KEY TAKEAWAYS OF SYSTEMS ARCHITECTURE

Systems Architecture Motivations
Design process of the architecture of a system
Stakeholder analysis
Needs analysis
Lifecycle analysis
Use cases & Operational scenarios analyses

Functional analysis
Functional mode analysis
Functional requirement analysis
Constructional analysis
Constructional requirement analysis
Trade-off analysis
Validation and verification

CESAM/COMMUNITY

3 Main categories of systemic problems

SYSTEMS ARCHITECTURE MOTIVATIONS

CESAM/COMMUNITY

3 Main categories of systemic problems

Modeling problems

Model and reality do not match

Problem type: the system design is based on a model which does not match with reality

Initial choice in design phase with late unexpected consequences

Problem type: the impact of a wrong desing choice appears late in a system life-cycle

3 Main categories of systemic problems

Integration problems

The robustness of a system is destroyed by a "domino effect"

Problem type: a local problem spreads step by step and has global consequences

The system has undesirable emergent properties

Problem type: an integrated system has unexpected or undesired emerging properties

CESAM/COMMUNITY

3 Main categories of systemic problems

Project problems

The project system has integration issues

Problem type: the engineering of the system is not done in a collaborative way

The mission of the product is diverted by the project system

Problem type: the project forgets the mission of the product and "indulges" itself

CESAM/COMMUNITY

KEY CONCEPTS OF SYSTEM ARCHITECTURE

SYSTEMS ARCHITECTURE MOTIVATIONS

CESAM/COMMUNITY

Key concepts

A **system** is a set of interrelated **components** (covering hardware, software and humanware) working together toward some **common mission**.

Every product or system that we develop is always used as part of a **larger system**.

Every project can benefit from **good systems architecture.**
Systems architecture is **not just for large** complex "solution" projects.

A mouse is a **system**

... so is an **aircraft!**

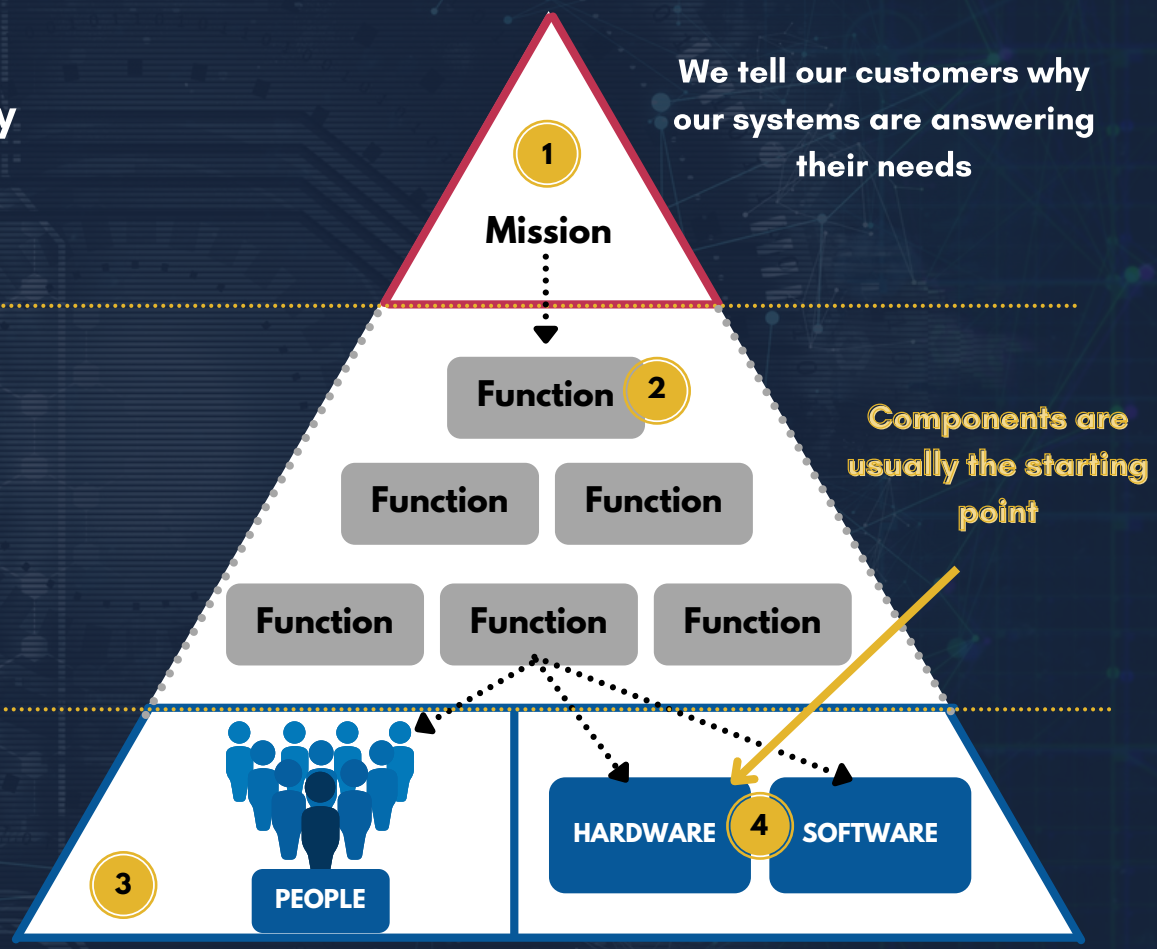
Key principle

Evolving from a technology-oriented or bottom-up approach...

... and ends with customers, **hopefully satisfied** by the system that we are proposing them

... which have certain functional behaviors ...

System design is traditionally a **bottom-up** process that starts with technology and possibly people ...



CESAM/COMMUNITY

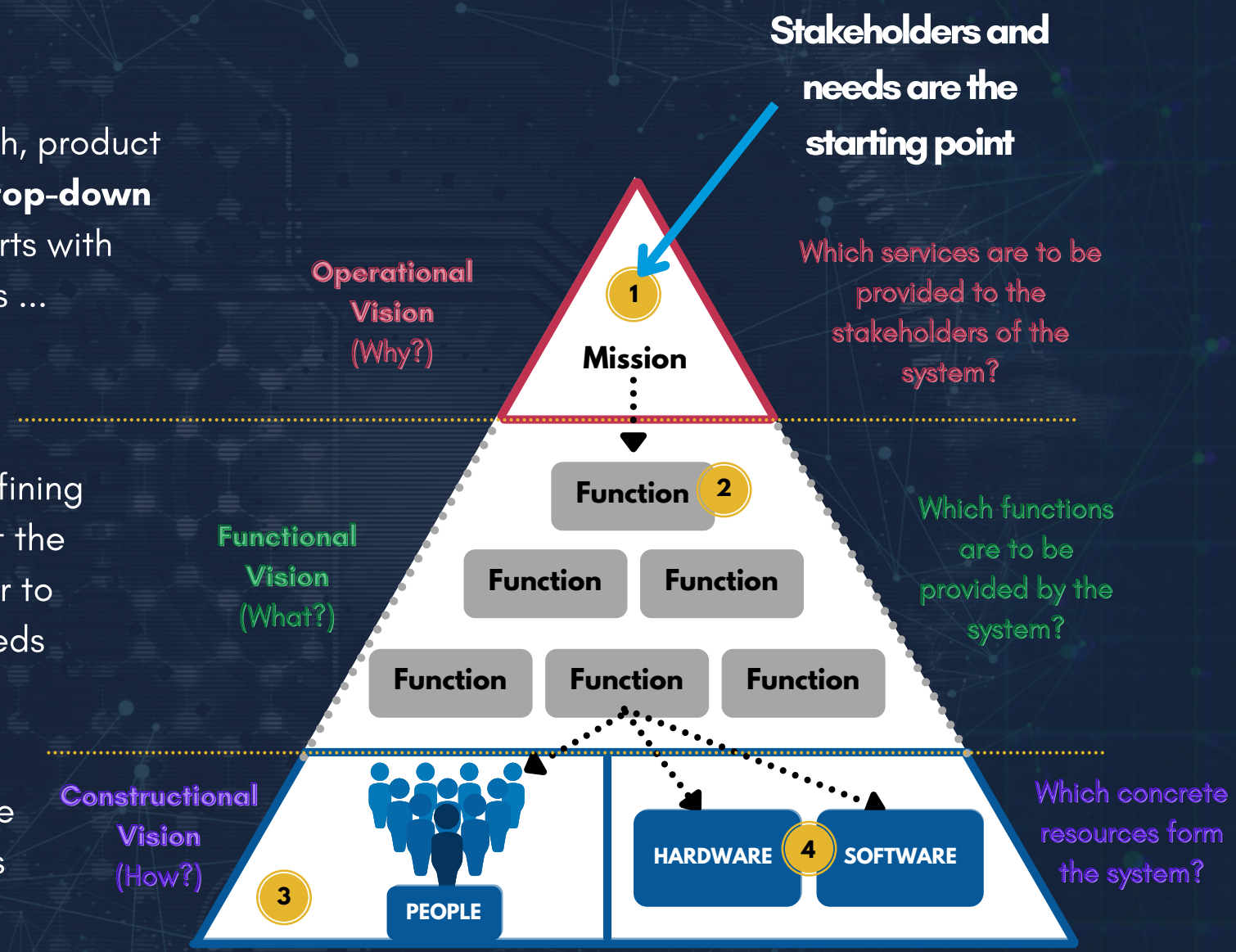
Key principle

... To a customer-focused and top-down system strategy

In a MBSE approach, product design is rather a **top-down** process that starts with stakeholders ...

... continues by defining the functions that the system shall offer to answer their needs

... and ends with the concrete resources that **effictively answer their needs**



CESAM/COMMUNITY

Design process of the architecture of a system

External constraints

Needs (often informal)

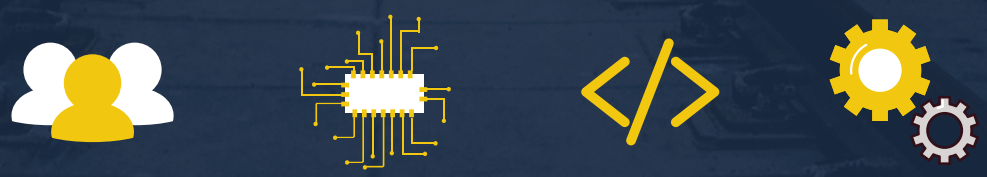
Expected features of the system described "in extension" (use cases, examples and models) using the language(s) of the "clients"

SYSTEMS ARCHITECTURE PROCESS

Architected technical solution

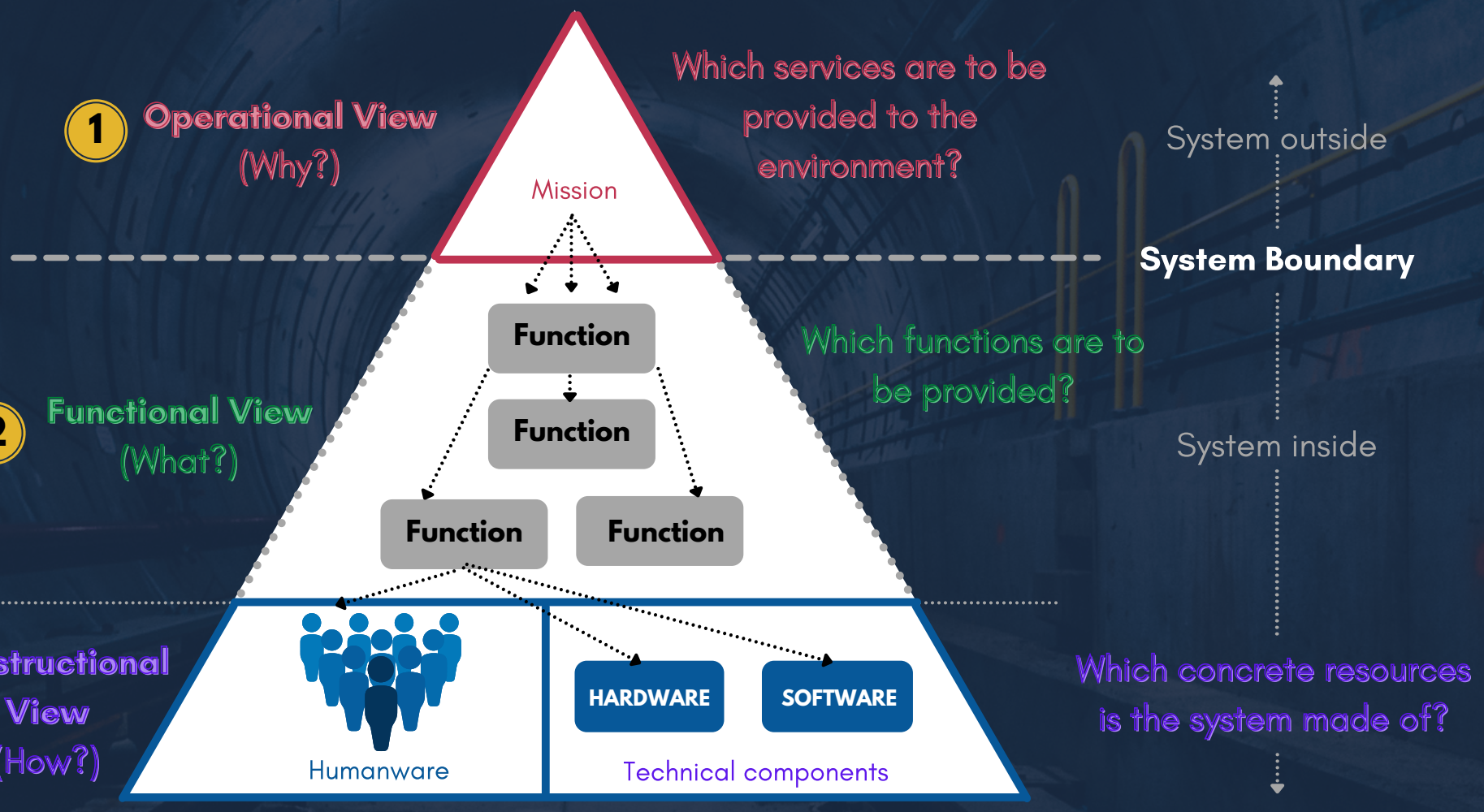
Actual system described "in extension" (informal, quasi-formal or formal models) using descriptive or executable languages

