



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

STAKEHOLDERS / CONOPS / DIAGRAMA DE CLASSES



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

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



AULA 04 Ciclo de Vida e CONOPs (Rev2)

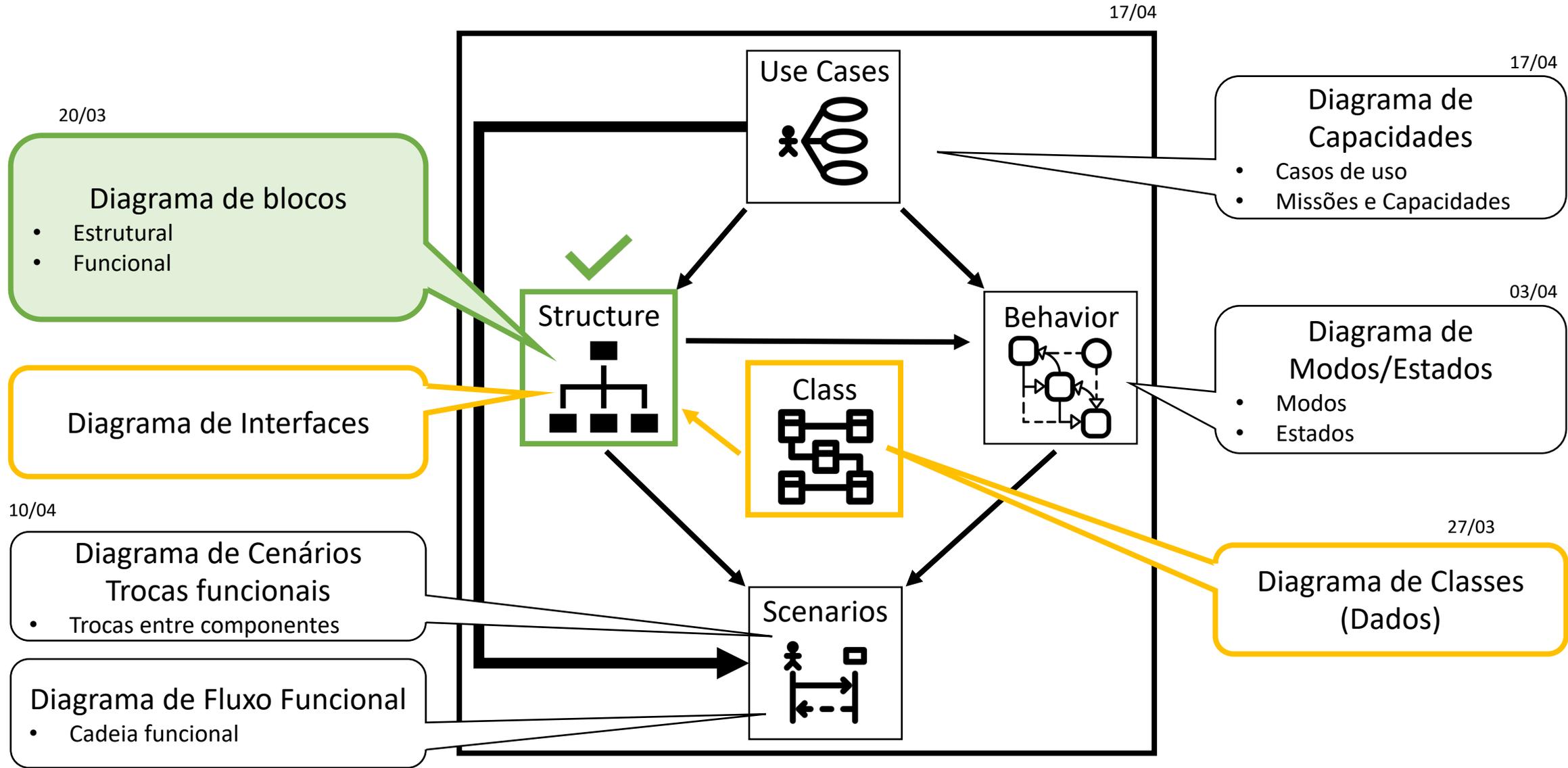
OBJETIVOS 04-01 - Definição de quem são interessados e os influenciados
 04-02 - Apresentar a a construção de CONOPs
 04-03 - Apresentar o diagrama de Classes e Componntes

DATA: 27-Mar

	TÍTULO	#	TÓPICO	ATIVIDADE INDIVIDUAL	ATIVIDADE EM GRUPO	
HORA 01	Stakeholders					
		1	Definição			
		2	Tipos			
		3	Elicitação			
		4	Processo de identificação		Descrever 3 (min) stakeholders da situação levantada: (2) stakeholders favoráveis e (1) stakeholder "inimigo".	
		5				
HORA 02	Ciclo de Vida e CONOPs					
		1	Ciclo de Vida			
		2	Modelos de Ciclo de Vida			
		3	Definição de Conceito			
		4	Formas de Representação de Conceito			
		5	Adiantar necessidades			
HORA 03	Diagrama de Classes e Componentes					
		1		Exercício de fixação	Descrever 4 interfaces harmonizadas entre os grupos, e uma que só existe na era em particular. Cada grupo deve fazer a estrutura de dados da troca funcional entre 1 stakeholder e o Delorean de uma das interfaces identificadas (apresentem entre sí para não repetir)	
		2				
		3				
		4				
		5				
		* Próximas atividades				



ROTEIRO DOS DIAGRAMAS





STAKEHOLDERS

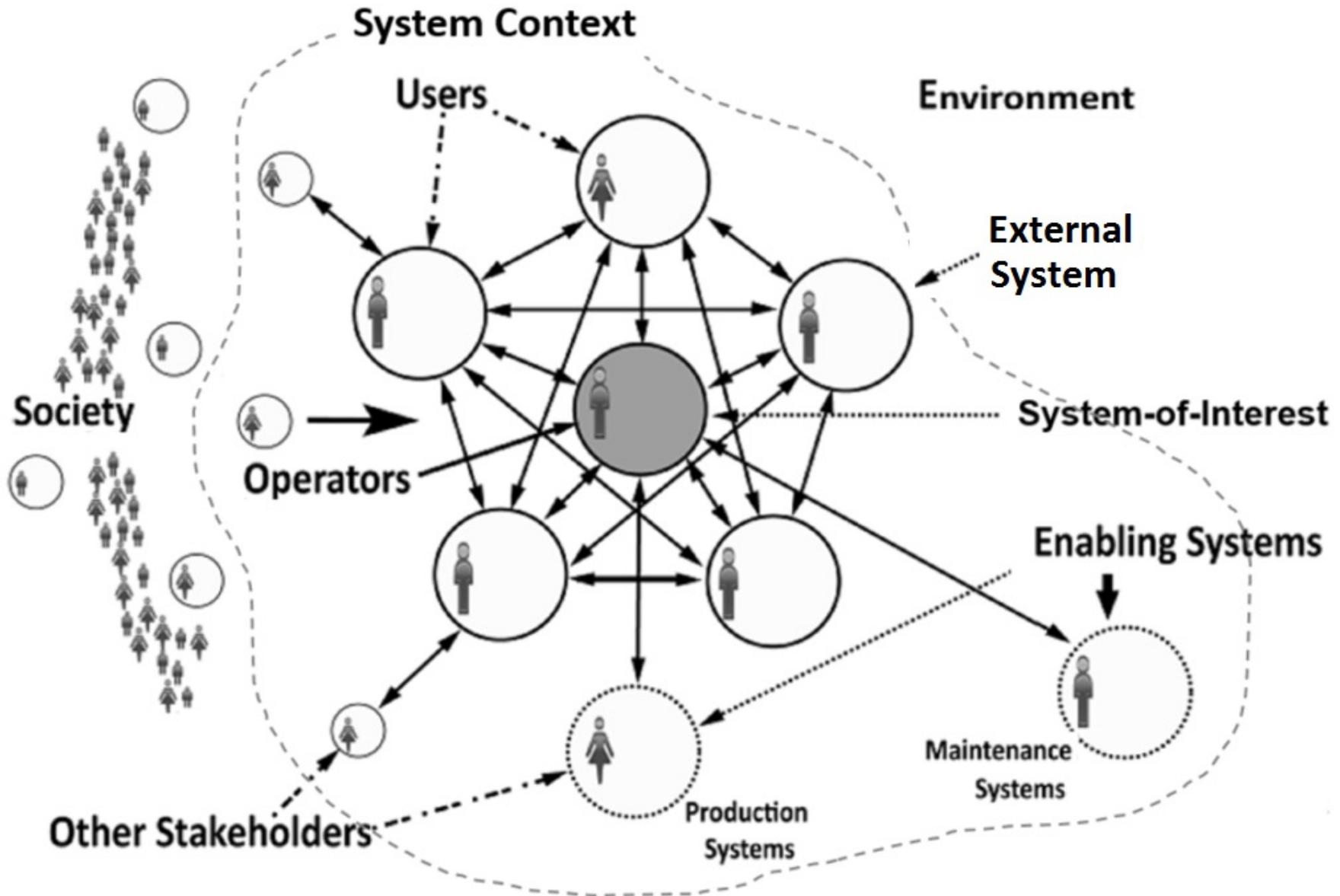


CONCEITOS



TOP LEVEL SYSTEM

STAKEHOLDER DEFINITION





STAKEHOLDERS

- Um stakeholder é qualquer indivíduo, grupo ou organização que possa afetar ou/e ser afetado por um projeto.





PROCESSO DE ESTRUTURAÇÃO

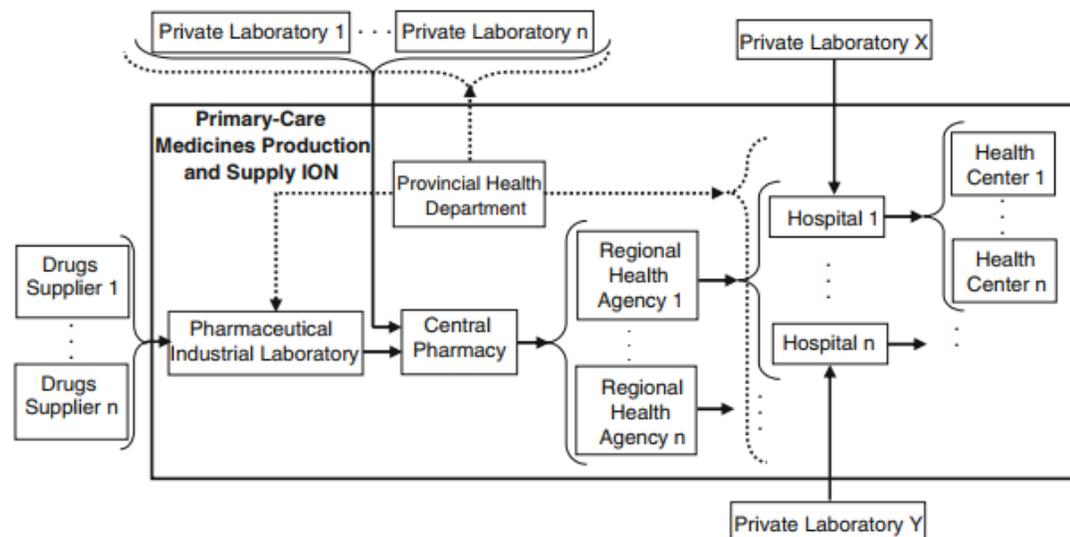


Method for stakeholder identification in interorganizational environments

Luciana C. Ballejos · Jorge M. Montagna

Received: 30 May 2007 / Accepted: 7 August 2008 / Published online: 9 September 2008
© Springer-Verlag London Limited 2008

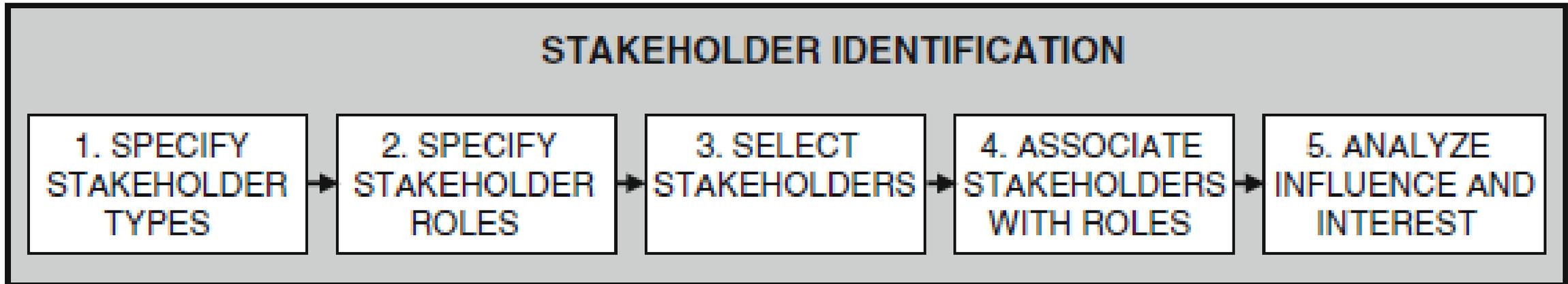
Abstract Stakeholders are the first emerging challenge in any software project. Their identification is a critical task for success. Nevertheless, many authors consider them as a default product of a non-explained identification process. Several aspects must be considered when the project is carried out in environments where multiple organizations





ETAPAS PARA AS PARTES INTERESSADAS IDENTIFICAÇÃO

STAKEHOLDER DEFINITION





1. TIPO

- O conceito de tipo de stakeholder é definido como "a classificação de conjuntos de stakeholders que compartilham as mesmas propriedades e atributos no que diz respeito à dimensão em análise"
- Exemplo:
 - **Funcional** – desempenham uma função na missão,
 - **Localização geográfica** – o posicionamento é a propriedade de dimensão mais importante ,
 - **Conhecimentos/capacidades** – ter um conhecimento sobre o domínio, mas não interagir com a função e
 - **Nível hierárquico** – ter um poder hierárquico



2. PAPÉIS

TABLE 3.2 – Stakeholder’s roles. (BALLEJOS; MONTAGNA, 2008)

Regulators:	They are generally appointed by government or industry to act as regulators of quality, security, costs or other aspects of the system. They generate guidelines and outlines that will affect the system development and/or operation.
Developers:	They are directly involved in IOS development (requirements engineer, analyst, designer, programmer, tester, project manager, etc.).
Beneficiaries:	Those that benefit from the system implementation.
Functional:	They benefit directly from the functions performed by the system and its products or results. Other information systems that interact with the new one can be included in this role, since the functionalities to be implemented would be beneficial to this exchange.
Financial:	They benefit indirectly from the system, obtaining financial rewards.



2. DESCRIÇÃO DOS PAPÉIS

TABLE 3.3 – Stakeholder role name description. (BALLEJOS; MONTAGNA, 2008)

Name:	Stakeholder role name:
Brief description:	Briefly describe the role and what it represents for the project. Generally, a stakeholder playing a particular role represents a group of stakeholders, some aspect of participating organizations, or some other affected business areas.
Responsibilities:	Summarize key responsibilities in relation to the project and the system to be developed. Specify the value the role will provide to the project team. For example, some responsibilities may be monitoring project progress, specifying expenditure levels and approving funds spending, etc.
Participation:	Briefly describe how they will be involved in the project and in which stages they will have greater influence.



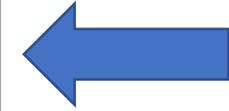
4. RELAÇÃO ENTRE TIPOS E PAPÉIS

DIMENSION			CRITERION	STAKEHOLDER	ROLES														
					BENEFICIARIES				NEGATIVES	RESPONSIBLES	DECISION-MAKERS	REGULATORS	OPERATORS	EXPERTS	CONSULTANTS				
					FUNCTIONAL	FINANCIAL	POLITICAL	SPONSORS											
INTERNAL	ORG	Functional	Physicians organization X		√														
			PIL agent						√										
			...																
		Geog. Locat.	Information systems administrator organis. Y, dependency 1		√									√	√				
			...																
	ION	Knowl/Ability	Information systems administrator organis. Y, dependency 1		√										√	√			
			...																
			Hierar. Level		...														
		Functional	Pharmacist region C		√											√			
			...																
EXTERNAL	Geog. Locat.	RHA coordinator (region B)						√		√									
		...																	
	Knowl/Ability	Pharmacist region C		√															
		...																	
EXTERNAL	Hierar. Level	Health department director										√	√						
		RHA coordinator (region B)											√	√					
	Functional	Private laboratory A		√											√				
		...																	
Geog. Locat.	Private laboratory A		√																
	...																		
Knowl/Ability	Specialist in process redesign															√	√		
	...																		
Hierar. Level	...																		
	...																		

