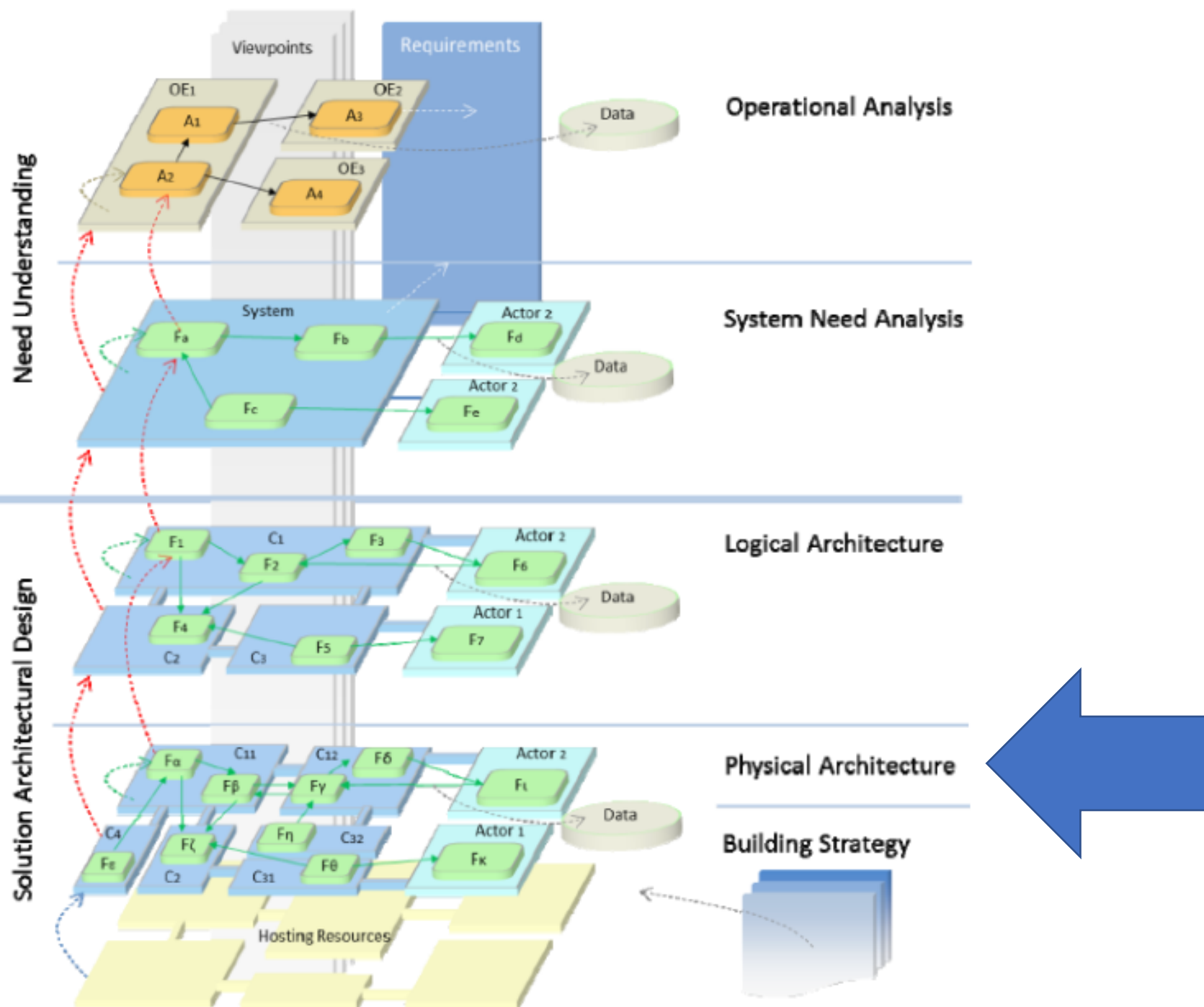




# ARQUITETURA CONCRETA





# O QUE ACONTECE NA ARQUITETURA CONCRETA PHYSICAL ARCHITECTURE (PA)?



# ARQUITETURA CONCRETA

*“Como o sistema será construído”*

- Essa perspectiva tem o mesmo objetivo da arquitetura lógica, exceto que ela define a **arquitetura finalizada do sistema**, como ele deve ser completado e integrado. Ele adiciona as funções requeridas pela implementação e **escolhas técnicas e revela os componentes comportamentais que executam essas funções**. Esses componentes comportamentais são então **implementados usando componentes de implementação**.
- Define a solução em um **nível de pormenor suficiente para especificar a evolução e as aquisições de todos os subsistemas (ou componentes) a implementar, e definir e orientar as fases de integração, verificação e validação (IVV) do sistema.**



- Muitas vezes, é apenas a este nível que são introduzidas escolhas e restrições relacionados com as **tecnologias de implementação e produção**, com os **elementos existentes a reutilizar**.
- Quaisquer ambiguidades ou imprecisões que ainda poderiam existir na arquitetura lógica (AL), devem ser resolvidas, de modo a **constituir contratos de desenvolvimento claros** para os componentes identificados.
- PA é o **lugar privilegiado de co-engenharia** com engenharia de subsistemas e componentes de software ou hardware.



# PRINCIPAIS ATIVIDADES A SEREM REALIZADAS PARA A DEFINIÇÃO DA PA FINALIZADA

1. definir os princípios estruturantes da arquitetura e do comportamento;
2. para detalhar e finalizar o comportamento esperado do sistema;
3. para construir e racionalizar uma ou mais arquiteturas de sistema possíveis;
4. para selecionar, completar e justificar a arquitetura do sistema mantida.



# 1. DEFINIÇÃO DOS PRINCÍPIOS ESTRUTURANTES DA ARQUITETURA E COMPORTAMENTO

- O principal objetivo da PA é minimizar a complexidade por meio da racionalização.
  - Um dos meios de racionalização mais utilizados consiste em **reduzir a diversidade e a heterogeneidade** dentro da solução, buscando semelhanças e, portanto, **possíveis invariantes da arquitetura (às vezes chamados de "padrões")** que podem ser aplicados mais de uma vez da mesma maneira – ou configuráveis.
  - Outra maneira clássica de superar a complexidade é baseada na **separação das preocupações e sua encapsulamento dentro de partes da arquitetura** o mais separadas possível umas das outras.



## 2. DETALHAMENTO E FINALIZAÇÃO DO COMPORTAMENTO ESPERADO DO SISTEMA

- Definir o comportamento esperado do sistema, com um **nível de detalhamento e validação suficiente para que cada um de seus componentes possa ser implementado (ou selecionado e adquirido), sem maiores riscos ou grandes questionamentos**; Esta definição deve demonstrar a conformidade com os requisitos, especialmente as não funcionais, às quais o sistema terá de respeitar quando utilizado em condições operacionais.