Resources of the architectural process



CESAM/COMMUNITY

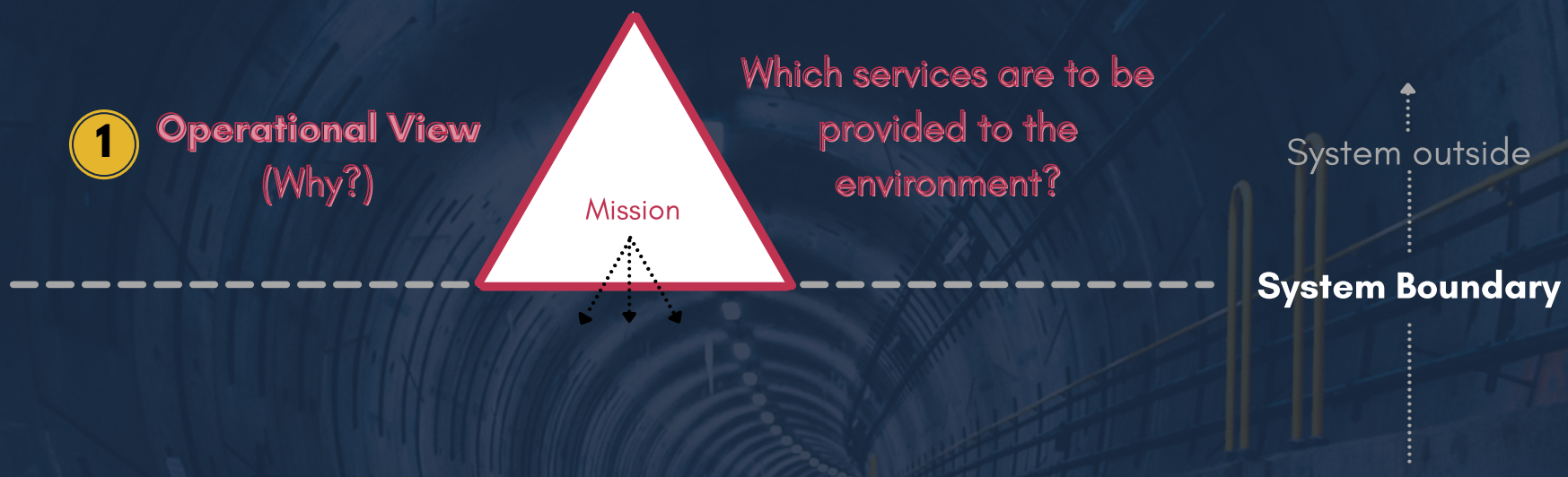
Systems Architecture framework



Architectural view pyramid

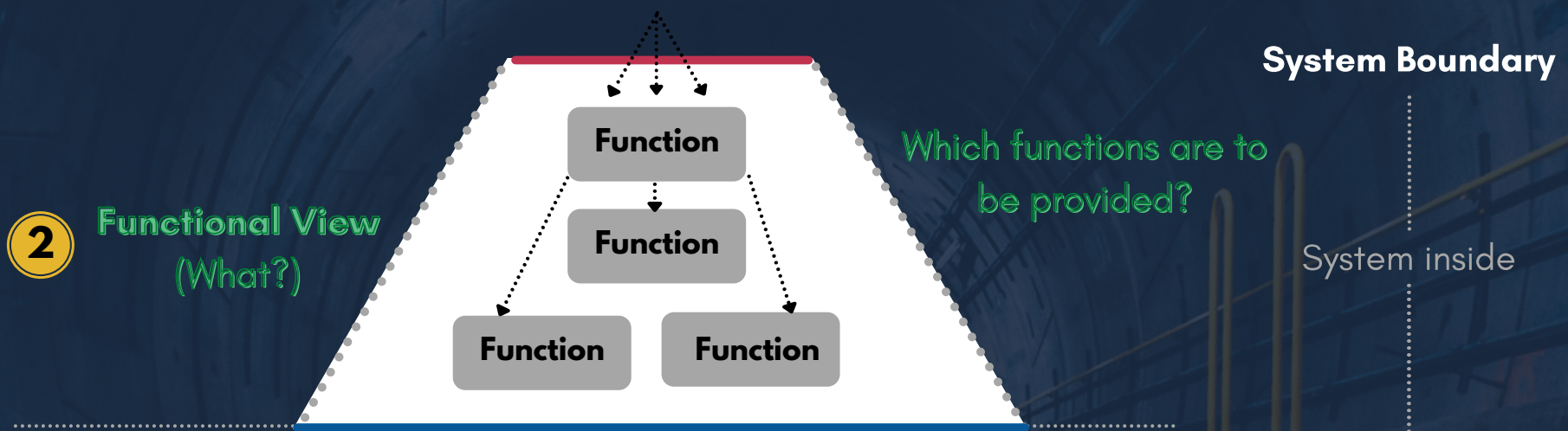
CESAM/COMMUNITY

Operational vision



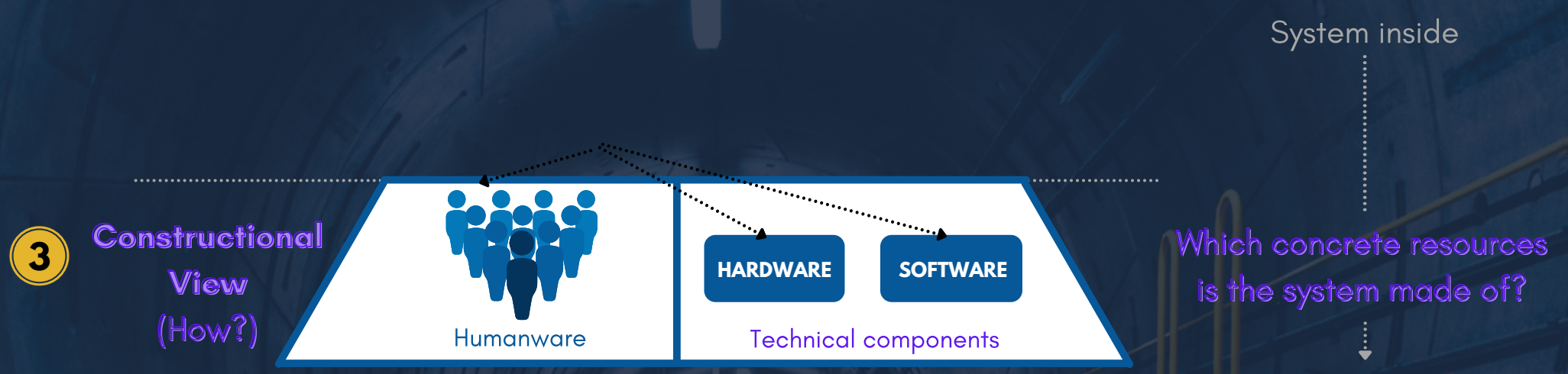
The **operational vision** of a system defines the **mission** of the system, analyzed here as a **black box** from the **external perspective** of the system **stakeholders**.

Functional vision



The **functional vision** of a system defines the **abstract functions** of the system, analyzed as a **grey box**, that are required to **deliver the system mission**.

Constructional vision



The **constructional vision** of a system defines the **concrete components & building blocks** of the system, analyzed as a **white box**, that **implement the functions** of the system.

EXTERNAL SYSTEM & STAKEHOLDER

Example of an electric toothbrush

An **external system** is an **external body that influences or interacts with the considered system**.

An external system:

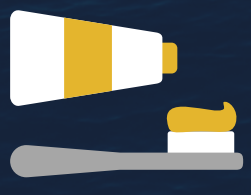
- is not necessarily a person,
- affects or is affected by the system (directly or indirectly),
- may come or not into contact with the system,
- has needs or imposes constraints relatively to the system

A stakeholder is a person who represents an External System.

World Health Organization

Conformity assessment body

Dentist



Toothpaste

Electronic toothbrush

Bathroom

User

CESAM/COMMUNITY

STAKEHOLDER ANALYSIS – PROCESS

Step 1

Generate a list with all possible stakeholders

Maximal systemic perimeter

Step 2

Identify stakeholders

List of stakeholders
(non relevant stakeholders removed)

Step 3

Organize and group

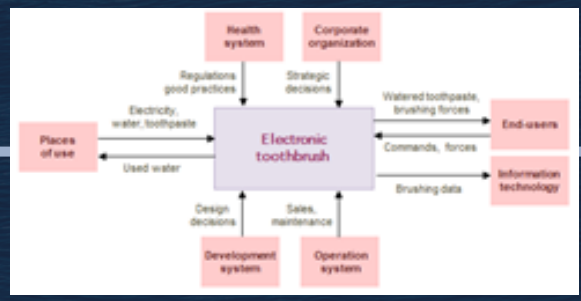
Stakeholder hierarchy

CESAM/COMMUNITY

STAKEHOLDER ANALYSIS – PROCESS

Step 4

Synthesize and describe external flows



Environment diagram

Step 5

Specify external interfaces

Interaction name	Interacting systems		Interface direction	Types of exchanged flows	Exchanged flows	Interface requirements
	Source	Destination				
Brushing interface	Electronic toothbrush	End-user and toothpaste	Bi-directional	Mechanical & chemical interactions	Toothpaste & friction forces	End-user teeth shall be in contact with toothbrush hairs covered by watered toothpaste

External interface specifications

Stakeholder analysis is the process that constructs the knowledge of the stakeholders and the environment of a system.

CESAM/COMMUNITY

NEEDS ANALYSIS

Definition, Process & Deliverables

Purpose:

Express in an unambiguous, measurable and testable way the expectations of all external systems and characterize their expected level of performance.
Needs are like a contract performance with all stakeholders for the system of interest.

CESAM/COMMUNITY

A **need** relative to a **system** is a **feature, expected** or **imposed** by **one or more stakeholders** of its environment that has an impact on the system of interest and that is necessary to respect to be accepted by the stakeholders.

The end-user expects a handy tool to brush his teeth

A stakeholder:
the end-user

Electronic toothbrush

Places of use have a high humidity rate

A stakeholder: the
places of use

Dentists want to improve the dental hygiene of their patients

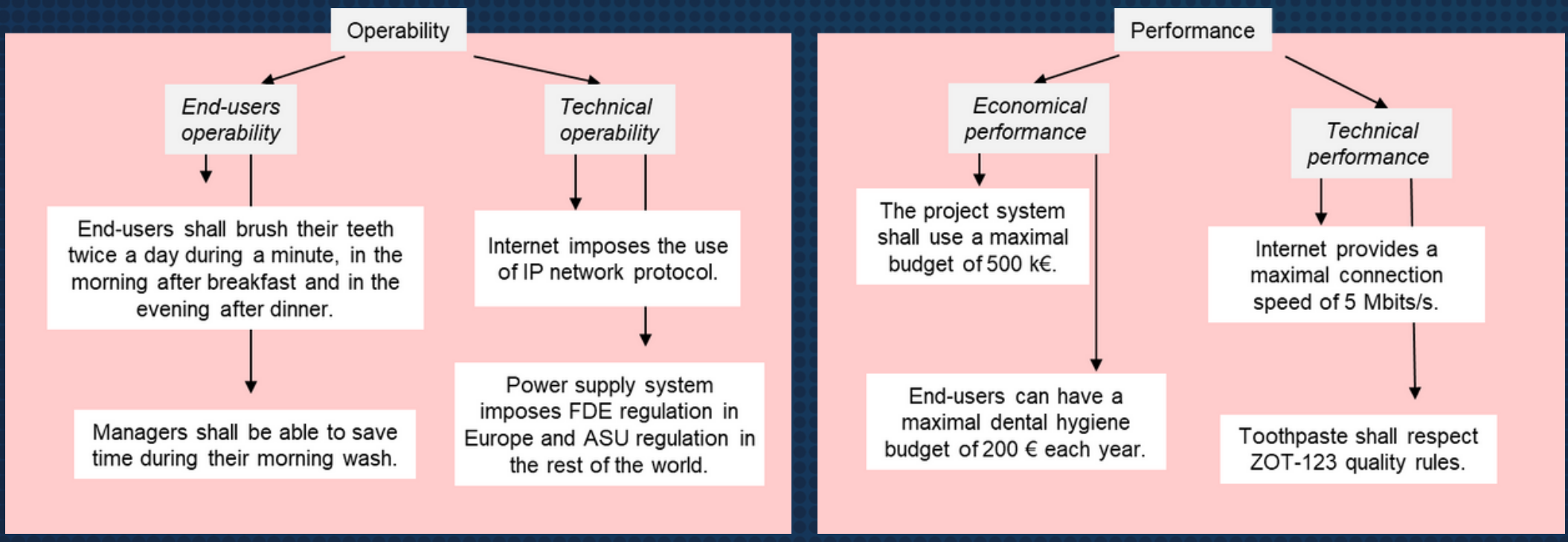
A stakeholder:
the dentist

The toothpaste respects the European healthcare policy

A stakeholder:
the toothpaste

CESAM/COMMUNITY

The key deliverable of the **needs analysis** process is the **need breakdown**.

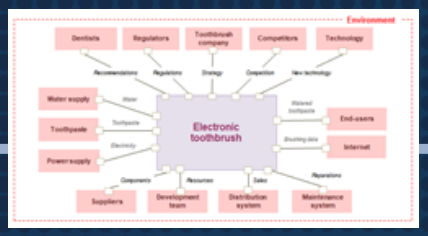


Need breakdown

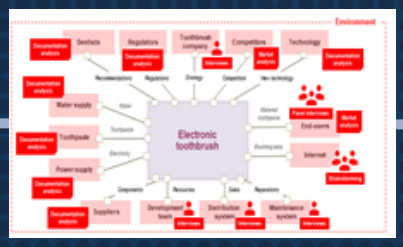
NEEDS ANALYSIS – PROCESS

Step 1

Identify “raw” needs capture methods



Environment diagram



“Raw” needs capture methods

Step 2

Capture elementary needs

End-users shall **brush their teeth twice a day during a minute, in the morning after breakfast and in the evening after dinner.**

