Managing Projects with openSE

Part 2

Pierre Bonnal & **Thijs Wijnands**

version **1.7**

Project Management

What are we going to see together?

Lifecycle

Processes

- Launching a project
- Defining requirements
- Planning & scheduling
- Managing risks
- Handling issues
- Ensuring quality
- Costing
- Reporting progress
- Finalizing a project

openSE

Roles

- Initiators
- Project Board
- Project Manager
- Project Team
- Study Team
- Study Leader
- Project Stakeholders

Artefacts

- Project Proposal
- Project Roadmap
- Project Mngt Plan
- Issue Record(s)
- WBS & RACI
- Coord. Schedule
- Cost Estimate
- Budget
- PBS
- Reqts. Register
- Risk Register
- Progress Report(s)
- Close-out Report

4

Project Quality Management

Project Management

What are we going to see together?

openSE

Lifecycle

Roles

Initiators

Project Board

Project Manager

Project Team

Study Team

Study Leader

Project Stakeholders

Processes

Launching a project

Ensuring quality

Defining requirements

Planning & scheduling

Costing

Managing risks

Reporting progress

Handling issues

Finalizing a project

Artefacts

Project Proposal

Project Roadmap

Project Mngt Plan

Issue Record(s)

WBS & RACI

Coord. Schedule

Cost Estimate

Budget

PBS

Reqts. Register

Risk Register

Progress Report(s)

Close-out Report

Quality

At a glance

Quality Planning

Quality Assurance

I say what I will do

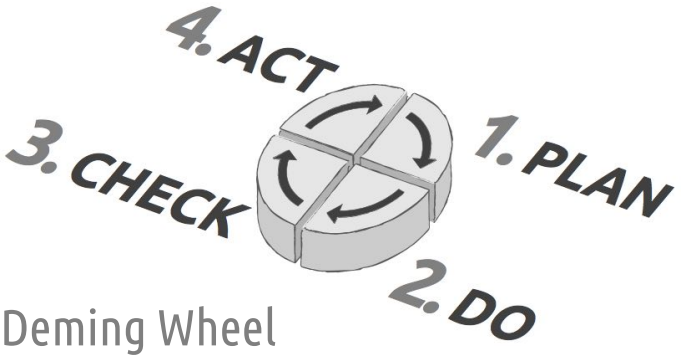
Someone checks that it is appropriate

I do what I have said

I provide evidence of compliance

Quality Control

I also identify defects in the processes and seize the opportunity to improve them



Deming Wheel

Quality Planning

At a glance, in a complex system development project context

- Defining which **standards** are applicable
- Defining which "**tools**" to use (incl. document **templates** and **forms**)
- Assigning **roles**, i.e. setting up a **project organization**
- Defining key managerial **processes**:
  - For releasing **documents** (incl. verification and validation)
  - For managing the **configuration** (i.e. the baselines)
  - For handling **issues** and **nonconforming outputs**
  - For planning, scheduling, costing, hiring project participants, managing risks, reporting the progress, buying supplies and services…
- More broadly, for **decision making**

ISO 9001:2015 § 8.7

Project Quality Assurance Framework

(Management) (System)

Quality Control

At a glance, in a complex system development project context

- Implementing straightforwardly the Project QA Framework provisions:
  - Proceeding systematically to the **verifications** and **validations** as they have been planned
  - Insuring the **traceability** of the tasks (how they were performed) and of the task's outputs/deliverables (specifications vs. actuals) by means of records
  - Releasing **change requests** when a baseline shall be modified
  - Releasing **non conformity** reports when an output or a deliverable is not as expected…

the **QA trade-off**

Resources to dedicate to QA

Value created by QA

4.1

Quality Planning

openSE

Applicable Standards

- Standards related to the SE / NPD **project management**
  e.g.: ISO 21500:2012 or ANSI PMBoK 5th ed. or 

- Standards related to the NPD **engineering processes**
  incl. PLM, CAD systems, geometrical tolerencing, etc.

- Standards related to the **product/system** to develop itself
  in the fields of materials, of communication, of energy, of interfaces,
  of software, of reliability, of availability, of maintenability, of safety, etc.

- Standards related to the **manufacturing and assembly processes**
  incl. supply chain, plant engineering, etc.

3.2
important results

4.1.2

SIMPLE approach

INTERMEDIATE approach

ADVANCED approach

Project Management Plan

openSE

Project Management Plan

- The "**entry point**" to project information
- The aim of the PMP is twofold:
  - Ensuring that the project participants agree upon and share a common framework for organizing their project
  - Giving the project board the assurance that the project expectations are well understood and that everything is done to ensure the operational success of the project
- A few possible approaches depending on the project participants maturity level w.r.t. project management processes

Project Management Plan

Typical Table of Contents
Simple Approach

1 **Project Overview** PMP Scope + Reformulation of the Project Roadmap

2 **Project Organization** Project Board, Project Team, roles, OBS

3 **Project Management Processes**

- 3.1 **Scope Management** WBS, Work Packages, Work Units, Activities
- 3.2 **Time Management** Master and Coordination Schedules
- 3.3 **Resource and Cost Management** Manpower, budgeting, EVM
- 3.4 **Quality Management** Document management, V&V, configuration management, issue and non conformity handling
- 3.5 **Communication Management** Meetings, reporting periodicity
- 3.6 **Risk Management** Project Risk Register, Project Continuity Plans
- 3.7 **Procurement and Contribution Management** Ordering, contracting

A **Applicable Standards**

Project Management Plan

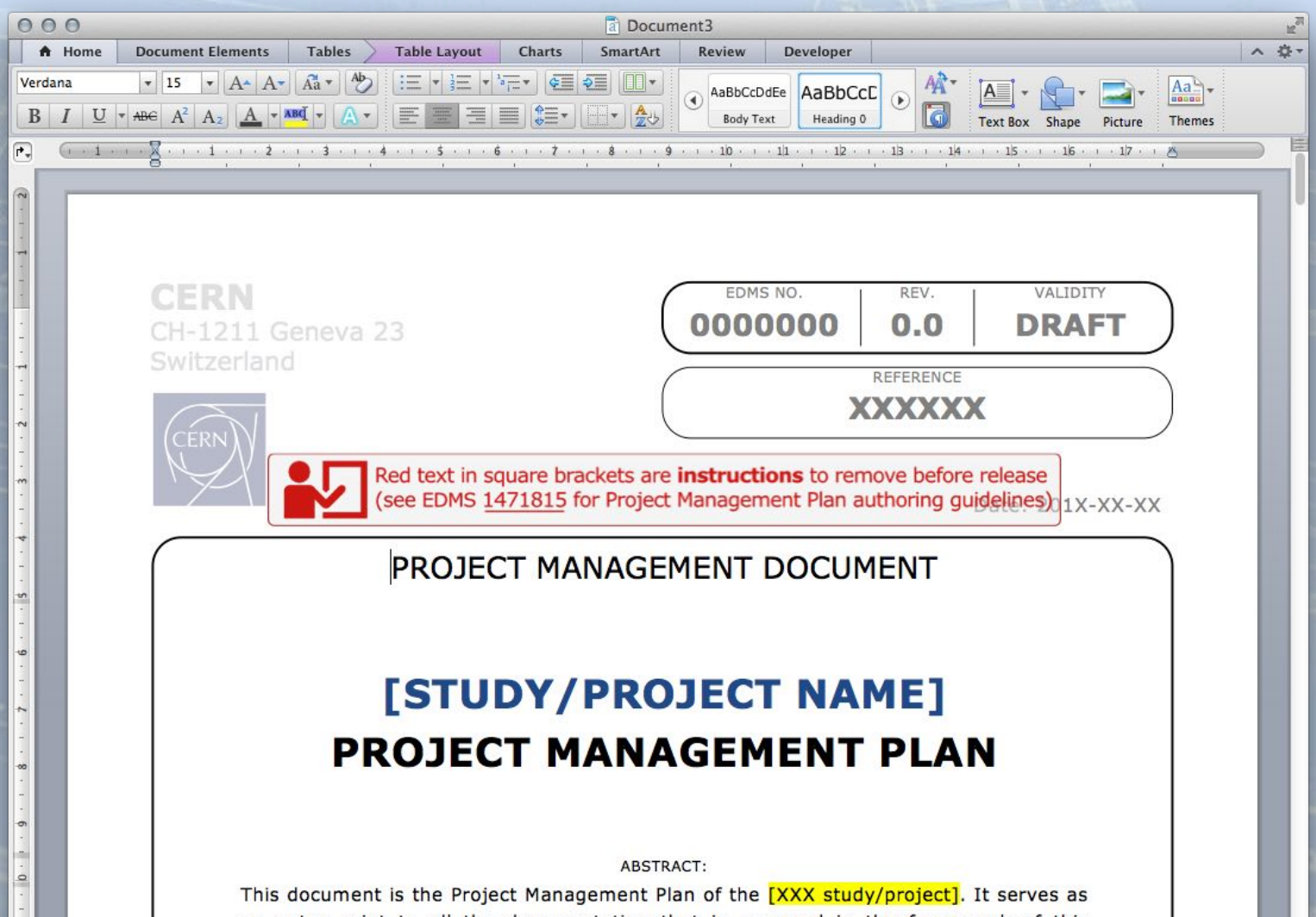
Editorial Process

Authoring: Project Manager 
+ a few Key Project Participants

Verification: Some other Key Project Participants + some Project Management Experts (e.g. members of the PMO)

Validation: **Project Manager** 

At **CERN** (in the A&T Sector)
Project Management Plan
Template and authoring guidelines → EDMS 1471815



3.3
important results

4.1.2

INTERMEDIATE approach

ADVANCED approach

Other PM Documents

openSE

Other Project Management Documents

Project Proposal

Project Roadmap

Project Mngt Plan

Master Schedule

Budget

WBS & RACI

Cost Estimate

Coord. Schedule

CBS

Risk Register

Issue Record(s)

PBS

Reqts. Register

Progress Report(s)

Close-out Report

4.2

Quality Assurance

openSE

Key Quality Assurance Processes

- Managing **documents**
  i.e. authoring, versioning, circulating (verification and validation), releasing and archiving project documents, but also 3D mock-ups and 2D drawings

- Managing the **configuration** and handling **issues** and **nonconforming outputs** i.e. managing baselines, managing change requests and orders, ensuring traceability

- Conducting **quality audits** and **reviews**

4.2.1

SIMPLE approach

INTERMEDIATE approach

ADVANCED approach

Document Management System

openSE

Project Document Register

Unique ID	Document title	Ver.	Date	Authored by	Verified by	Validated by
100	Project Roadmap					
		0.1	2014-01-13	Alberte		
		0.2	2014-01-20	—	Ursule, Yvone	
		1.0	2014-01-22	—	—	Xavier, Zélie
101	Project Management Plan					
		0.1	2014-02-05	Alberte, Barnabé		
102	Project Work Breakdown Structure					
103	Project Cost Estimate					
104	Project Budget					
105	Project Master Schedule					
		0.1	2014-02-07	Alberte, Cyprien		
106	Project Coordination Schedule					
107	Project RACI Matrix					
108	Project Risk Register					

At **CERN** (in the A&T Sector)
Project Document Registers

- **EDMS** → 100% engineering and PM documents
- **EDMS/CDD** → 2D drawings
- **CATIA/SmarTeam** → 3D models
- **CDS** → Scientific publications (reports, notes)
- **Indico** → Presentations
- **SharePoint** or **Drupal** → General project information
- **DFS** → Nothing! Very bad practice
- **CFU/CDS** → Released procurement documents

No project-wide document register!