Types



Stks



Roles



5. INTERESSE

STAKEHOLDER DEFINITION

Stakeholder	Interests	Estimated Project Impact	Estimated Priority
<i>Primary</i>	Owner	Achieve targets Liability (avoid at all costs) Increase sales margin	Med + High - Med + 1
	Sponsor	Successfully addresses needs of adjunct customer Appears competent among peers Provides new market to expand ventures	Low + Low - Med + 3
	Team Memebers	New product excitement Demand end-of-year bonus Retain and expand skill level Strike (if basic demands aren't met with new process)	Med + ? Med + High - 2
	Project Manager		
<i>Secondary</i>			



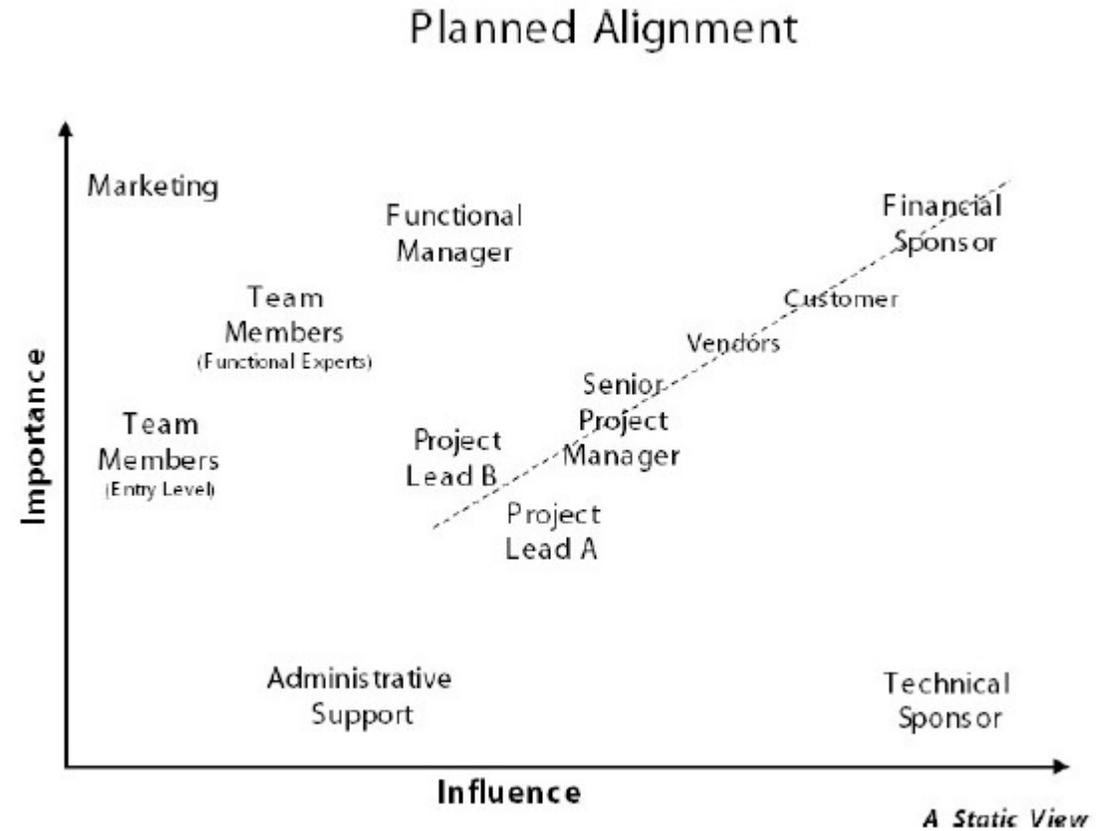
5. PARTICIPAÇÃO E ALINHAMENTO

STAKEHOLDER DEFINITION

Type of Participation

		Inform	Consult	Partnership	Control
Stage in Lifecycle	Initiation <i>Identification</i>	S _A	S _H	S _C	S _I
	Planning	S _K	S _B	S _J	S _D
	Execution <i>Implementation</i>	S _M	S _F	S _N	S _G
	Controlling <i>Monitoring and Evaluation</i>			S _L	S _O
	Closing		S _D		S _A

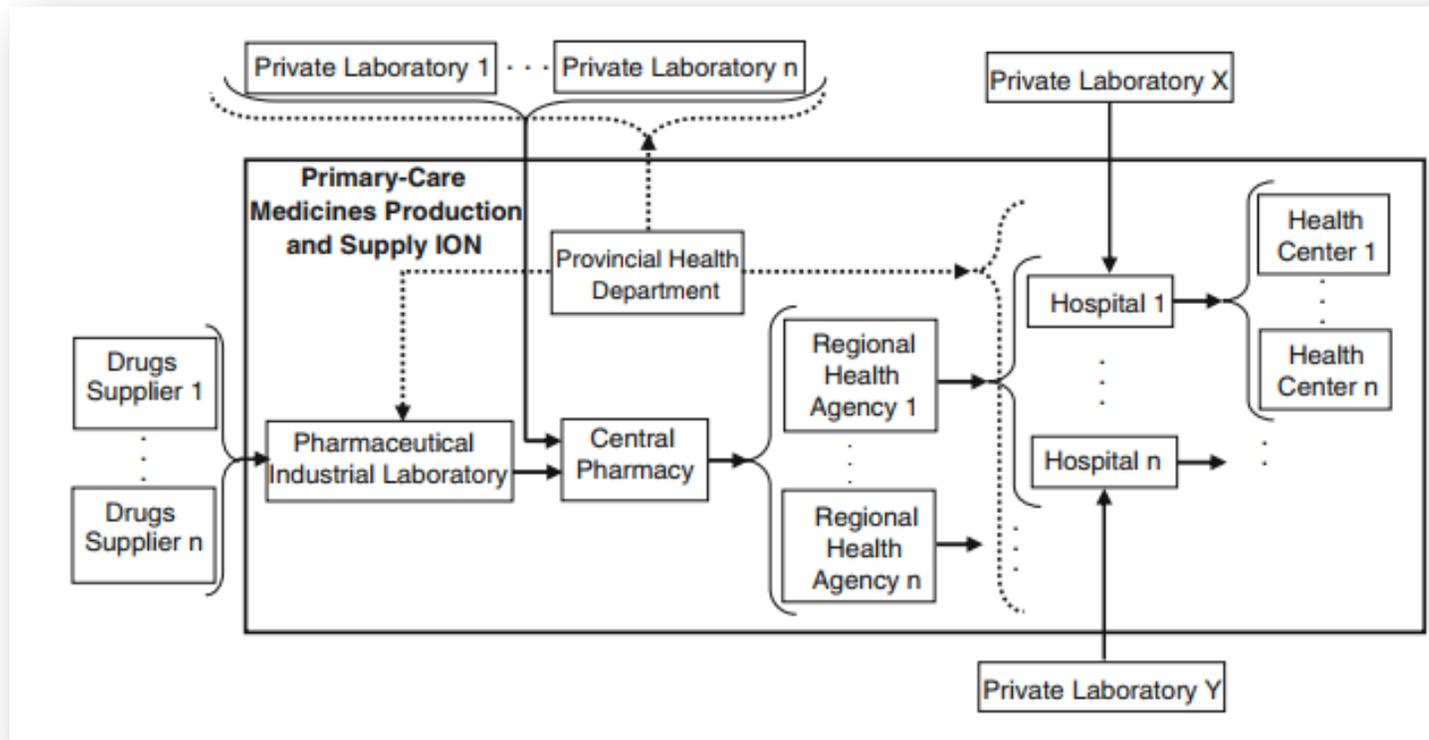
S_x = Stakeholder X





NO FIM... REDE DE STAKEHOLDERS

- Descrevem as necessidades, desejos, expectativas e restrições dos stakeholders
 - Define a rede de valor a ser criado para as partes interessadas





CICLO DE VIDA



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**.





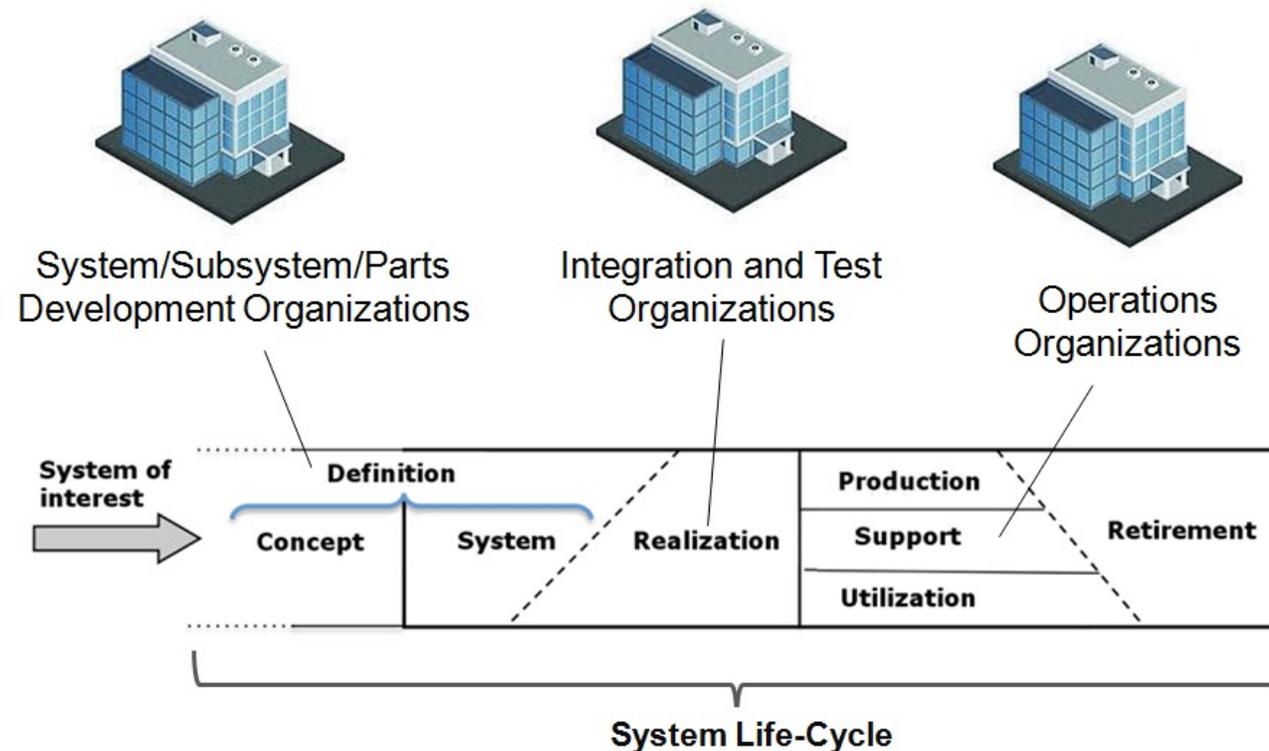
CICLO DE VIDA DO SISTEMA

- **ECSS-E-ST-10-06C:** A **time interval** between the **conceptual exploration** of the product introduction to its **withdrawal from service**.
- **ISO/IEC/IEEE 15288:** A system progresses through its life cycle as **the result of actions**, performed and **managed by people** in organizations, using processes for execution of these actions.



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.



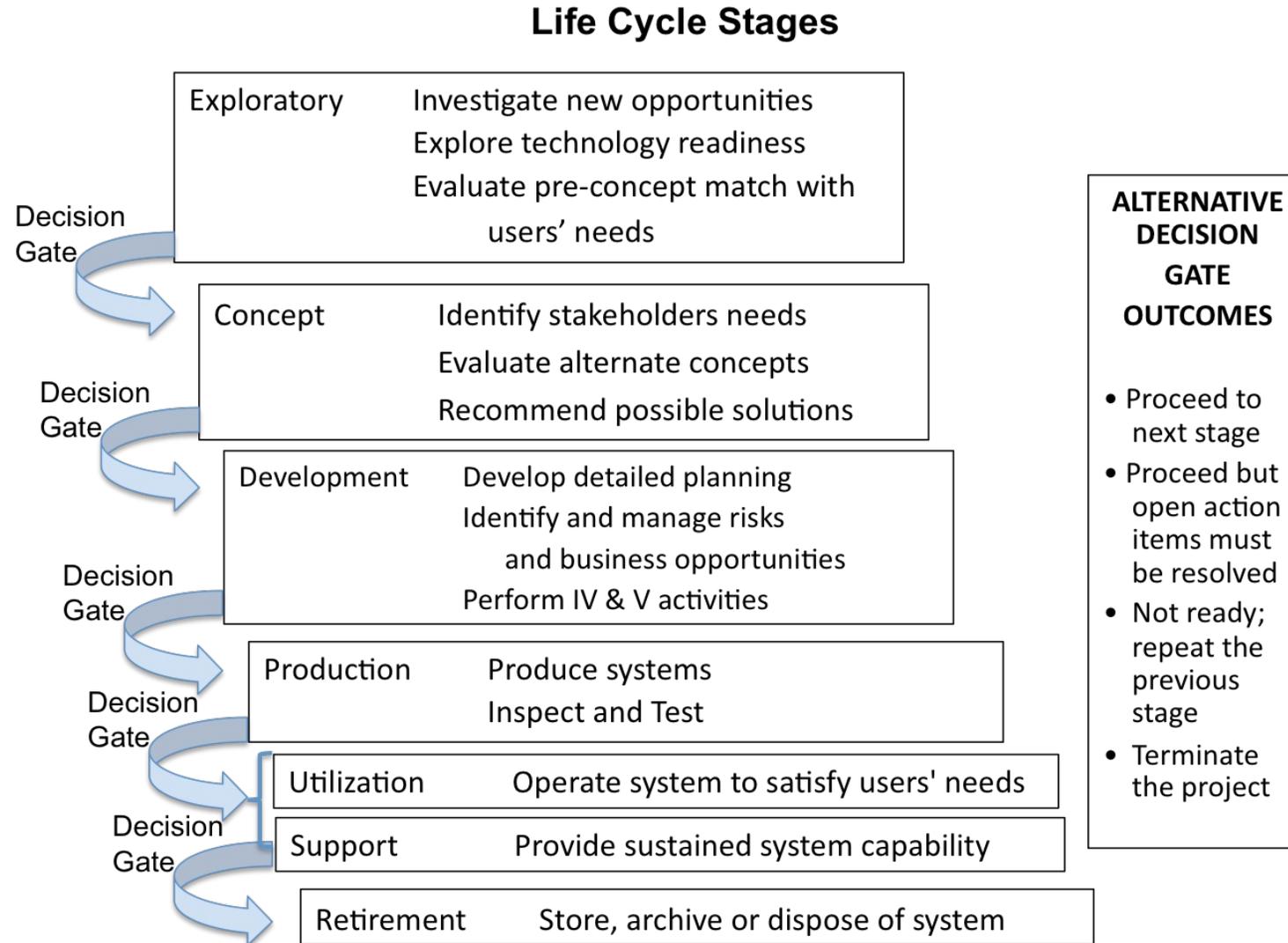


TIPOS DE CICLO DE VIDA

[https://sebokwiki.org/wiki/Systems Lifecycle Approaches](https://sebokwiki.org/wiki/Systems_Lifecycle_Approaches)



CASCATA (WATERFALL)





Generic life cycle (ISO/IEC/IEEE 15288:2015)

Concept stage	Development stage	Production stage	Utilization stage	Retirement stage
			Support stage	

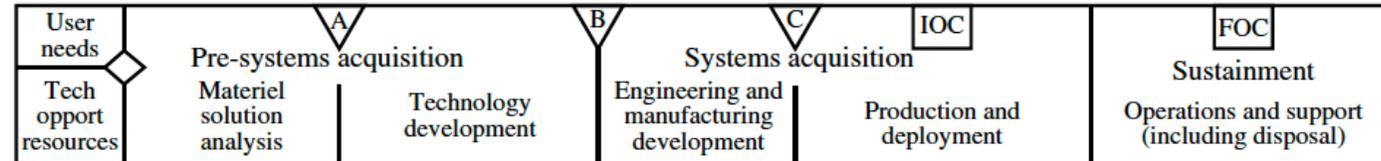
Typical high-tech commercial systems integrator

Study period				Implementation period			Operations period		
User requirements definition phase	Concept definition phase	System specification phase	Acq prep phase	Source select. phase	Development phase	Verification phase	Deployment phase	Operations and maintenance phase	Deactivation phase

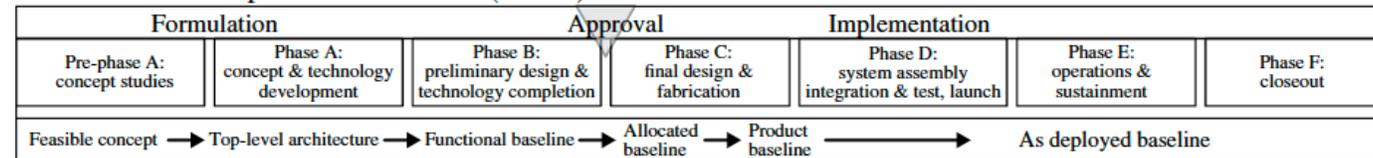
Typical high-tech commercial manufacturer

Study period			Implementation period			Operations period		
Product requirements phase	Product definition phase	Product development phase	Engr. model phase	Internal test phase	External test phase	Full-scale production phase	Manufacturing, sales, and support phase	Deactivation phase

US Department of Defense (DoD)



National Aeronautics and Space Administration (NASA)



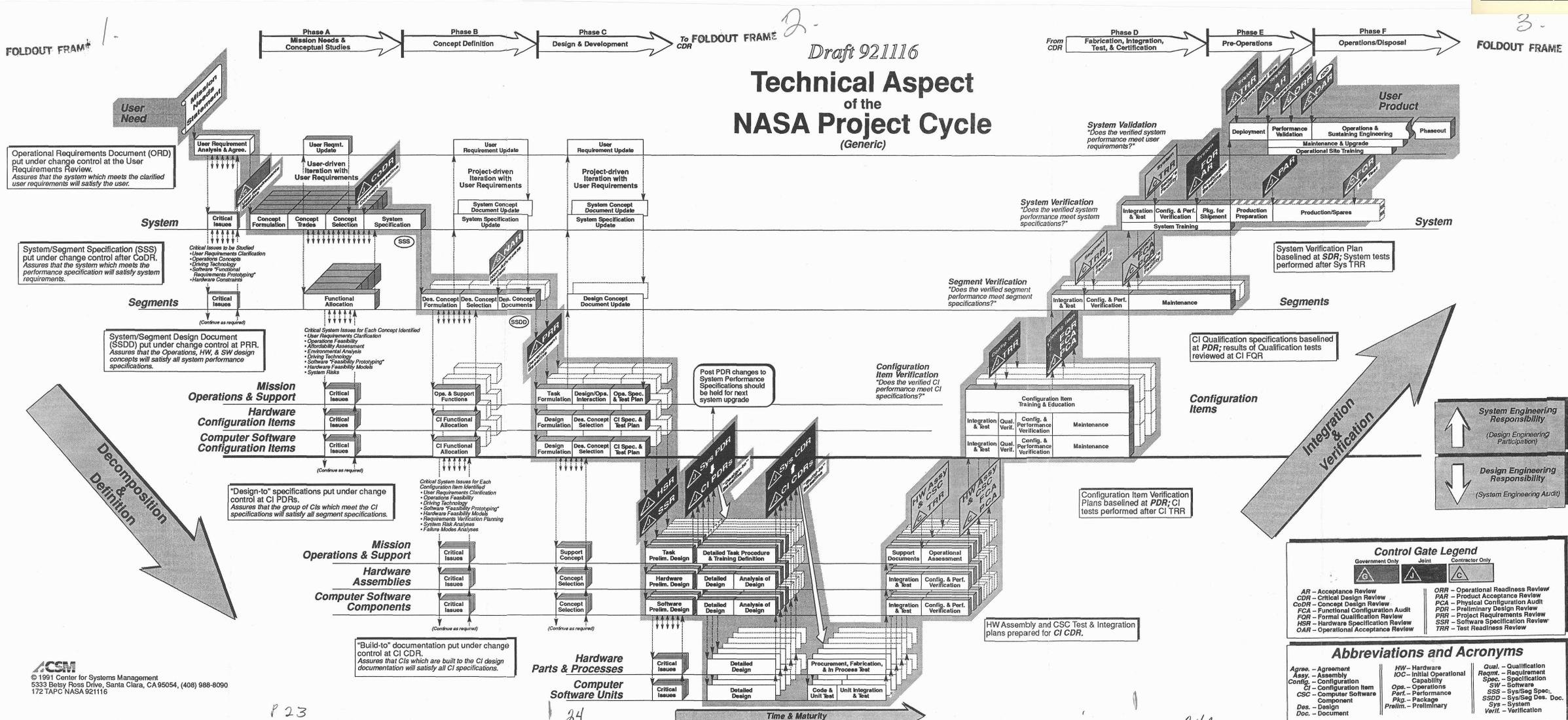
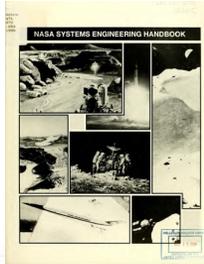
US Department of Energy (DoE)