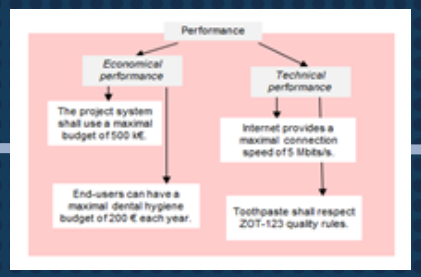
Formalized elementary needs

CESAM/COMMUNITY

NEEDS ANALYSIS – PROCESS

Step 3

Classify needs into homogeneous subsets



Homogeneous sets of needs

Step 4

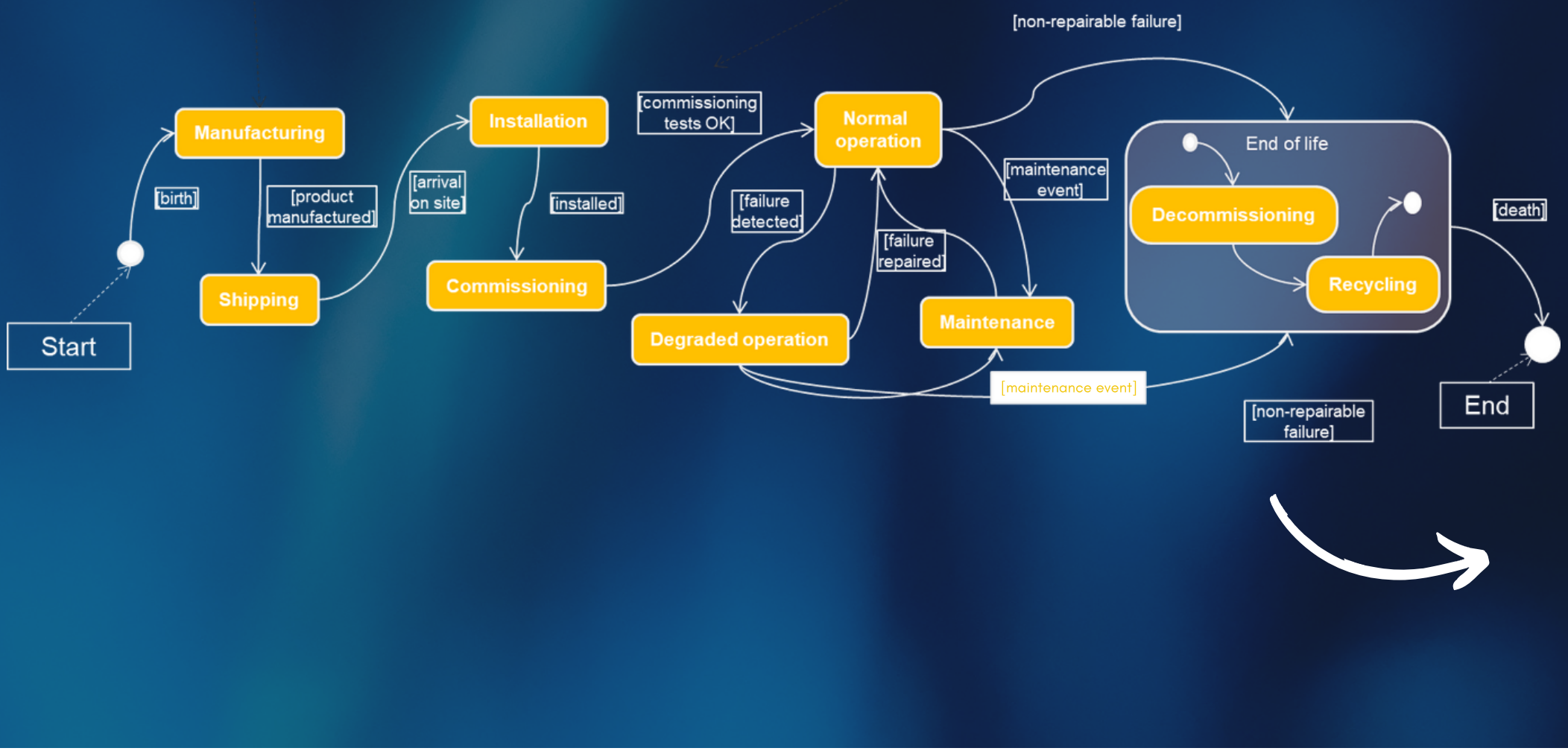
Verify coherence of needs breakdown

Needs breakdown

CESAM/COMMUNITY

LIFECYCLE ANALYSIS

Definition, Process & Deliverable



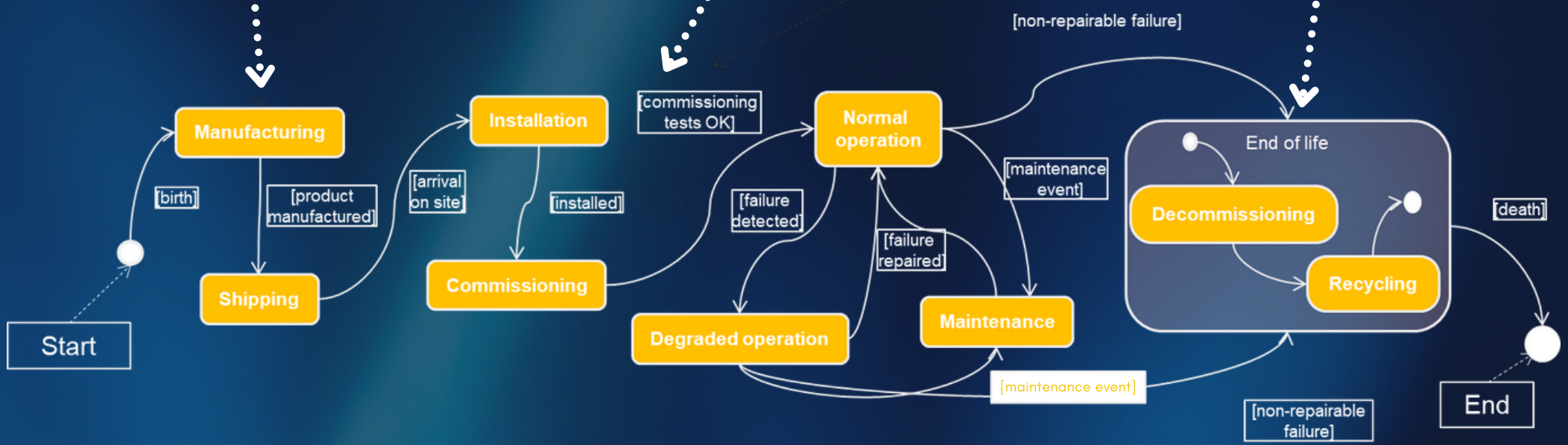
CESAM/COMMUNITY

LIFECYCLE ANALYSIS – DEFINITION

This is a lifecycle phase, that is to say a period of time (duration > 0) during which the system environment has some stability

This is a transition, that is to say an event (duration = 0) that makes switch from one lifecycle phase to another, and marks a major change in the system environment

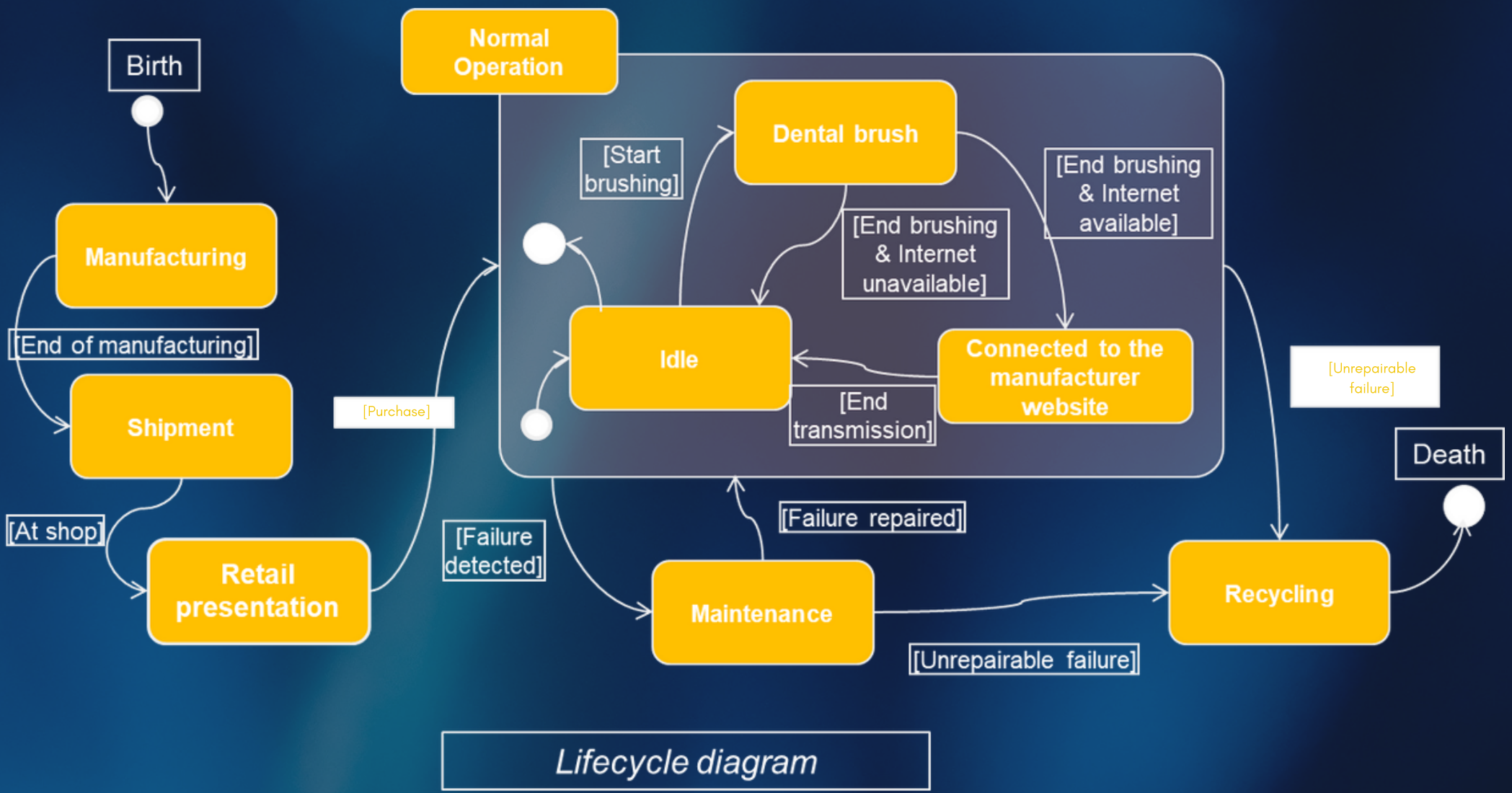
A lifecycle phase can contain several sub phases



A **lifecycle phase** of a system is a **homogeneous period of time** from the perspective of the stakeholders of the system. Its **lifecycle** models the **succession of all lifecycle phases** and the **transitions** between lifecycle phases among time, from birth to death of the system.

CESAM/COMMUNITY

LIFECYCLE ANALYSIS – DELIVERABLE



Lifecycle diagram

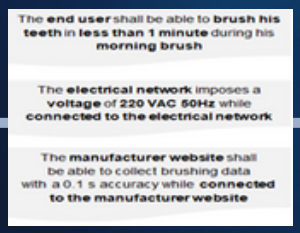
The key deliverable of the lifecycle analysis process is the lifecycle diagram.

CESAM/COMMUNITY

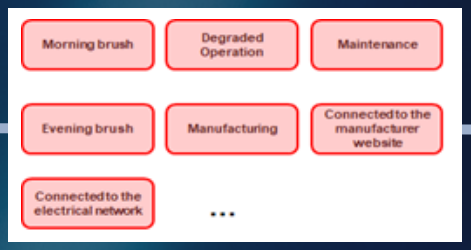
LIFECYCLE ANALYSIS – PROCESS

Step 1

Identify lifecycle phases



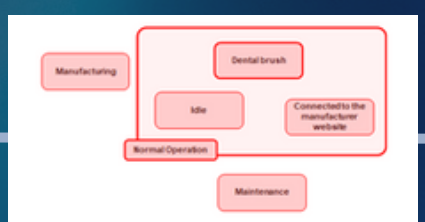
Need breakdown



Flat list of lifecycle phases

Step 2

Organize lifecycle phases



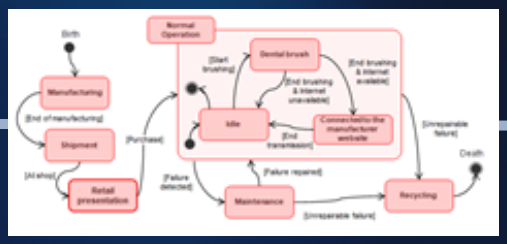
Organized lifecycle phases

CESAM/COMMUNITY

LIFECYCLE ANALYSIS – PROCESS

Step 3

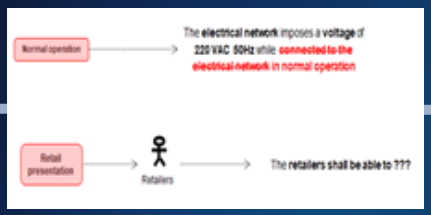
Identify transitions



Lifecycle diagram

Step 4

Align with need and stakeholder analyses

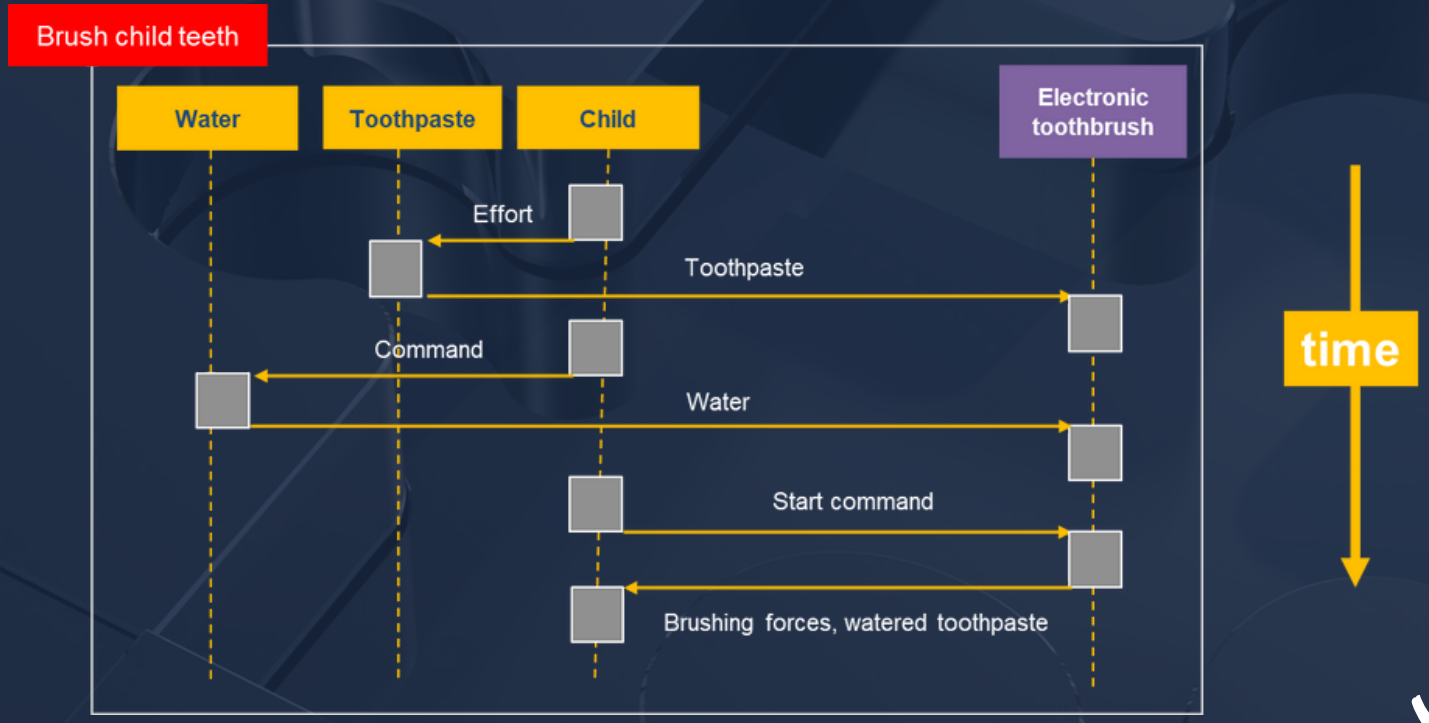


Aligned need and stakeholder analyses

CESAM/COMMUNITY

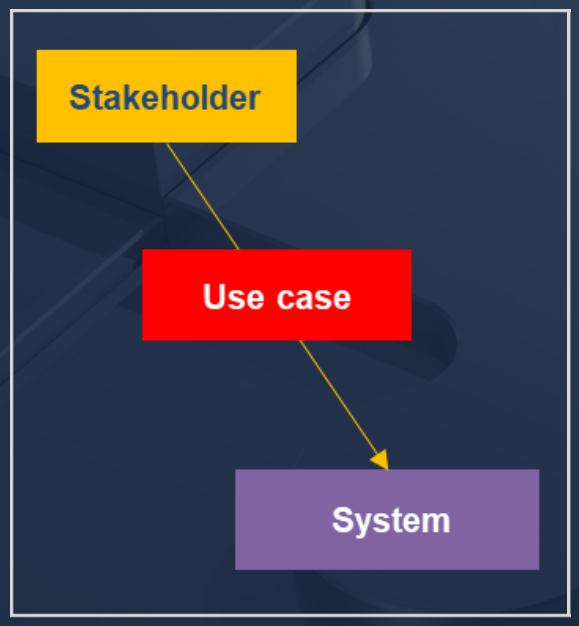
USE CASES & OPERATIONAL SCENARIOS ANALYSES

Purpose : Understand how the system will be used by its stakeholders and interact with its stakeholders.