Project Document Template

Unique ID	Version	Status	Date
101	**0.3**	**DRAFT**	2014-02-22

the whatever project

PROJECT MANAGEMENT PLAN

Authored by:	Verified by:	To be validated by:
Alberte Barnabé	Cyprien Denise	Ernest

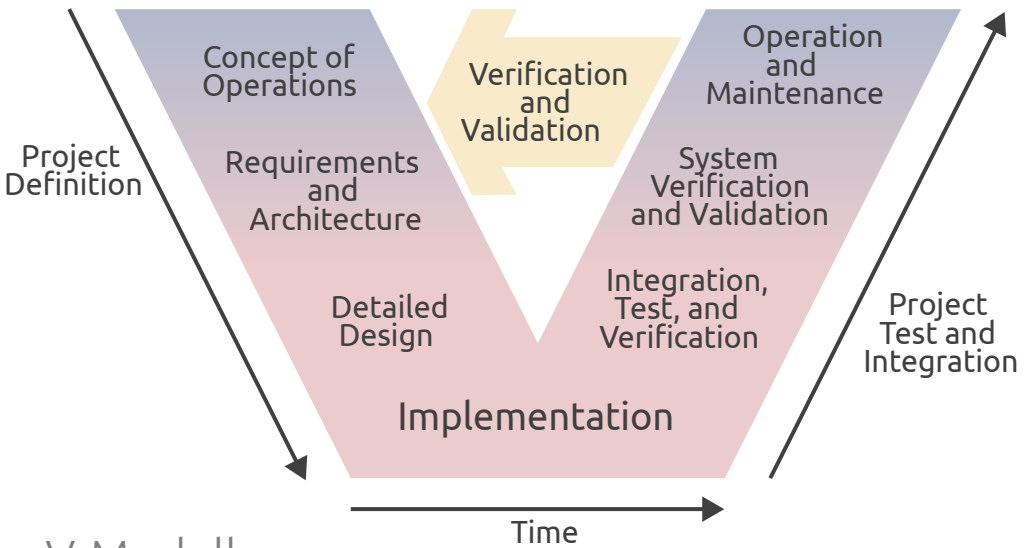
This document is uncontrolled when printed.
Check the Project Document Register to verify
that this is the correct version before use

Verification vs. Validation

Check vs. Approval

From Software Engineering but also widely applied to document lifecycle

Concept introduced by **Barry W. Boehm** (1981)



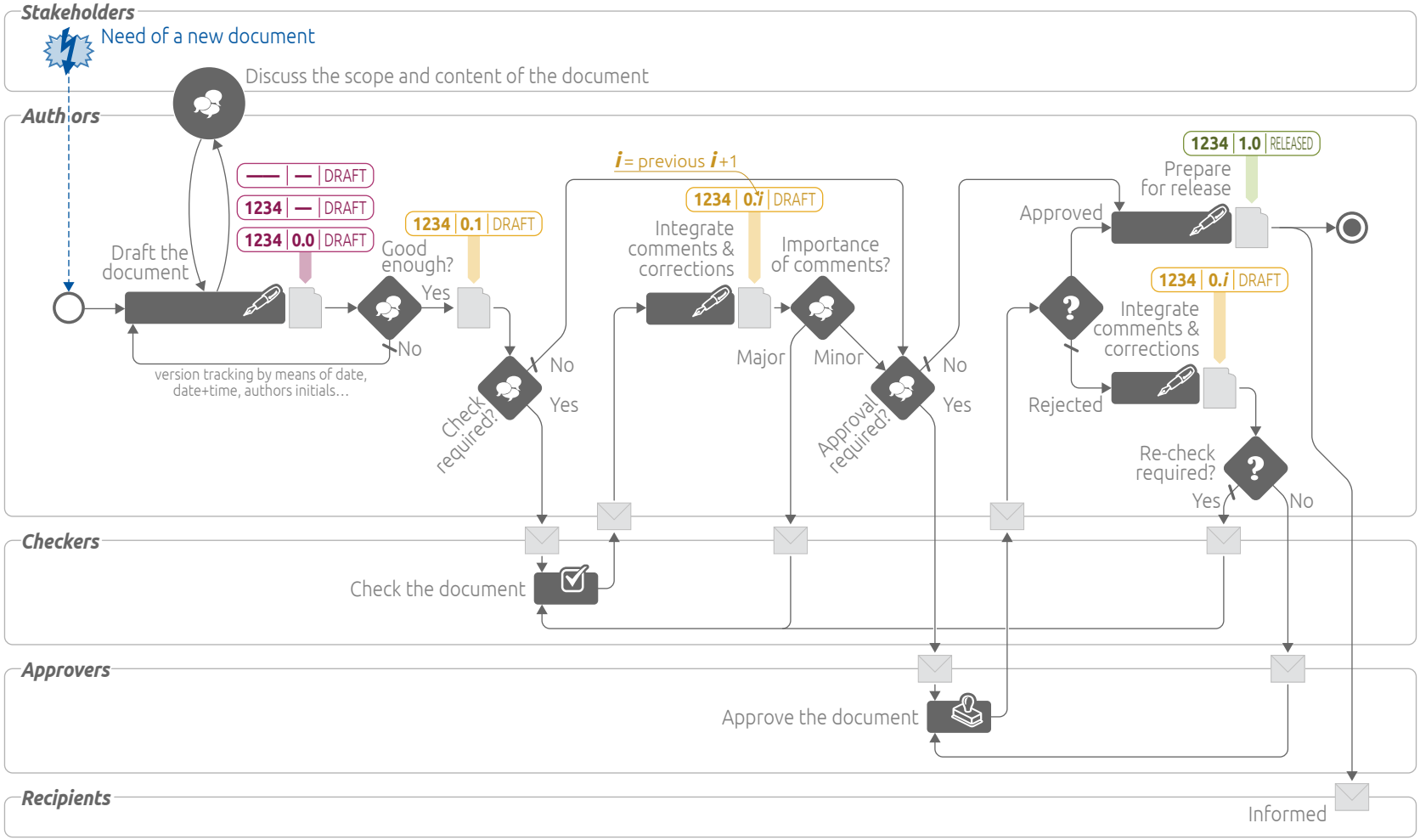
V-Modell

Verification:
Are we building the product right?
Are we solving the equation right?

Validation:
Are we building the right product?
Are we solving the right equation?

Document Lifecycle

Document authoring, circulating and versioning



4.2.2

INTERMEDIATE approach

ADVANCED approach

Issue Management System

openSE

Project Management

What are we going to see together?

Lifecycle

Roles

Initiators

Project Board

Project Manager

Project Team

Study Team

Study Leader

Project Stakeholders

openSE

Processes

Launching a project

Ensuring quality

Defining requirements

Costing

Planning & scheduling

Managing risks

Reporting progress

Handling issues

Finalizing a project

Artefacts

Project Proposal

Project Roadmap

Project Mngt Plan

Issue Record(s)

WBS & RACI

Coord. Schedule

Cost Estimate

Budget

PBS

Reqts. Register

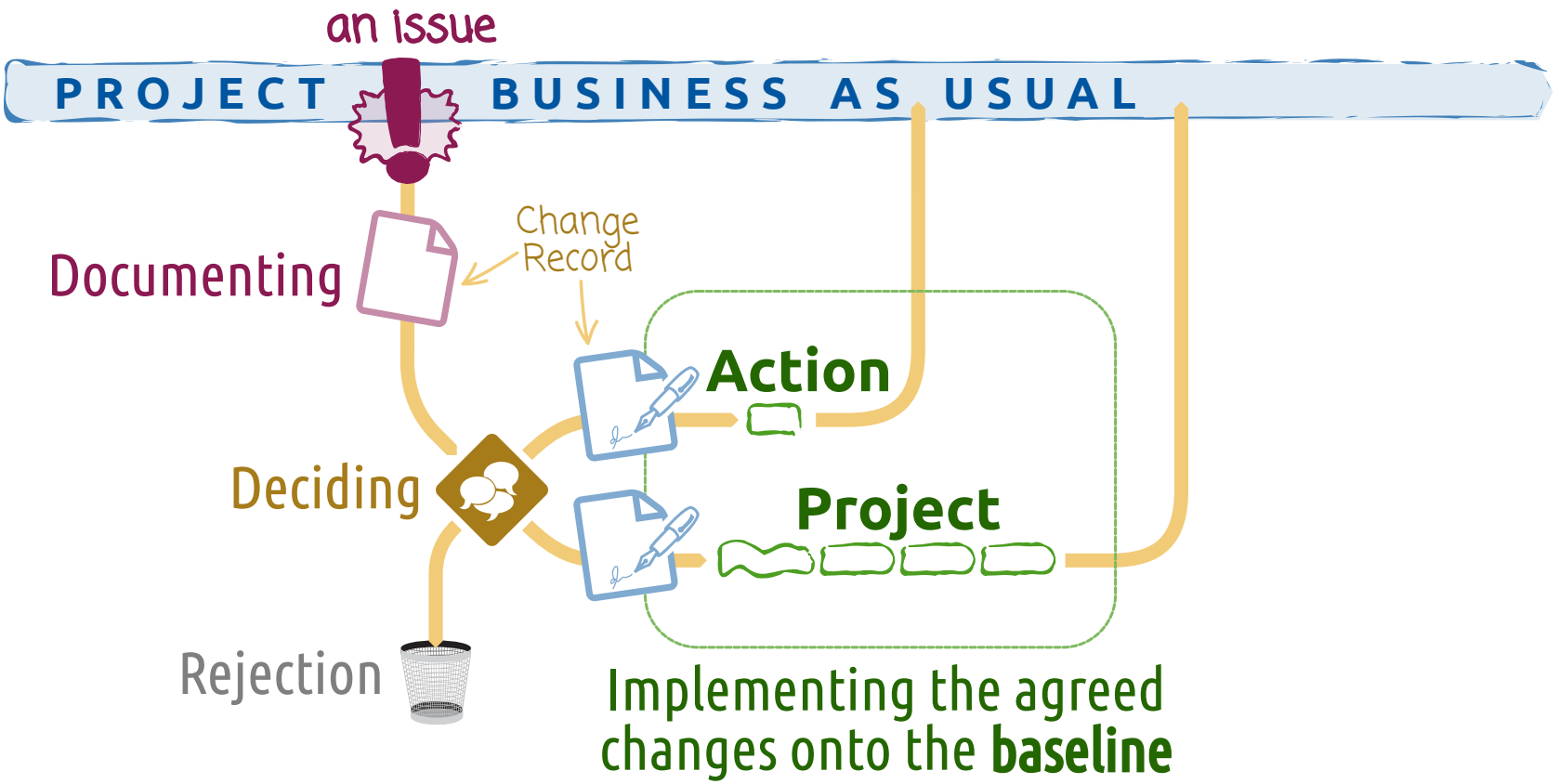
Risk Register

Progress Report(s)

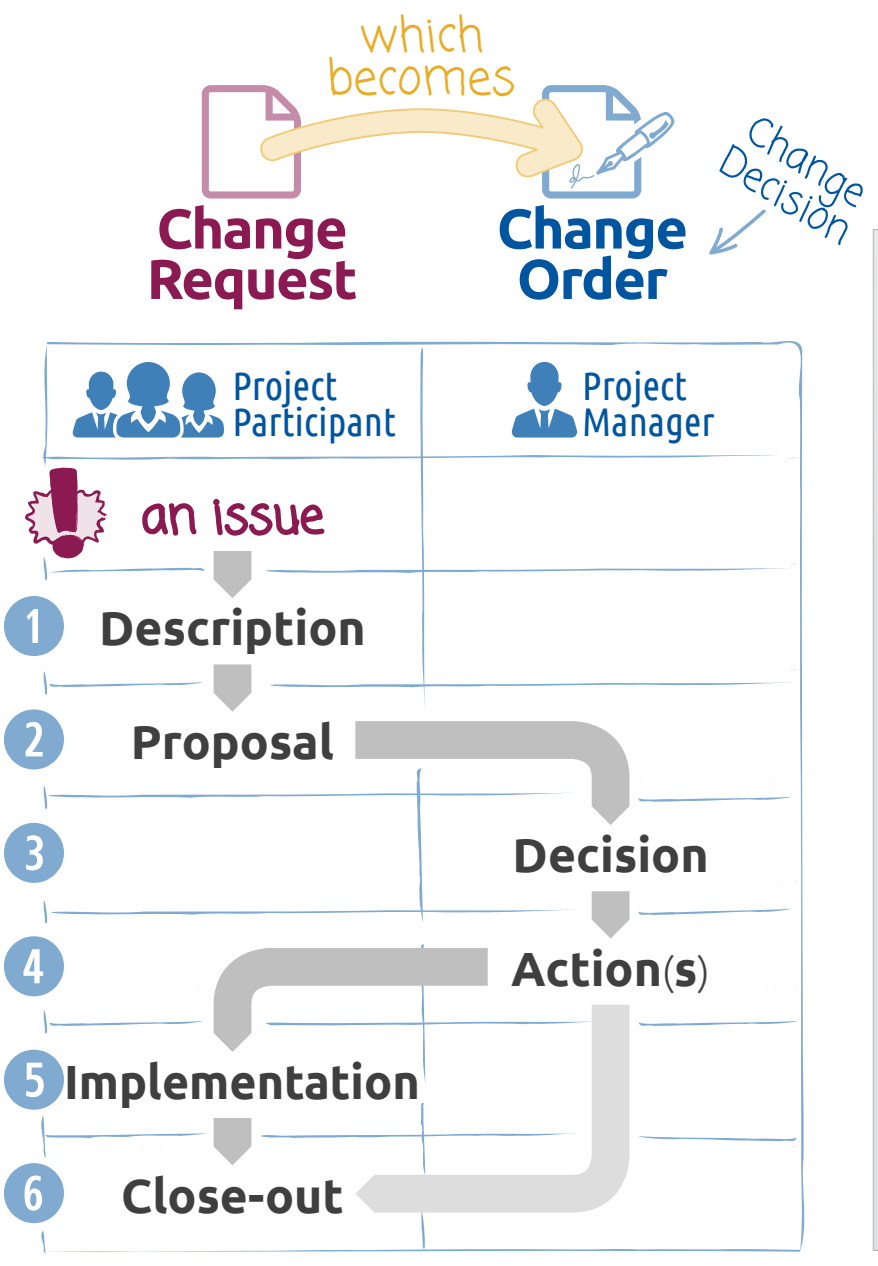
Close-out Report

Managing Issues

Change Records (**CR**) featuring requests and orders



Change Record



Unique ID: 234 | Version: 0.1 | Status: DRAFT | Date: 2014-02-26

CHANGE RECORD

1. Description

2. Proposal

3. Decision
   - ☐ Rejection
   - ☐ Action(s)
   - ☐ Project(s)

   Comments

4. Action(s) | Resp. | Date | 5 Done

6. Close-out

4.3

Quality Control

5

Requirements Engineering

Adapted from SE best practices:
ISO/IEC/IEEE 29148:2011
and a few major textbooks

openSE

Project Management

What are we going to see together?