Project planning period			Project execution			Mission	
Pre-project	Preconceptual planning	Conceptual design	Preliminary design	Final design	Construction	Acceptance	Operations





(CLÁSSICO) VEE MODEL



Technical Aspect of the NASA Project Cycle (Generic)

Draft 921116

Integration & Verification

Decomposition & Definition

CSM
© 1991 Center for Systems Management
5333 Dutey Mesa Drive, Santa Clara, CA 95054, (408) 988-8090
172 TAPC NASA 921116

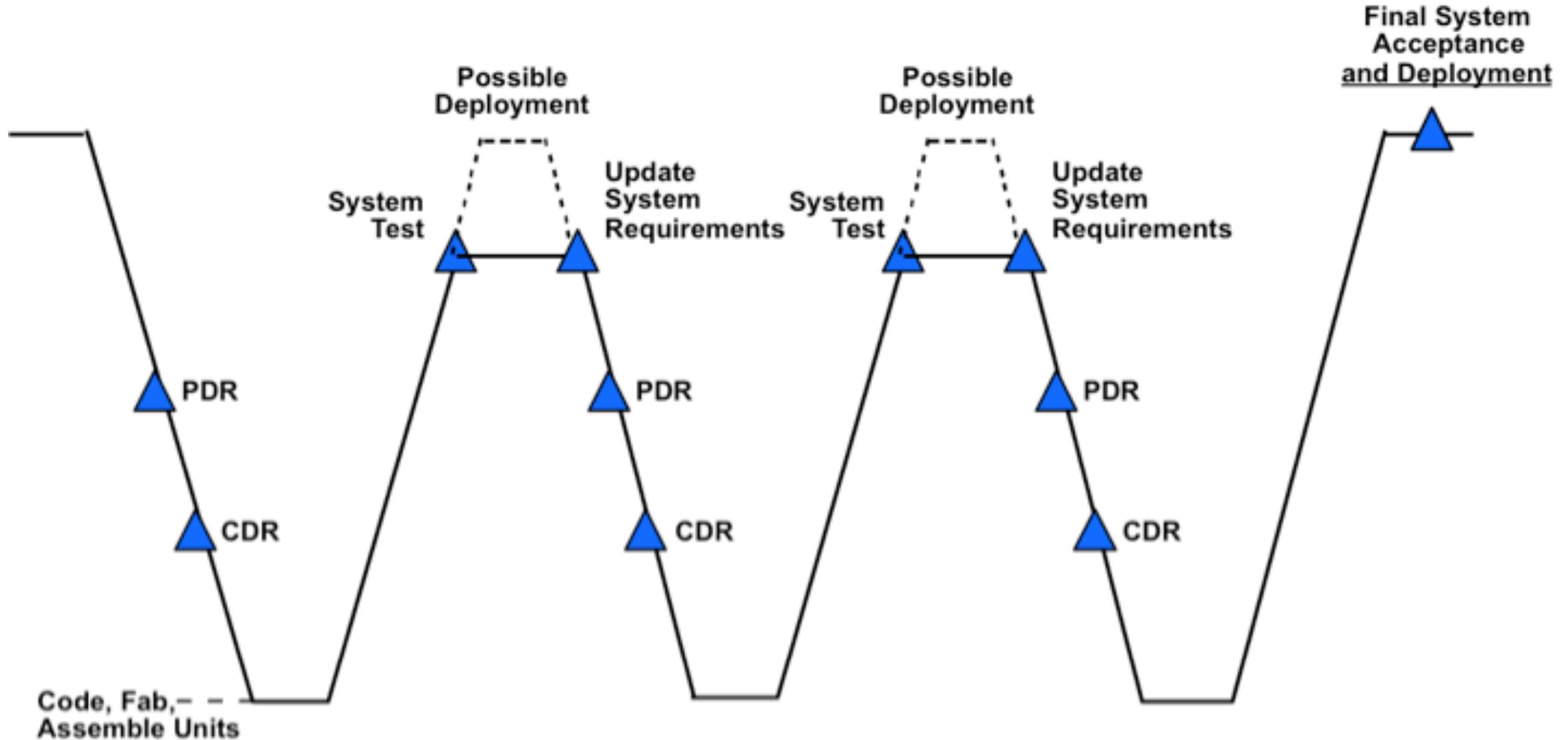
P 23

24

24A

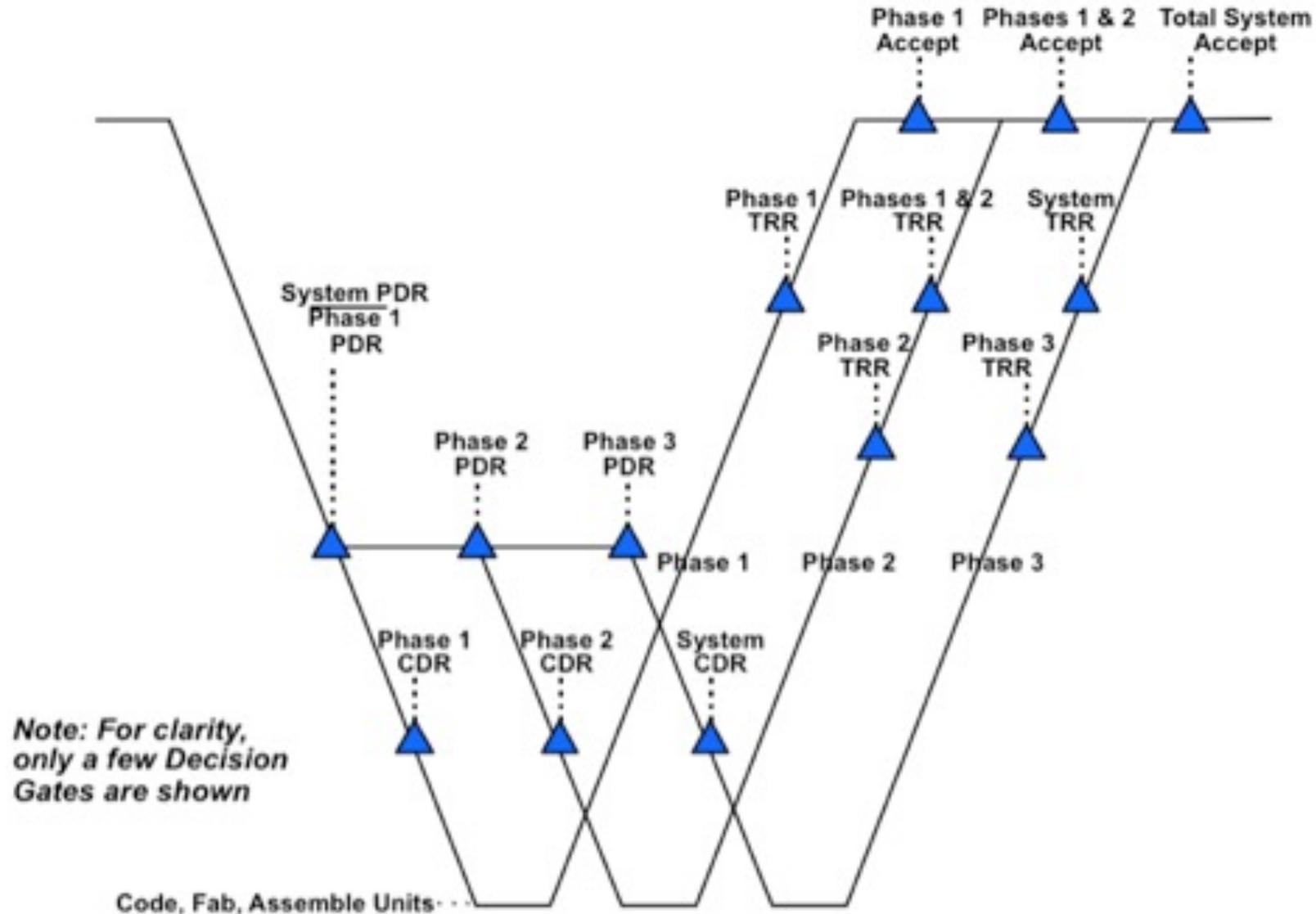


PROCESSO EVOLUTIVO



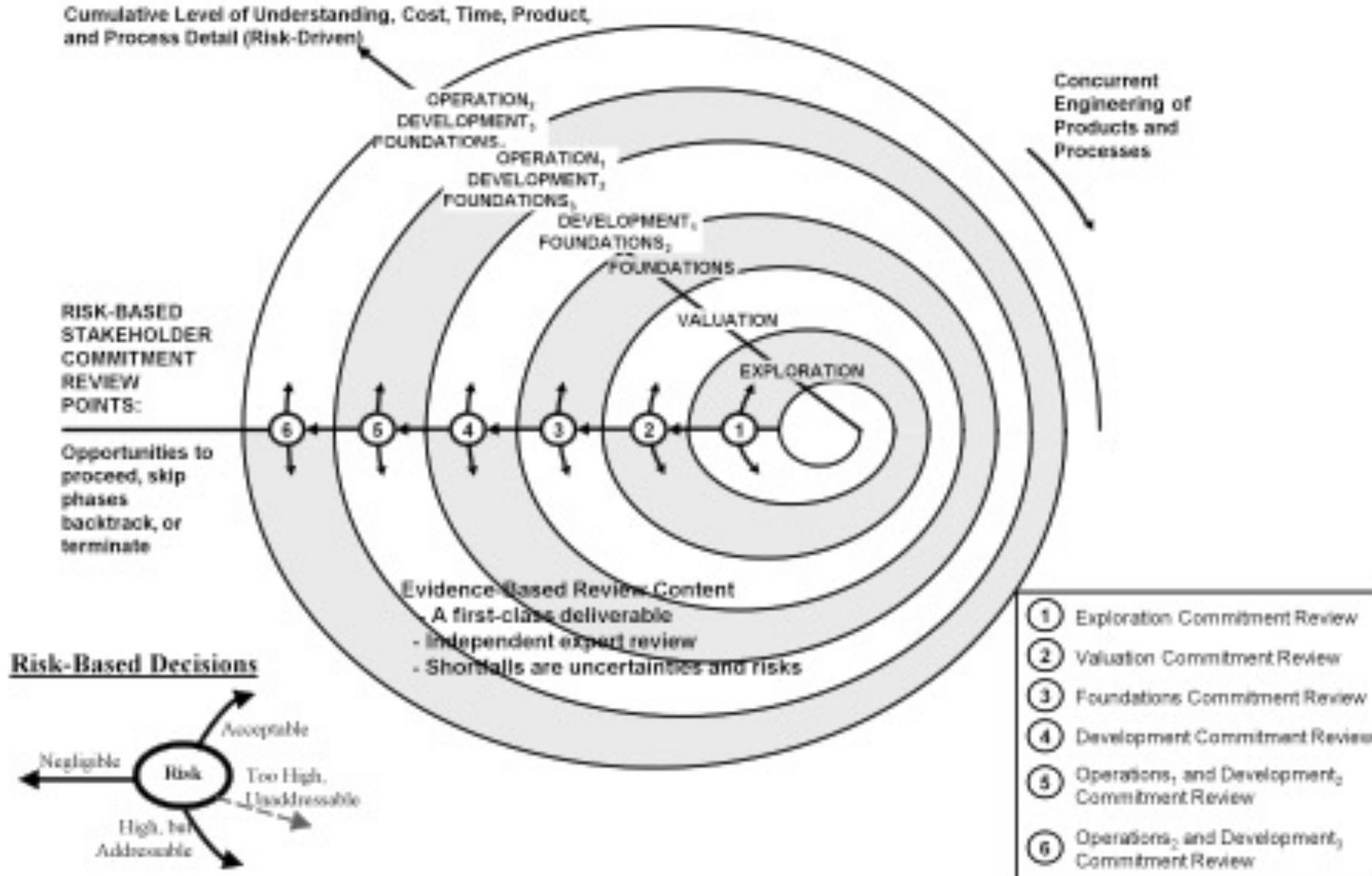


DESENVOLVIMENTO INCREMENTAL COM MÚLTIPLAS ENTREGAS





MODELO ESPIRAL





SCRUM

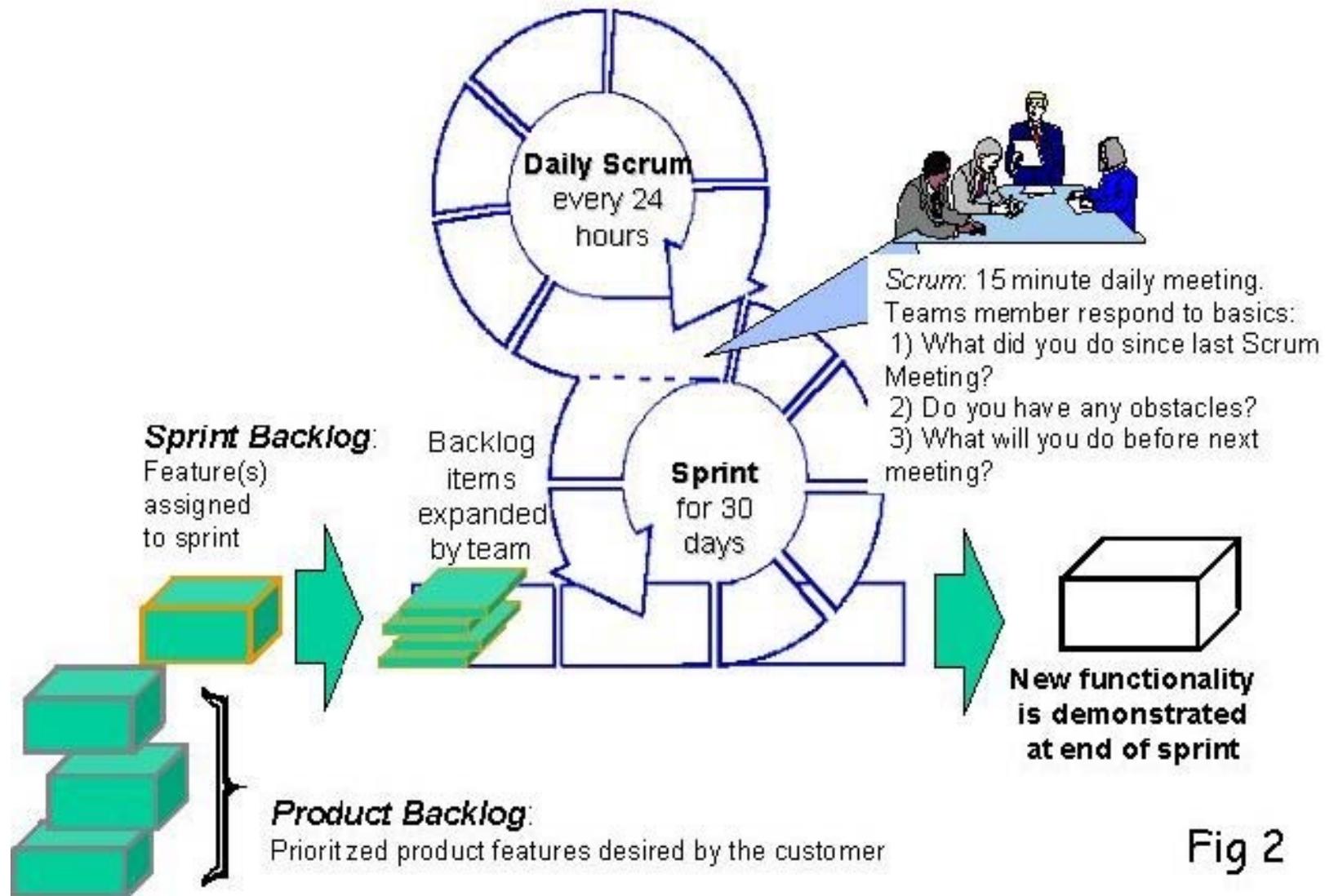
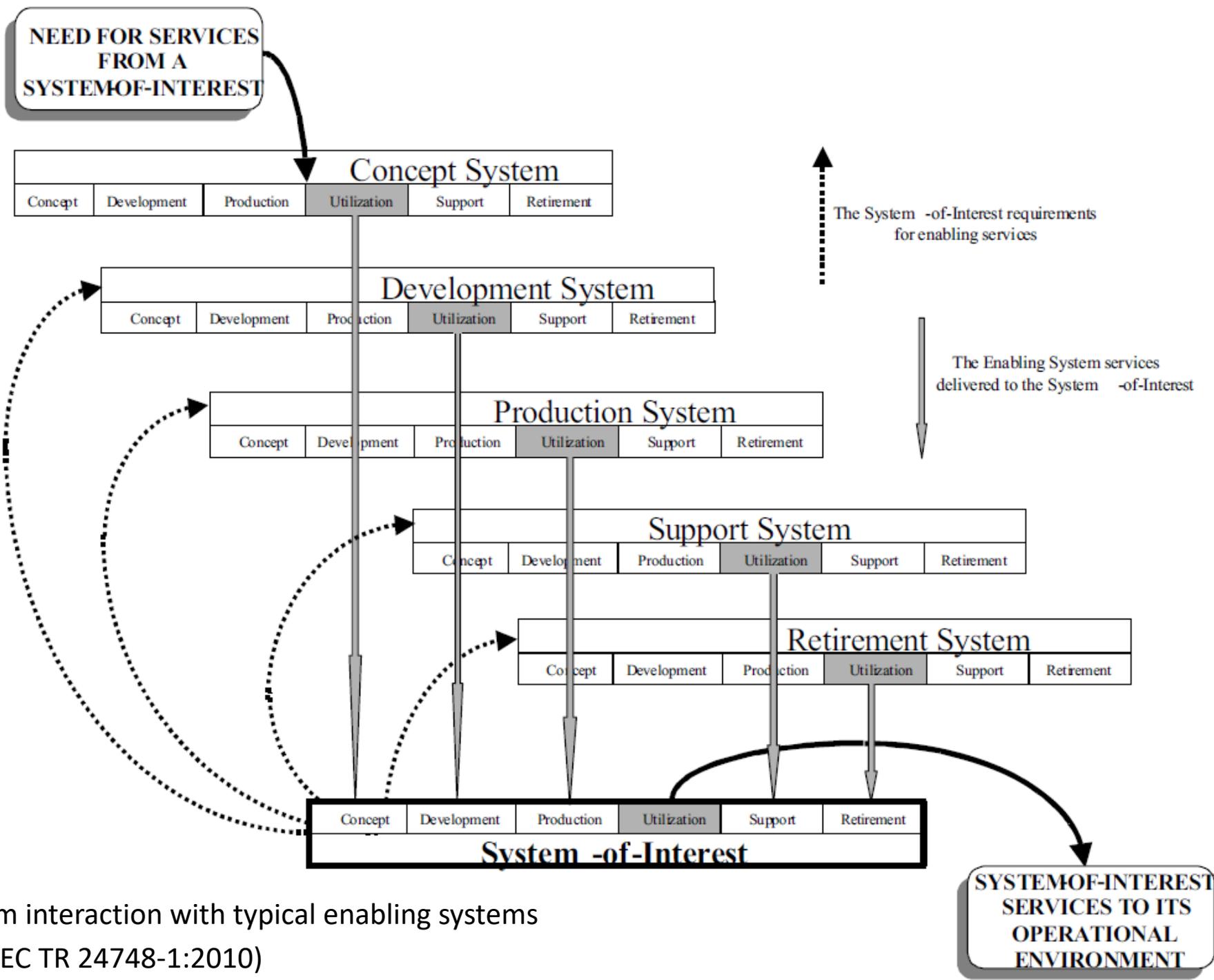


