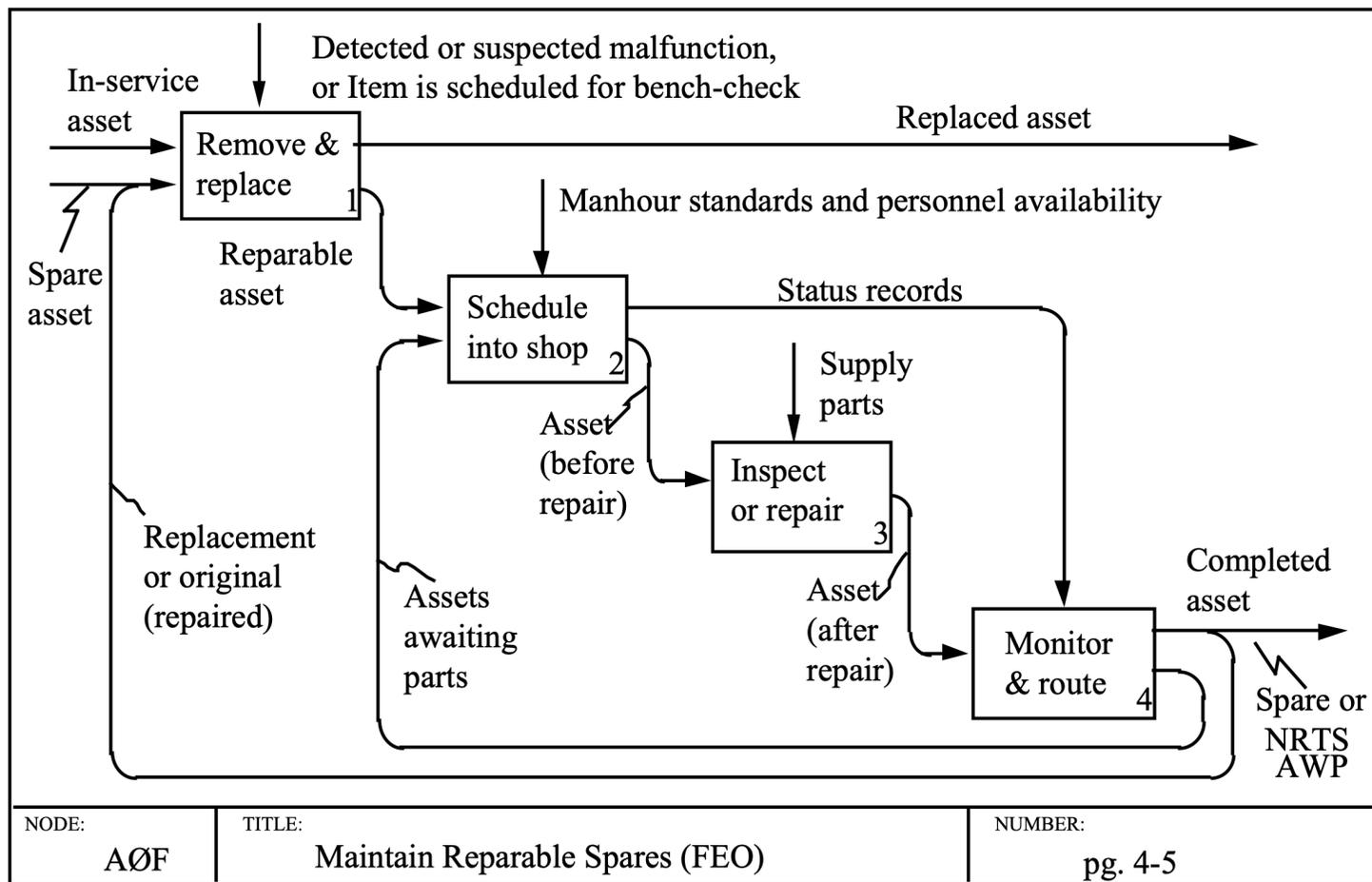


Figure 4.3-4 Example of an N2 diagram



# IDEFO





# BEHAVIOR DIAGRAM

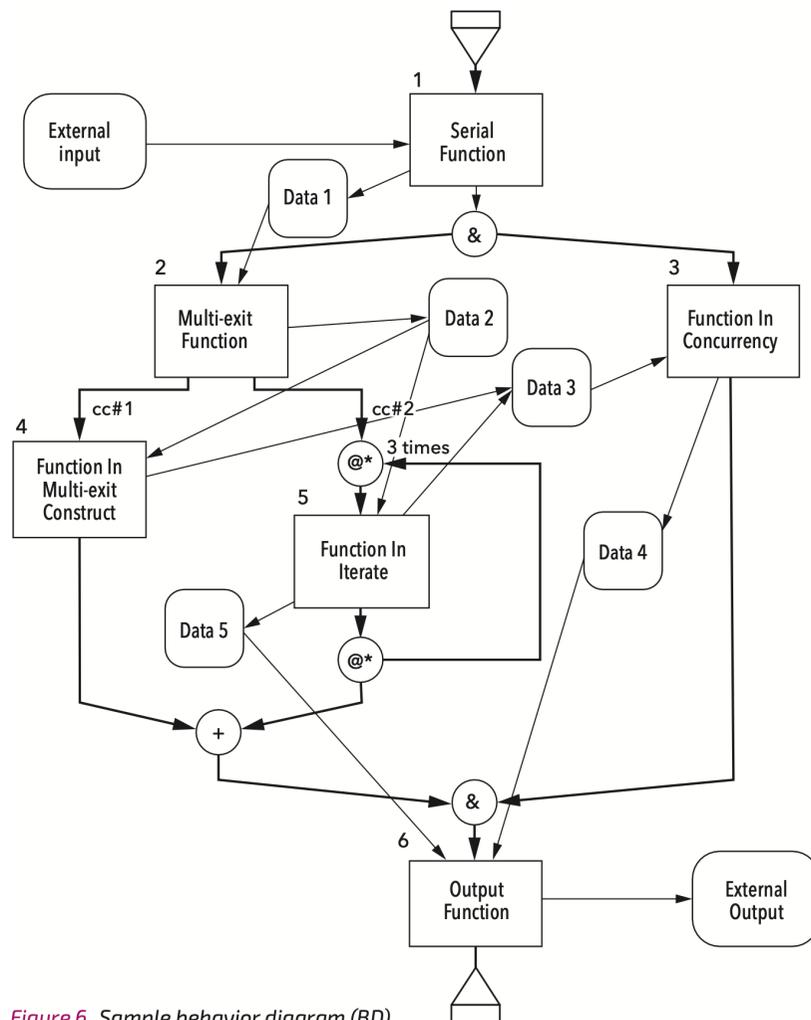


Figure 6. Sample behavior diagram (BD)



# CONSIDERAÇÕES FINAIS



# RESUMO

- “Transformação” de ES para ES(BM)
- Arquitetura
- Diagramas clássicos
  
- Exercícios para a próxima aula:
  - Ex.2.1 – ind – resumir, em 1 slide, cada uma das metodologias de MBSE (cap3): [http://www.omgSysML.org/MBSE Methodology Survey RevB.pdf](http://www.omgSysML.org/MBSE_Methodology_Survey_RevB.pdf)
  - Ex.2.2 – ind – lista de exercício sobre diagramas clássicos da Engenharia de Sistemas (incluindo arquitetura funcional).
  
- Pitch (1.1) ficou para a próxima aula:
  - explicação livre do contexto da situação (período que passa a cena, elementos da cena, onde está acontecendo o problema)