openSE

Lifecycle

Roles

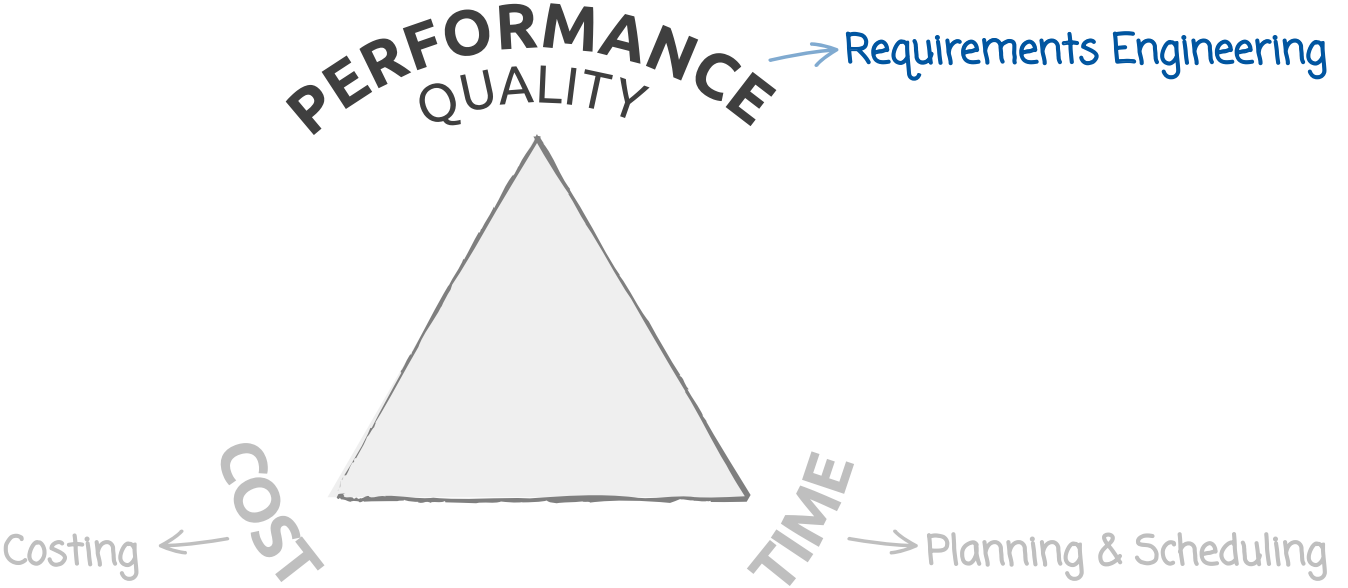
Processes

- Launching a project
- Ensuring quality
- Defining requirements
- Costing
- Planning & scheduling
- Managing risks
- Reporting progress
- Handling issues
- Finalizing a project

Artefacts

Requirement(s) Engineering

"Project triangle"



Why?

1. to set a reference 
2. to assess uncertainties
3. to follow up progress

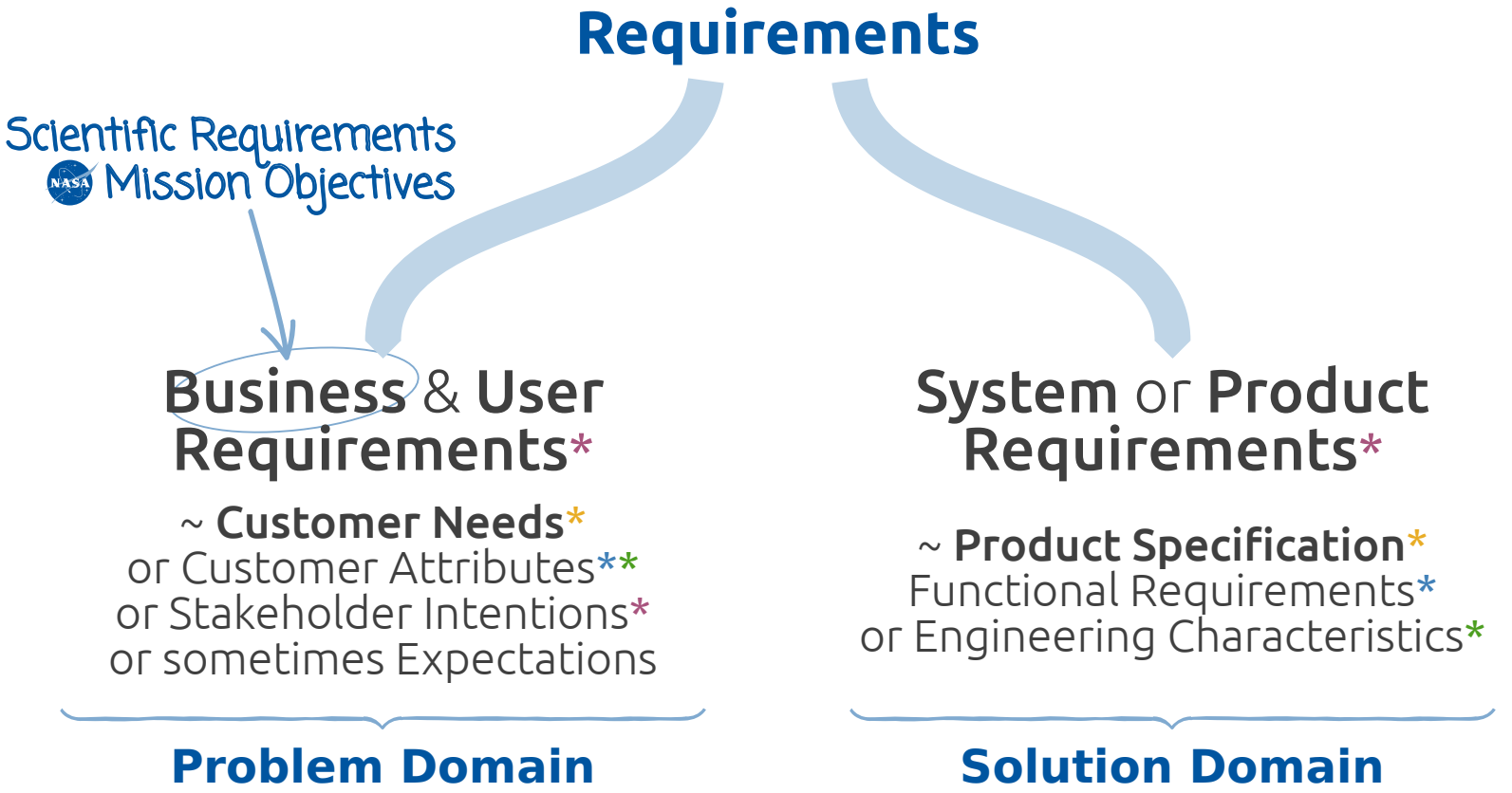
Requirement(s) Engineering

The process of documenting, analyzing, tracing, prioritizing and agreeing on requirements and then controlling change and communicating to relevant stakeholders

en.Wikipedia.org

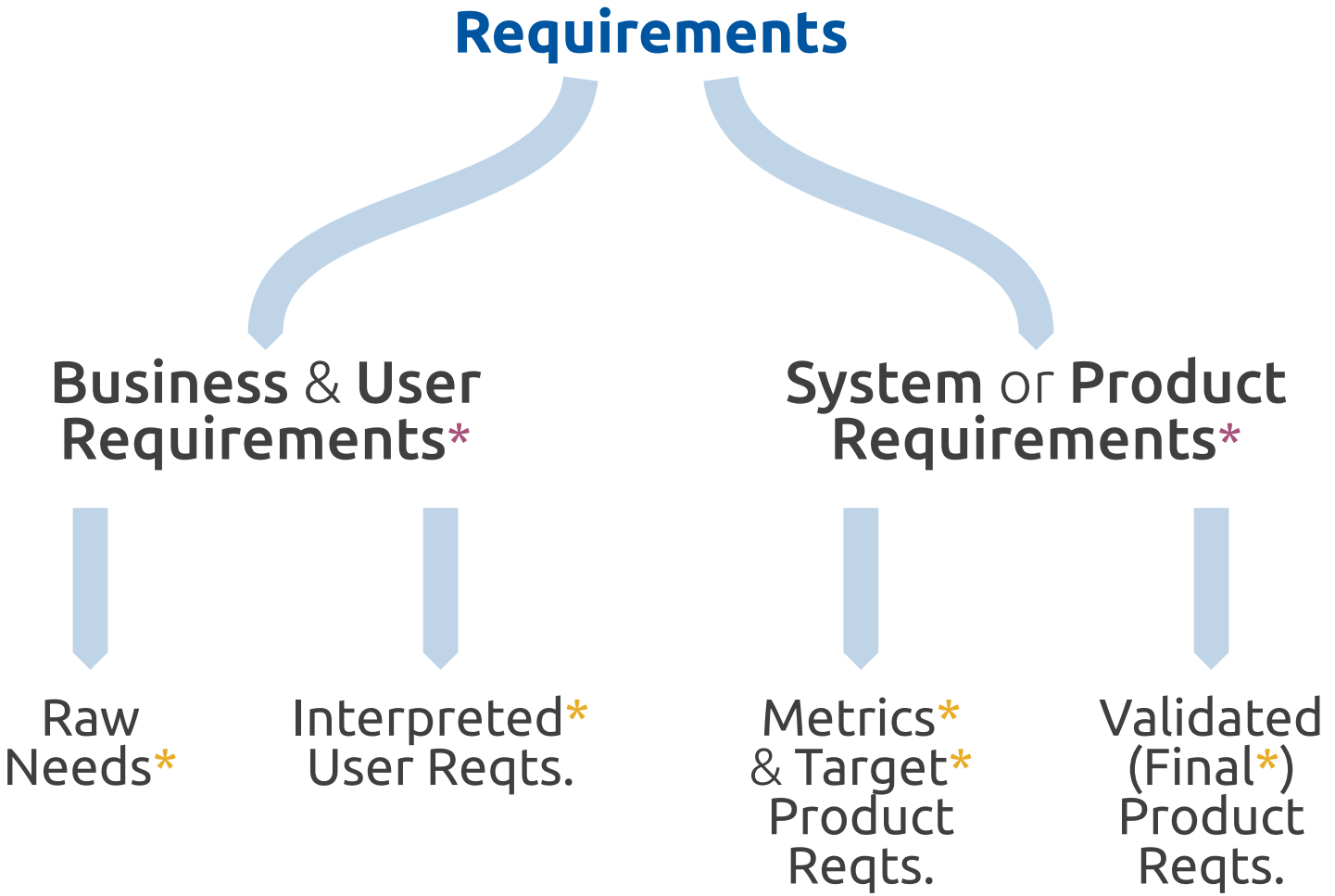
- Procurement and Purchasing → **technical specification** writting
- Quality Management → **QFD** (Quality Function Deployment) and the ***House of Quality*** ← 60's-70's in Japan
- New Product Development → gathering **customers needs** and translating them into **specifications** or specification items ← 80's
- Software Engineering → capturing **users requirements** ← 90's
- Systems Engineering → identifying **users** vs. **functional** vs. **non-functional requirements** ← ca. 2005