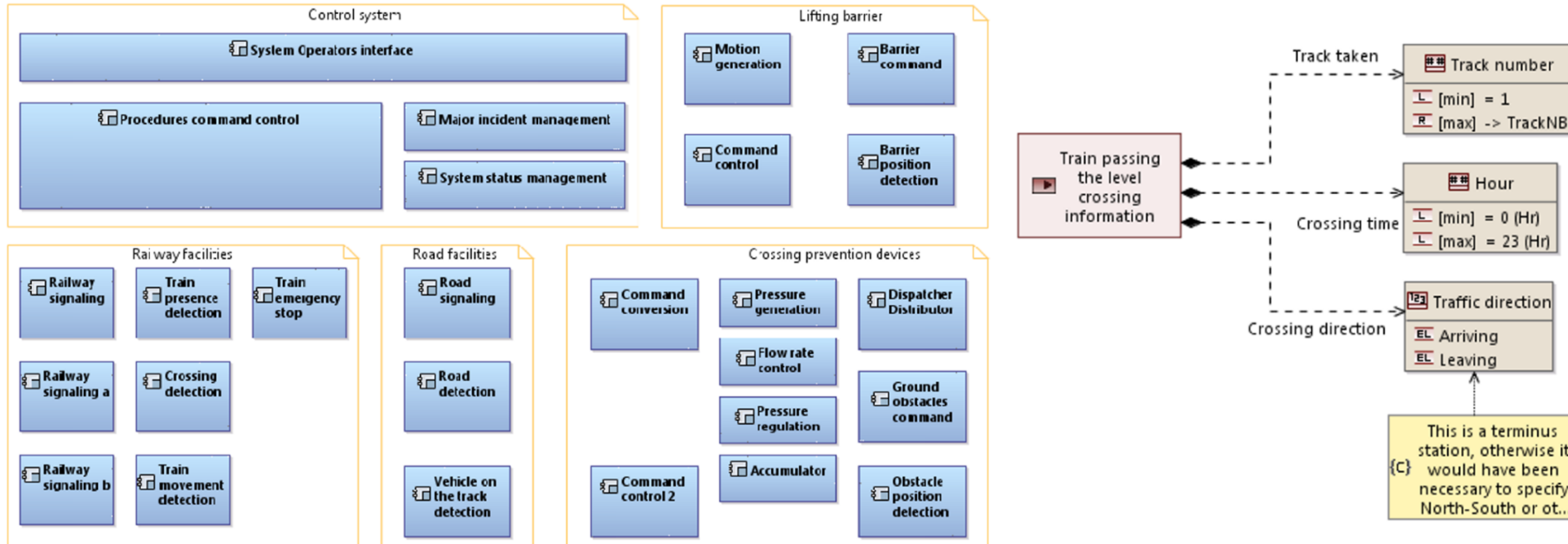


- Em particular, o comportamento finalizado não deve ser considerado como um simples refinamento daquele definido na Arquitetura Conceitual.
- A finalização do comportamento escolhido, de fato, muitas vezes constitui um **reprojeto**, que deve resultar da comparação entre o comportamento principal da Arq. Funcional e as implicações dos princípios escolhidos na Arq. Final: **escolhas tecnológicas e adoção de padrões, princípios estruturantes, etc.**



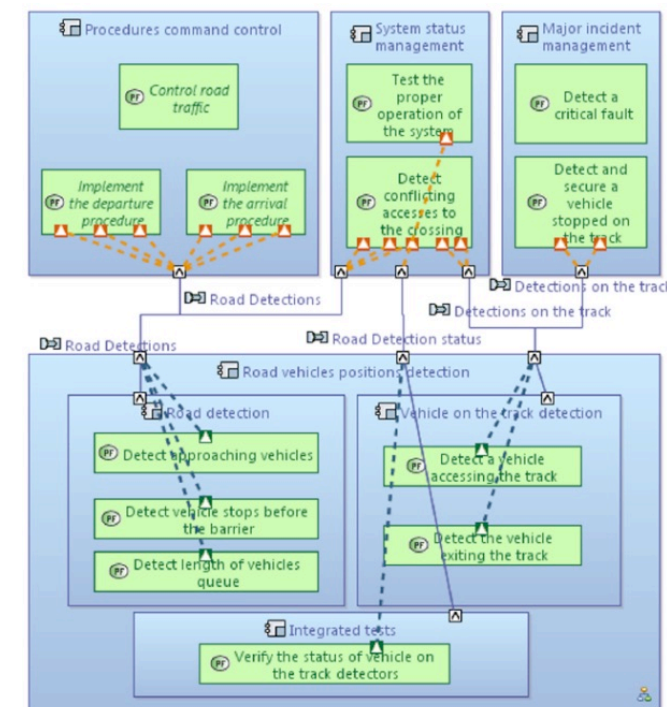
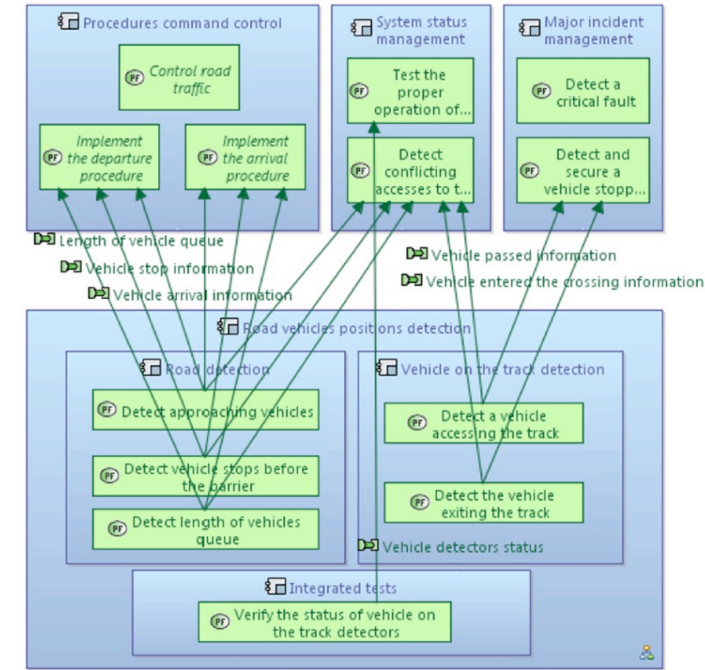
### 3. CONSTRUÇÃO E RACIONALIZAÇÃO DE UMA OU MAIS ARQUITETURAS DE SISTEMAS POSSÍVEIS

- Esta etapa tem como objetivo definir uma ou mais soluções refletindo os princípios estruturantes definidos na Arq. Conceitual, o comportamento finalizado prévio, satisfazendo as restrições esperadas e aplicando opções tecnológicas e de reutilização decididas de acordo com os princípios estruturantes adotados.