CESAM/COMMUNITY

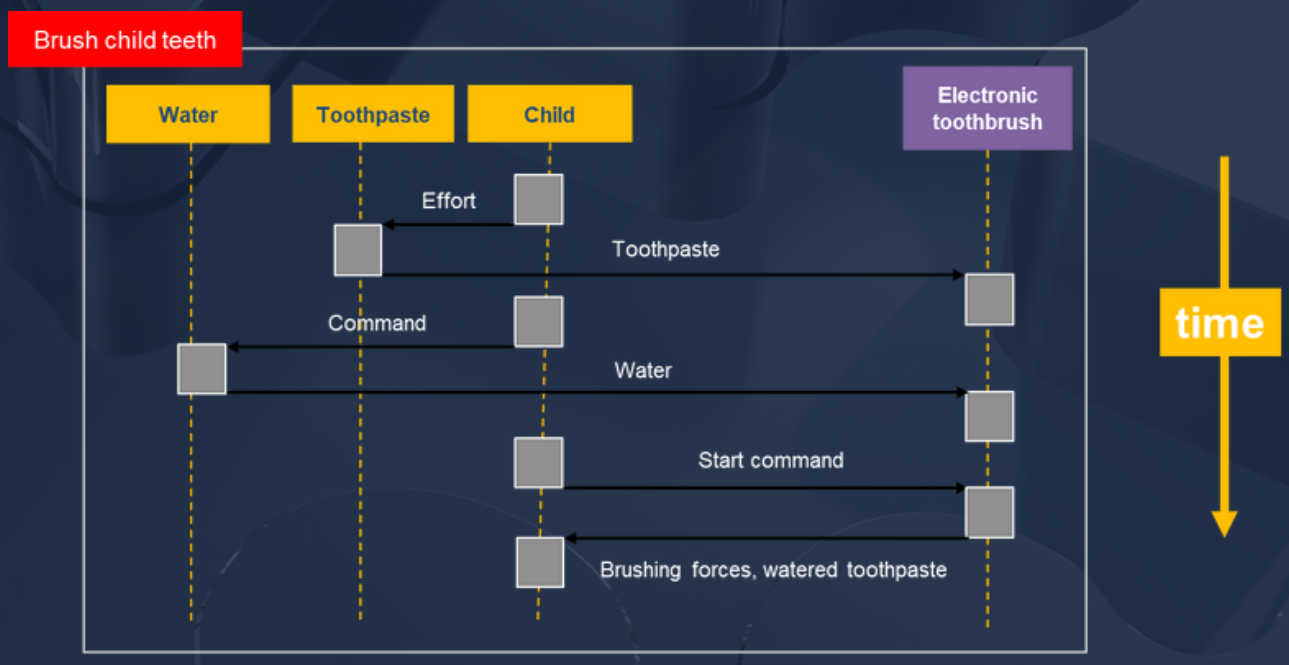
KEY CONCEPTS



Use case definition

A **use case** describes an action that can be performed by one or several stakeholders when using the system.

A use case can be described through an **operational scenario** which specifies – using a sequence diagram – the sequence of activities and the external exchanges that take place between the system of interest, considered here as a black box, and the stakeholders during the considered use case.

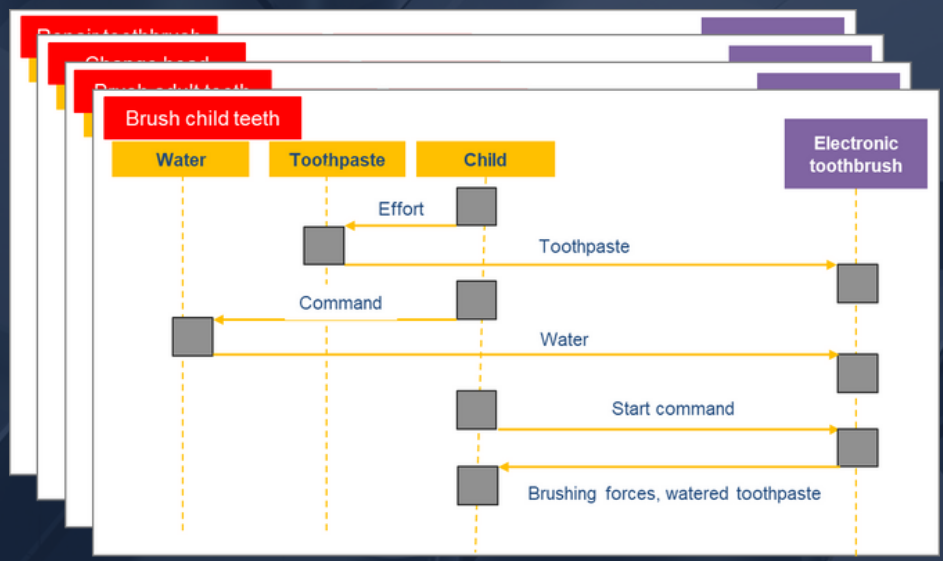


Operational scenario associated with the "Brush child teeth" use case

CESAM/COMMUNITY

DELIVERABLES

Lifecycle phases	Use cases	Involved stakeholders
Retail presentation	UC1: Identify product	Retailers
Idle	UC2: Recharge toothbrush	Bathroom, electrical system
	UC3: Change head	Bathroom, end-users
Dental brush	UC4: Brush adult teeth	Bathroom, end-users, water, toothpaste
	UC5: Brush child teeth	Bathroom, end-users, water, toothpaste
Connected to manufacturer website	UC6: Retrieve brushing data	Internet, manufacturer website, dentists
Maintenance	UC7: Repair toothbrush	Maintenance system, end-users
	UC8: Locate & identify failure	Maintenance system
	UC9: Dismount system	Maintenance system, end-users



The first key deliverable of the use case analysis process is the **operational breakdown structure** of the system of interest where all its use cases are classified according to its lifecycle phases.

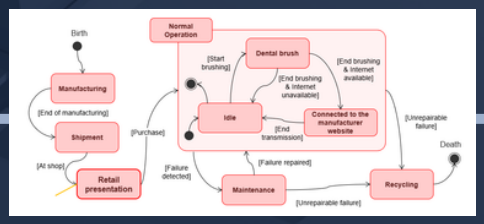
The second key deliverables of the use case analysis process are the **operational scenarios** which are associated with each use case of the system of interest as listed in the operational breakdown structure.

CESAM/COMMUNITY

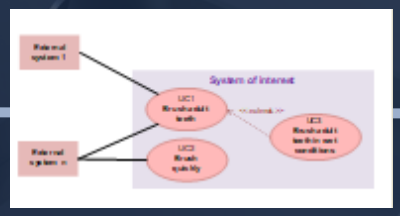
PROCESS

Step 1

Identify all possible use cases



Lifecycle



Use cases

Step 2

Classify use cases

Lifecycle phases	Use cases	Involved stakeholders
Retail presentation	UC1: Identify product	Retailers
Idle	UC2: Recharge toothbrush	Bathroom, electrical system
	UC3: Change head	Bathroom, end-users
Dental brush	UC4: Brush adult teeth	Bathroom, end-users, water, toothpaste
	UC5: Brush child teeth	Bathroom, end-users, water, toothpaste
Connected to manufacturer website	UC6: Retrieve brushing data	Internet, manufacturer website, dentists
Maintenance	UC7: Repair toothbrush	Maintenance system, end-users
	UC8: Locate & identify failure	Maintenance system
	UC9: Dismount system	Maintenance system, end-users

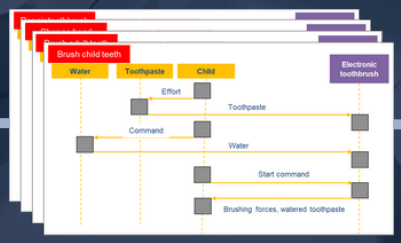
Operational breakdown structure

CESAM/COMMUNITY

PROCESS

Step 3

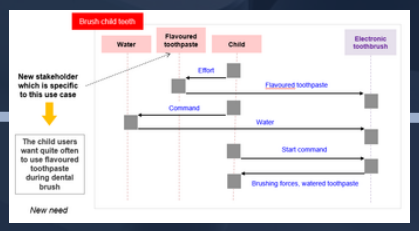
Define operational scenarios



Operational scenarios

Step 4

Align with stakeholder and needs analyses

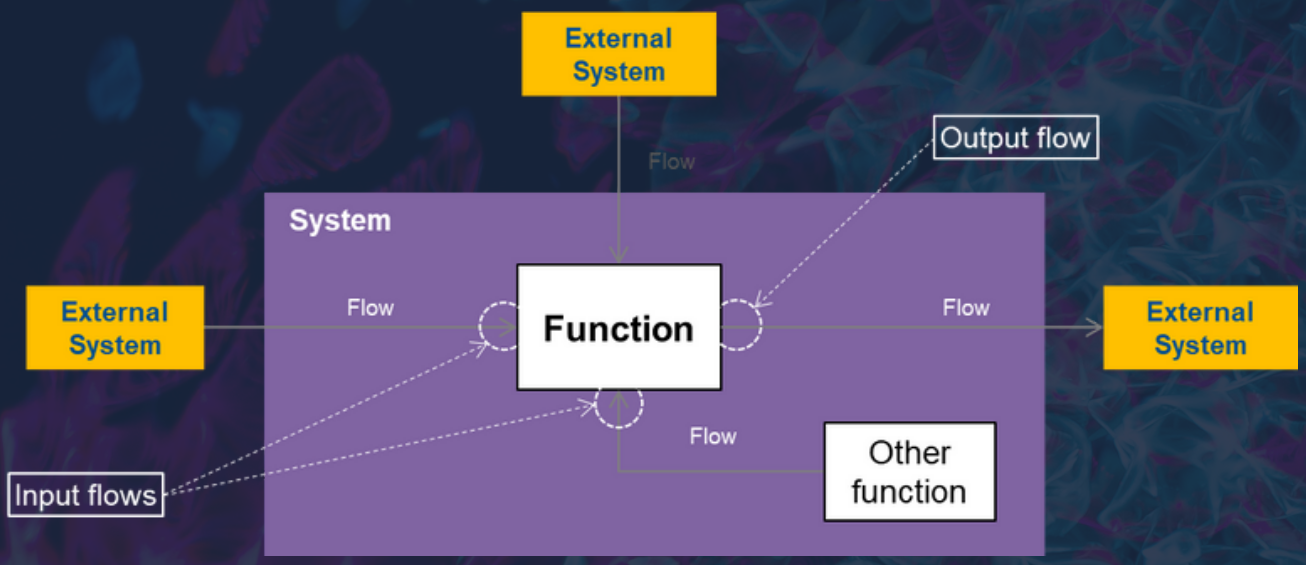


Aligned need and stakeholder analyses

CESAM/COMMUNITY

FUNCTIONAL ANALYSIS

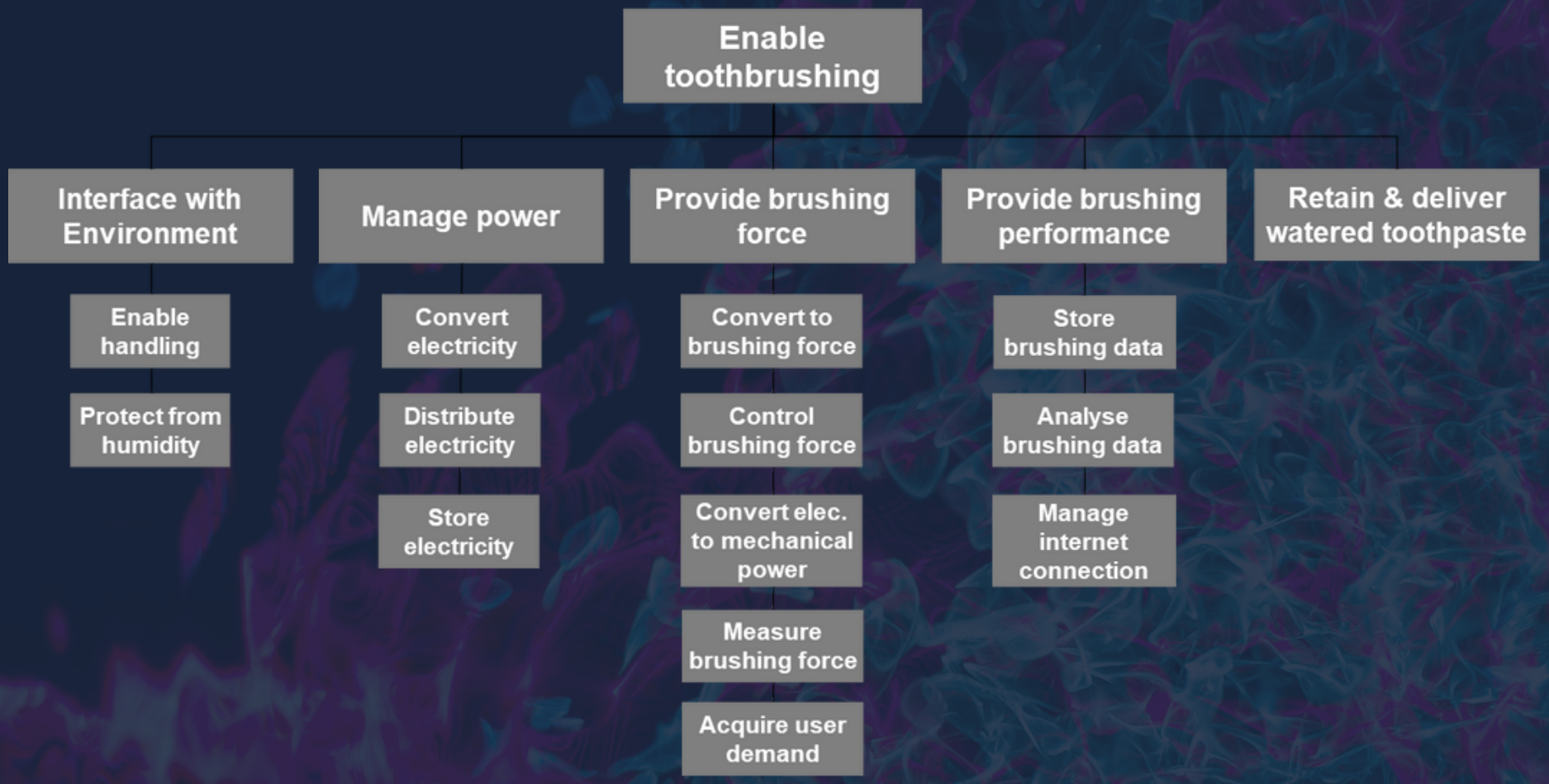
Purpose : Have a comprehensive view of the behaviour of a system.