Fig 2

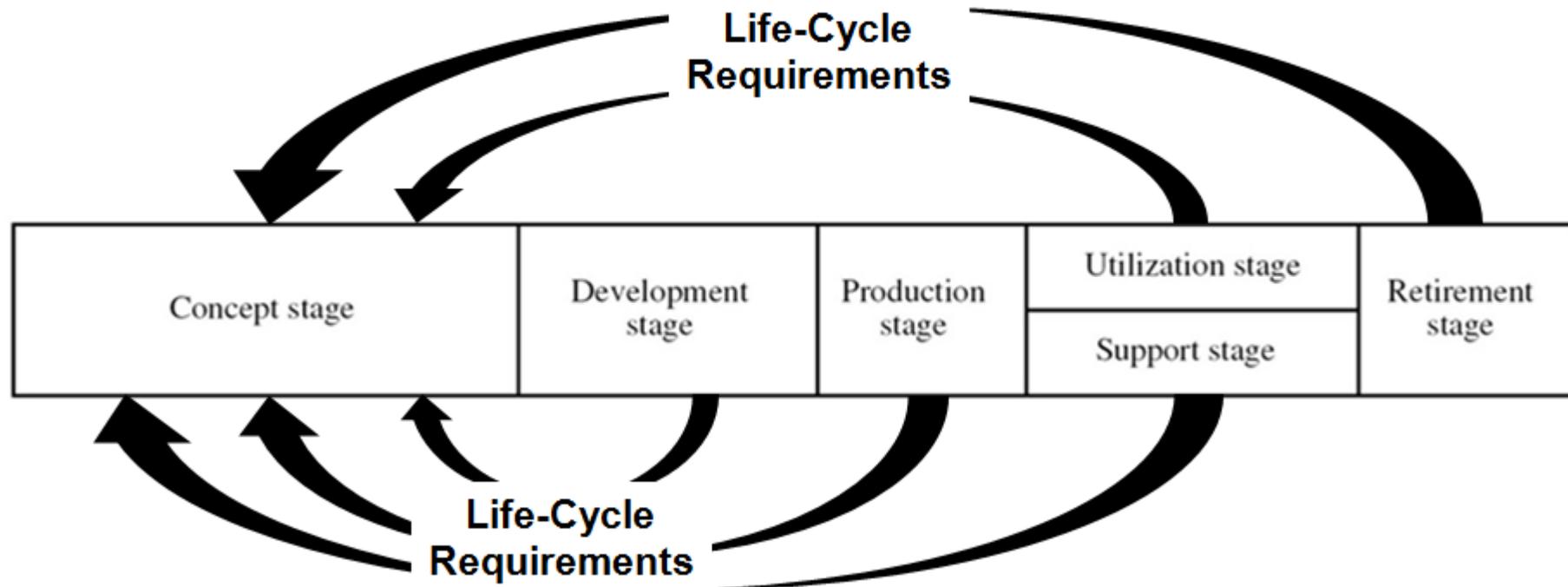


- System interaction with typical enabling systems
- (ISO/IEC TR 24748-1:2010)



REQUISITOS DO CICLO DE VIDA

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



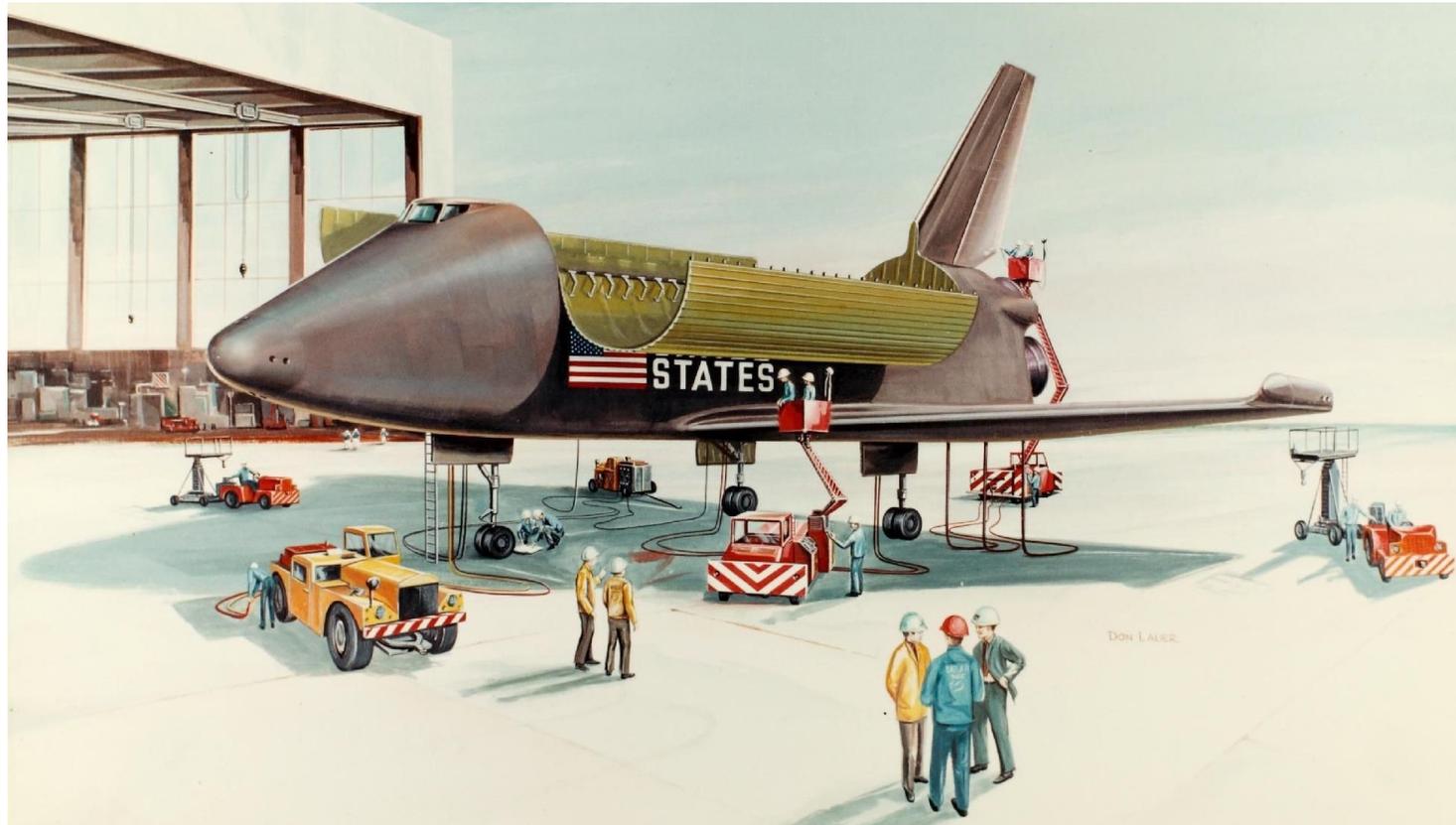


CONCEITOS BÁSICOS DE CONCEITO DE OPERAÇÕES (CONOPS)



CONCEITO

- No início da atividade de desenvolvimento do sistema, um sistema é de natureza conceitual.

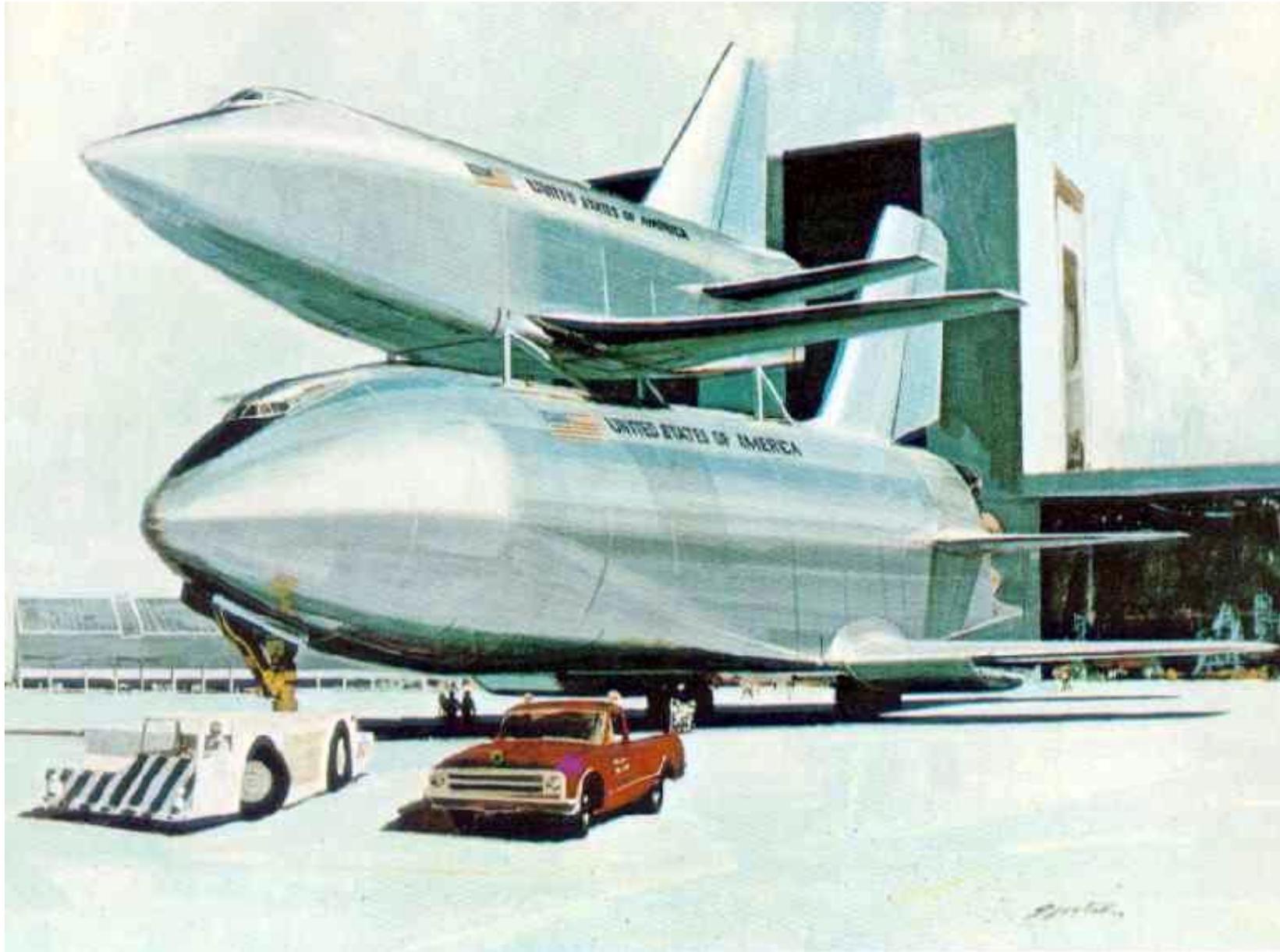




CONCEITO

- À medida que o esforço de desenvolvimento continua, o sistema se torna hardware, software, materiais, pessoal, instalações e processos.



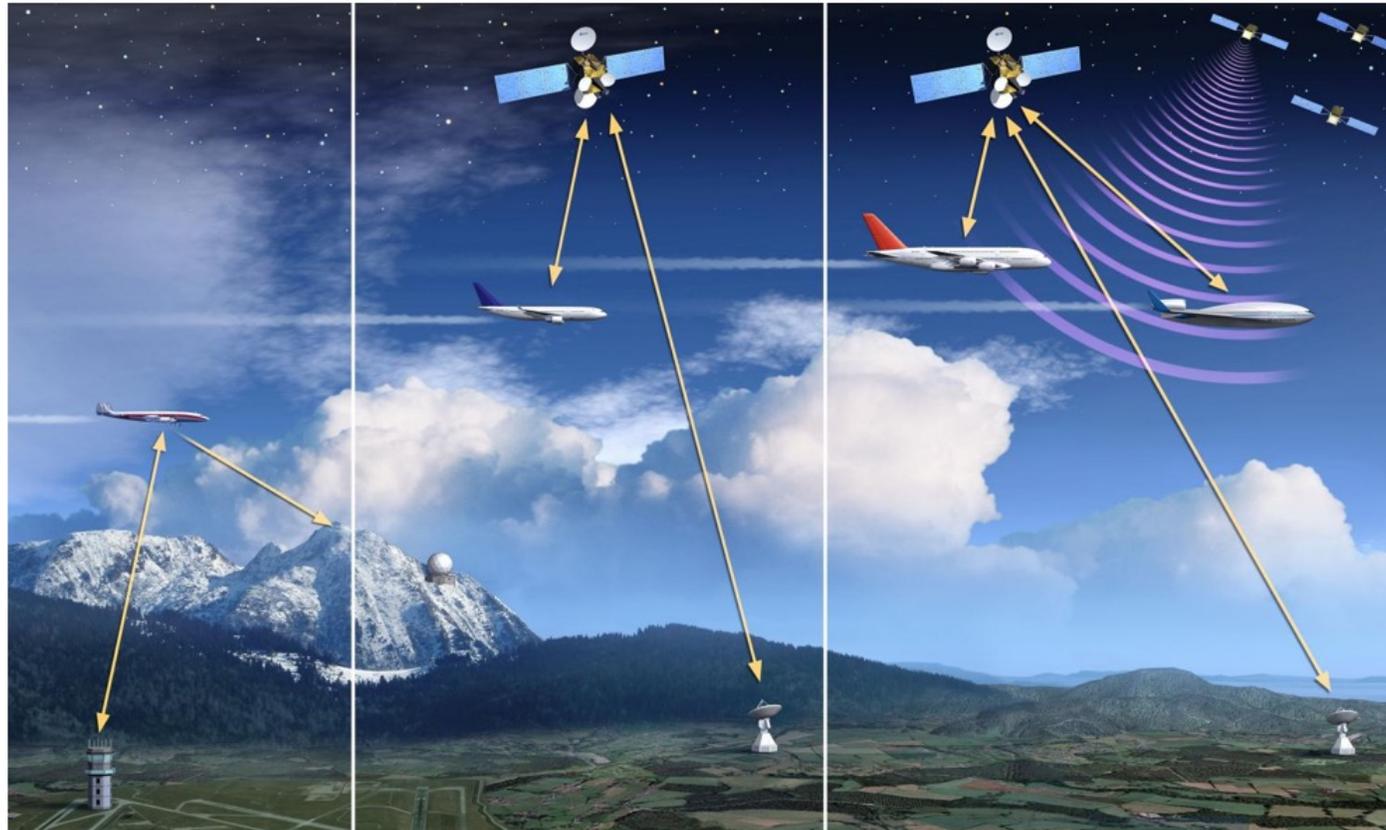






CONCEITO DE OPERAÇÕES -> CONOPS

- Descreve as características de um sistema que está sendo proposto, do ponto de vista de seus operador.





POR QUE O CONOPS É IMPORTANTE?

- Direciona o **desenvolvimento**
 - Mantém o **contexto das necessidades** em linguagem cotidiana e informal
 - Pensar nos ConOps e nos casos de uso **revela requisitos e funções** que, de outra forma, poderiam ser negligenciados
- Coloca todos na mesma página sobre **o que é o projeto e o que ele fará**
- Identifica as **interfaces do usuário** com antecedência
- Identifica as **principais necessidades** do stakeholders para definir, projetar e implementar o produto final
- Fornece orientação para **o desenvolvimento da documentação de definição do Sistema.**



DOCUMENTO DO CONOPS

IEEE Std 1362™-1998 (R2007)

(Incorporates
IEEE Std 1362a-1998)

IEEE Guide for Information Technology—System Definition— Concept of Operations (ConOps) Document

Sponsor
**Software Engineering Standards Committee
of the
IEEE Computer Society**

Approved 19 March 1998
Reaffirmed 5 December 2007

IEEE-SA Standards Board

Abstract: The format and contents of a concept of operations (ConOps) document are described. A ConOps is a user-oriented document that describes system characteristics for a proposed system from the users' viewpoint. The ConOps document is used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements (for example, training, facilities, staffing, and maintenance). It is used to describe the user organization(s), mission(s), and organizational objectives from an integrated systems point of view.

Keywords: buver, characteristics, concept of operation, concepts of operations document, ConOps,

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1998 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 31 December 1998. Printed in the United States of America.