Typology of Requirements



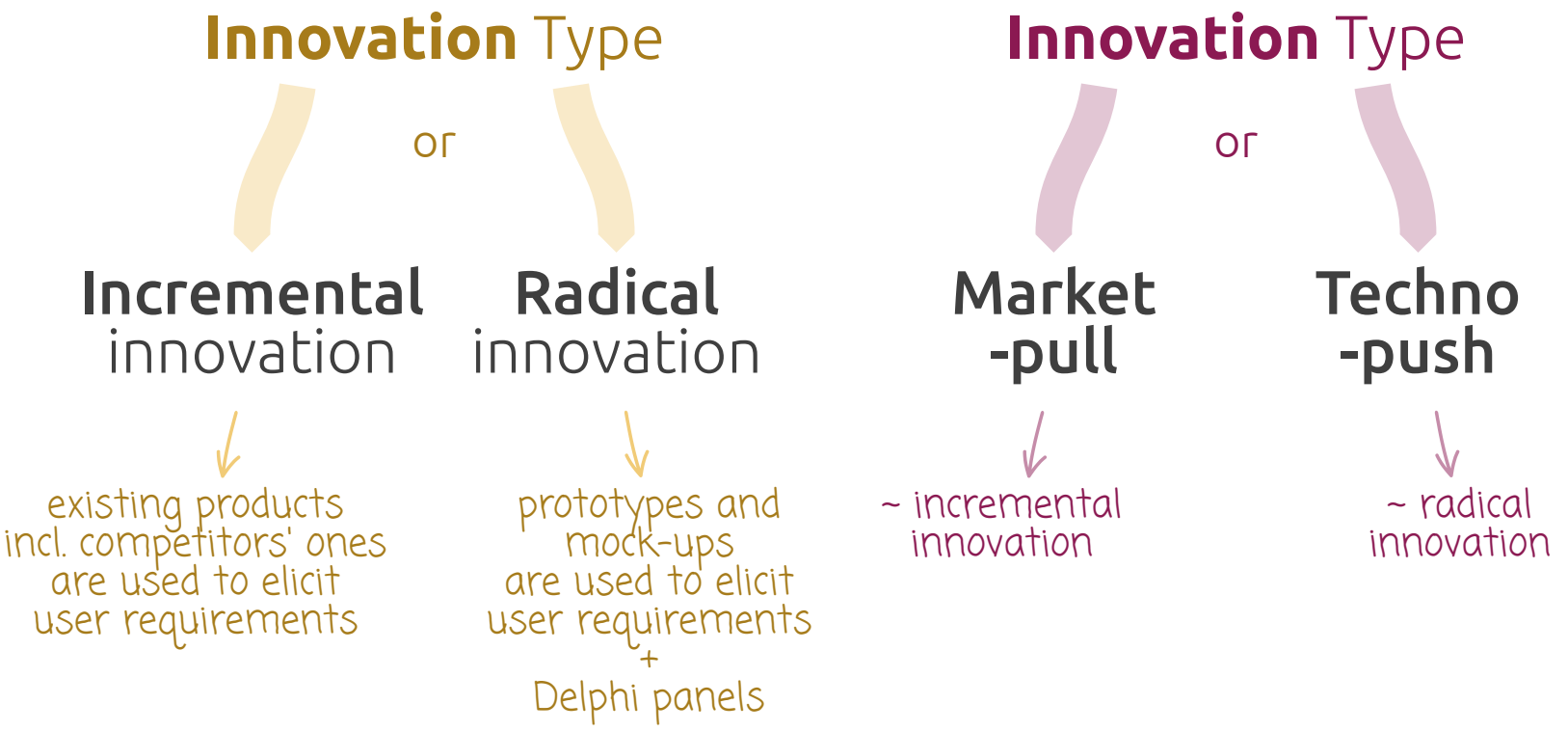
- *ISO/IEC/IEEE 29148:2011 Requirements Engineering
- *Karl Ulrich, Steve Eppinger (2011) Product Design and Development. McGraw-Hill/Irwin
- *Nam-pyo Suh (1990) Principles of Design. Oxford University Press
- *John Hauser, Don Clausing (1988) The House of Quality. HBR

Typology of Requirements



- *ISO/IEC/IEEE 29148:2011 Requirements Engineering
- *Karl Ulrich, Steve Eppinger (2011) Product Design and Development. McGraw-Hill/Irwin

Triggers to Requirements Engineering



Basic Types of Requirements

Requirements

Functional Requirements

What the product shall **do**

i.e. Function of **FFF**

e.g. to hold two parts together

Mixed Requirements

quite often Fit of **FFF**

e.g. pan head parallel to part surfaces

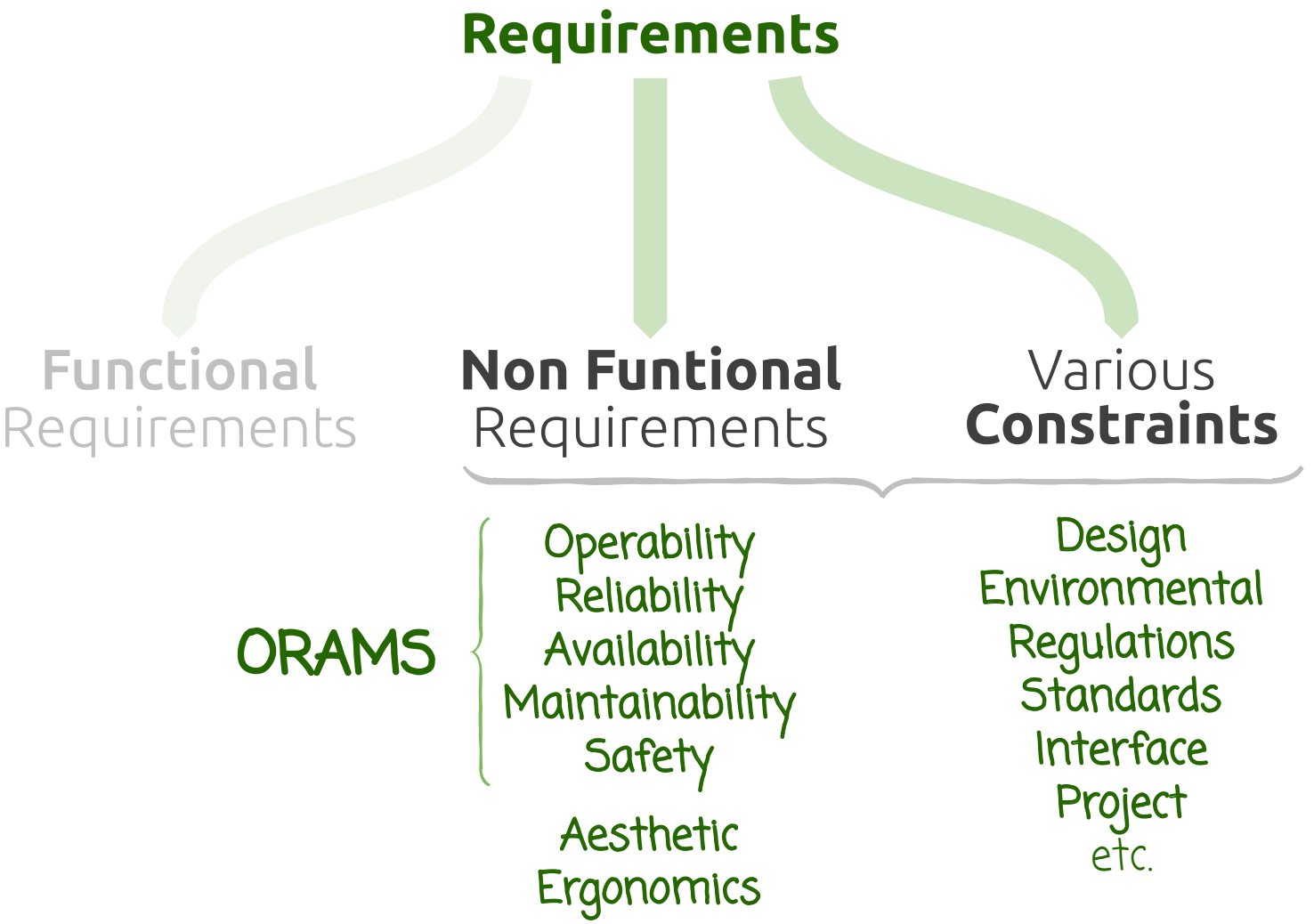
Non Fonctional Requirements

How the product shall **be**

i.e. Form of **FFF**

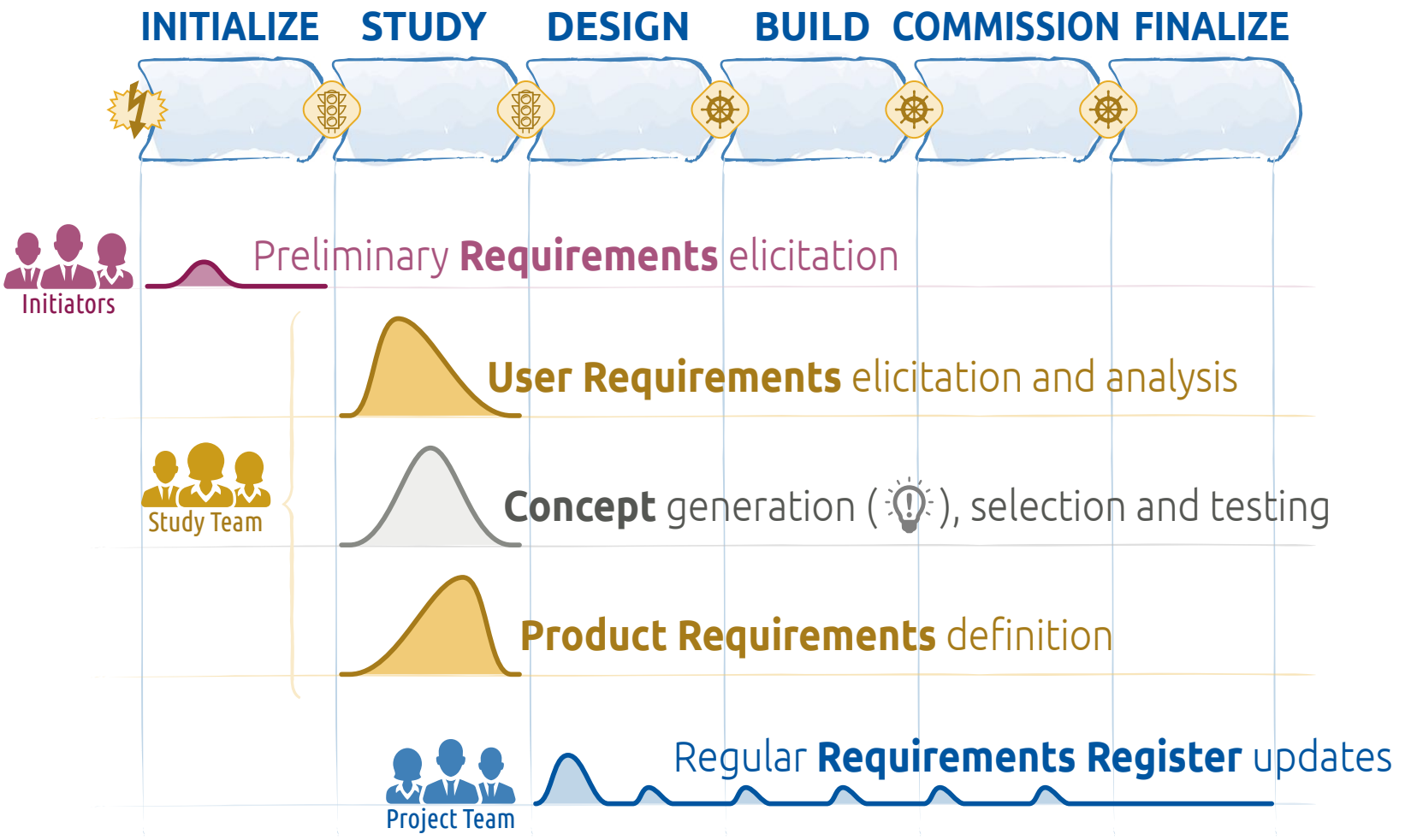
e.g. screw, pan head, M5 x 0.8, 20 mm, 316 SS + washer + bolt

Basic Types of Requirements



Requirements Engineering

When and which effort?