A **function** describes a **transformation** performed by the system between its **input and output flows** in order to provide an **adequate answer** to use cases

- **Input flows** can come from external systems or other functions of the system
- **Output flows** can go to external systems or other functions of the system

DELIVERABLES

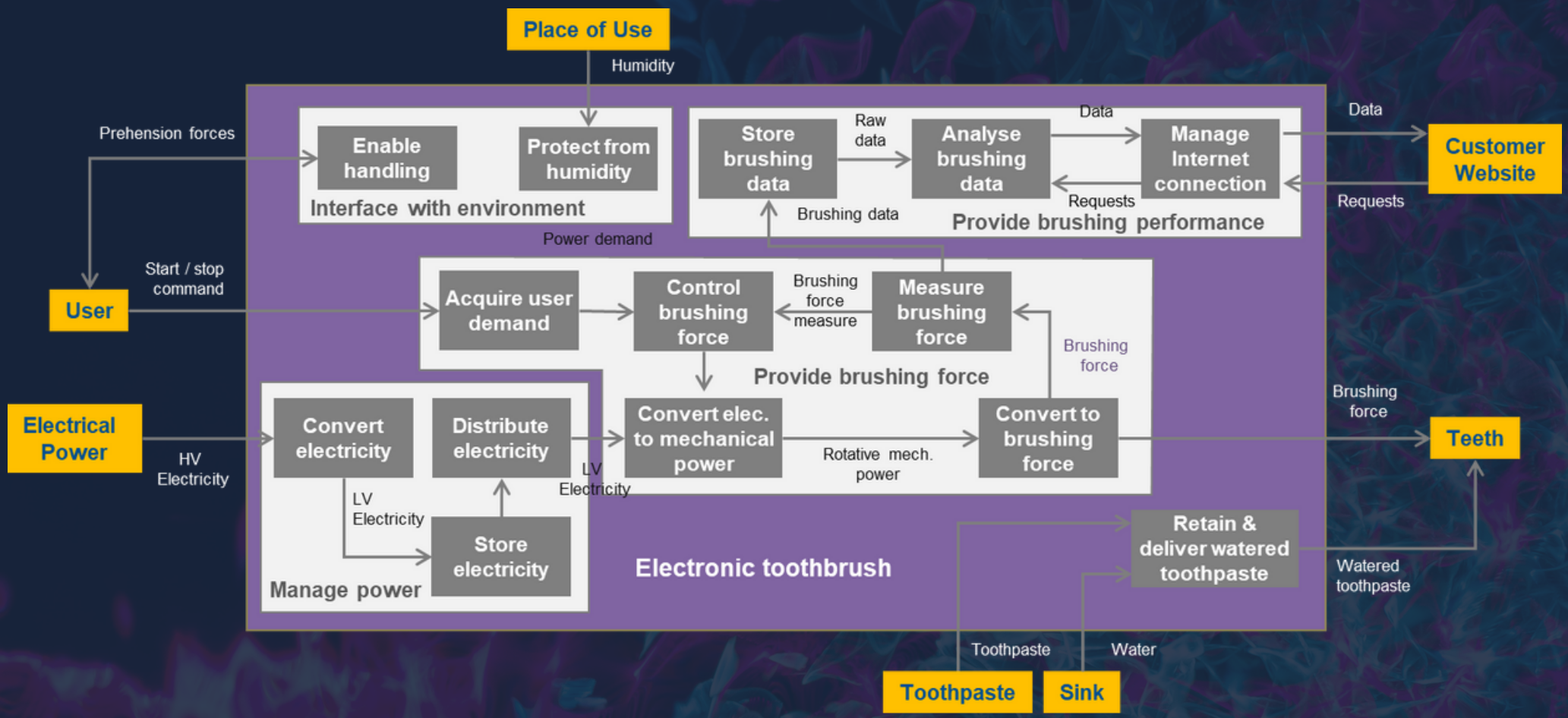


Functional Breakdown Structure

The **functional breakdown structure** of a system is a tree-like diagram that represents the breakdown structure of the functions of the system.

CESAM/COMMUNITY

DELIVERABLES



Functional Interaction Diagram

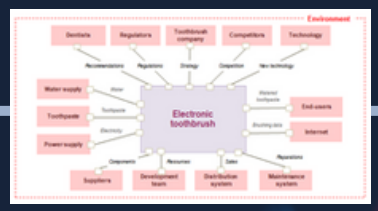
The **functional interaction diagram** represents the **functions** of a system and the **functional flows** exchanged between the system functions and with the stakeholders.

CESAM/COMMUNITY

PROCESS

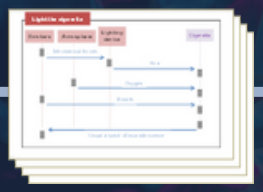
Step 1

Identify functions



Environment diagram

Use cases

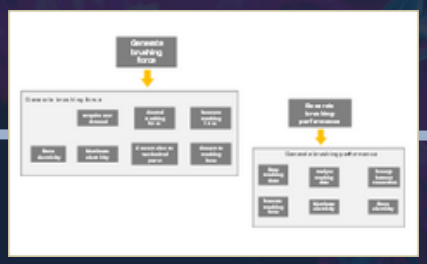


Operational scenarios

System level functions

Step 2

Identify low-level functions



Low level functions

CESAM/COMMUNITY

PROCESS

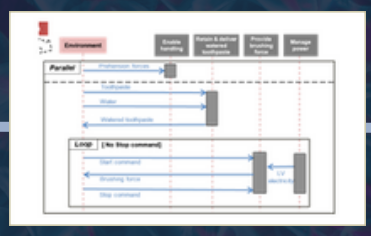
Step 3

Organize Functions hierarchically

Functional breakdown structure

Step 4

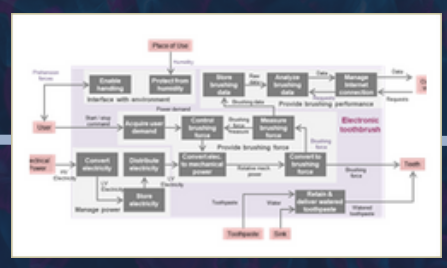
Analyse functional scenarios



Functional scenarios

Step 5

Describe interactions between functions

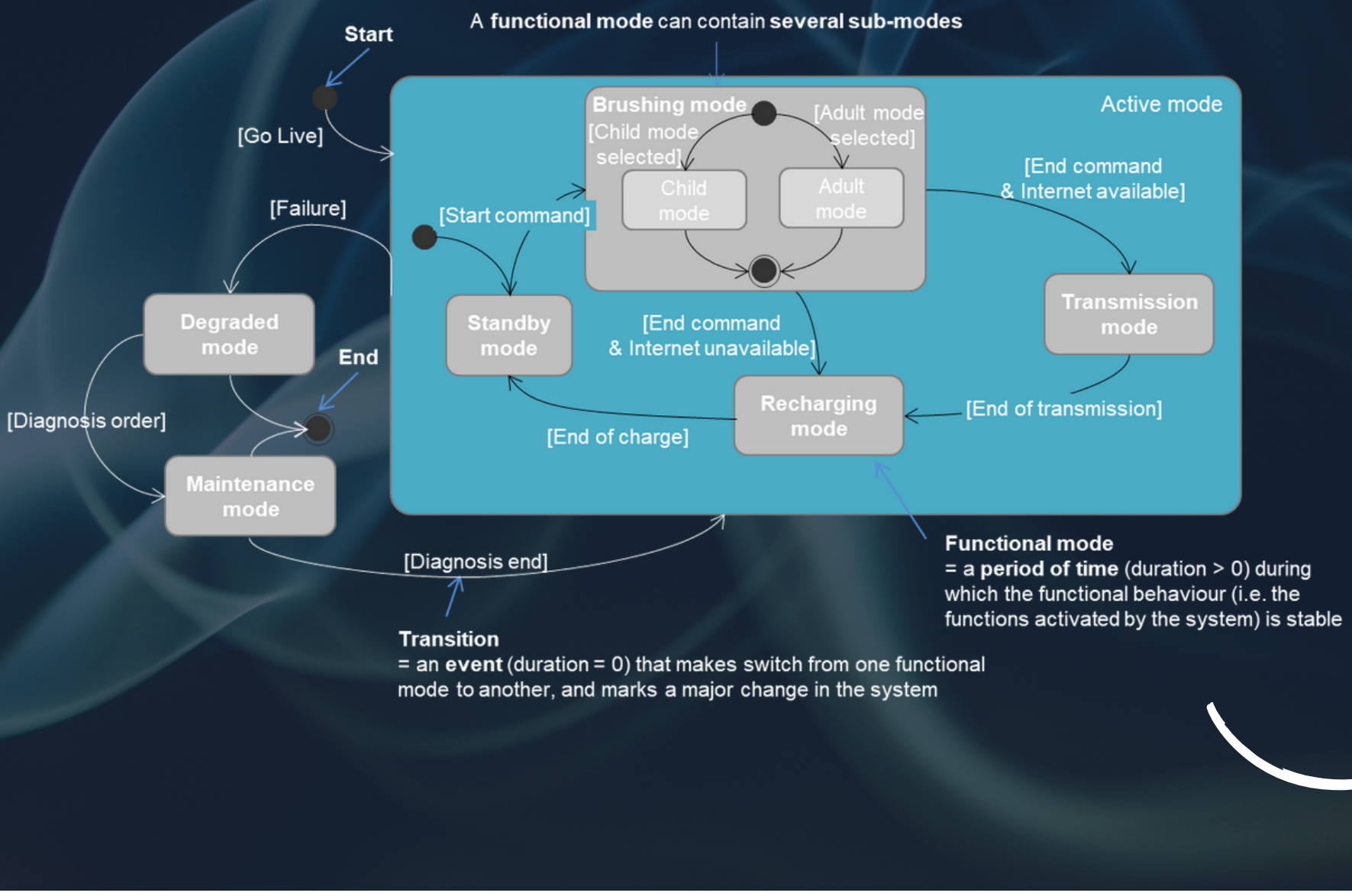


Functional interactions

CESAM/COMMUNITY

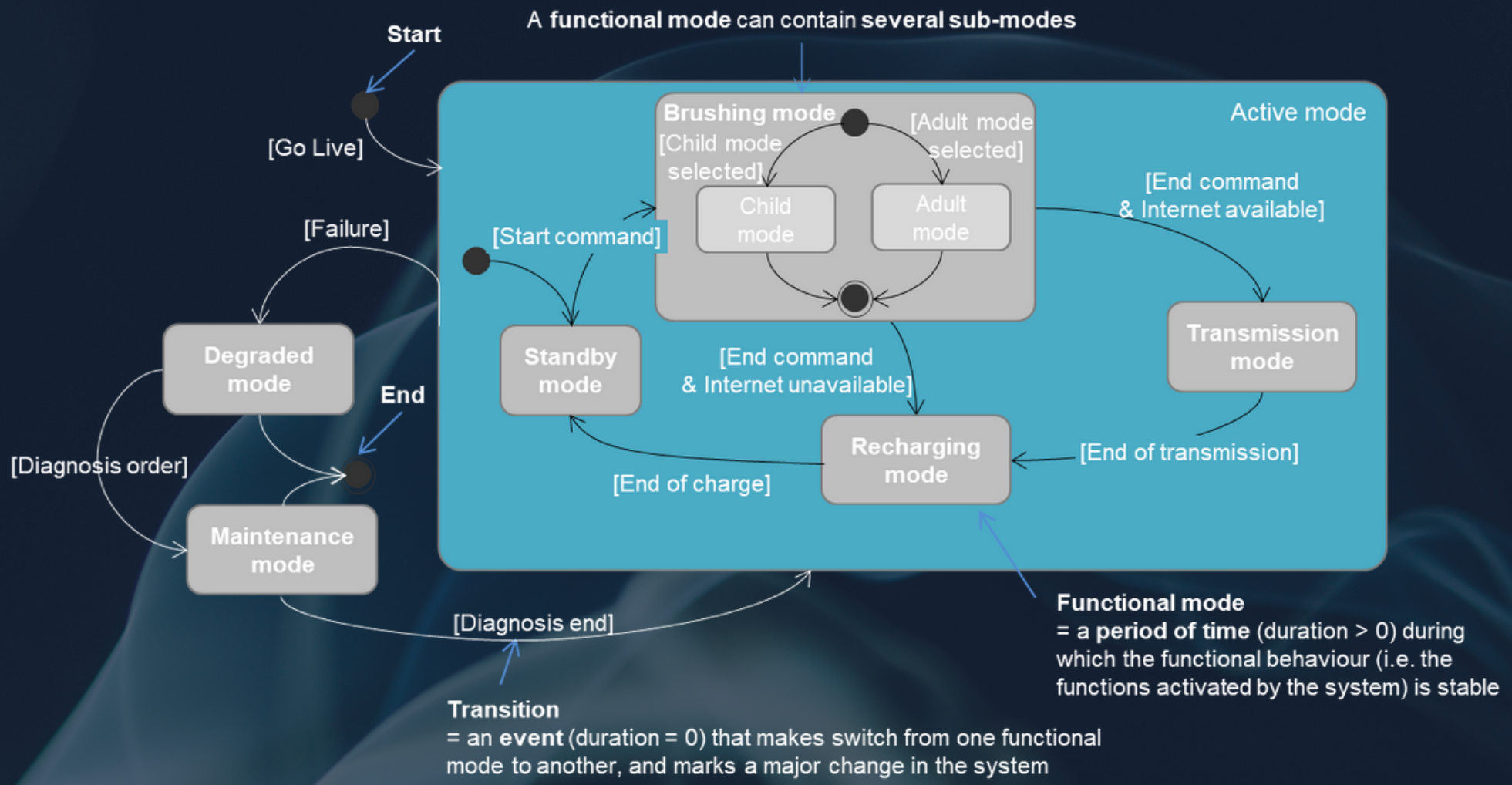
FUNCTIONAL MODE ANALYSIS

Purpose : Understand the evolutions of what the system should do with time.



CESAM/COMMUNITY

KEY CONCEPTS

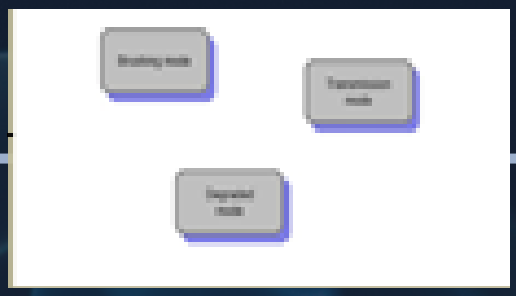


A **functional mode** of a system is a **functionally coherent period of life** of the system, i.e. a period of time which is characterized in an unambiguous way by the set of functions that the system is using during it.

PROCESS

Step 1

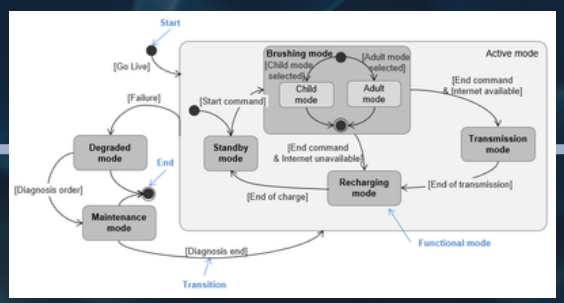
Identify stable functionnal behaviours



Functional modes

Step 2

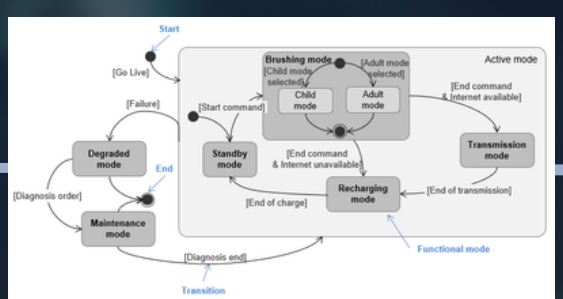
Refine and identify transitions



Functional modes and transitions

Step 3

Align with lifecycle phases



Functional mode diagram

CESAM/COMMUNITY

FUNCTIONAL REQUIREMENT ANALYSIS

Template, Process & Deliverable

Purpose:

Express in an unambiguous, measurable and testable way how the expected functions of the system answer to stakeholders' needs and characterize the level of performances of these functions.