Print: ISBN 0-7381-0185-2 SH94615
PDF: ISBN 0-7381-1407-3 SS94615

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

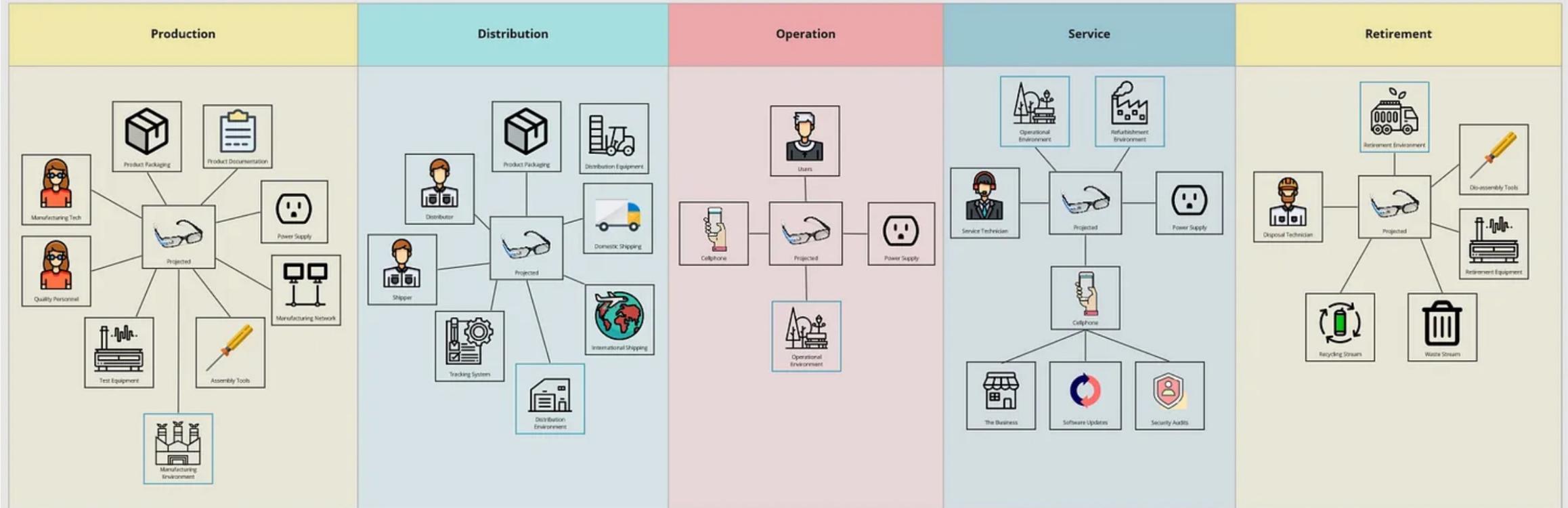
Title page
Revision chart
Preface
Table of contents
List of figures
List of tables
1. Scope
1.1 Identification
1.2 Document overview
1.3 System overview
2. Referenced documents
3. Current system or situation
3.1 Background, objectives, and scope
3.2 Operational policies and constraints
3.3 Description of the current system or situation
3.4 Modes of operation for the current system or situation
3.5 User classes and other involved personnel
3.6 Support environment
4. Justification for and nature of changes
4.1 Justification of changes
4.2 Description of desired changes
4.3 Priorities among changes
4.4 Changes considered but not included
5. Concepts for the proposed system
5.1 Background, objectives, and scope
5.2 Operational policies and constraints
5.3 Description of the proposed system
5.4 Modes of operation
5.5 User classes and other involved personnel
5.6 Support environment
6. Operational scenarios
7. Summary of impacts
7.1 Operational impacts
7.2 Organizational impacts
7.3 Impacts during development
8. Analysis of the proposed system
8.1 Summary of improvements
8.2 Disadvantages and limitations
8.3 Alternatives and trade-offs considered
9. Notes
Appendices
Glossary



EXEMPLOS



Projected Lifecycles



<https://systemthink.substack.com/p/how-to-make-a-conops>

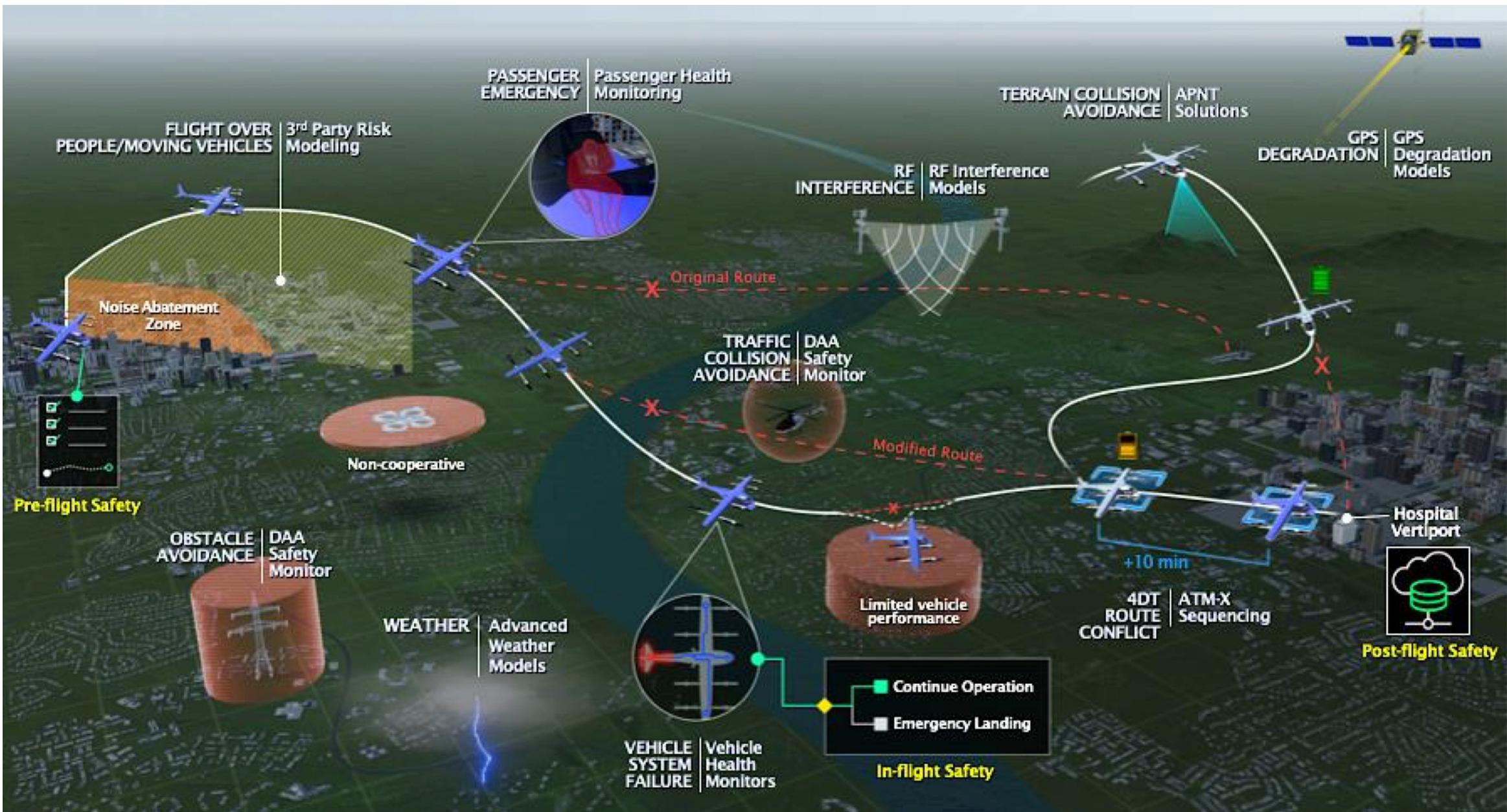


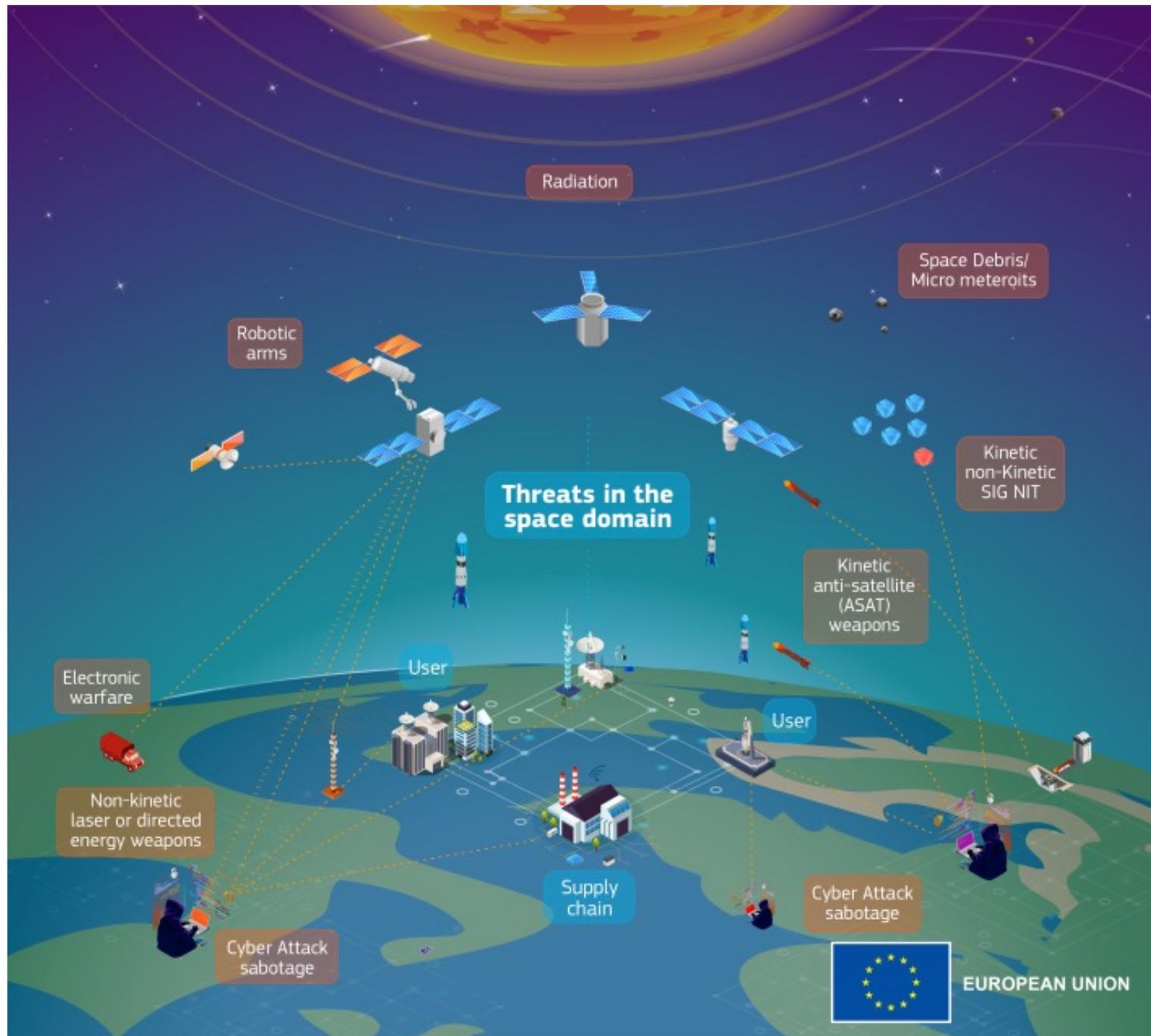
Fig. 1 IASMS Operational View.



FireSat Operational Concept



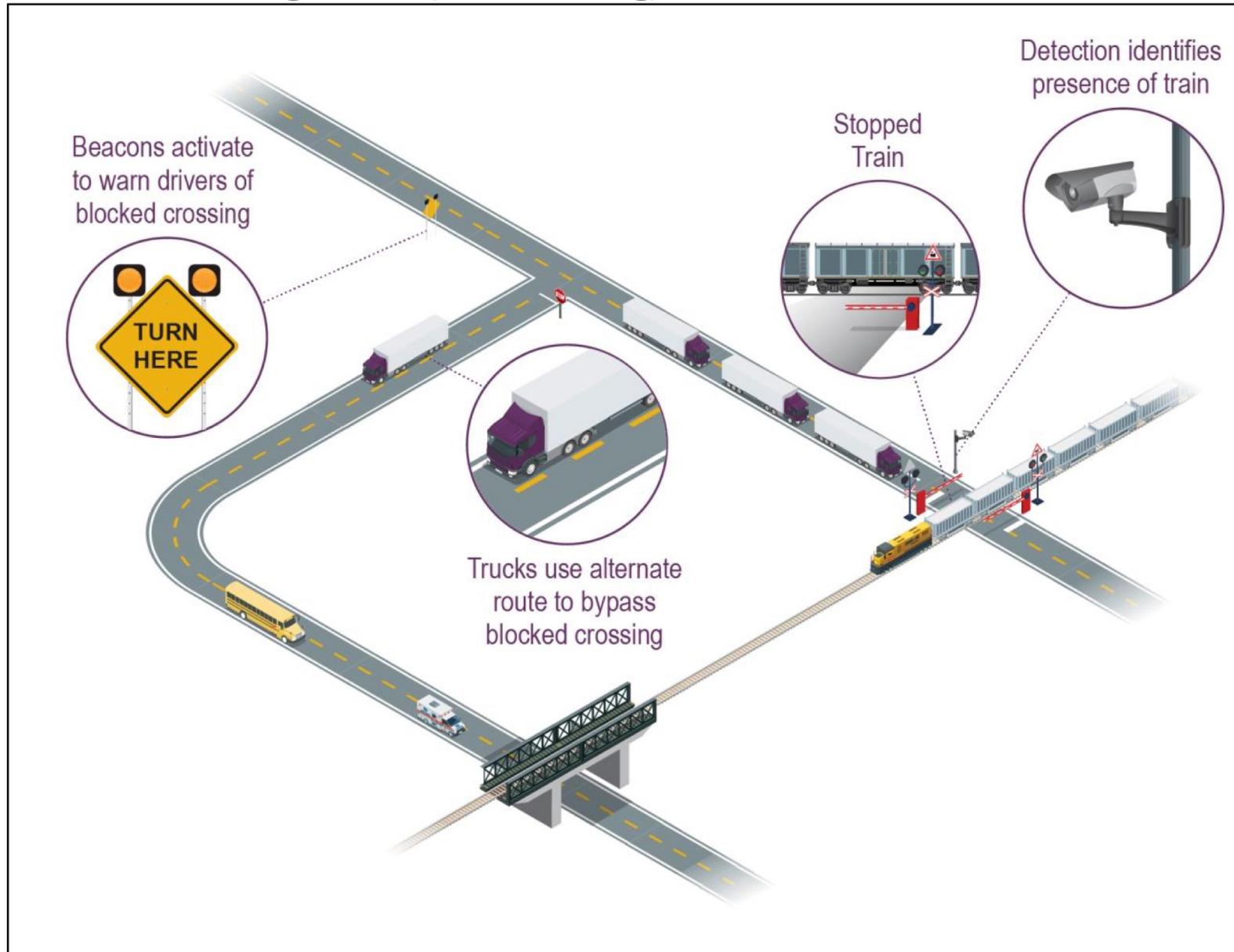
http://johnadamdsis.d.weebly.com/uploads/3/8/1/7/38172527/lessonn_6_unit_8.pdf



<https://spacewatch.global/2023/03/eu-space-strategy-for-security-and-defense-announced-by-the-european-commission/>



Exhibit 27: Illustrative Example of Blocked Rail Crossing Traffic Management System Strategy



<https://ftp.txdot.gov/pub/txdot/tpp/freight-planning/fntop/concepts/blocked-rail-crossing-traffic-management-system.pdf>



10.1 Use Case 1: Morning Backups

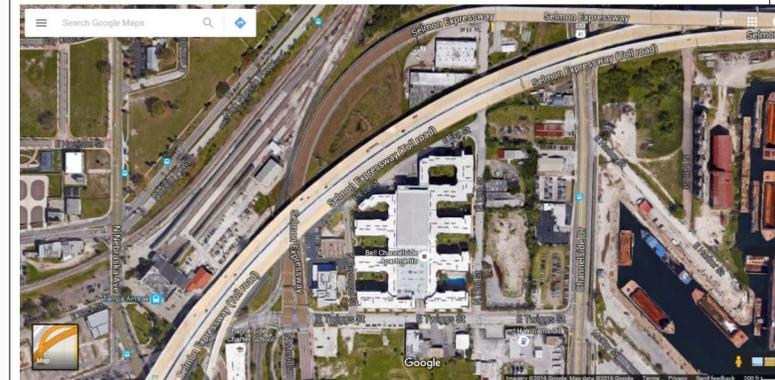
Table 11: Morning Backup Use Case Scenario 1: Normal Conditions

Use Case	Morning Backups
&	UC1-S1: CV Normal Conditions

This scenario describes the normal conditions where there is a “no problem” or “no issue” with the Drivers exiting the Selmon Expressway REL at the intersection of Meridian Avenue and Twigg Street in the morning peak period. Traffic exits the Selmon Expressway REL Monday to Friday 6 – 10 AM plus split operation from 10 AM – 1 PM. Although the site is equipped with the proposed CSW, EEBL and FCW technologies, normal events do not initiate the use of the proposed CV technology in the vehicle.

Current Situation:

At this site, a driver is proceeding west on the REL exit to a signalized intersection at Meridian Avenue and may turn right or left onto east-west Twigg Street, or may proceed south through on Meridian without incident.



Source: Google Maps

Proceeding west at mile marker 6.2, the REL begins a sweeping left curve while descending to ground level with limited forward visibility towards the queue at the Twigg Street traffic signal.