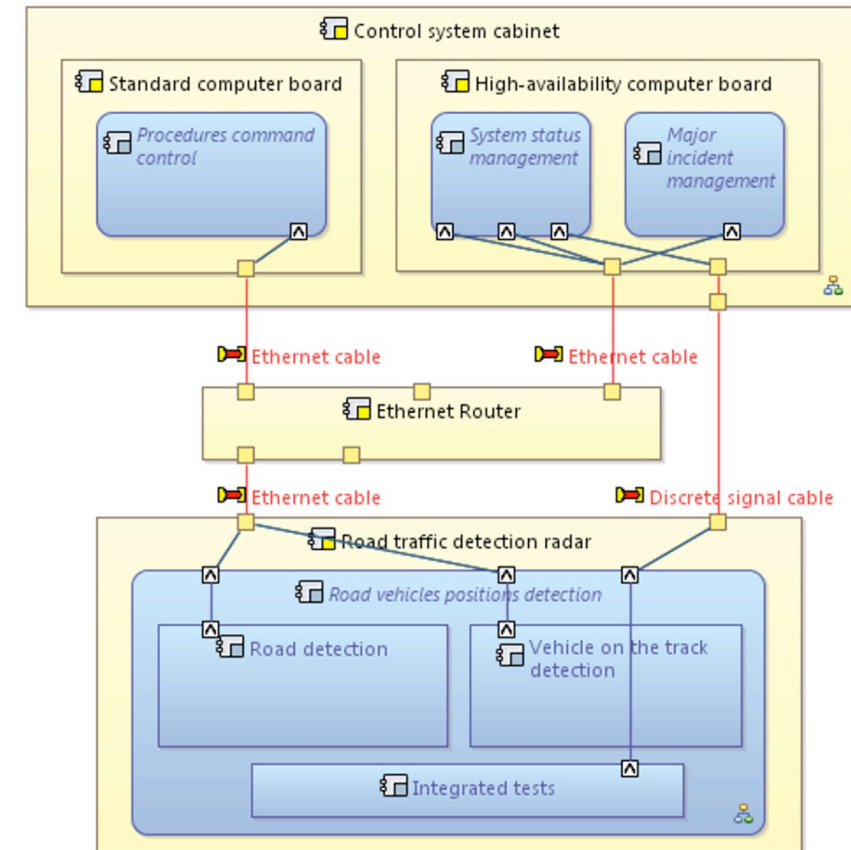


- Nos casos mais simples, ou em sistemas com um dominante físico ou elétrico, os itens de troca são muitas vezes simples em sua descrição e uso neste nível de engenharia e modelagem.
- No entanto, para itens de troca mais complexos, envolvendo um grande número de elementos de conteúdo de troca, **é desejável poder estruturar uma lista de itens de troca que pode ser extensa, agrupando-os por tipo de serviço realizado, por exemplo.** Esse é o papel do conceito de interface (também presente principalmente no design de software).





- A Arq. Final complementa essa descrição comportamental por meio da definição dos componentes de implementação, ou **hosting physical components, contendo componentes comportamentais e formando a infraestrutura do sistema; Os componentes comportamentais são implantados nesses componentes de host**, que fornecem recursos necessários para seu comportamento e hardware (links) para suas comunicações. Pode, assim, consistir em computadores de alto desempenho, recursos para processamento digital ou analógico, sistemas mecânicos, evaporadores, fornos, reatores químicos, etc.
- **Os componentes físicos de hospedagem são conectados por links físicos**, refletindo os meios que canalizam as trocas entre componentes comportamentais (uma rede cabeada, um link de satélite, uma tubulação ou um eixo mecânico, por exemplo).
- Os mesmos processos de racionalização devem ser realizados para hospedar componentes físicos e comportamentais, em conformidade com os princípios estruturantes estabelecidos.





## 4. SELEÇÃO, PREENCHIMENTO E JUSTIFICATIVA DA ARQUITETURA DO SISTEMA

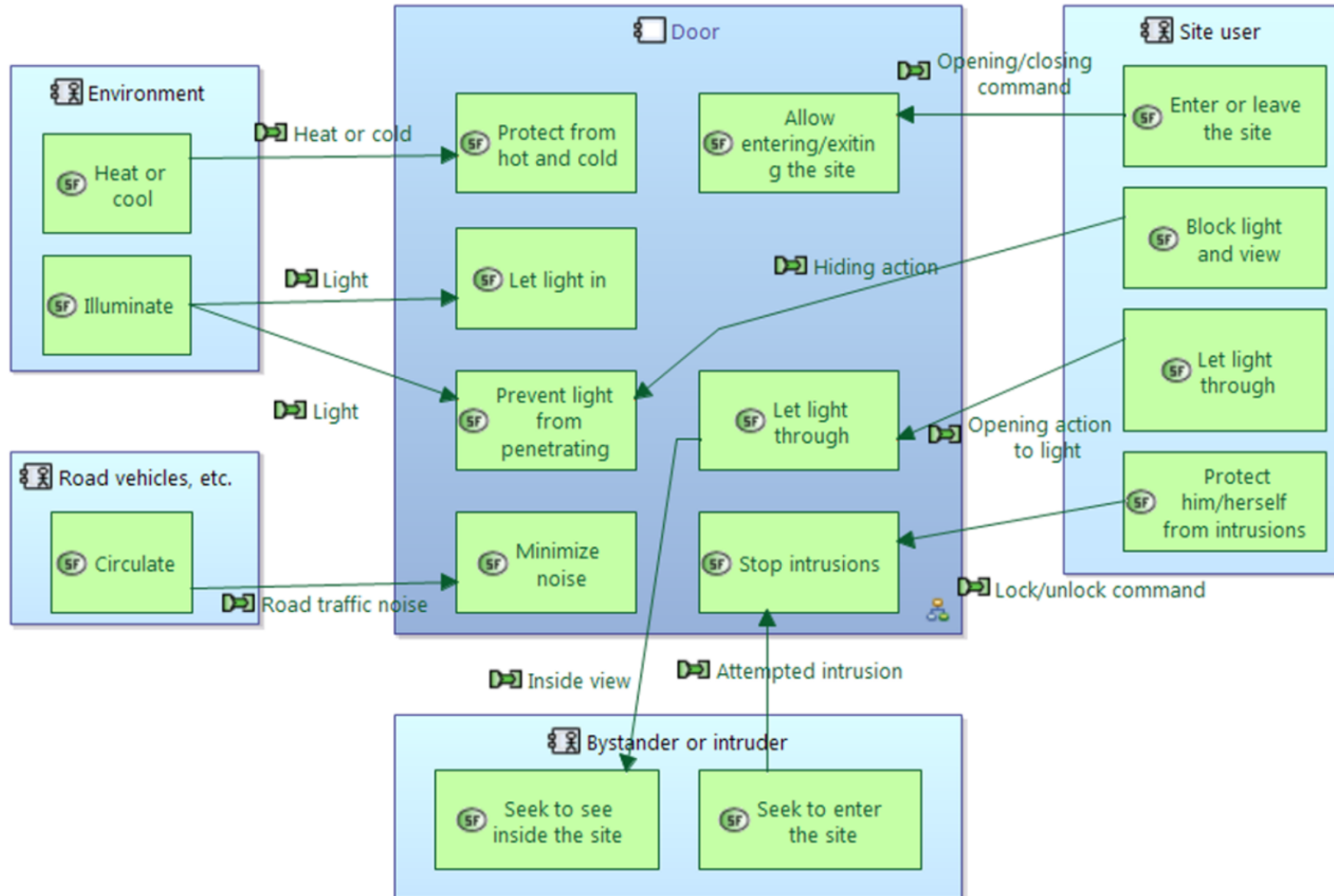
- Finalizar as escolhas entre as **alternativas potenciais e verificar se a alternativa retida satisfaz, possivelmente por meio de um trade-off aceitável, todas as necessidades e restrições que lhe foram impostas.**
- Por exemplo, os recursos de implementação disponíveis podem não ser suficientes para suportar um comportamento esperado ou propriedades associadas (carga computacional muito alta para um determinado processo em computadores que o suportam, temperatura e pressão muito altas para um determinado tubo, etc.).
- Isso levará a um redesenho da arquitetura, incluindo uma recomposição e uma distribuição diferente de componentes comportamentais, ou o uso de outros recursos de implementação (computadores mais potentes, tubos mais robustos).