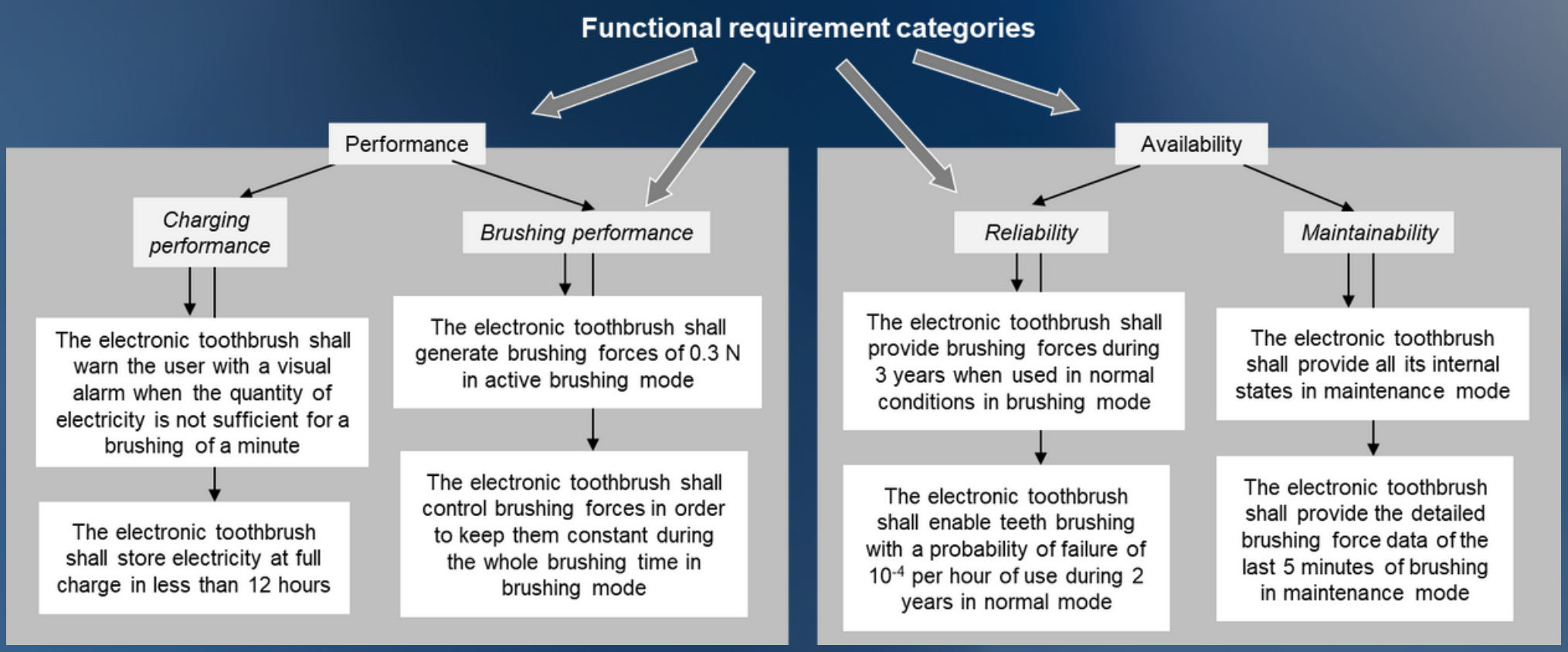
CESAM/COMMUNITY

TEMPLATE

Functional Requirement			
Domain	Main category to which the requirement belongs	**Reference**	A unique code for the Requirement
Statement			
Functional requirement pattern to respect The <**SYSTEM**> (who) shall <**DO SOMETHING**> (what) with an <**EXPECTED LEVEL OF PERFORMANCE**> (how much) in a <**GIVEN FUNCTIONAL MODE**> (when and/or where).			
Satisfaction criteria			
How does one measure and quantify that the functional requirement is really fulfilled?			

Remember the deliverable is a list of quantified functional requirement statements.

DELIVERABLE



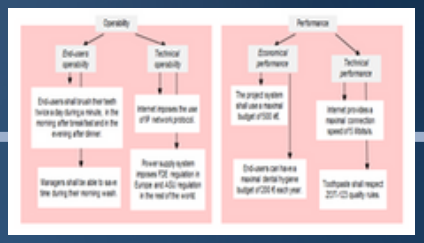
Functional Requirement Breakdown

A functional requirement breakdown is a hierarchical classification of all functional requirements relative to the system of interest according to relevant functional requirement categories.

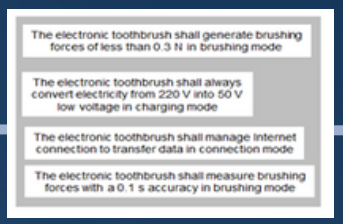
PROCESS

Step 1

Derive needs into functional requirements



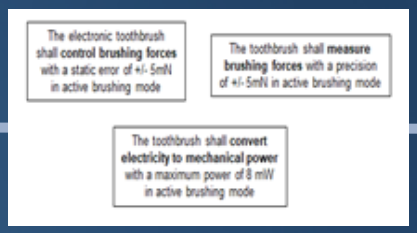
Needs breakdown



System level
functional requirements

Step 2

Derive low-level functional requirements



Lower level
functional requirements

CESAM/COMMUNITY

PROCESS

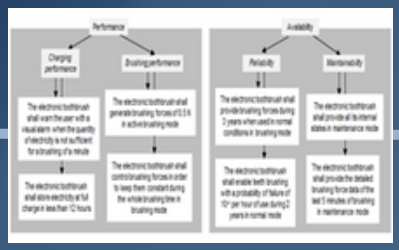
Step 3

Classify functional requirements into homogeneous sets

First version of functional requirement breakdown

Step 4

Verify coherence of functional requirement breakdown



Coherent version of functional requirement breakdown

Functional requirements analysis is the process that builds the knowledge of the functional requirements of the system of interest and constructs its functional requirement breakdown.

CESAM/COMMUNITY

CONSTRUCTIONAL ANALYSIS

Key concepts, Process & Deliverables

Purpose:

Find an optimal solution that results from a trade-off between what is desired (functions, style...) and what is possible (feasibility, technology, physical constraints,...)

Component

Product breakdown structure

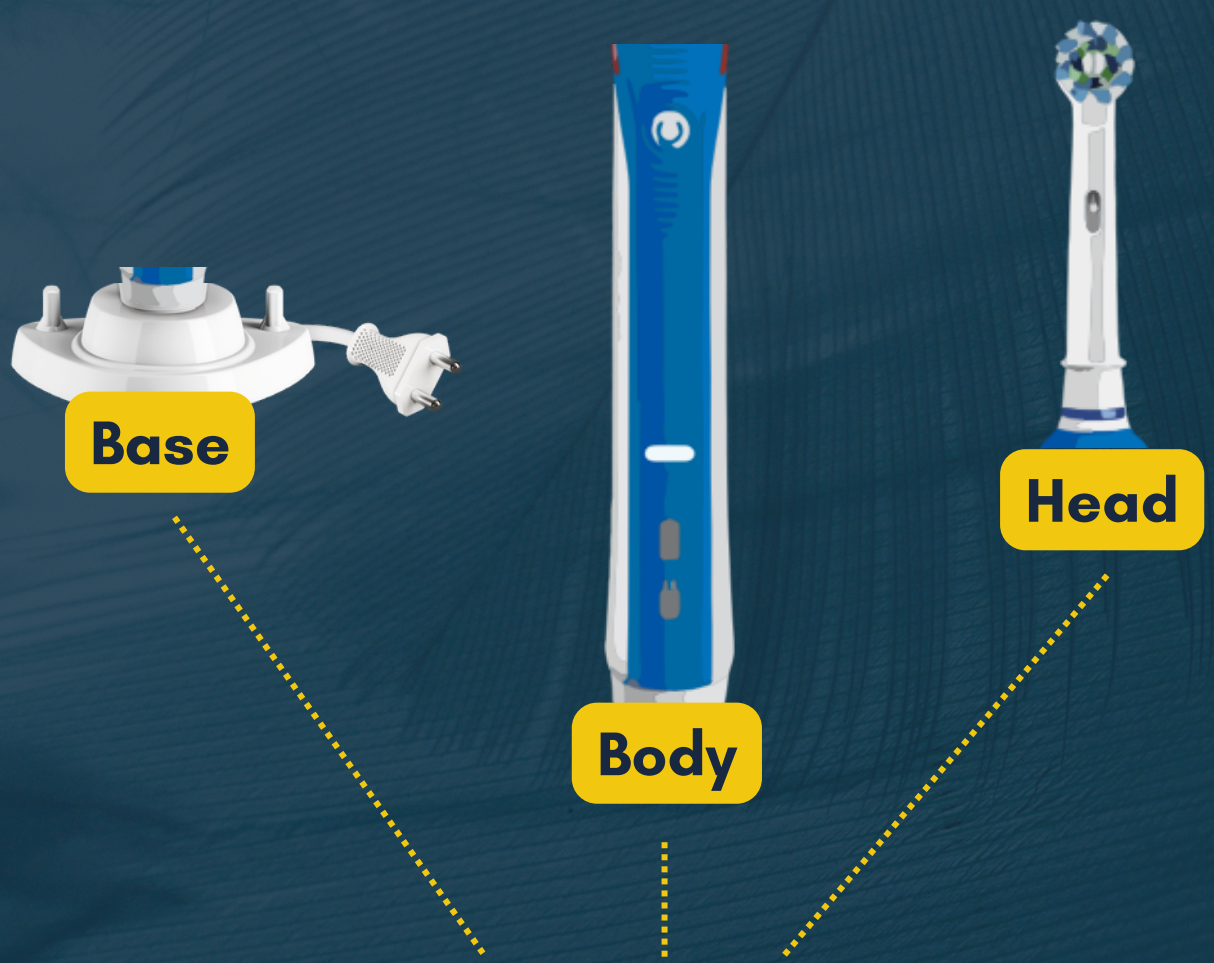
Constructional interactions diagram

CESAM/COMMUNITY

KEY CONCEPTS

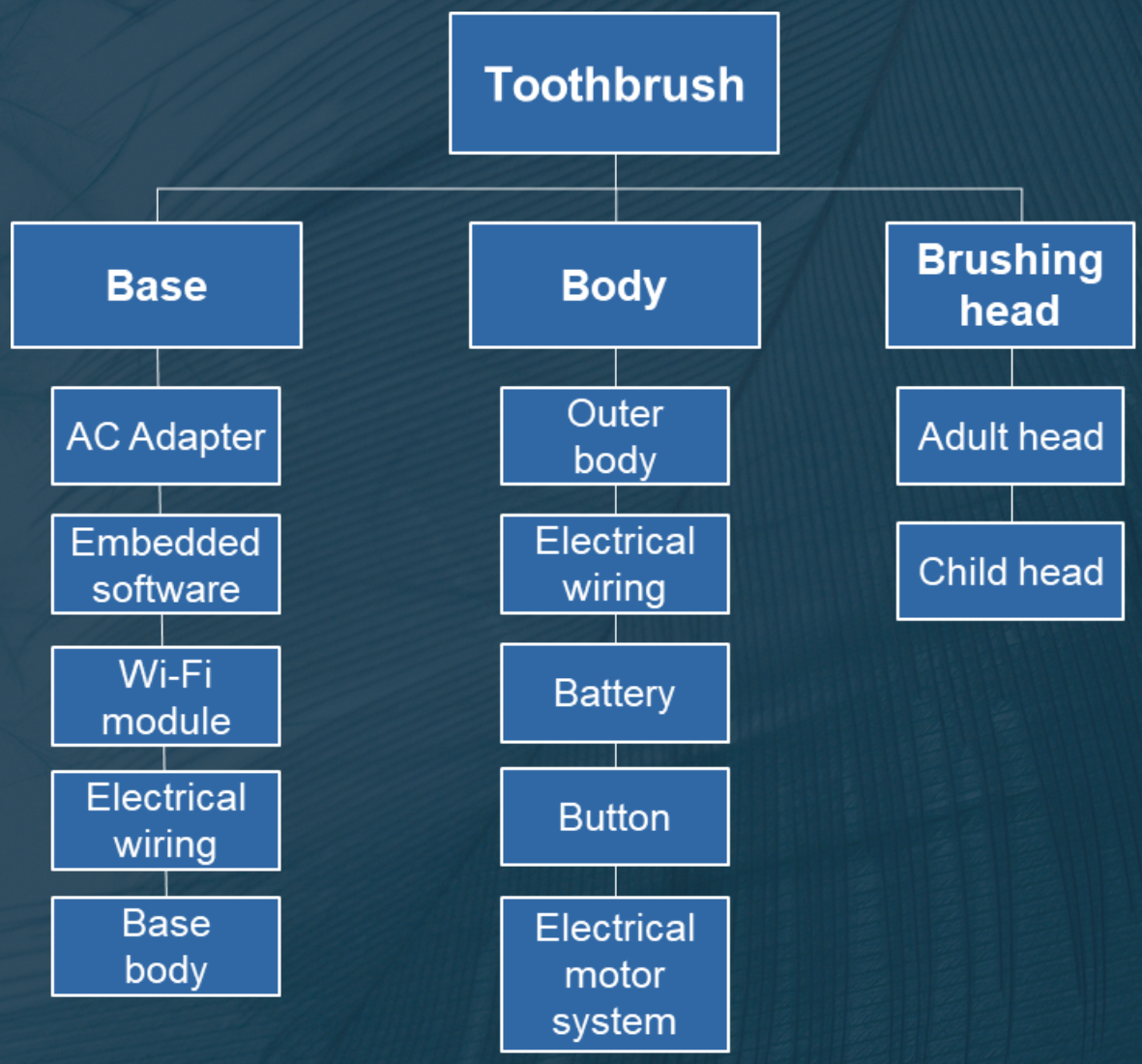
A **component** is a concrete building block of the system

- Components refer to the nature of the considered part of the system
- A component can itself be considered as a system.



These are some components of the electronic toothbrush

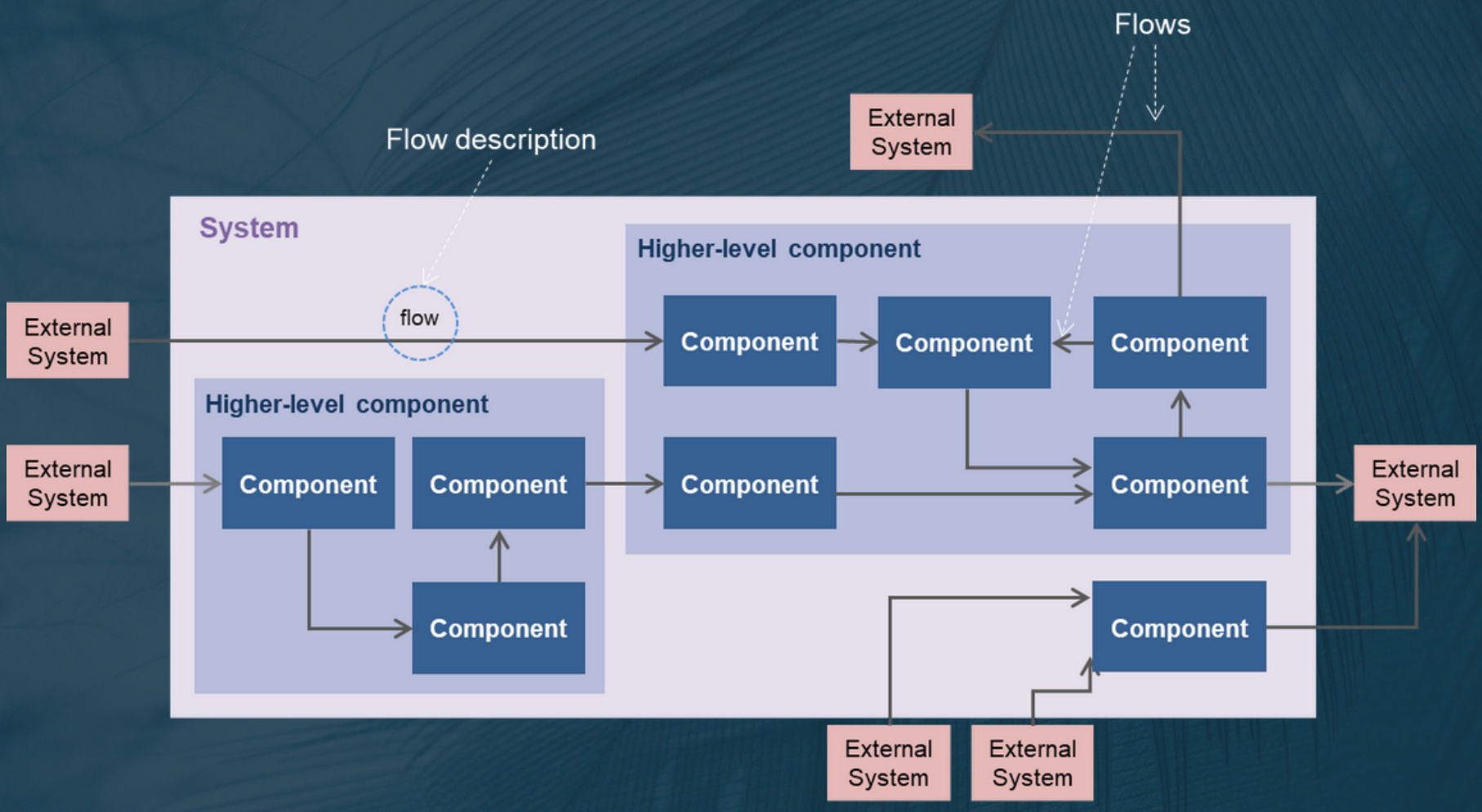
DELIVERABLES



Product breakdown structure

The **Product breakdown structure** of a system is a tree-like diagram that represents the breakdown structure of system and its components.