<https://www.its.dot.gov/pilots/events.htm>

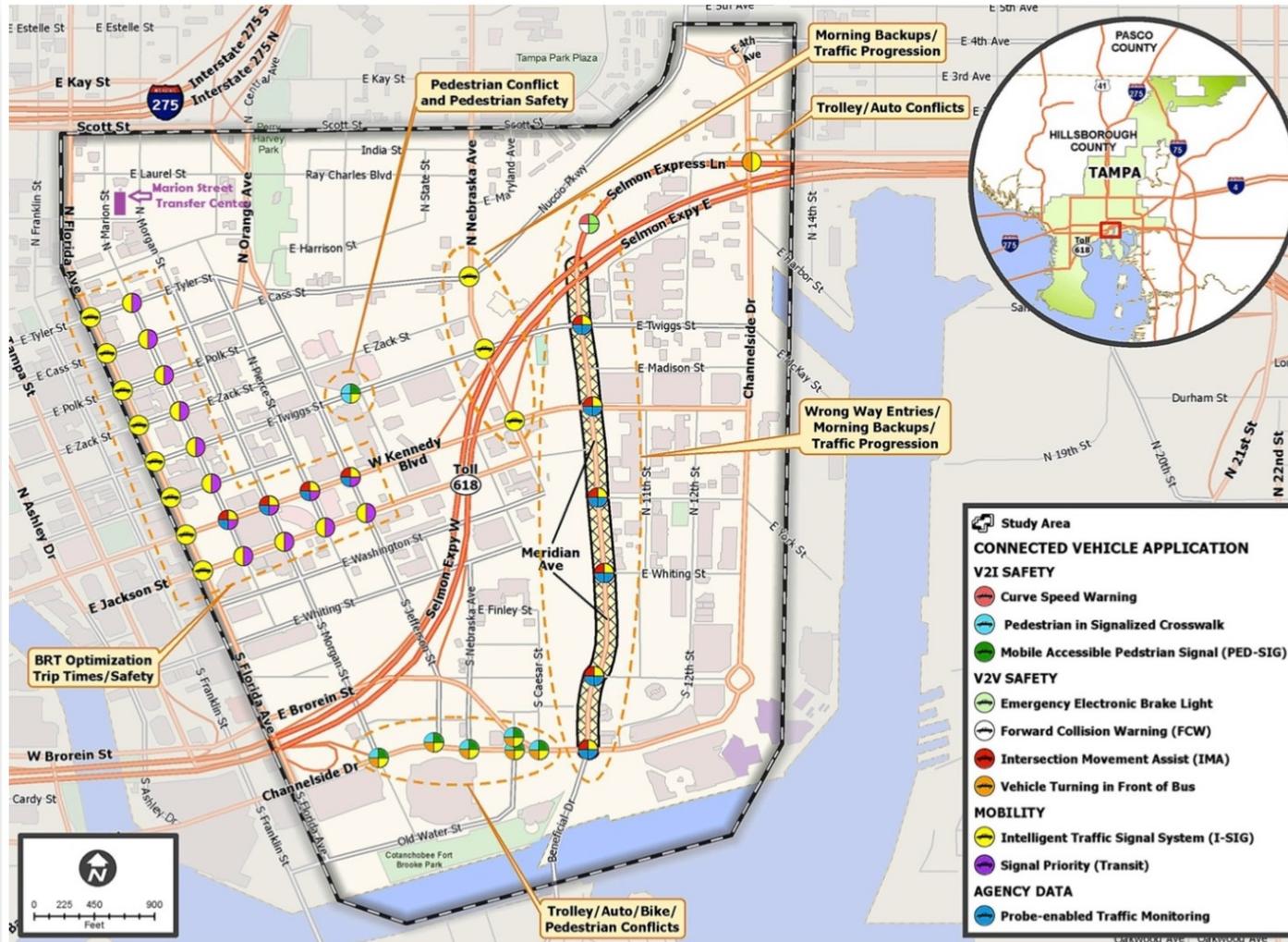


Figure 4: THEA CV Pilot Deployment Locations

Source: Googlemaps.com, HNTB



UMS Operation Concept

Surface Surveillance and Reconnaissance

#1

UMS is launched, and when an unidentified surface target approaches, UMS identifies the target.

#2

Automatic target tracking and data transmission. Conduct engagement mission following the engagement order from mother-ship





The New Maritime digital landscape



- Today tasks are moving ashore
- Shore provides advice and enshrines best practice
- The trend will continue.....



Rolls-Royce Proprietary Information

[https://wwwcdn.imo.org/localresources/en/MediaCentre/IMOMediaAccreditation/Documents/MSC%20100%20special%20session%20presentations/20181203 Technology Progression In MASS IMO Final For PDF.pdf](https://wwwcdn.imo.org/localresources/en/MediaCentre/IMOMediaAccreditation/Documents/MSC%20100%20special%20session%20presentations/20181203%20Technology%20Progression%20In%20MASS%20IMO%20Final%20For%20PDF.pdf)



MODELAGEM DE DADOS DA ARQUITETURA



MODELAGEM DE DADOS

- Uma parte importante da engenharia de sistemas envolve garantir a coerência entre os dados gerenciados no sistema e as **trocas** internas e com atores externos. A fim de descrever inequivocamente essas trocas, os dados, informações, fluxo de matéria, etc., devem ser formalizados.
- É preciso mecanismos para **modelar as estruturas de dados** e vincular às: Trocas Funcionais, Portas de Função ou Componente, Interfaces, etc.



RELACIONAMENTO COM A ARQUITETURA

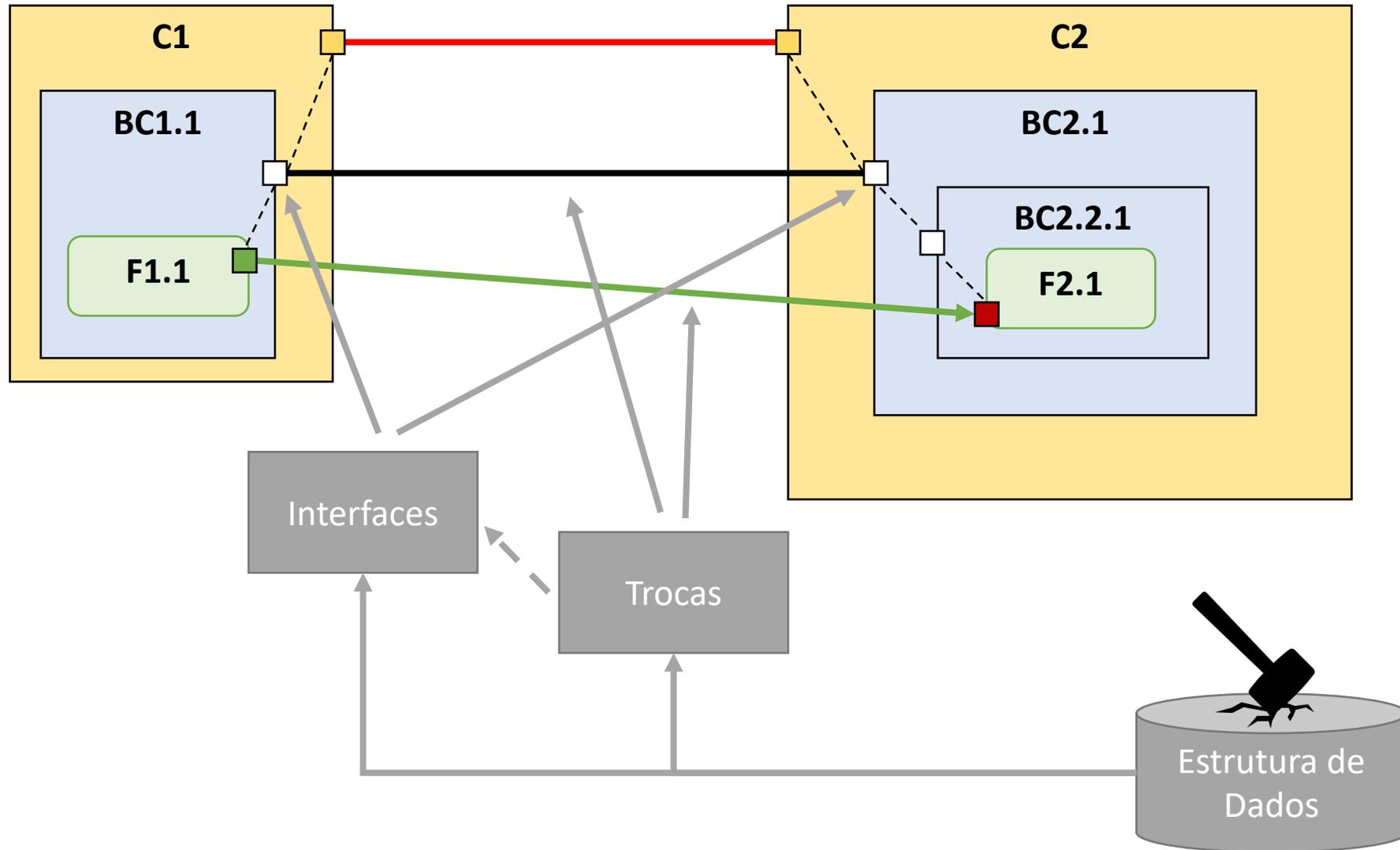




DIAGRAMA DE CLASSES



CONCEITO DE CLASSES

<https://www.youtube.com/watch?v=UI6lqHOVHic>





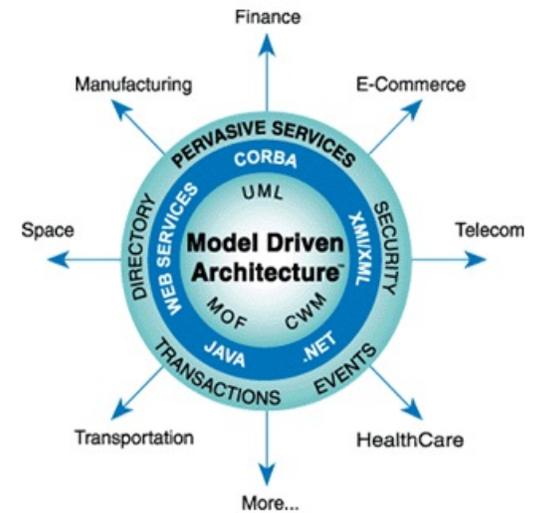
DIAGRAMA DE CLASSES



<http://www.uml.org/>

- 60'-70' – COBOL/FORTRAN/C (Estruturado)
- 80'-início 90' – SmallTalk/ADA/C++ (Início da OO)
- 90' – Java/UML (proliferação da OO)

- Unified Modeling Language (UML)
- Vocabulário para descrição de modelos.
- Apelo para “*Model Driven Architecture*”



<http://www.omg.org/mda/>



OBJETO?!!

- Interface para uma estrutura de dados + seus serviços
- Representação abstrata



Dados / Atributos
(PRIVADOS)

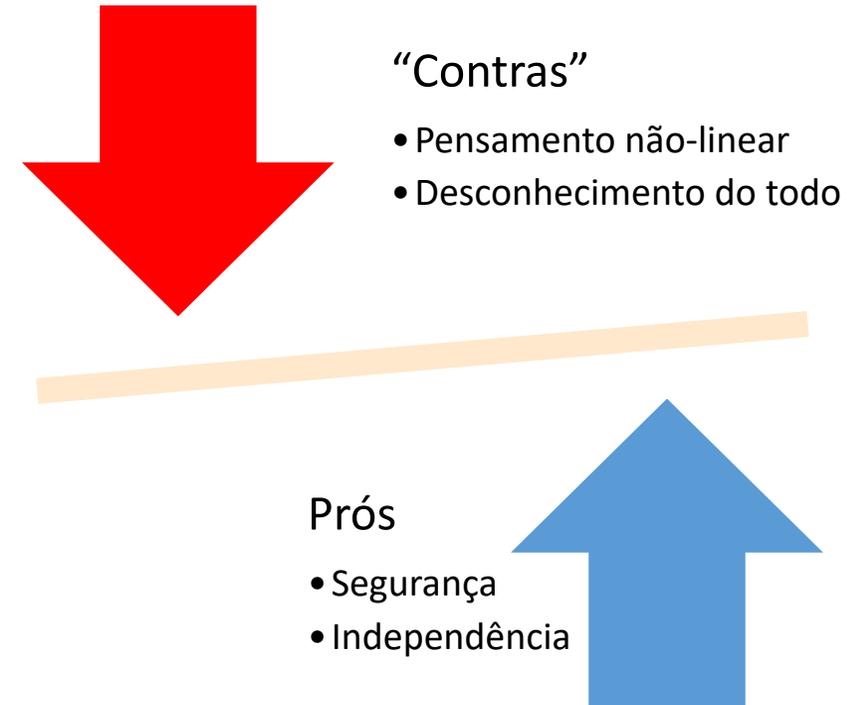
Operações / Métodos / Serviços / Interfaces
(PÚBLICOS)



ENCAPSULAMENTO

- Abstração do sistema
- Quem usa não precisa saber como funciona.
- Precisa saber como utilizar.
- Uso pelas interfaces

CSE-310



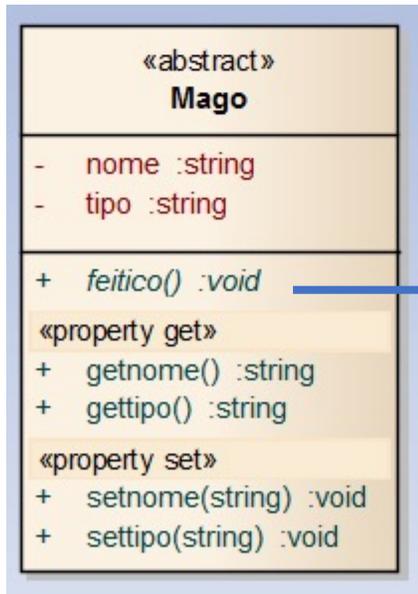


MODELO OBJETO (UML)



- Descreve Relacionamentos/Atributos/Operações
- Forma (MODELO) para o objeto

CSE-310



Identificação

Atributos

Métodos

Visibilidade:

+ → public: (todos veem)

- → private: (só ele vê)

→ protected: (só ele e os filhos)

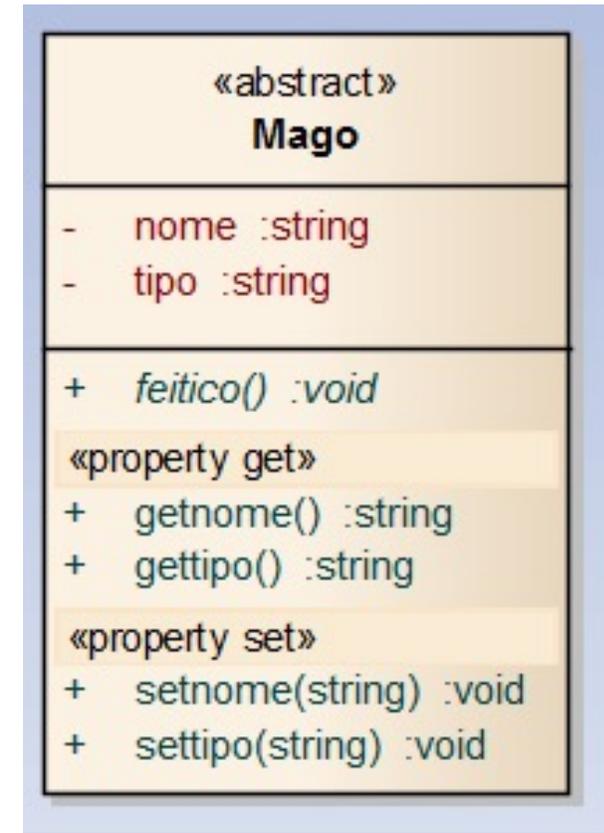
Itálico – método abstrato

→ Torna a classe abstrata

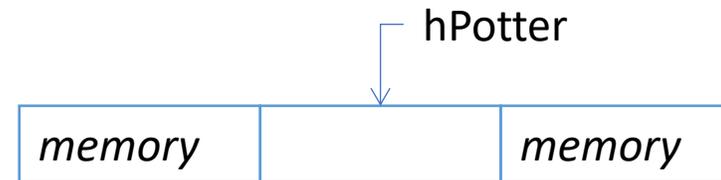
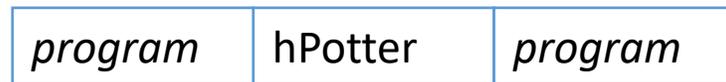


INSTÂNCIA

- Realização de uma classe (da forma)
- C++
 - `Mago *hPotter = new Mago();`
 - `Mago hPotter;`
- Java
 - `Mago hPotter = new Mago();`



CSE-310





ATRIBUTOS

- **Atributos** são definidos ao nível da **classe**, enquanto que os **valores** dos atributos são definidos ao nível do **objeto**.
- Ex.:
 - Uma pessoa (classe) tem os atributos nome, peso.
 - João (objeto) é uma pessoa com nome “João” e peso “80kg”
- **Uma classe não deve ter dois atributos com o mesmo nome.**
- Obs.:
 - Atributos estáticos – tem um único valor para todas as instâncias.
No UML o atributo estático é sublinhado
 - Pode-se usar um **valor inicial**, caso tenha sido omitido.



MÉTODOS

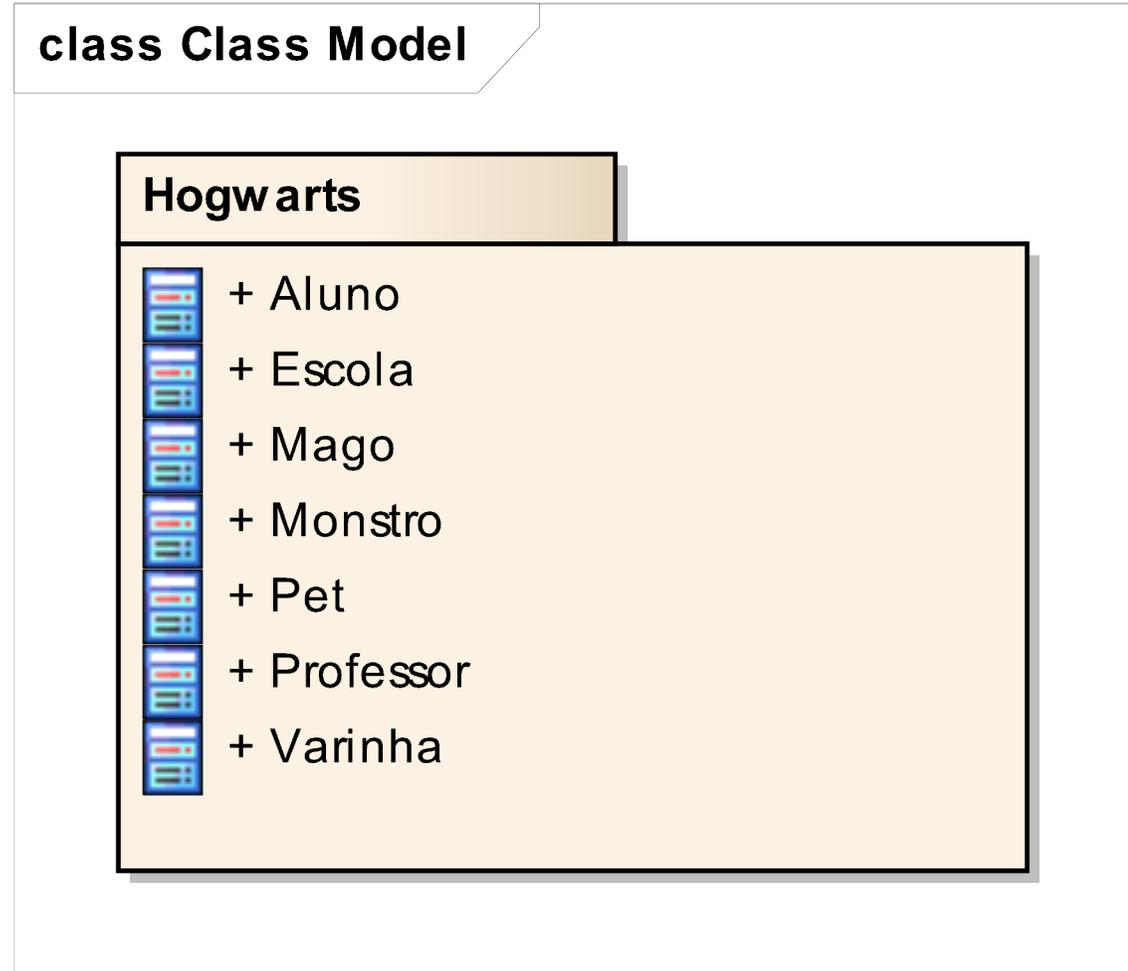
- Comportamento dos objetos é **modelado nas operações**.
 - É algo que se pode pedir para ser feito a um objeto.
 - Objetos da mesma classe têm as mesmas operações
- Padrão: Acesso e alteração dos valores do atributos através de operações. (ex.: GETS/SETS)



PACOTES

- Organiza as classes em grupos
- Hierarquiza (organiza) o projeto
- *Feels* de um endereço web.
 - br.com.cscerqueira.hogwards

CSE-310





OUTRAS INFOS

- *Classes Abstratas* – são classes que **não podem** ser instanciadas (não podem criar nada)
- *Classes Concretas* – são classes que podem ser instanciadas (que podem criar alguma coisa)
- **<<Interfaces>>** – Classes sem atributos e métodos sem implementação. A implementação vem de uma outra classe.
- Sobrecarga – Métodos com o mesmo nome, mas com **assinaturas** diferentes (parâmetros de entrada).