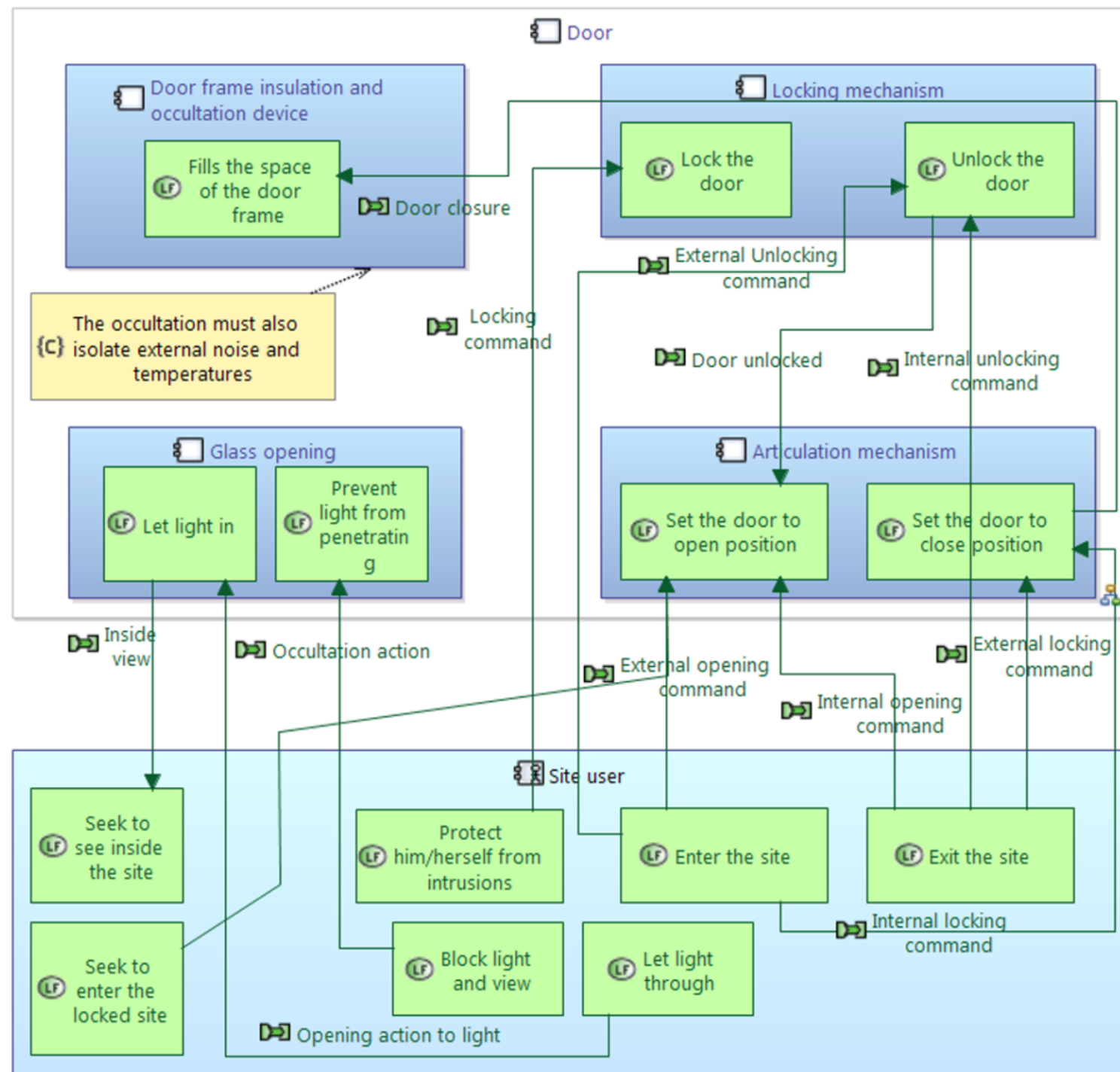
# SUMMARY

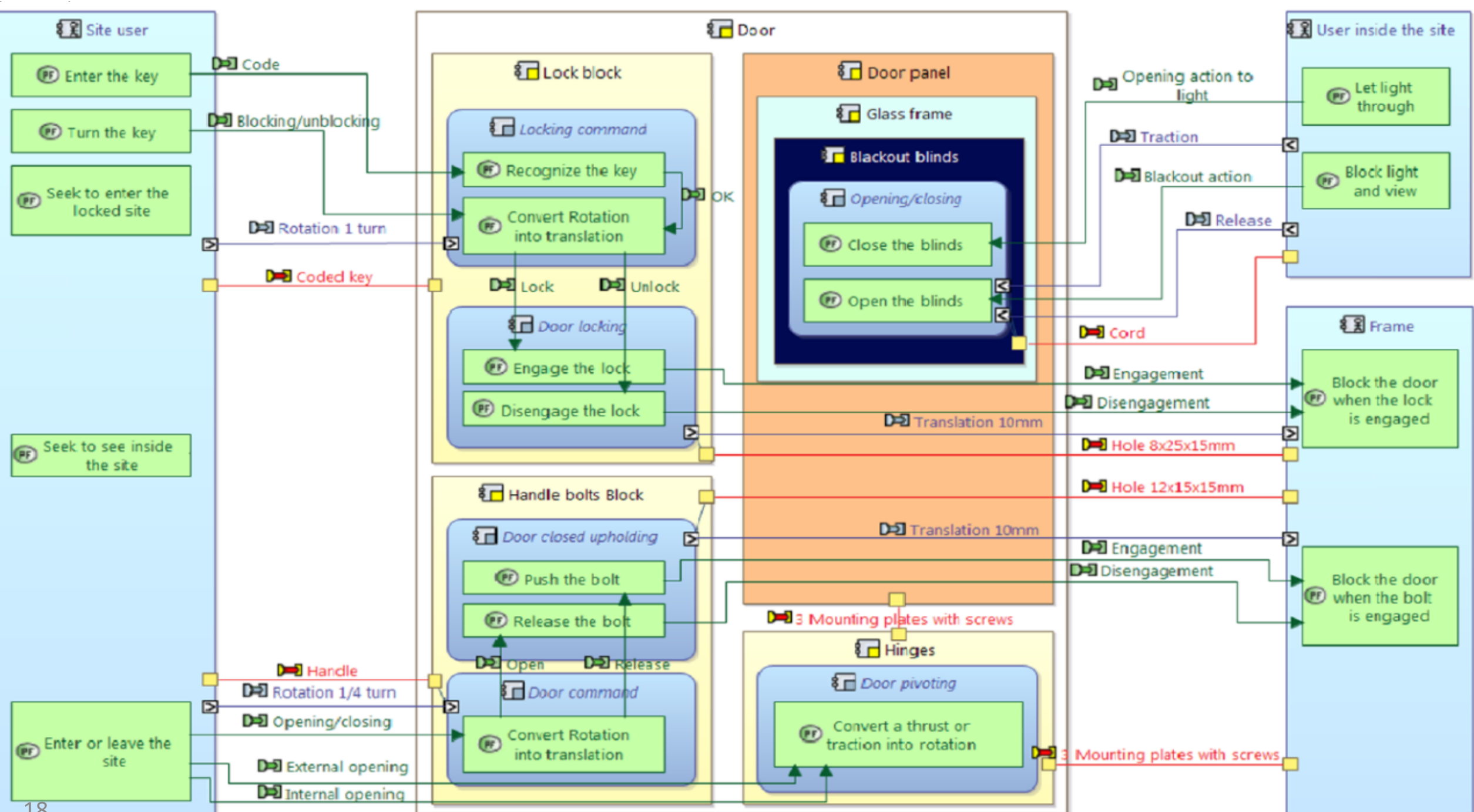
<b>define the structuring principles of the architecture and behavior;</b>	Factors impacting or constraining the definition of the architecture, as well as the viewpoints and structuring design choices mentioned earlier, equally apply to this level of architecture and are to be taken into account in a similar way.
<b>detail and finalize the expected system behavior;</b>	define the expected behavior of the system, to a level of detail and validation enough so that each of its components can be implemented (or selected and purchased), without any further risk or major questioning
<b>build and rationalize one or more possible system architectures</b>	define one or more solutions reflecting the structuring principles defined in the LA, the previous finalized behavior, satisfying the expected non-functional constraints and applying technology and reuse choices decided in accordance with the structuring principles adopted.
<b>select, complete and justify the system architecture retained.</b>	finalize the choices among potential alternatives, and verify that the retained alternative satisfies, possibly by means of an acceptable trade-off, all of the needs and constraints that have been imposed thereon.