DELIVERABLES



Constructional interactions diagram

The **Constructional interactions diagram** represents the components of a system and the concrete flows exchanged between these components and with the environment.

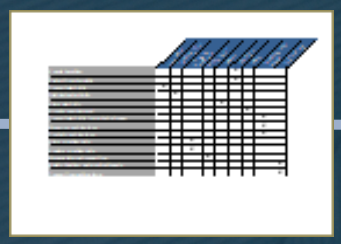
PROCESS

Step 1

Functional architecture

Identify components

Functional allocation



Step 2

Define constructional breakdown

Constructional breakdown

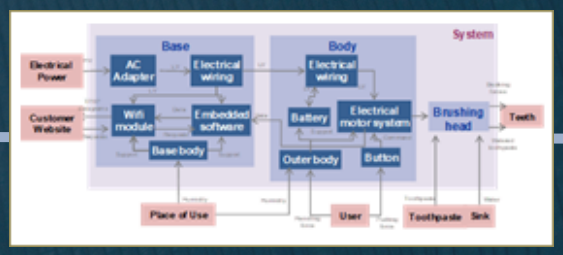
CESAM/COMMUNITY

PROCESS

Step 3

Define constructional interactions

Constructional interactions diagram



Step 4

Specify constructional interfaces

Constructional interfaces

Step 5

Align operational and functional architectures

Constructional architecture

CESAM/COMMUNITY

CONSTRUCTIONAL REQUIREMENT ANALYSIS

Template, Process & Deliverable

Purpose:

Express in an unambiguous, measurable and testable way how the components of the system answer to stakeholders' needs and system's functions and characterize the level of performance of these components.

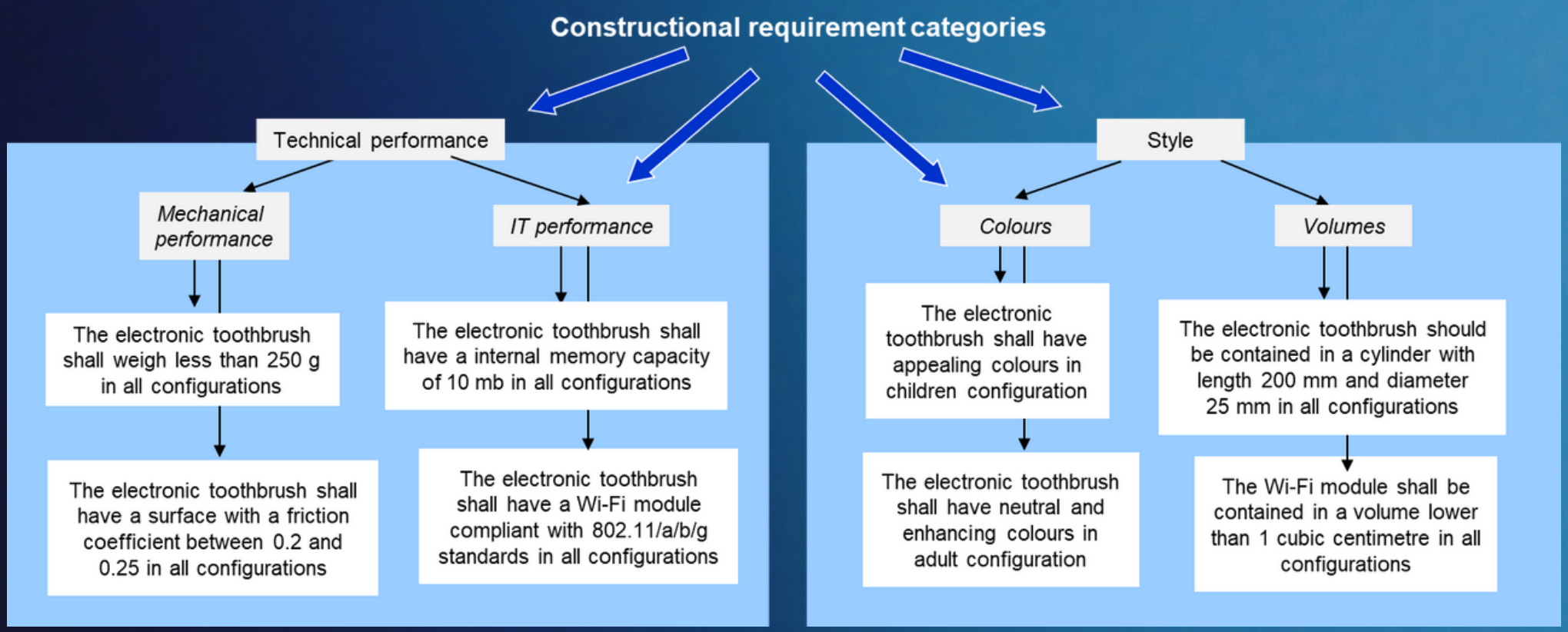
CESAM/COMMUNITY

TEMPLATE

Constructional requirement			
Domain	Main category to which the requirement belongs	**Reference**	A unique code for the Requirement
Statement			
Constructional requirement pattern to respect The **<SYSTEM>** (who) shall **<BE / BE MADE OF SOMETHING>** (how) with an **<EXPECTED LEVEL OF PERFORMANCE>** (how much) in a given **<TECHNICAL CONFIGURATION>** (when and/or where).			
Satisfaction criteria			
How does one measure and quantify that the constructional requirement is really fulfilled?			

Remember the deliverable is a list of quantified constructional requirement statements.

DELIVERABLE



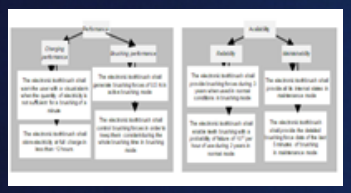
Constructional requirement breakdown

A constructional requirement breakdown is a hierarchical classification of all constructional requirements relative to the system of interest according to relevant constructional requirement categories.

CESAM/COMMUNITY

PROCESS

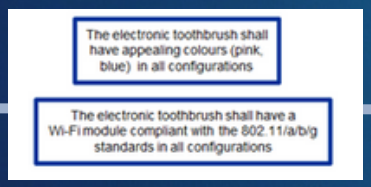
Needs breakdown



Functional requirement breakdown

Step 1

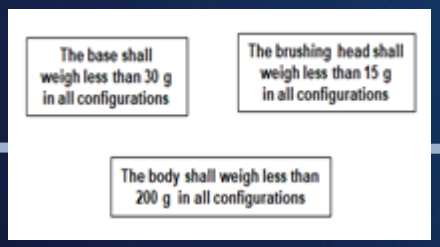
Derive needs and functional requirements into constructional requirements



System level constructional requirements

Step 2

Derive low-level constructional requirements



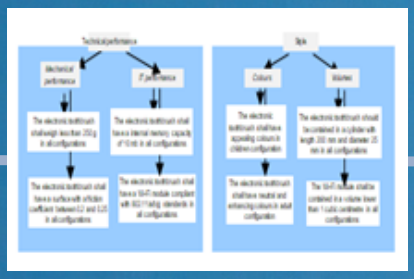
Lower level constructional requirements

CESAM/COMMUNITY

PROCESS

Step 3

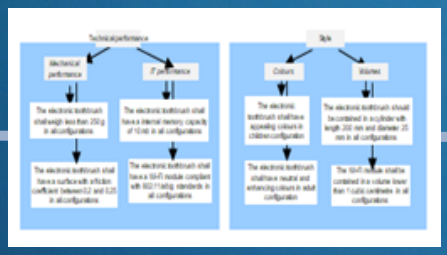
Classify constructional requirements into homogeneous sets



First version of functional requirement breakdown

Step 4

Verify coherence of constructional requirement breakdown



Coherent version of the constructional requirement breakdown

Constructional requirements analysis is the process that builds the knowledge of the **constructional requirements** of the system of interest and constructs its **constructional requirement breakdown**.

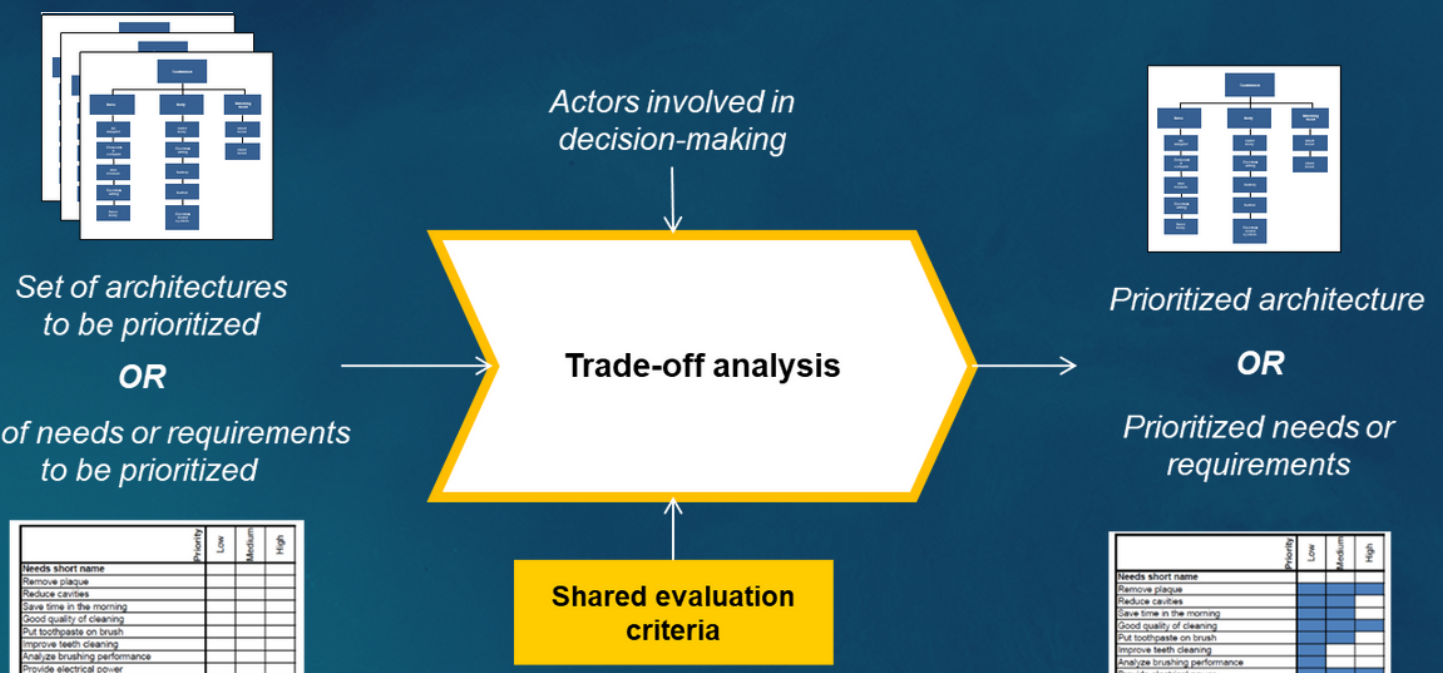
CESAM/COMMUNITY

TRADE-OFF ANALYSIS

Definition, Process & Deliverables

Purpose:

Help the actors involved in a decision making process to prioritize an architecture or a set of needs or requirements in a rational way using shared evaluation criteria.



Key inputs and outputs of an architectural trade-off analysis

CESAM/COMMUNITY

KEY CONCEPTS

Set of architectures to be prioritized

OR

Set of needs or requirements to be prioritized

Needs short name	Priority	Low	Medium	High
Remove plaque				
Reduce cavities				
Save time in the morning				
Good quality of cleaning				
Put toothpaste on brush				
Improve teeth cleaning				
Analyze brushing performance				
Provide electrical power				

Actors involved in decision-making

Trade-off analysis

Shared evaluation criteria

Prioritized architecture

OR

Prioritized needs or requirements

Needs short name	Priority	Low	Medium	High
Remove plaque				
Reduce cavities				
Save time in the morning				
Good quality of cleaning				
Put toothpaste on brush				
Improve teeth cleaning				
Analyze brushing performance				
Provide electrical power				

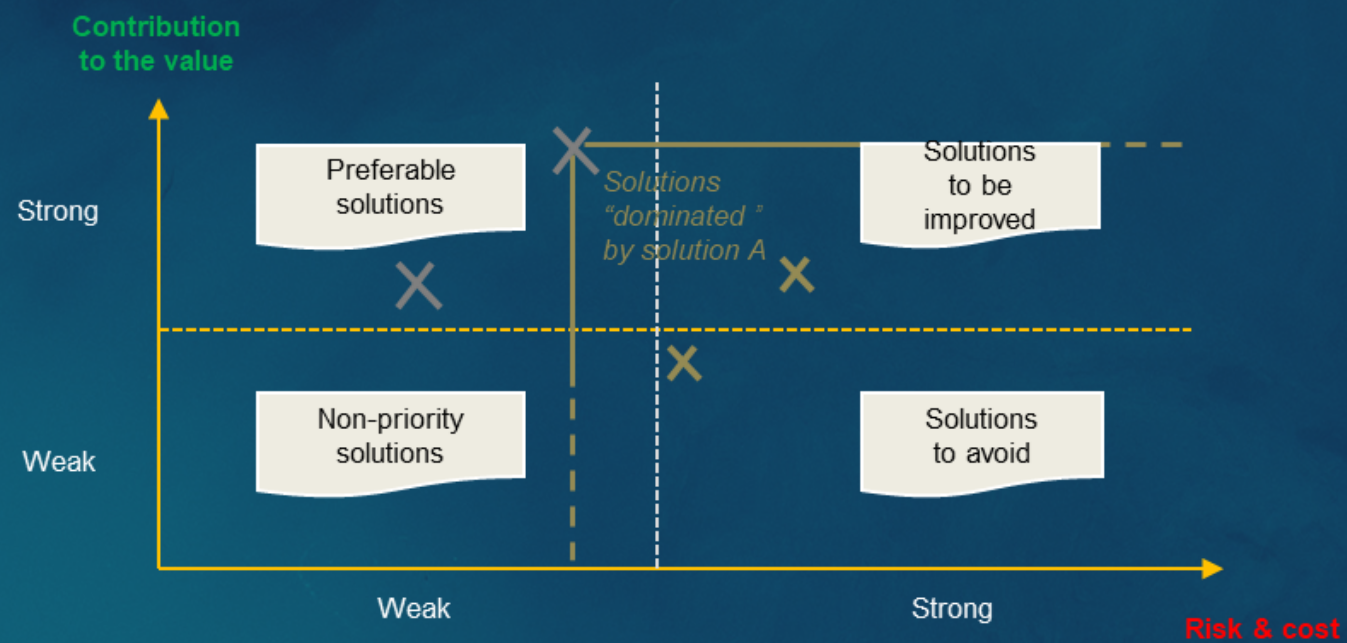
Key inputs and outputs of an architectural trade-off analysis

A trade-off analysis has for objective to help all the actors involved in a decision making process to prioritize an architecture or a set of needs or requirements in a rational way, using shared evaluation criteria.