Requirements Engineering

A nine-step process ➔ **incremental** innovations / **market-pull** products

1. Identifying the **stakeholders** (end users, key users, customers, etc.)
2. Elicitating the **user* requirements** (~ needs gathering*)
3. Searching for **solutions** (~ concept generation*)
4. Translating the **user requirements** into **target product requirements** (~ target specifications setting*)
5. Benchmarking the **solutions** (~ concept selection and testing*)
6. Setting the **final product requirements** (~ final specifications setting*)
7. Developing the **solution**, the **product**, the **service** or the **facility** (i.e. going through the **DESIGN** and **BUILD** phases)
8. Verifying the product w.r.t. the **product requirements**
9. Validating the end product w.r.t. the **user requirements**

*User requirements include business/scientific requirements

*Karl Ulrich, Steve Eppinger (2011) Product Design and Development. McGraw-Hill/Irwin

Requirements Engineering

A nine-step process → **radical** innovations / **techno-push** products

1. Searching for **solutions** (~ concept generation*)
2. Identifying the **stakeholders** (end users, key users, customers, etc.)
3. Elicitating the **user* requirements** (~ needs gathering*)
4. Translating the **user requirements** into **target product requirements** (~ target specifications setting*)
5. Benchmarking the **solutions** (~ concept selection and testing*)
6. Setting the **final product requirements** (~ final specifications setting*)
7. Developing the **solution**, the **product**, the **service** or the **facility** (i.e. going through the **DESIGN** and **BUILD** phases)
8. Verifying the product w.r.t. the **product requirements**
9. Validating the end product w.r.t. the **user requirements**

*User requirements include business/scientific requirements

*Karl Ulrich, Steve Eppinger (2011) Product Design and Development. McGraw-Hill/Irwin

5.1

User Requirements

User Requirements

1. Identifying the **stakeholders** (end users, key users, customers, etc.)
2. Elicitating the **user requirements**
   - 2.1 Gathering **raw needs**

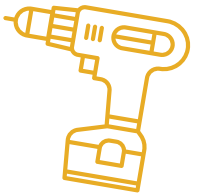
When and why do you (or will you) **use** this product?
Walk us through a **typical usage** of it
What do you **like** (👍) about the (existing) product?
What do you **dislike** (👎) about the (existing) product?
What issues do/will you consider when acquiring it?
What **improvements** would you make to it?

   - 2.2 Translating raw data into **interpreted user requirements**
   - 2.3 Organizing the IUR's into a list → prelim. **Requirements Register***

*Stakeholder Requirements Specification (StRS) or preliminary Systems Requirements Specification (SyRS)

User Requirements

2.1 Gathering **raw needs**



cordless screwdriver
incremental innovation
market-pull

I need to drive screws fast, faster than by hand.
I sometimes do duct work; use sheet metal screws.
A lot of electrical; switch covers, outlets,
fans, kitchen appliances.
I like the pistol grip; it feels the best.
I like the magnetized tip.
I don't like it when the tip slips off the screw.
I would like to be able to lock it so I can use it with a dead battery.
Can't drive screws into hard wood.
Sometimes I strip tough screws.
An attachment to allow me to reach down skinny holes.
A point so I can scrape paint off of screws.
Would be nice if it could punch a pilot hole.

User Requirements

2.2 Translating raw data into **interpreted user requirements**

- Raw needs → *"in any vernacular spoken by the users"*
- Requirements → in a formal language*, a.k.a. *"shall-statements"* or "**deontic statements**"

  this applies to all types of requirements

  - "**Shall**" indicates **mandatory** or **binding** requirements strictly to be followed in order to conform and from which no deviation is permitted
    ("shall" equals "is required to")
  - "**Should**" indicates that among several possibilities one is **recommended** as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required
    ("should" equals "is recommended that")

*ISO/IEC/IEEE 29148:2011 Requirements Engineering → § 5.2.4 Requirements Constructs

User Requirements

2.2 Translating raw data into **interpreted user requirements**

- "**May**" is used to indicate a course of **action permissible**, of **allowance** or **suggestion**
  ("may" equals "is permited to")
- "**Can**" is used for statements of **possibility** and **capability**, whether material, physical, or causal
  ("can" equals "is able to")
- "shall" or "should" or "may" or "can"

Mandatory — Desirable — Optional — Possible (future enhancement)

- ~~"must"~~

"It is best to avoid using the term 'must' due to potential misunderstanding as a requirement"*

- "**Will**" is used for **statement of fact**, **futurity**, or **declaration of purpose**

*ISO/IEC/IEEE 29148:2011 Requirements Engineering → p. 10

User Requirements

2.2 Translating raw data into **interpreted user requirements**

- "**Is/are**" are used for **non requirement statements**
  "Is/are-statements" aim at providing information in the StRS or SyRS
  → "**epistemic statements**" (opposite to "**deontic statements**")

- The **active voice** shall be preferred to the passive voice
  (passive voices promotes ambiguity and leads to needlessly complex sentences)

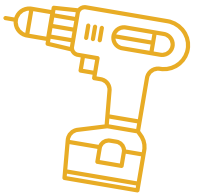
- **Positive** "shall-statements" shall be preferred to negative ones
  (such as "the system shall not do this")

- One shall be carefull to possible multiple meanings

  "Flying aircraft may be hazardous"
  ('flying' may act as a noun, an adjective or a verb!)

User Requirements

2.2 Translating raw data into **interpreted user requirements**



cordless screwdriver
incremental innovation
market-pull

The CSD **shall** drive screws faster than by hand
The CSD **shall** drive sheet metal screws into metal duct work
The CSD **can** be used for screws on electrical devices
The CSD **shall** be comfortable to grip
The CSD tip **shall** retain the screw before it is driven
The CSD tip **shall** remain aligned with the screw head without slipping
The user **can** apply torque manually to the CSD to drive a screw
The CSD **can** drive screws into hard wood
The CSD **shall** not strip screw heads
The CSD **can** access screws at the end of deep, narrow holes
The CSD **shall** allow the user to work with screws that were painted over
The CSD **can** be used to create a pilot hole

User Requirements

2.3 Organizing the IUR's into a list → prelim. **requirements register**

- Merging all interpreted user requirements in a list
- From a few dozens to several hundred IUR's
- Eliminating redundant “shall-statements”
- Flagging them: **M**andatory, **D**esirable, **O**ptional, **P**ossible
- Grouping them according to the similarities of the needs they express

⚠ UR's (and IUR's) can be contradictory! → "the product shall be red"
"the product shall be blue"

- Requirements breakdown into more focused requirements

Diagrams

9 diagrams for modelling systems

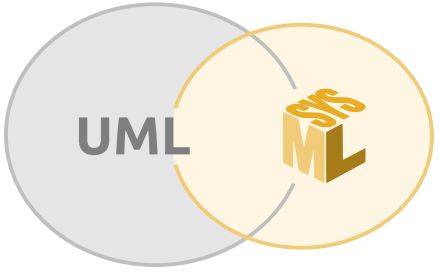
Behaviour Diagrams

- Activity Diagram
- Sequence Diagram
- State Machine Diagram
- Use Case Diagram

Requirements Diagram

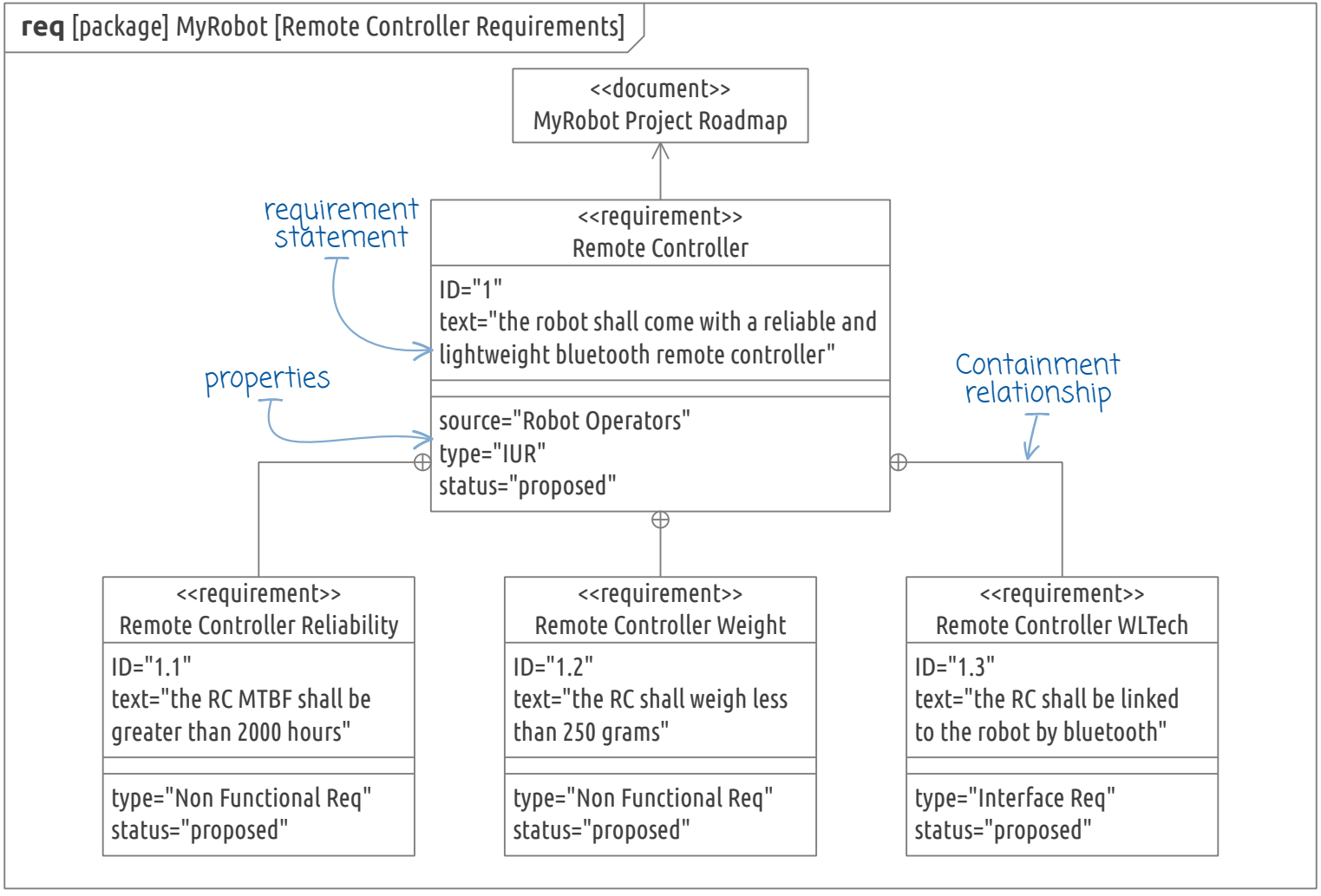
Structure Diagrams

- Block Definition Diagram
- Internal Block Diagram
  - Parametric Diagram
- Package Diagram

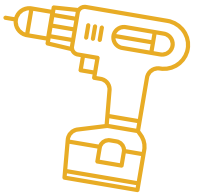


User Requirements

2.3 Organizing the IUR's into a list → requirements diagram



The CSD shall provide plenty of power to drive screws
The CSD shall maintain power for several hours of heavy use
The CSD can drive screws into hardwood
The CSD shall drive sheet metal screws into metal ductwork
The CSD shall drive screws faster than by hand
The CSD shall make it easy to start a screw
The CSD shall retain the screw before it is driven
The CSD can be used to create a pilot hole
The CSD shall work with a variety of screws
The CSD can turn Phillips, Torx, socket, and hex head screws
The CSD can turn many sizes of screws
The CSD can access most screws
The CSD can be maneuvered in tight areas
The CSD can access screws at the end of deep, narrow holes
The CSD should turn screws that are in poor condition
The CSD can be used to remove grease and dirt from screws
The CSD shall allow the user to work with painted screws
The CSD shall feel good in the user's hand
The CSD shall be comfortable when the user pushes on it
The CSD shall be comfortable when the user resists twisting
The CSD shall be balanced in the user's hand
The CSD shall be equally easy to use in right or left hands
The CSD weight should be just right
The CSD shall be warm to touch in cold weather
The CSD shall remain comfortable when left in the sun
The CSD shall be easy to control while turning screws
The user can easily push on the CSD
The user can easily resist the CSD twisting
The CSD can be locked on.
The CSD speed can be controlled by the user while turning a screw
The CSD shall remain aligned with the screw head without slipping
The user can easily see where the screw is



The CSD shall not strip screw heads
The CSD shall be easily reversible
The CSD shall be easy to set up and use
The CSD shall be easy to turn on
The CSD shall prevent inadvertent switching off
The user can set the maximum torque of the CSD
The CSD shall provide ready access to bits or accessories
The CSD power shall be convenient
The CSD can be attached to the user for temporary storage
The CSD shall be easy to recharge.
The CSD can be used while recharging
The CSD shall recharge quickly
The CSD batteries shall be ready to use when new
The user can apply torque manually to the CSD to drive a screw
The CSD last a long time
The CSD tip shall survive heavy use
The CSD can be hammered
The CSD can be dropped from a ladder without damage
The CSD shall be easy to store
The CSD shall fit in a toolbox easily
The CSD can be charged while in storage
The CSD shall resist corrosion when left outside or in damp places
The CSD shall maintain its charge after long periods of storage
The CSD shall maintain its charge when wet
The CSD shall prevent damage to the work
The CSD shall prevent damage to the screw head
The CSD shall prevent scratching of finished surfaces
The CSD shall have a pleasant sound when in use
The CSD shall look like a professional quality tool
The CSD shall be safe
The CSD can be used on electrical devices
The CSD shall not cut the user's hands