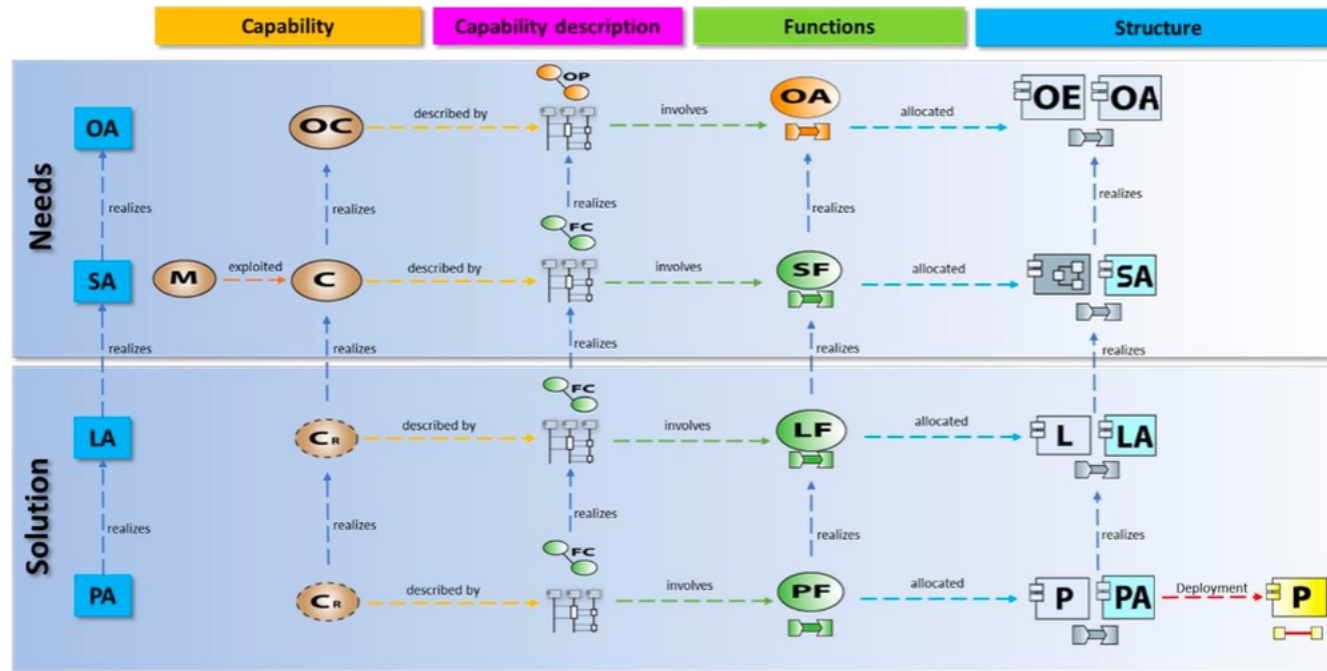


Figure 2.3: Arcadia ontology traceability






























































Arcadia layer	Requirements	Capability	Capability description	Functional	Structure	Modes and States	Data	Interfaces
Operational Analysis	R-OA	OA1	OA2	OA3	OA4	M&S-OA5	D-OA6	I-OA7
	Capture stakeholder requirements	Define Operational Capabilities	Define processes and scenarios	Define Operational Activities and interactions	Capture Operational Entities and Actors. Allocate Operational Activities to Operational Actors, Entities	Define operational modes and states	Define operational data model	Define interfaces and describe interfaces scenarios
	 				 	 	 	 
								
System Analysis	R-SA	SA1	SA2	SA3	SA4	M&S-SA5	D-SA6	I-SA7
	Derive Stakeholder requirements and capture System requirements	Define System Missions and System Capabilities	Define Functional Chains and Scenarios.	Define System Functions. Define Functional Exchanges and components	Allocate System Functions to System and Actors	Define system modes and states	Define system data model	Define interfaces and describe interfaces scenarios Enrich Logical Scenarios.
	 	 				 	 	 
								
Logical Architecture	R-LA	LA1	LA2	LA3	LA4	M&S-LA5	D-LA6	I-LA7
	Derive system requirements and Capture components requirements	Transition Capabilities Realization from system layer	Define Functional Chains and scenarios	Derive System Functions and define Logical Functions. Define Functional Exchanges and components.	Allocate Logical Functions to Logical Components	Define logical components modes and states	Define logical data model	Delegate System Interfaces and create Logical Interfaces. Enrich Logical Scenarios.
	 				 	 	 	 
								
Physical Architecture	R-PA	PA1	PA2	PA3	PA4	M&S-PA5	D-PA6	I-PA7
	Derive logical requirements and capture physical requirements	Transition Capabilities Realization from logical layer	Define Functional Chains, Scenarios, and Physical Path	Derive Logical Functions and define Physical Functions. Define Functional Exchanges and components.	Define Physical Nodes and refine Behavioural Physical Components. Allocate Behavioural Components.	Define physical nodes modes and states	Define physical data model	Delegate Logical Interfaces and create Physical Interface. Enrich Physical Scenarios.
	 				 	 	 	 