DELIVERABLES

Solutions to be evaluated	Contribution of solution to decision criteria	Decision criteria to be used to evaluate the solutions					Score
		A. User friendliness	B. Short development time	C. Reuse of existing technology	D. Cost of non quality reduction	E. Improvement of services provided to customers	
		2/40	13/40	3/40	16/40	6/40	
Solution 1	*Weak (1 pt.)*	1	2	0	1	0	4
	Medium (3 pt.)	2	1	0	4	0	
	Strong (9 pt.)	3	3	6	1	6	
...	---	...	---	---		...	---
Solution N	*Weak (1 pt.)*	0	0	0	0	0	3
	Medium (3 pt.)	6	6	6	3	6	
	Strong (9 pt.)	0	0	0	3	0	

Global evaluation for each solution



Prioritization of each solution

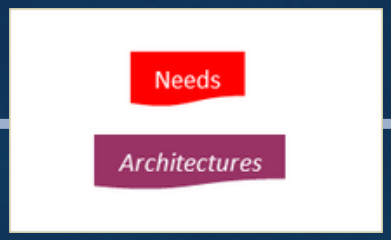
The two possible deliverables of the trade-off analysis process are the evaluation of all possible solutions and a prioritization of all possible solutions.

CESAM/COMMUNITY

PROCESS

Step 1

Identify the actors to involve in the trade-off process

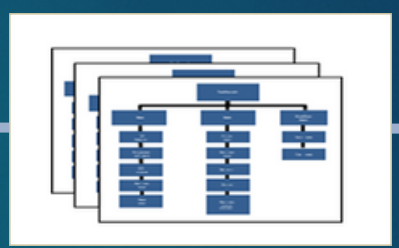


Needs and architectures

Set of actors to be involved in the decision process

Step 2

Identify & share the items to evaluate



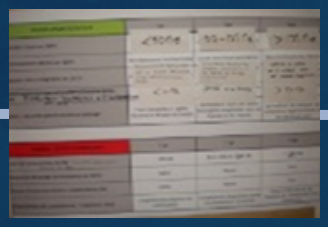
Set of architectures or needs to be prioritized

CESAM/COMMUNITY

PROCESS

Step 3

Define & share the decision criteria to be used



Shared decision criteria

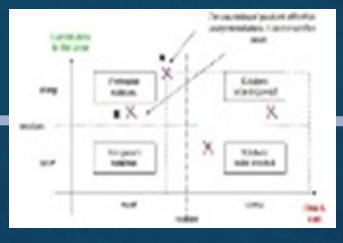
Step 4 - Option 1

Evaluate each solution according to the decision criteria

Solutions evaluation

Step 4 - Option 2

Prioritize the evaluated solutions using prioritization techniques



Solutions prioritization

Trade-off analysis is the process that aims at selecting the best solution among several possibilities.

CESAM/COMMUNITY

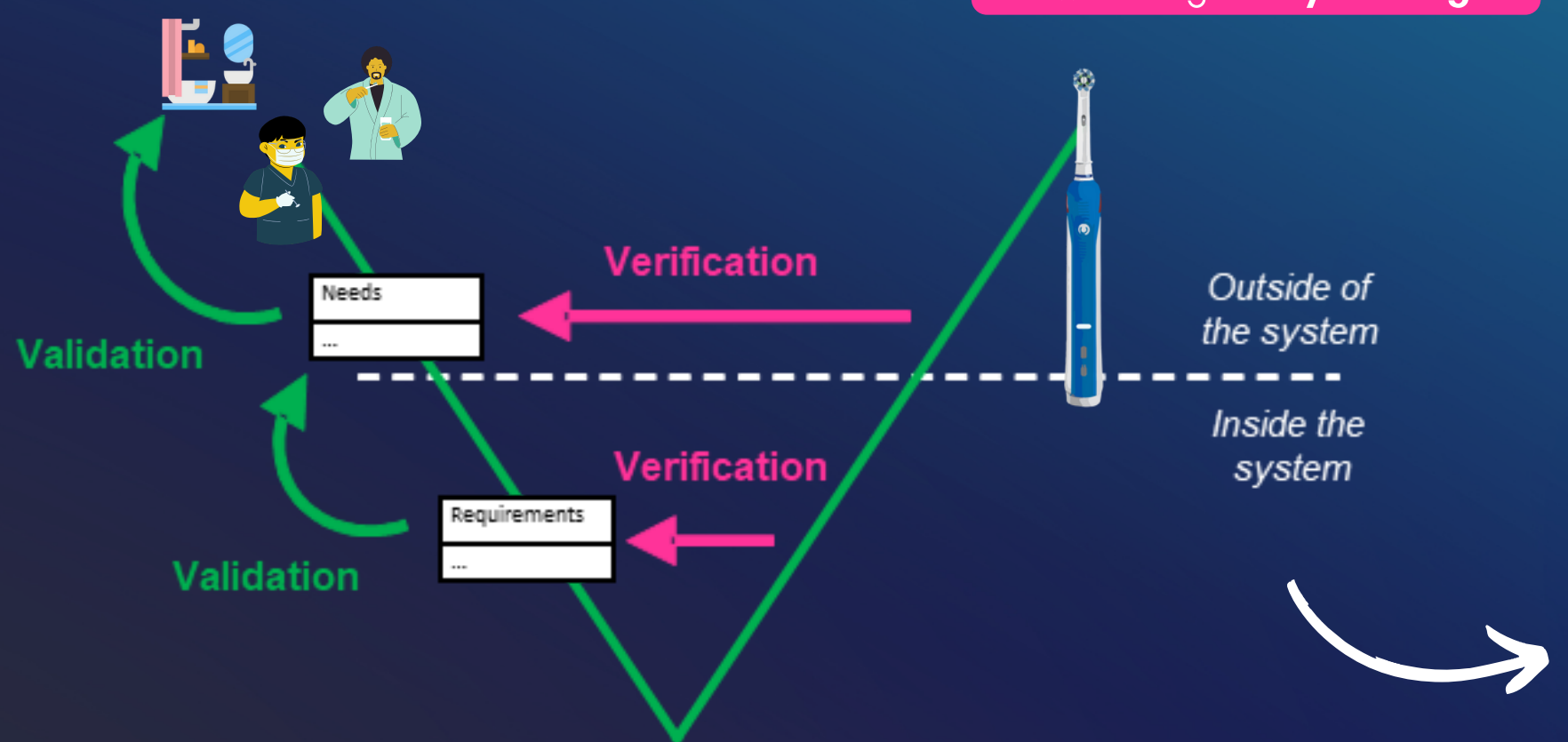
VALIDATION & VERIFICATION

Purpose:

Guarantee that the system is operationally, functionally and constructionally consistent and takes correctly into account all its expected properties.

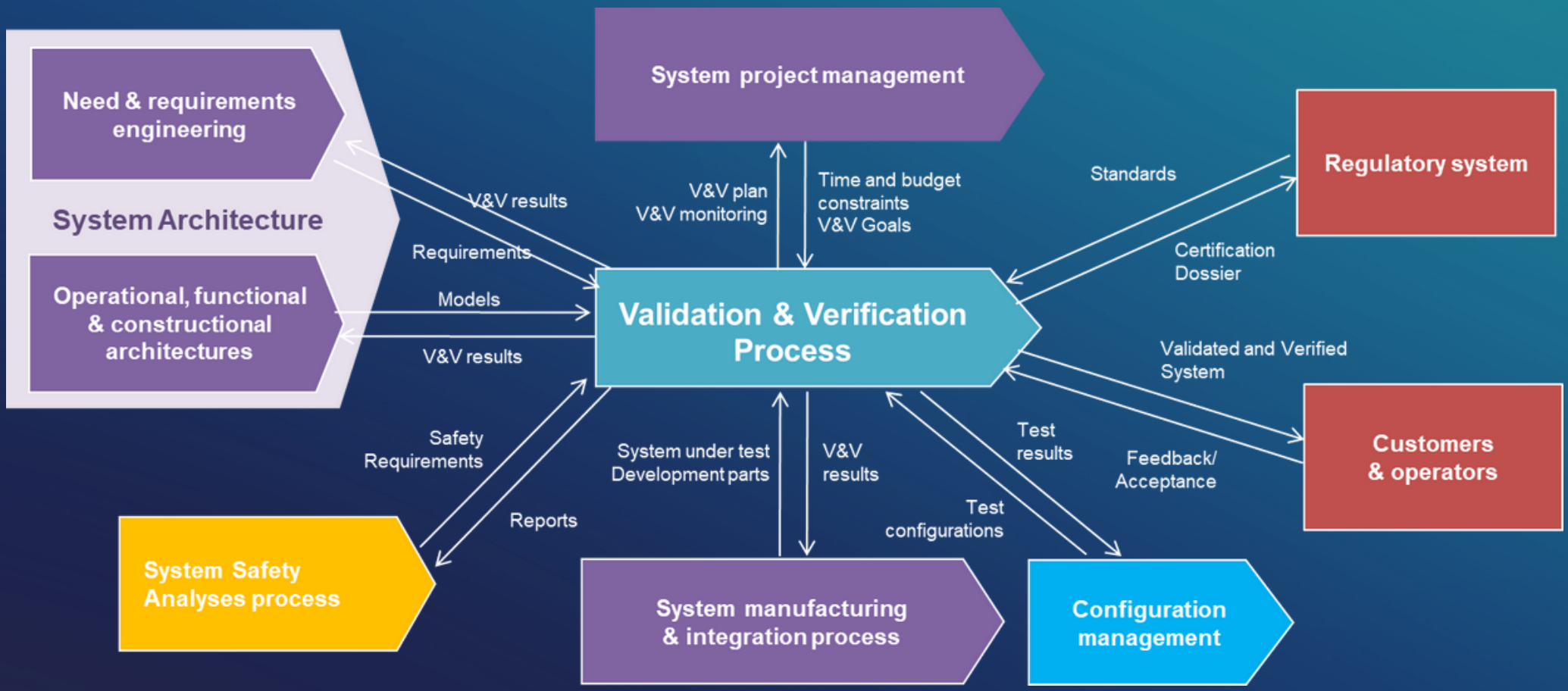
Validation answers the question:
are we doing **the right system?**

Verification answers the question
are we doing the **system right?**



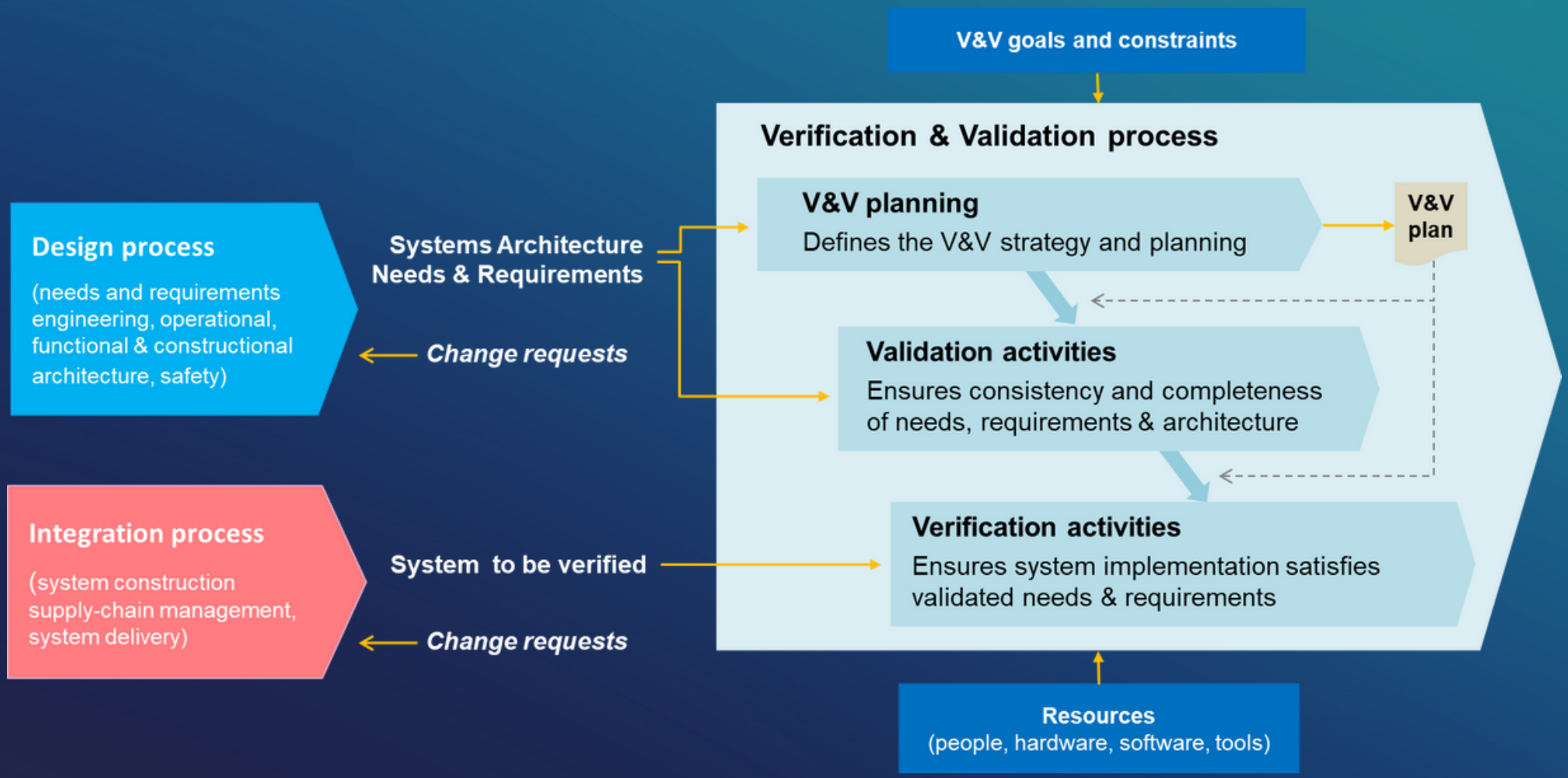
CESAM/COMMUNITY

ENVIRONMENT



External processes within the environment of the Validation & Verification process of system.

PROCESS



The Verification & Validation processes interact iteratively with the system Design & Integration processes.

CESAM/COMMUNITY

GOOD PRACTICES

V&V method	Model-oriented V&V practices	Integration-oriented V&V practices
Analysis	▪ Manual or automatic analyses of a model (syntactic rules verification, crossed analyses, completeness analysis, etc.)	▪ Functional demonstrations (e.g. users interfaces, components behaviours, etc.) ▪ Prototyping (e.g. for safety analyses, etc.)
Review	▪ Model self-examinations ▪ Specifications pear reviews (quality & completeness of needs, requirements & descriptions)	▪ Pear reviews of the integrated system ▪ More or less formal reviews of the integrated system by the stakeholders ▪ Returns on experience
Test	▪ Simulations (e.g. using MATLAB & Simulink)	▪ Unitary and integration tests of the integrated system components (at each systemic level) ▪ Formal qualification of the integrated system with its stakeholders

Verification & validation practices that can be deployed during the different **development phases** of a system.