5.2

Solutions / Concepts

openSE

Requirements Engineering

A nine-step process ➔ **incremental** innovations / **market-pull** products

1. Identifying the **stakeholders** (end users, key users, customers, etc.)
2. Collecting the **user requirements** (~ needs gathering)
3. Searching for **solutions** (~ concept generation)
4. Translating the **user requirements** into **target product requirements** (~ target specifications setting)
5. Benchmarking the **solutions** (~ concept selection and testing)
6. Setting the **final product requirements** (~ final specifications setting)
7. Developing the **solution**, the **product**, the **service** or the **facility** (i.e. going through the **DESIGN** and **BUILD** phases)
8. Verifying the product w.r.t. the **product requirements**
9. Validating the end product w.r.t. the **user requirements**

5.2.1

INTERMEDIATE approach

ADVANCED approach

Product Breakdown Structure

openSE

Coordination Planning & Scheduling

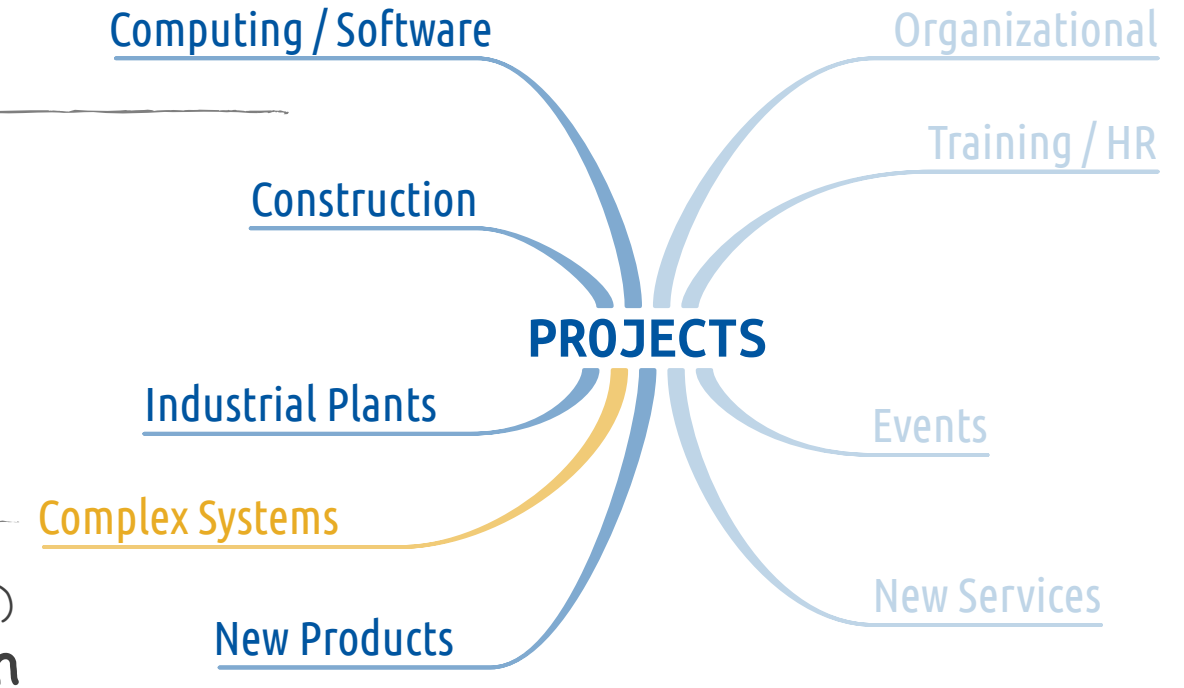
1 Identifying the project **activities** → analytical approach

1.1 Describing the final **deliverable**(**s**)

the **software**

the **building**
the **facility**
the **system**
the **prototype**

the **product** (item)
+ the **supply chain**
(i.e. the manufacturing facility)

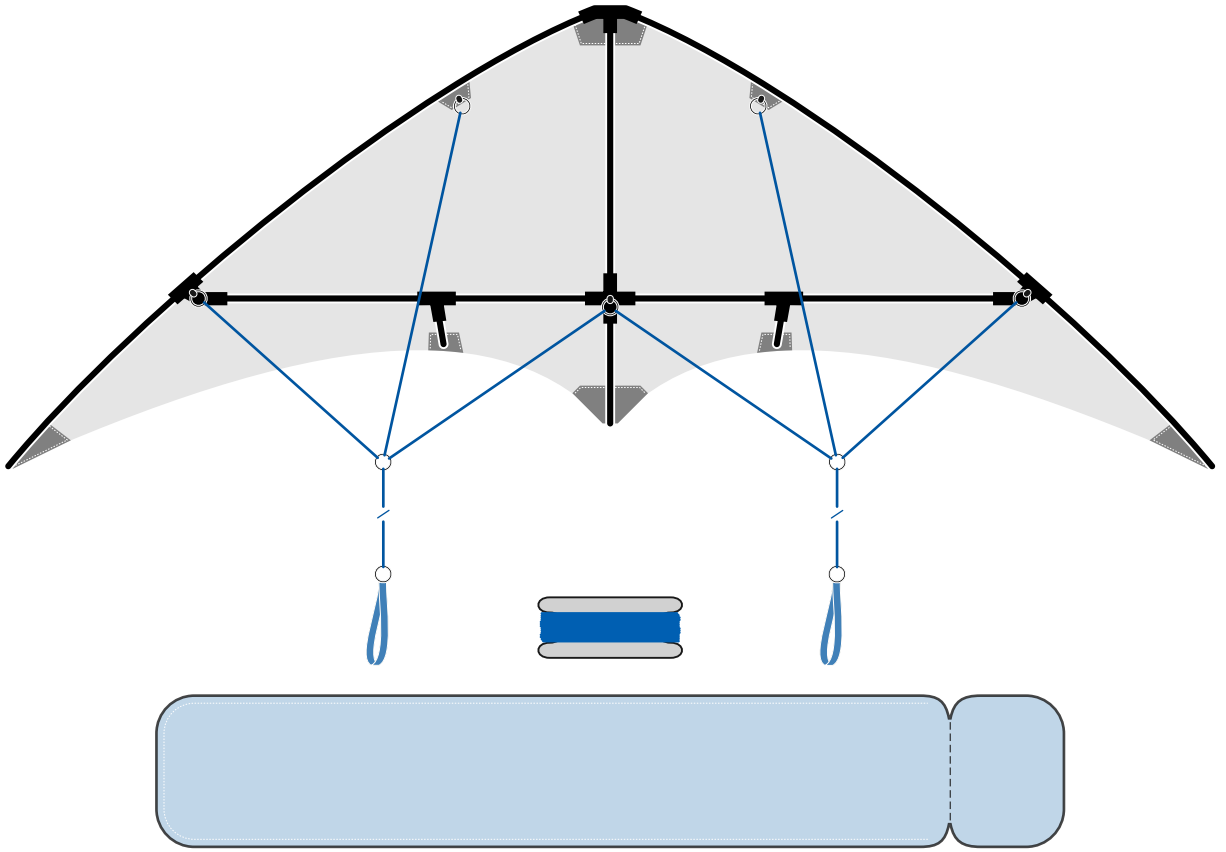


⚠ It can become a rather complex mix of “objects”

Coordination Planning & Scheduling

1 Identifying the project **activities** → analytical approach

1.1 Describing the final **deliverable**(**s**)



User's manual

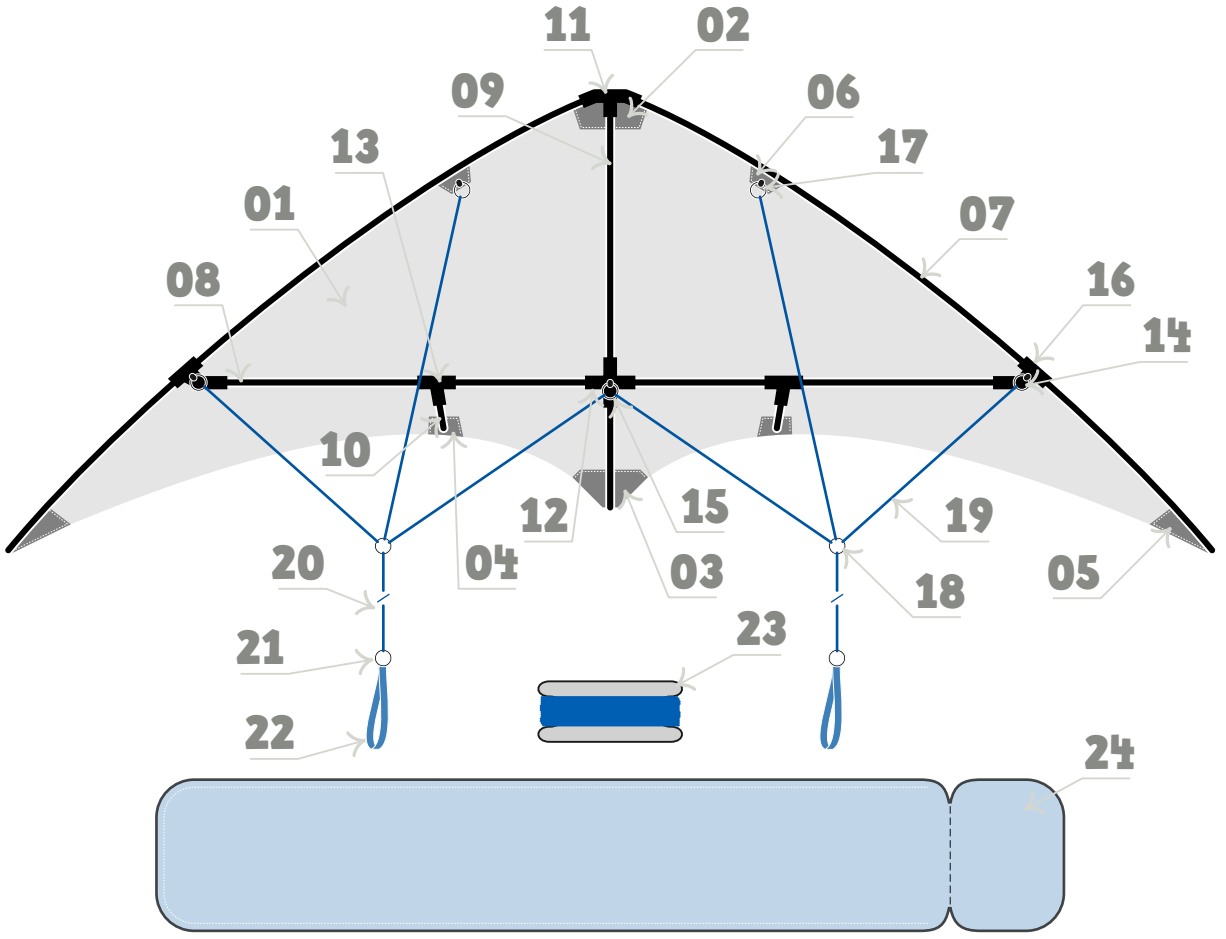
Safety instr.