RELAÇÕES ENTRE CLASSES

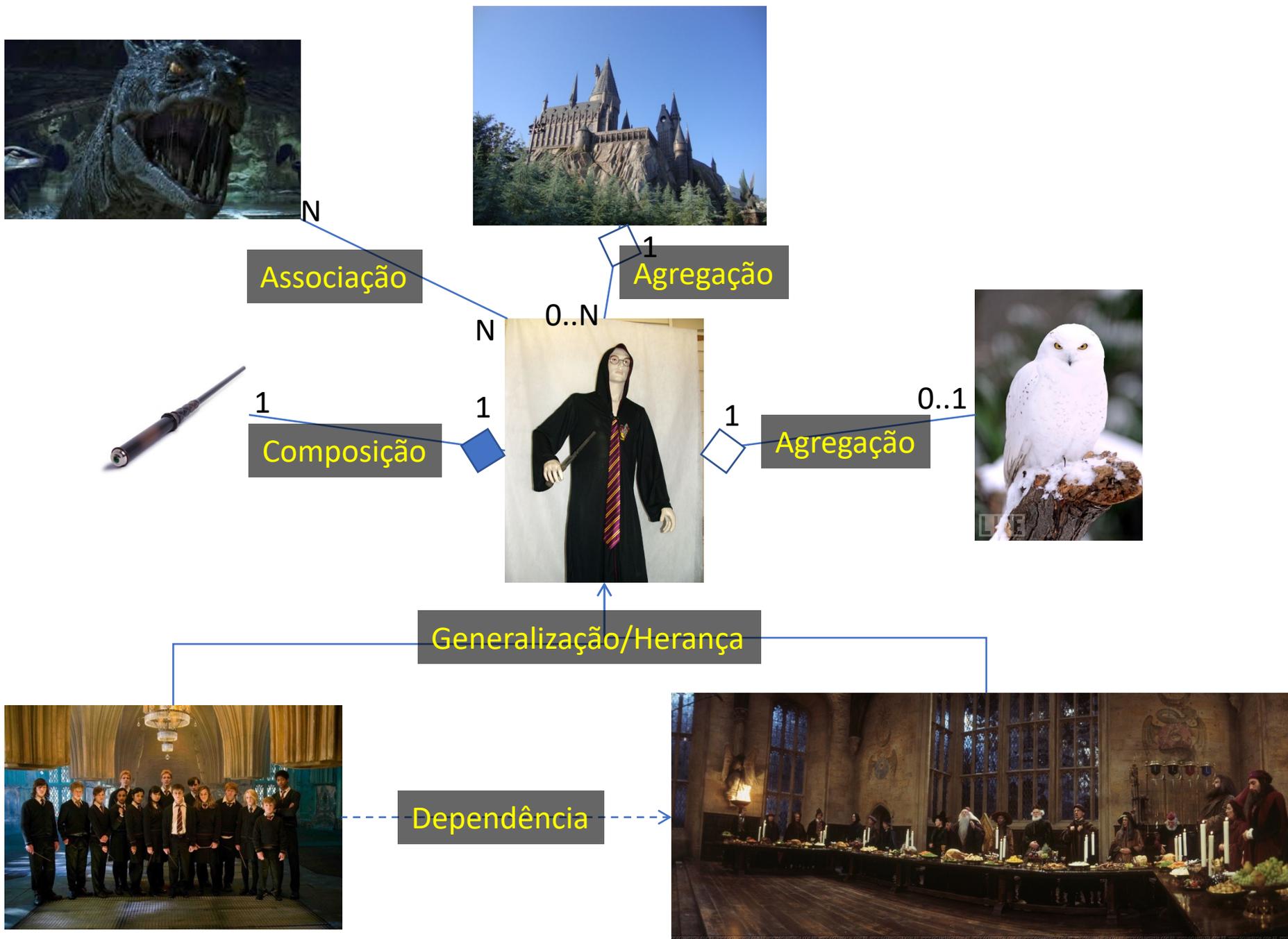


LIGAÇÕES / RELAÇÕES

- Os objetos se relacionam → a forma (classe) também tem que relacionar.
 - Relacionamentos precisam ser modelados.
- Exs.:
 - Um mago pode ser aluno ou professor (generalização/especialização)
 - Um mago joga um feitiço num monstro. (associação)
 - Um mago possui um pet. Uma escola possui magos. (agregação)
 - Um mago possui uma varinha, e a varinha conhece seu mago. (composição)
 - Alunos dependem dos professores para aprender (dependência)



CSE-310



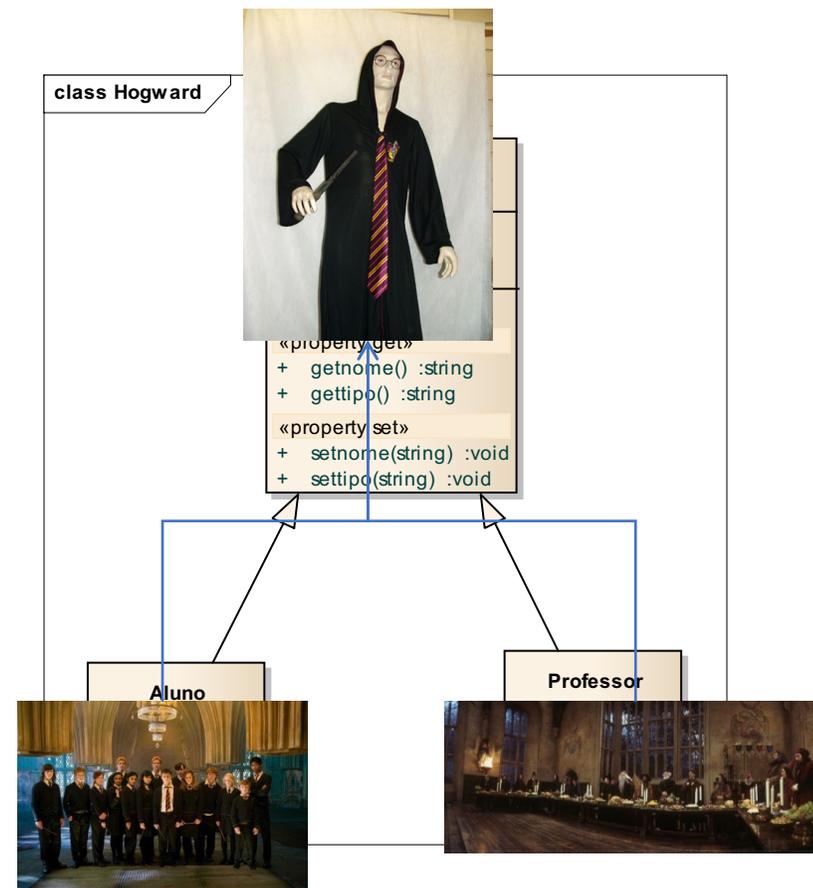


HERANÇA

- Relação semântica de “é um(a)”
 - **Sub-classes** herdam propriedades
 - **Super-classes** – propriedades comuns
- Operação abstrata é realizada na classe concreta.
 - **Herança simples** – apenas uma super classe
 - **Herança múltipla** – mais de uma super classe

CSE-310

Um mago pode ser aluno ou professor



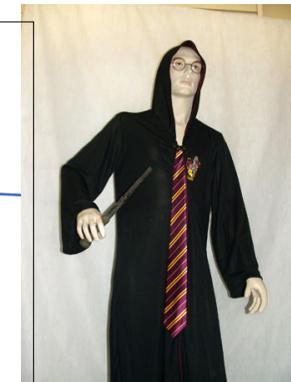
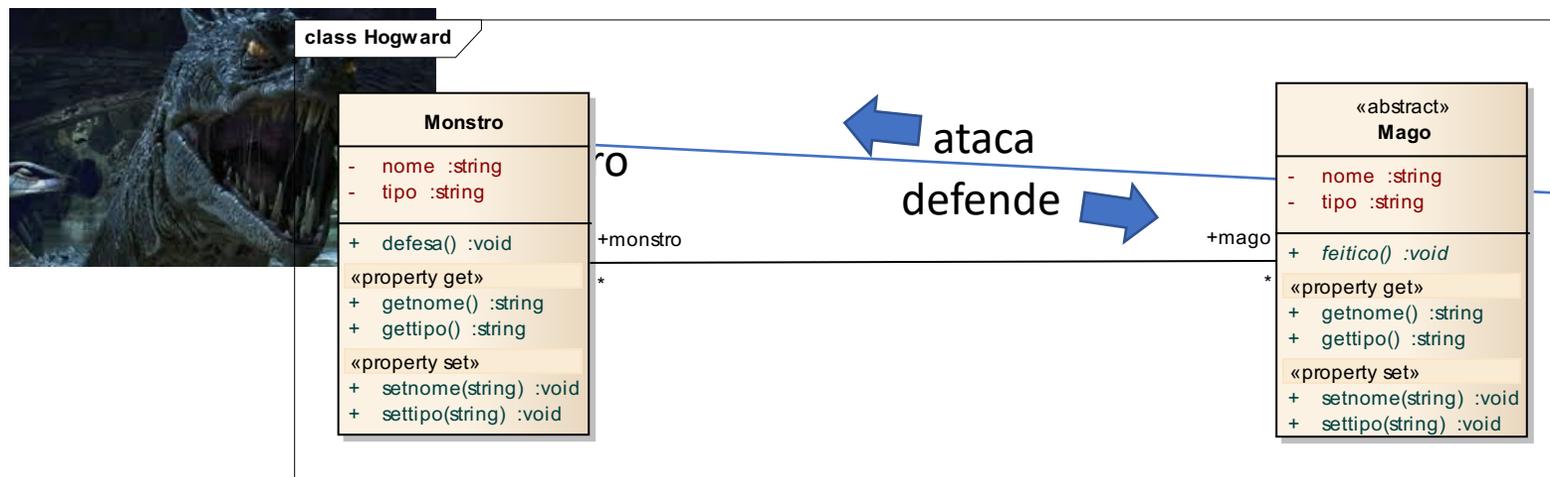


ASSOCIAÇÃO

- É a relação entre objetos das classes.
 - Implementação através de uma referencia.
 - Pode haver mais de uma associação entre as mesmas classes.
 - Papéis indicam a semântica, multiplicidade e visibilidade da associação.

CSE-310

Um mago joga um feitiço num monstro





ASSOCIAÇÃO - MULTIPLICIDADE

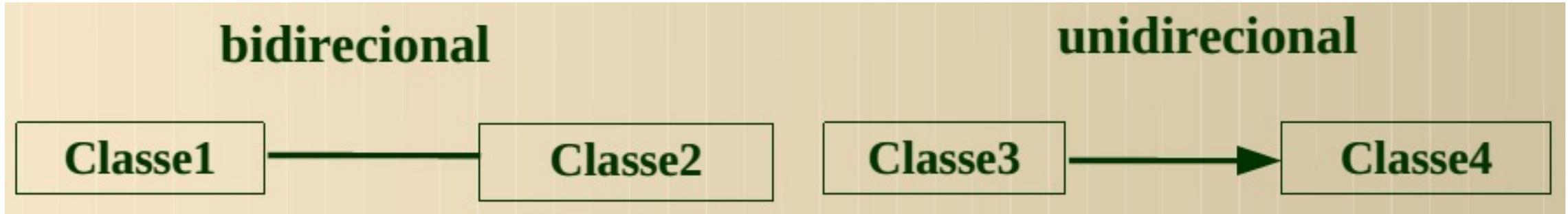
- Muitos para Muitos - * *
- Muitos para um - * 1
- Um para um - 1 1
- 1 - exatamente um
- 0..1 - zero ou 1
- * - zero ou mais
- 0..* - zero ou mais
- 1..* - um ou mais
- 1, 3..5 - um ou três a 5
- Quando um objeto recebe múltiplas instancias de outro é necessário controlar por estruturas de dados.



ASSOCIAÇÃO - DIREÇÃO

- Associações são classificadas quanto a navegabilidade:
 - **Bidirecional** – ambos objetos possuem referência
 - **Unidirecional** – classe que recebe a seta recebe a referência

CSE-310



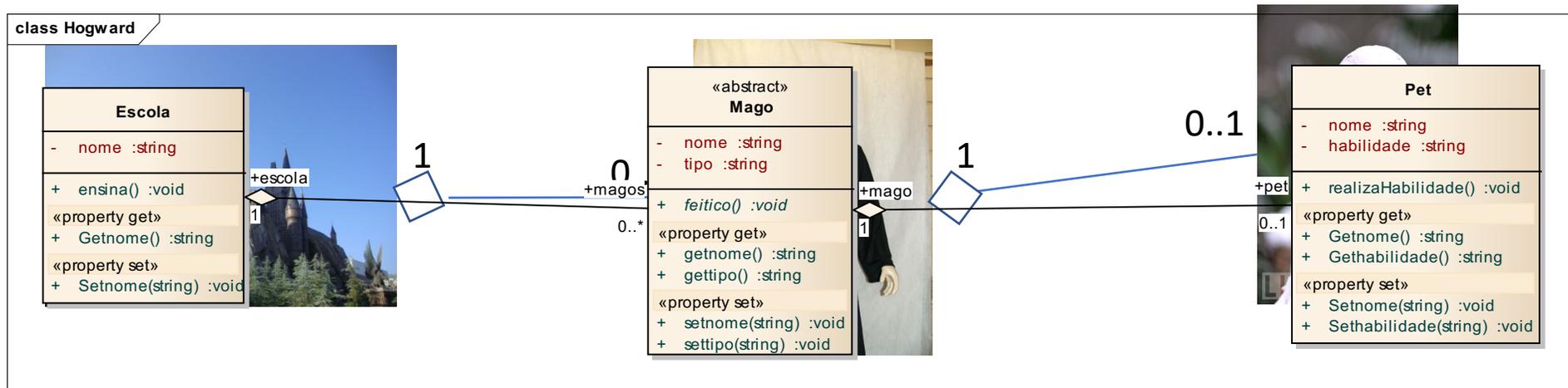


AGREGAÇÃO

- Associação com significado de contém / faz parte de.
- Relação de inclusão

CSE-310

Um mago possui um pet.
Uma escola possui magos.

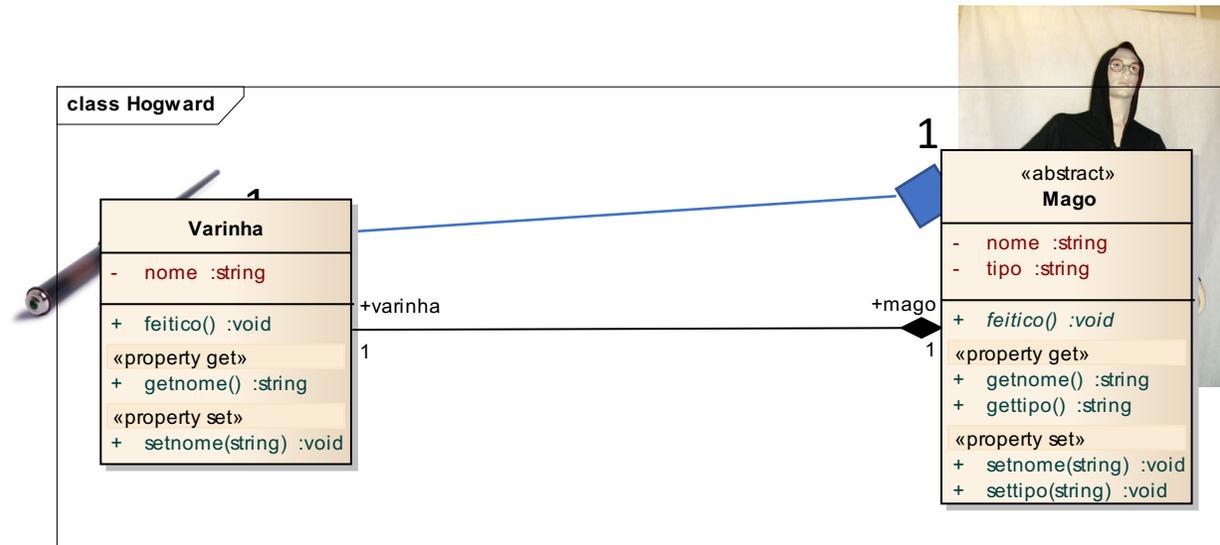




COMPOSIÇÃO

- Forma mais forte de agregação.
- Forte grau de dependência, cada parte só pode fazer parte de um todo, o todo e as partes tem o mesmo tempo de vida.

CSE-310





DEPENDÊNCIA (NÃO EXISTE NO SYSML/CAPELLA)

- Relação de uso, em que uma mudança na especificação do elemento usado pode afetar o elemento utilizador.
- Objeto usa outro como parâmetro de um método.