					 			



Table 3.2: Arcadia matrix activities



Arcadia layer	Requirements	Capability	Capability description	Functional	Structural	Modes and States	Data	Interfaces
<b>Operational Analysis</b>	<b>R-OA</b> No dedicated diagram	<b>OA1</b> [OCB] Operational Capabilities	<b>OA2</b> [OAS] Operational Activity Scenario [OPD] Operational Process Scenario [OES] Operational Entity Scenario	<b>OA3</b> [OABD] Operational Activity Breakdown Diagram [OAIB] Operational Activity Interaction Blank	<b>OA4</b> [OEBD] Operational Entities Blank Diagram [ORB] Operational Roles Blank [OAB] Operational Architecture Blank	<b>M&amp;S-OA5</b> [MSM] Modes and States	<b>D-OA6</b> [CDB] Class Diagram	<b>I-OA7</b> [IDB] Interface Definition Blank [CEI] Component External Interfaces [IS] Interface Scenario [CDI] Component Detailed Interface
<b>System Analysis</b>	<b>R-SA</b> No dedicated diagram	<b>SA1</b> [MCB] Mission and Capabilities Blank [CC] Contextual Capability	<b>SA2</b> [FS] System Functional Scenario [ES] System Entity Scenario [SFCD] System Functional Chain Description	<b>SA3</b> [SFBD] System Functional Breakdown Diagram [SDFB] System Data Flow Blank	<b>SA4</b> [CSA] Contextual System Actor [SAB] System Architecture Blank	<b>M&amp;S-SA5</b> [MSM] Modes and States	<b>D-SA6</b> [CDB] Class Diagram	<b>I-SA7</b> [IDB] Interface Definition Blank [CEI] Component External Interfaces [IS] Interface Scenario [CDI] Component Detailed Interface
<b>Logical Architecture</b>	<b>R-LA</b> No dedicated diagram	<b>LA1</b> [CRB] Capabilities Realization Blank [CRI] Contextual Capability Realization Involvement	<b>LA2</b> [FS] Logical Functional Scenario [ES] Logical Entity Scenario [LFCD] Logical Functional Chain Description	<b>LA3</b> [LFBD] Logical Functional Breakdown Diagram [LDFB] Logical Data Flow Blank	<b>LA4</b> [LCBD] Logical Component Breakdown Diagram [LAB] Logical Architecture Blank	<b>M&amp;S-LA5</b> [MSM] Modes and States	<b>D-LA6</b> [CDB] Class Diagram	<b>I-LA7</b> [IDB] Interface Definition Blank [CEI] Component External Interfaces [IS] Interface Scenario [CDI] Component Detailed Interface
<b>Physical Architecture</b>	<b>R-PA</b> No dedicated diagram	<b>PA1</b> [CRB] Capabilities Realization Blank [CRI] Contextual Capability Realization Involvement	<b>PA2</b> [FS] Physical Functional Scenario [ES] Physical Entity Scenario [PFCD] Physical Functional Chain Description	<b>PA3</b> [PFBD] Physical Functional Breakdown Diagram [PDFB] Physical Data Flow Blank	<b>PA4</b> [PCBD] Physical Component Breakdown Diagram [PAB] Physical Architecture Blank	<b>M&amp;S-PA5</b> [MSM] Modes and States	<b>D-PA6</b> [CDB] Class Diagram	<b>I-PA7</b> [IDB] Interface Definition Blank [CEI] Component External Interfaces [IS] Interface Scenario [CDI] Component Detailed Interface

Table 3.3: Arcadia diagrams matrix





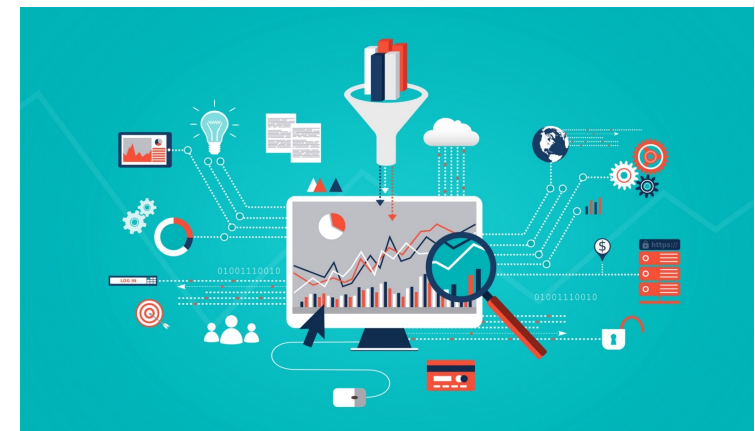
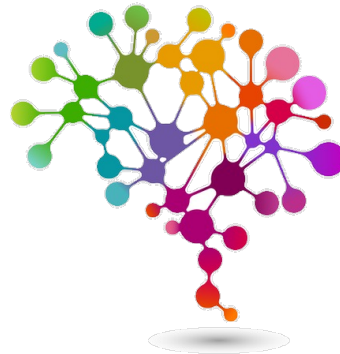
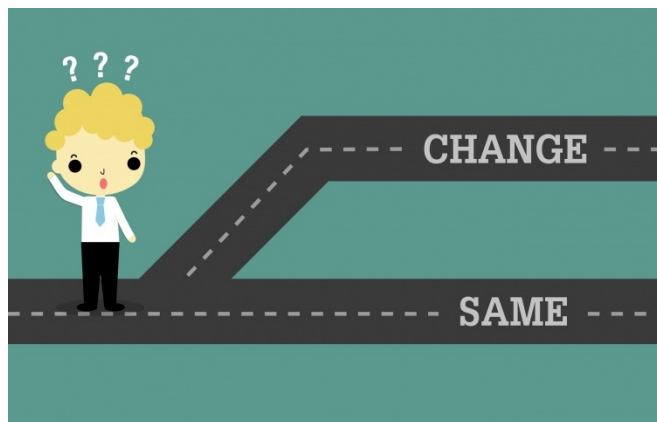
# VOCABULÁRIO DA PA



- Neste nível, os principais conceitos propostos pela Arcádia são semelhantes aos da Arquitetura Conceitual: Função Física, Troca Funcional, Componente Físico, Ator Físico, etc.
- No entanto, existem alguns conceitos adicionais, notadamente:



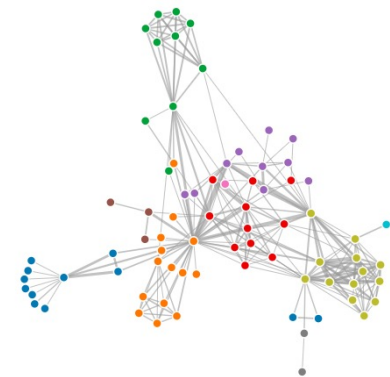
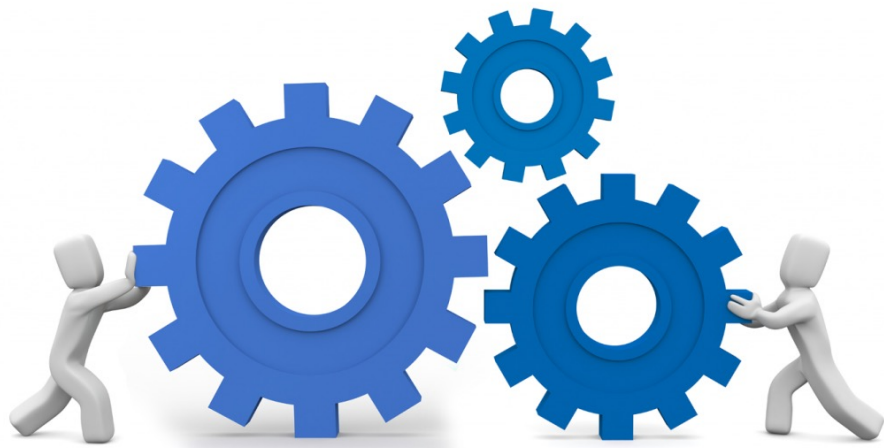
- **Componente Físico do Comportamento (Behavior Physical Component):** **Componente Comportamental encarregado das Funções Físicas** e, portanto, realizar parte do comportamento do Sistema (por exemplo, componente de software, servidor de dados, etc.);







- **Componente físico do nó (ou implementação) (Node (or Implementation) Physical Component):** Componente físico que fornece os recursos materiais necessários para um ou vários componentes de comportamento (por exemplo, processador, roteador, sistema operacional, etc.).