BRZ·KITE

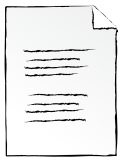
Coordination Planning & Scheduling

1 Identifying the project **activities** → analytical approach

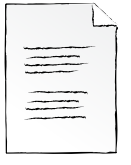
1.1 Describing the final **deliverable**(s)



25 User's manual



26 Safety instr.



BRZ·KITE

ID	Component description	Qty	Component specication
01	Wing surface (canopy)	1	1800 mm × 700 mm ; 0.5 m^2 ; Polyamide 5.5 nylon
02	Nose reinforcement piece	1	Reinforced polyamide nylon
03	Tail reinforcement piece	1	Reinforced polyamide nylon
04	Wisker reinforcement piece	2	Reinforced polyamide nylon
05	Wing end reinforcement piece	2	Reinforced polyamide nylon
06	Wing side reinforcement piece	2	Reinforced polyamide nylon
07	Wing side yard	2	Ø6 mm × 1100 mm ; carbon rod
08	Rear yard	2	Ø6 mm × 600 mm ; carbon rod
09	Longitudinal yard	1	Ø6 mm × 550 mm ; carbon rod
10	Wisker	2	Ø6 mm × 120 mm ; carbon rod
11	Nose yard junction tee	1	Cycolac ABS
12	Central cross	1	Cycolac ABS
13	Wisker junction tee	2	Cycolac ABS
14	Wing side junction tee	2	Cycolac ABS
15	Central tying ring	1	Ø10 mm × 0.4 mm^2 ; stainless steel
16	Wing end tying ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
17	Wing side tying ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
18	Line attachment ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
19	Tie	6	0.1 mm^2 × ca. 800 mm ; nylon rope
20	Line	2	0.15 mm^2 × ca. 800 mm ; nylon rope
21	Handle ring	2	Ø20 mm × 1 mm^2 ; stainless steel
22	Handle	2	20 mm width nylon strap
23	Line winder	1	Cycolac ABS
24	Storage bag	1	Transparent nylon
25	User's manual	1	Printed material ; A6 format
26	Safety instructions	1	Printed material ; A6 format

BRZ·KITE

ID	Component description	Qty	Component specication
01	Wing surface (canopy)	1	1800 mm × 700 mm ; 0.5 m^2 ; Polyamide 5.5 nylon
02	Nose reinforcement piece	1	Reinforced polyamide nylon
03	Tail reinforcement piece	1	Reinforced polyamide nylon
04	Wisker reinforcement piece	2	Reinforced polyamide nylon
05	Wing end reinforcement piece	2	Reinforced polyamide nylon
06	Wing side reinforcement piece	2	Reinforced polyamide nylon
07	Wing side yard	2	Ø6 mm × 1100 mm ; carbon rod
08	Rear yard	2	Ø6 mm × 600 mm ; carbon rod
09	Longitudinal yard	1	Ø6 mm × 330 mm ; carbon rod
10	Wisker	2	Ø6 mm × 120 mm ; carbon rod
11	Nose yard junction tee	1	Cycolac ABS
12	Central cross	1	Cycolac ABS
13	Wisker junction tee	2	Cycolac ABS
14	Wing side junction tee	2	Cycolac ABS
15	Central tying ring	1	Ø10 mm × 0.4 mm^2 ; stainless steel
16	Wing end tying ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
17	Wing side tying ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
18	Line attachment ring	2	Ø10 mm × 0.4 mm^2 ; stainless steel
19	Tie	1	0.1 mm^2 × ca. 800 mm ; nylon rope
20	Line	2	0.15 mm^2 × ca. 800 mm ; nylon rope
21	Handle ring	2	Ø20 mm × 1 mm^2 ; stainless steel
22	Handle	2	20 mm width nylon strap
23	Line winder	1	Cycolac ABS
24	Storage bag	1	Transparent nylon
25	User's manual	1	Printed material ; A6 format
26	Safety instructions	1	Printed material ; A6 format

Fabrics

Carbon rods

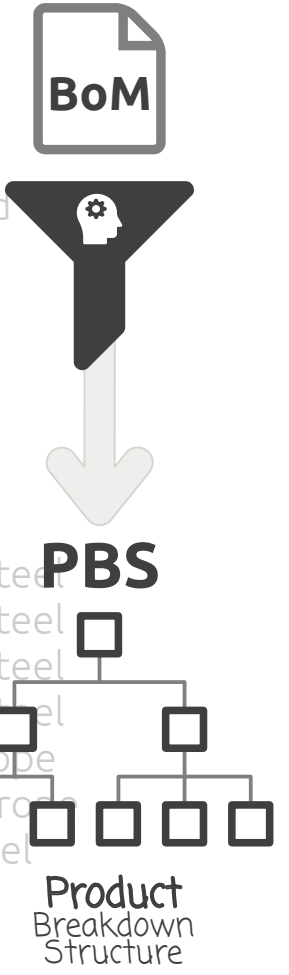
Moulded ABS parts

Rings (COTS)

Rope & strap

Bag

Printed material



BRZ-KITE

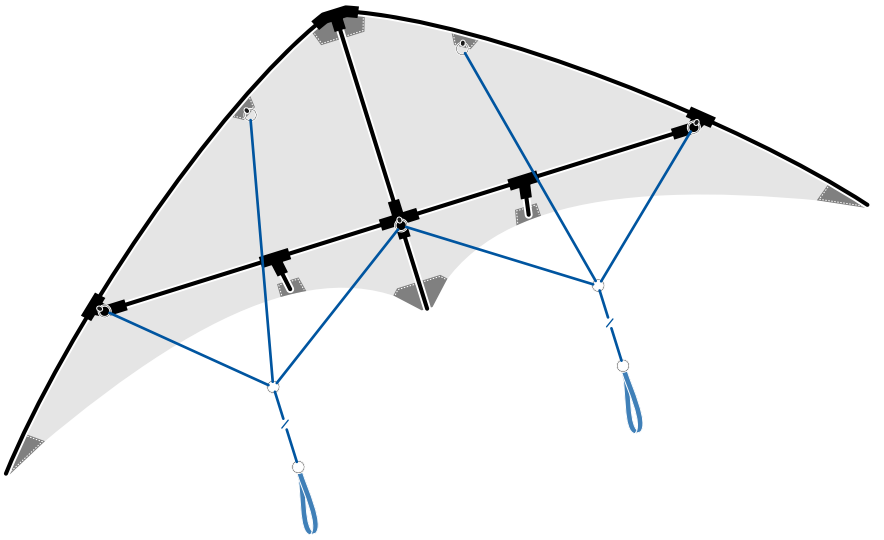
Coordination Planning & Scheduling

1 Identifying the project **activities** → analytical approach

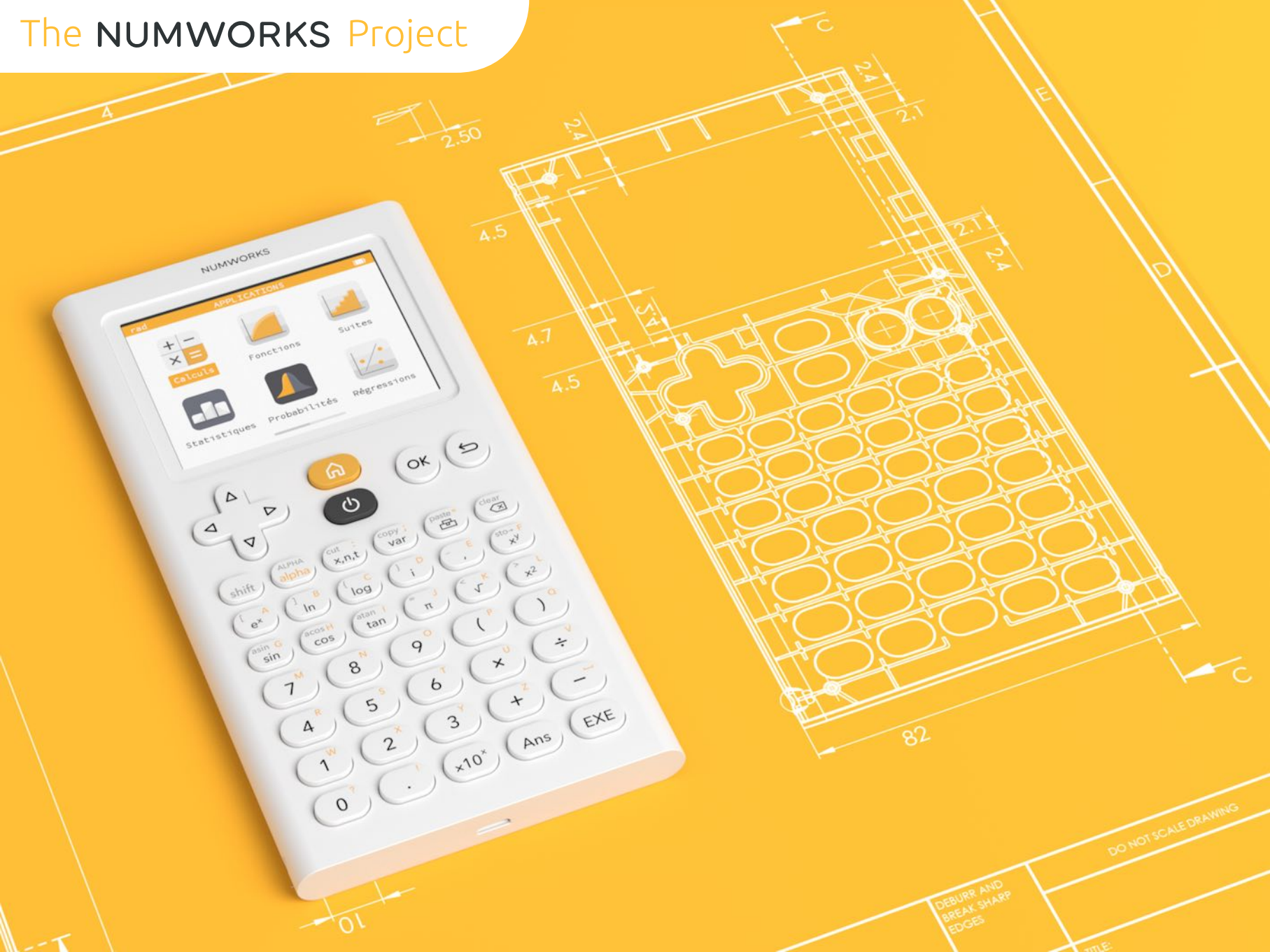
1.1 Describing the final **deliverable**(**s**)

Product Breakdown Structure

- BRZ·KITE
  - Fabrics
  - Carbon rods
  - Moulded ABS parts
  - Rings (COTS)
  - Rope & strap
  - Storage bag
  - Printed material



The NUMWORKS Project



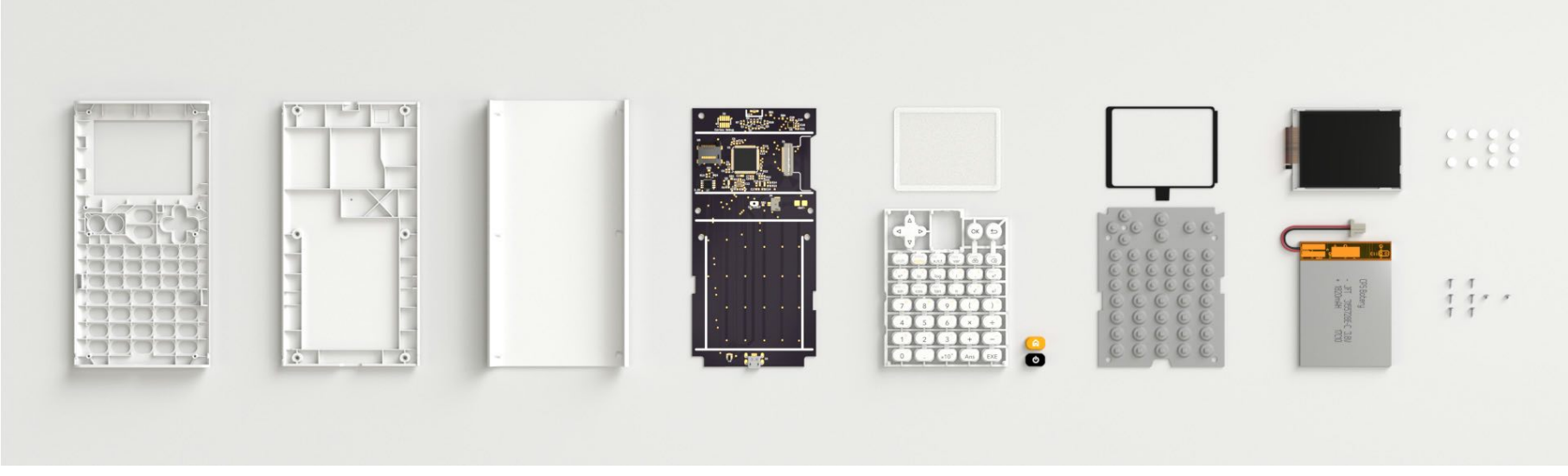
*Preliminary Bill of Materials (**BoM**)*