CSE-310



Alunos dependem dos professores para aprender



DIAGRAMA DE COMPONENTES

<https://www.omg.org/spec/UML/2.5.1/PDF>

<https://developer.ibm.com/articles/the-component-diagram/>



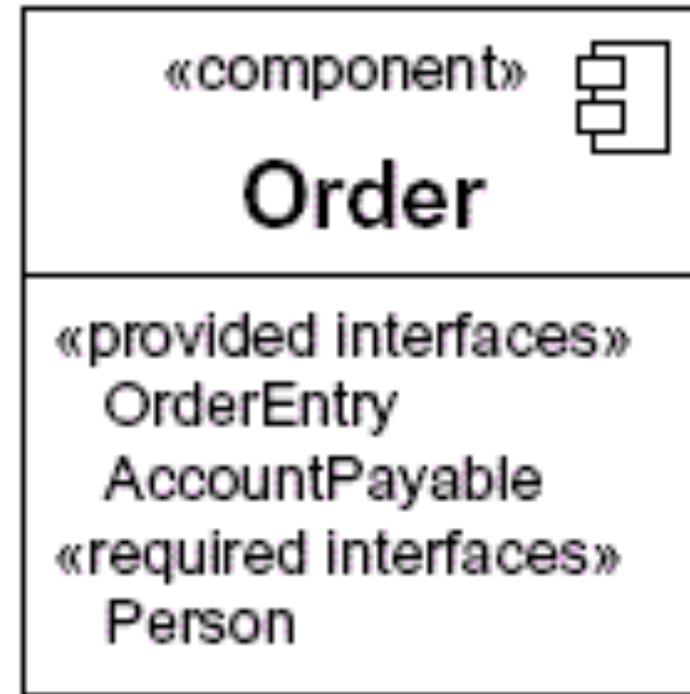
COMPONENTES

- Um Componente **representa uma parte modular** de um sistema que encapsula seu conteúdo.
- O principal objetivo do diagrama de componentes é **mostrar as relações estruturais** entre os componentes de um sistema.
- .



REPRESENTAÇÃO

- Desenhar um componente em UML 2 é muito semelhante a desenhar uma classe em um diagrama de classes.
- No componente *Order* do exemplo, o componente **fornece** as interfaces de *OrderEntry* e *AccountPayable*. Além disso, o componente também **requer** que outro componente forneça a interface *Pessoa*.





REPRESENTAÇÃO

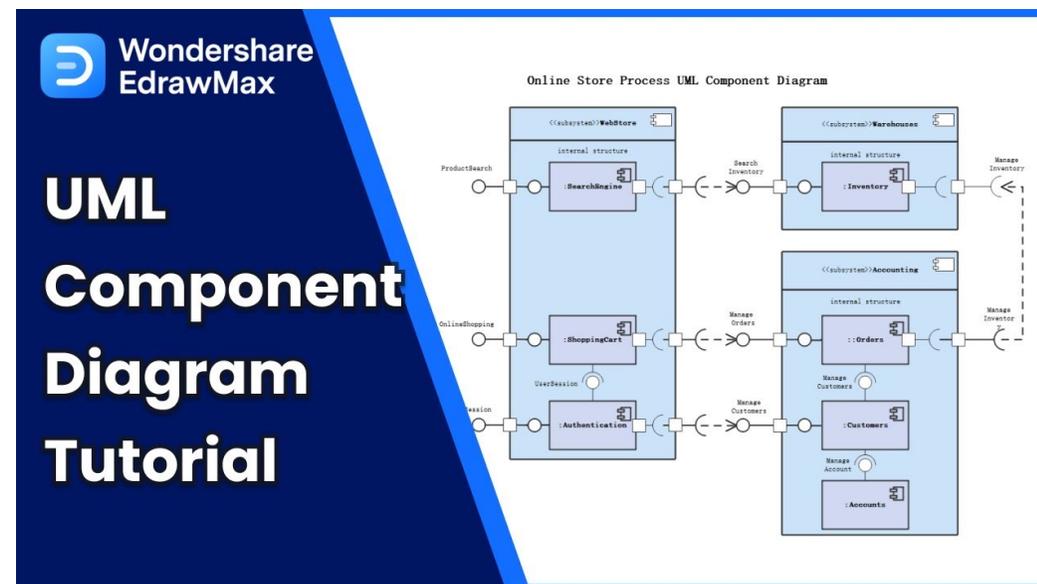
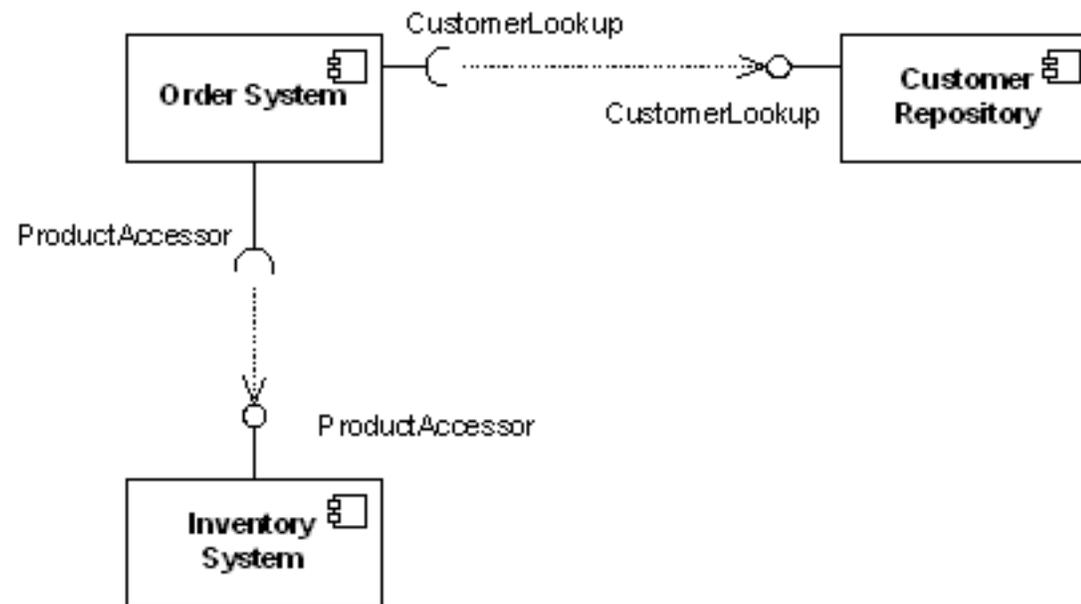
- A UML 2 também introduziu uma maneira de mostrar as interfaces fornecidas e necessárias de um componente.





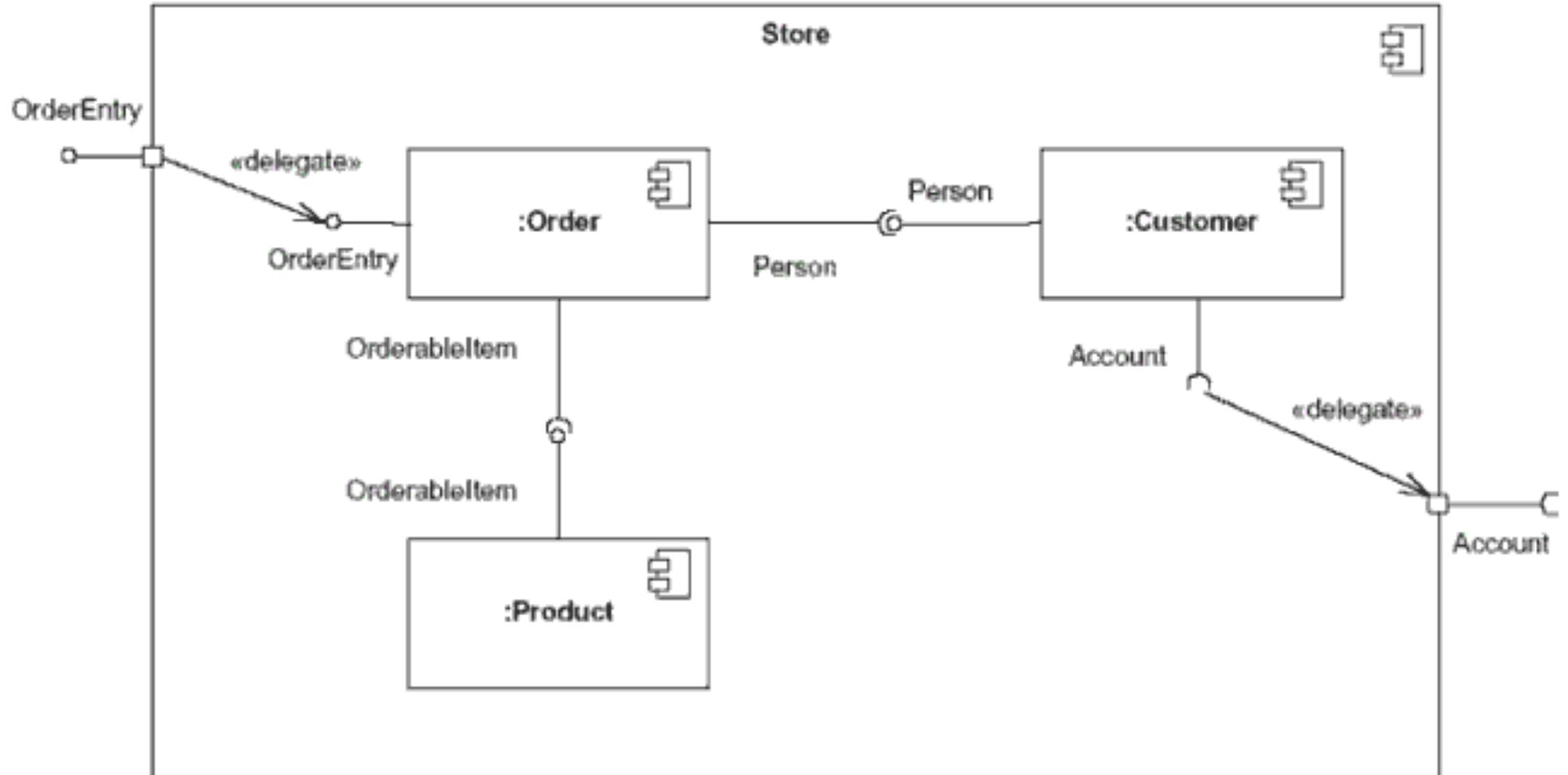
RELAÇÕES

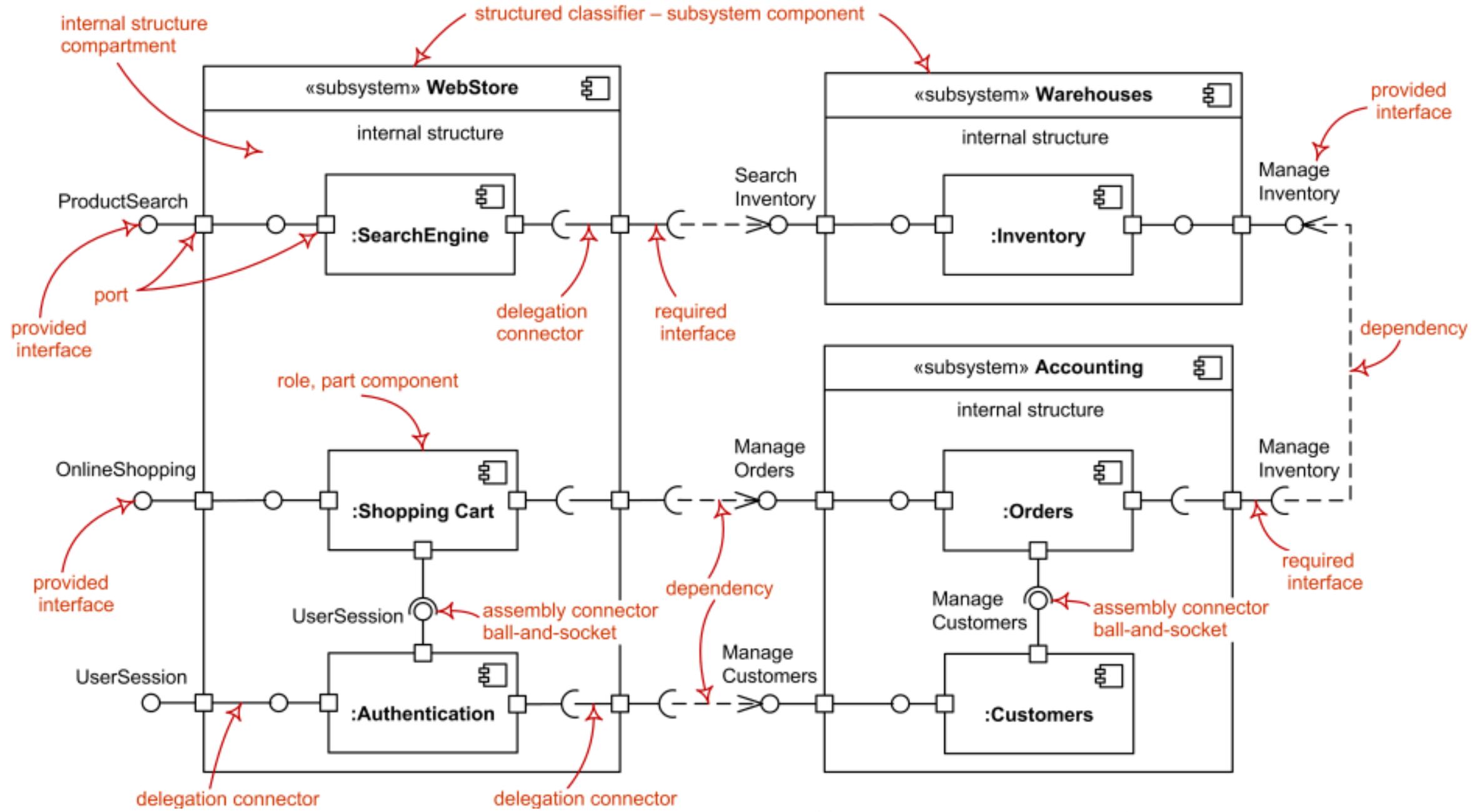
- Ao mostrar a relação de um componente com outros componentes, a **notação de pirulito (lollipop) e soquete (socket)** também deve incluir uma seta de dependência.
 - Em um diagrama de componente com pirulitos e soquetes, observe que a seta de dependência sai do **soquete de consumo (exige / require)** e sua cabeça de seta se conecta ao **pirulito do provedor (provide)**,





MESMA LÓGICA PARA DECOMPOSIÇÃO INTERNA



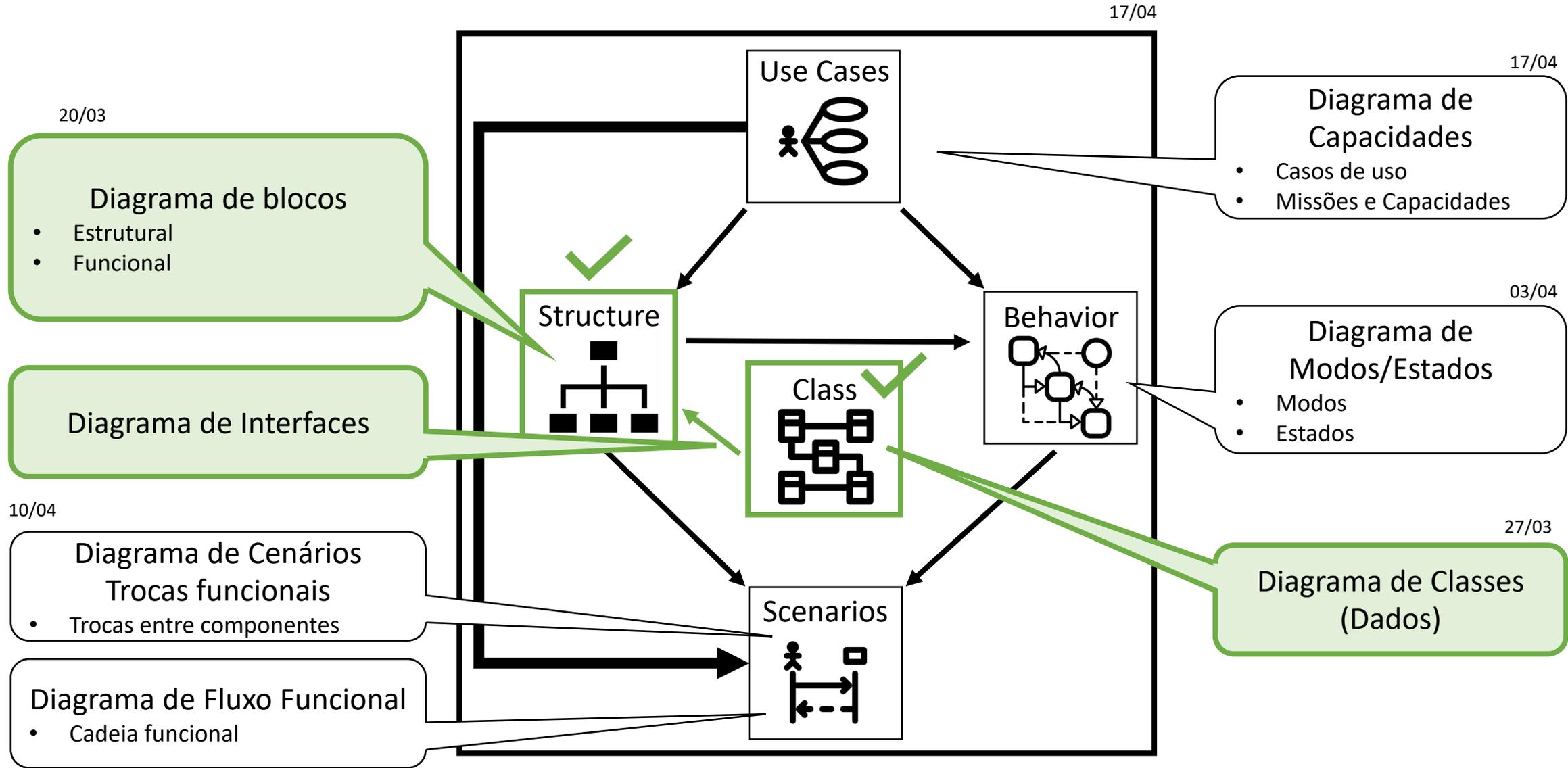




CONSIDERAÇÕES FINAIS



ROTEIRO DOS DIAGRAMAS





CONSIDERAÇÕES FINAIS

- Resumo:
 - Conceitos: Stakeholder / Ciclo de Vida / Conceito de Operação
 - Diagramas: Classe / Componente
- Exercícios
 - 4.1 – (ind) Praticar diagrama de classes.
 - 4.2 – (grupo) Levantar stakeholders.
 - 4.3 – (grupo) Criar diagrama de componentes (interfaces) do Delorean e usar o diagrama de classes para descrever a estrutura de dados das trocas.