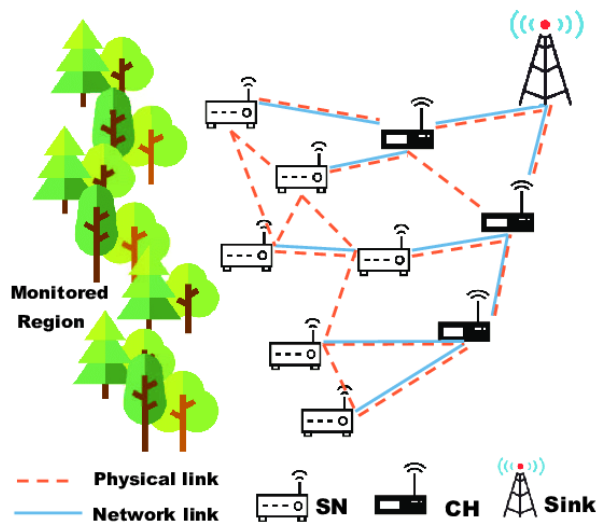


- **Porta Física (Physical Port):** porta não orientada que pertence a um componente de implementação (ou nó).





- **Link Físico (Physical Link):** conexão material não orientada entre os componentes de implementação (ou nós). O Component Exchange permanece uma conexão entre os componentes de comportamento. Um link físico permite que uma ou várias trocas de componentes ocorram (por exemplo, cabo Ethernet, cabo USB, etc.);

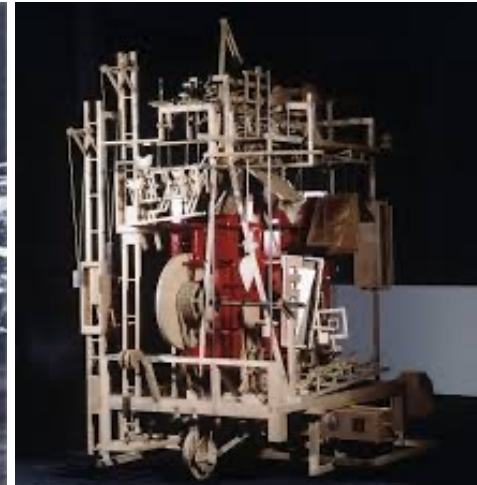
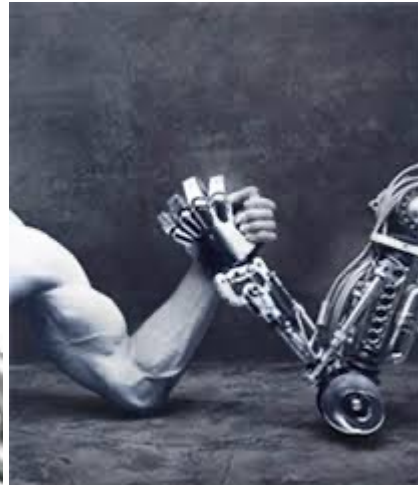


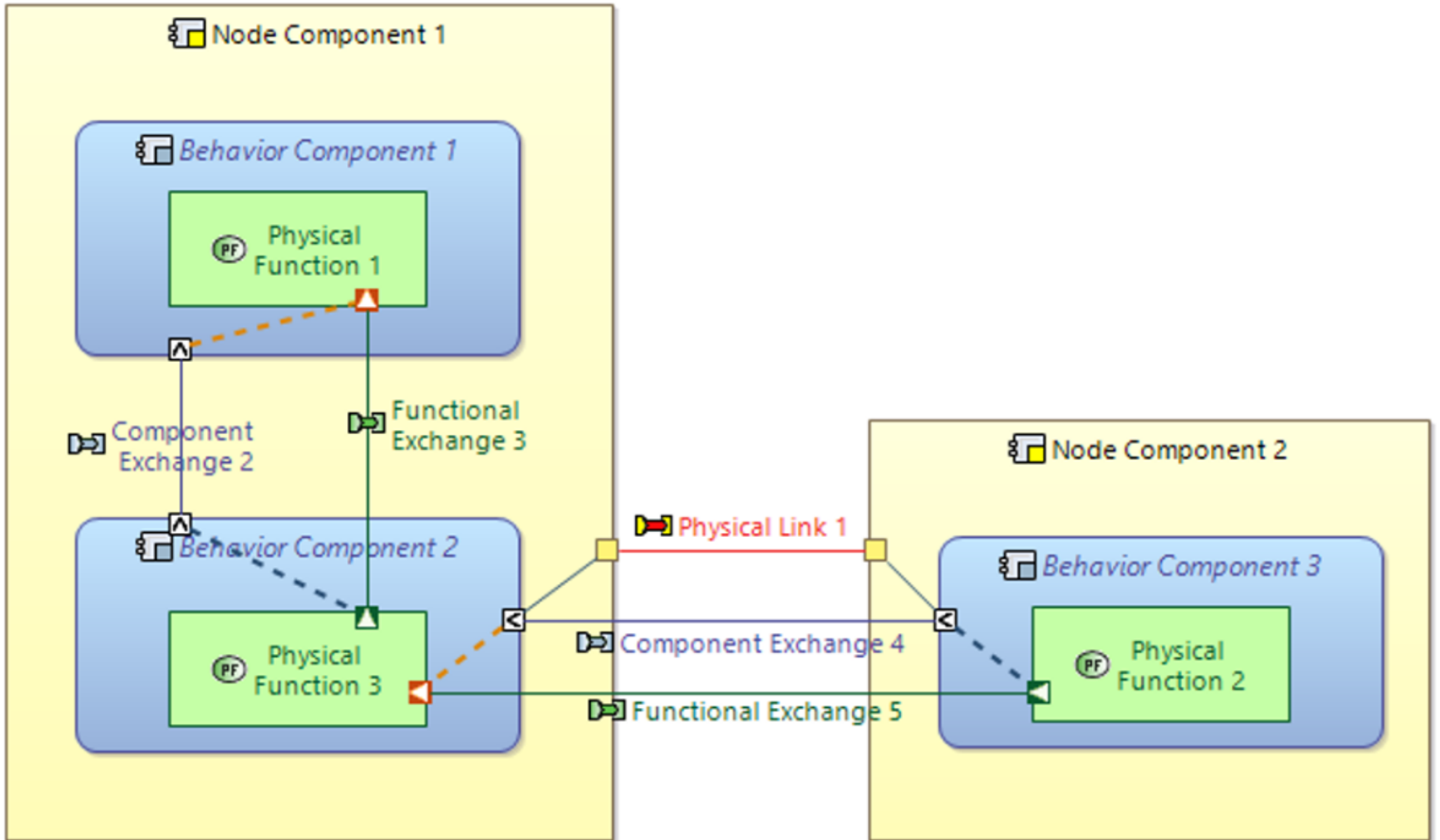
**PHYSICAL LINK  
COMMUNICATIONS**





- **Caminho Físico (Physical Path):** sucessão organizada de Links Físicos, permitindo que um Component Exchange passe por vários Componentes de Implementação (ou Nós).







# DIAGRAMAS



#### ▼ Transition from Logical Functions



[Perform an automated transition of Logical Functions](#)



[Create Traceability Matrix](#)

#### ▼ Refine Physical Functions, describe Functional Exchanges



[\[PFBD\] Create a new Functional Breakdown diagrams](#)



[\[PDFB\] Create a new Functional Dataflow Blank diagram](#)



[\[FS\] Create a new Functional Scenario](#)

#### ▼ Define Physical Components and Actors, Manage deployments



[Perform an automated transition of External Logical Actors](#)



[Perform an automated transition of Logical System](#)



[\[PCBD\] Create a new Physical Component Breakdown diagram](#)



[\[PAB\] Create a new Physical Architecture diagram](#)



[Create a new Physical Component / Logical Component Matrix](#)

Inicialização e atualização automatizada das funções físicas de acordo com as funções lógicas

As ferramentas de transição criam um primeiro mapeamento de rastreabilidade 1-1 entre a Arquitetura Física e a Arquitetura Lógica. Use matrizes de rastreabilidade dedicadas para modificar as relações de rastreabilidade.

Enriqueça e detalhe a quebra funcional com novas funções físicas.

Descrever os fluxos de dados entre funções físicas e identificar cadeias funcionais específicas.

A inicialização e atualização automatizada dos atores físicos pode ser realizada automaticamente de acordo com os atores lógicos.

Defina os componentes físicos. Um componente físico é uma representação física de uma entidade no sistema (hardware, software, firmware, pessoal, instalações, dados, materiais, serviços e processos). Ele é responsável pela implementação de um ou vários componentes lógicos. Um componente físico pode ser Nó ou Comportamento.



#### ▼ Allocate Physical Functions to Physical Components



[\[PAB\] Create a new Physical Architecture diagram](#)



[\[ES\] Create a new Exchange Scenario](#)



[Create a new allocation Physical Component / Physical Function Matrix](#)

#### ▼ Delegate Logical Interfaces and create Physical Interfaces



[\[CII\] Create a new Contextual Internal Interface diagram on the Physical System](#)

#### ▼ Enrich Physical Scenarios



[Perform an automated transition of Logical Architecture Capabilities](#)



[\[IS\] Create a new Interface Scenario](#)

Os componentes físicos comportamentais são responsáveis pela implementação das funções físicas. Gerencie essas alocações usando um diagrama de arquitetura e deduza as trocas de componentes implementando as trocas funcionais.

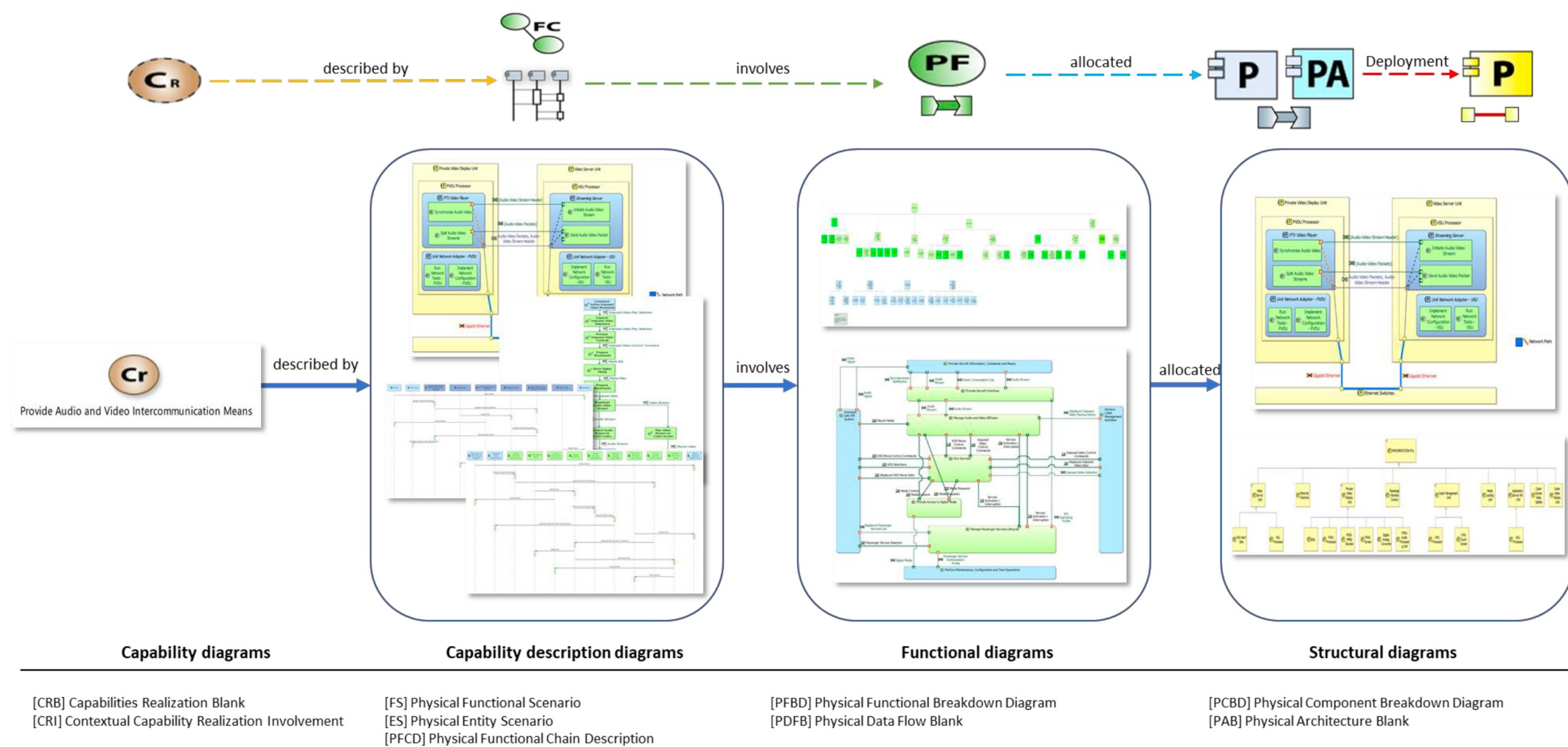
Gerencie a implantação de componentes de comportamento em componentes de nó e deduza links físicos e caminhos. Criar cenários de fluxos de dados para ilustrar trocas funcionais entre os componentes.

Delegue cada interface lógica a um componente físico. Criar novo interfaces físicas entre componentes.

Especifique o comportamento dinâmico dos componentes físicos completando as seqüências de interação provenientes da Arquitetura Lógica. O enriquecimento das seqüências de interação e a identificação das novas interfaces físicas são duas atividades muito estreitas e iterativas.

O processo de refinamento do cenário é iterativo, cada atualização em uma origem pode ser propagada automaticamente para o destino.





Capability diagrams

Capability description diagrams

Functional diagrams

Structural diagrams

[CRB] Capabilities Realization Blank  
[CRI] Contextual Capability Realization Involvement

[FS] Physical Functional Scenario  
[ES] Physical Entity Scenario  
[PFCD] Physical Functional Chain Description

[PFBD] Physical Functional Breakdown Diagram  
[PDFB] Physical Data Flow Blank

[PCBD] Physical Component Breakdown Diagram  
[PAB] Physical Architecture Blank

Figure 7.7: Physical Architecture traceability flow