packaging
& documents

software

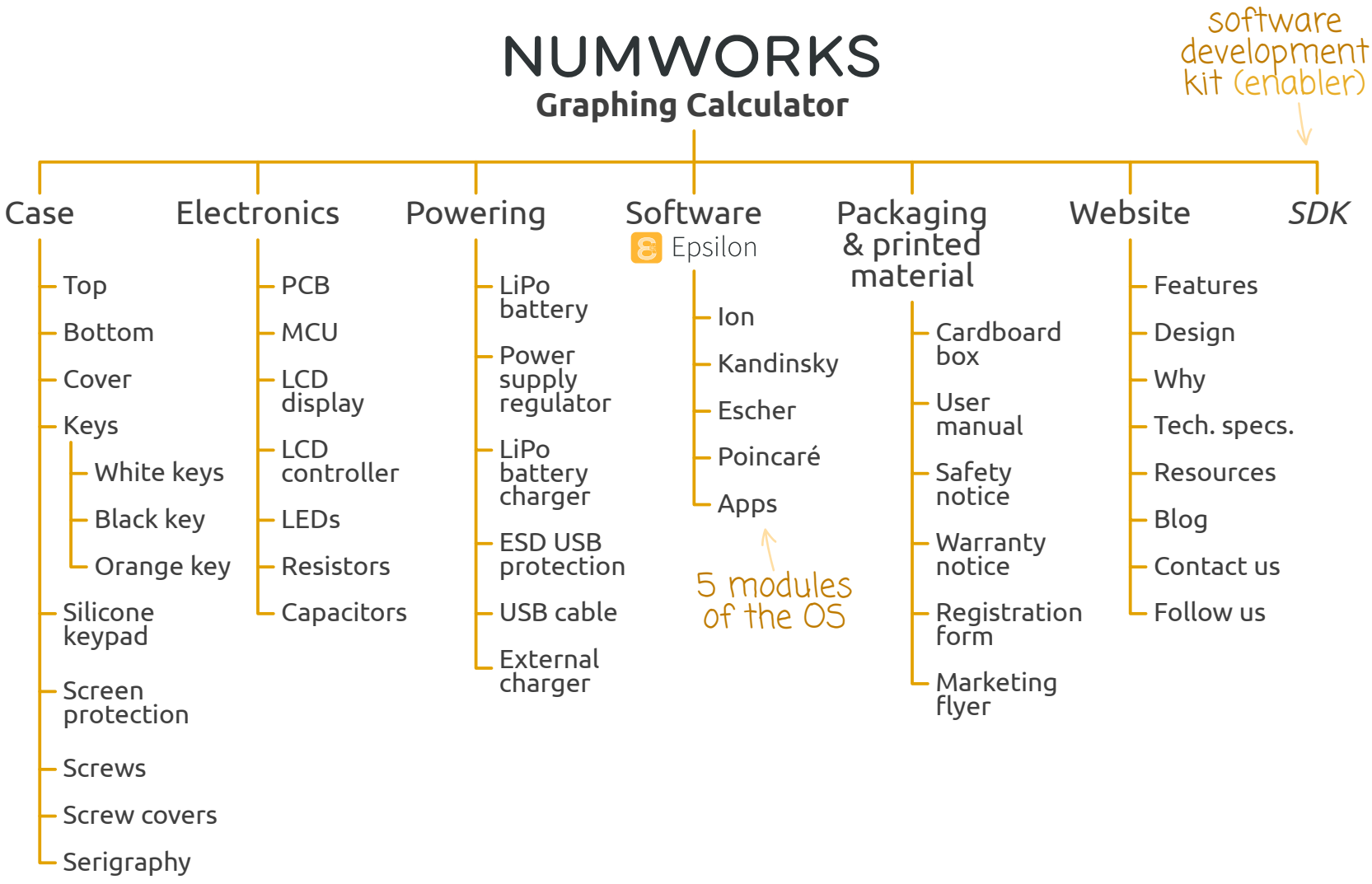
Epsilon

cable & charger

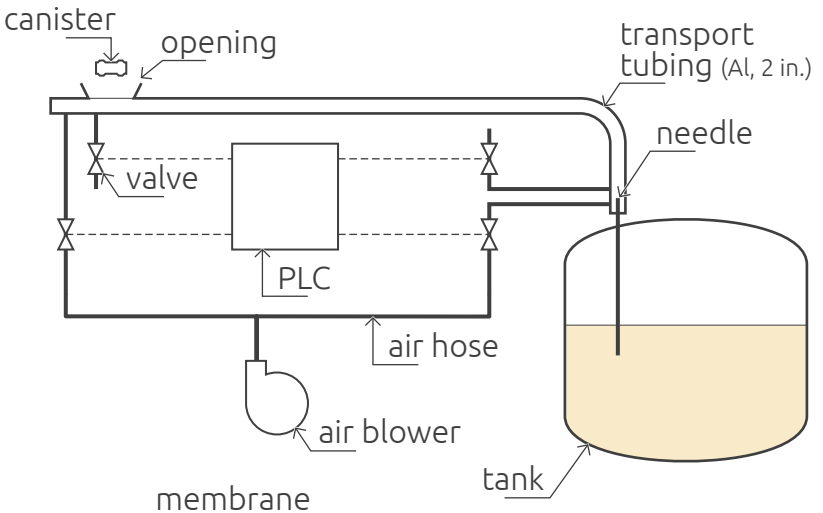


product website

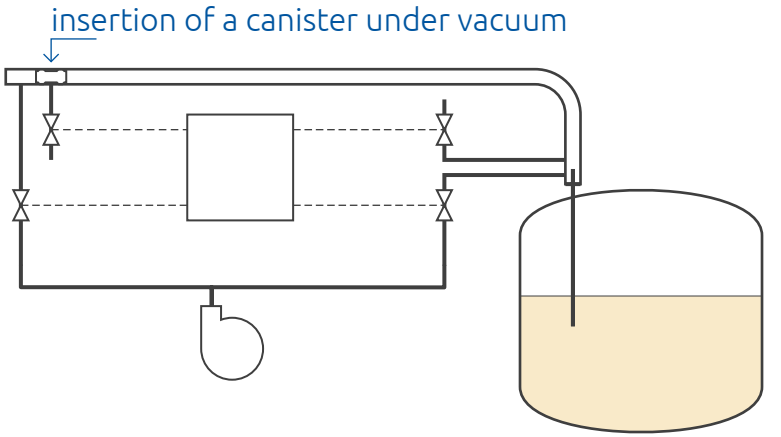
*Product Breakdown Structure (**PBS**)*

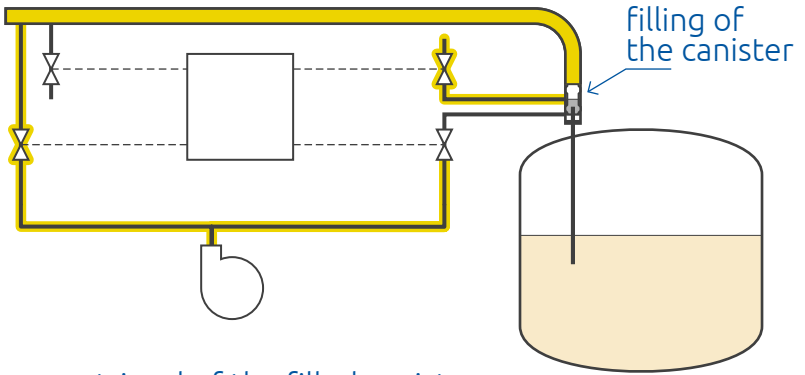


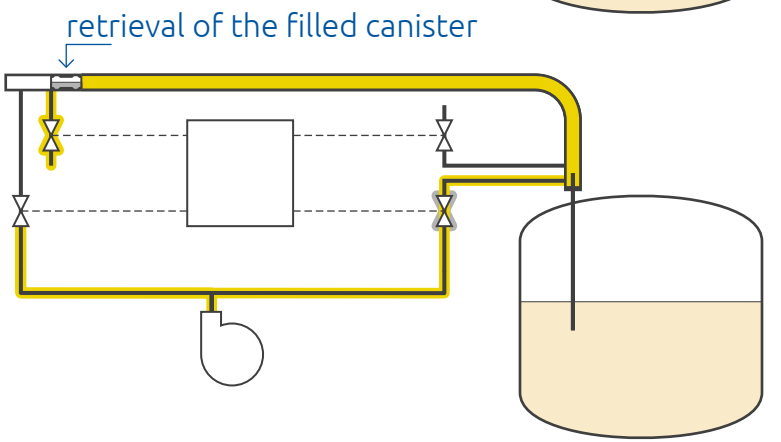
The process


canister
opening
transport
tubing (Al, 2 in.)
needle
valve
PLC
air hose
air blower
tank

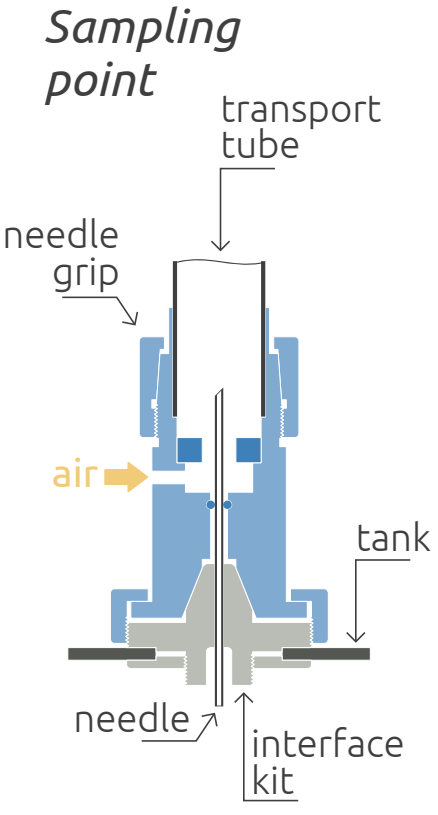

membrane

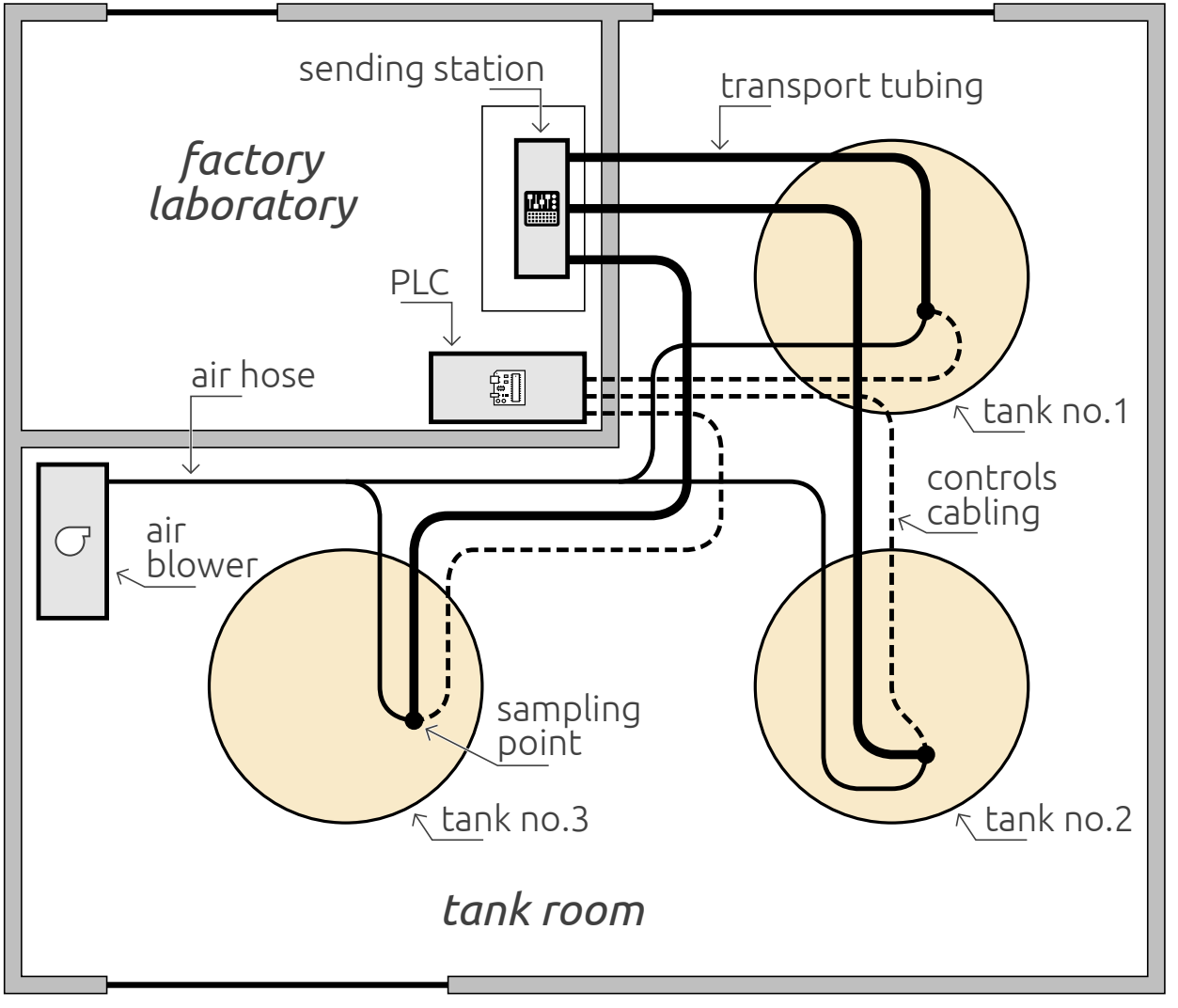

insertion of a canister under vacuum


filling of
the canister

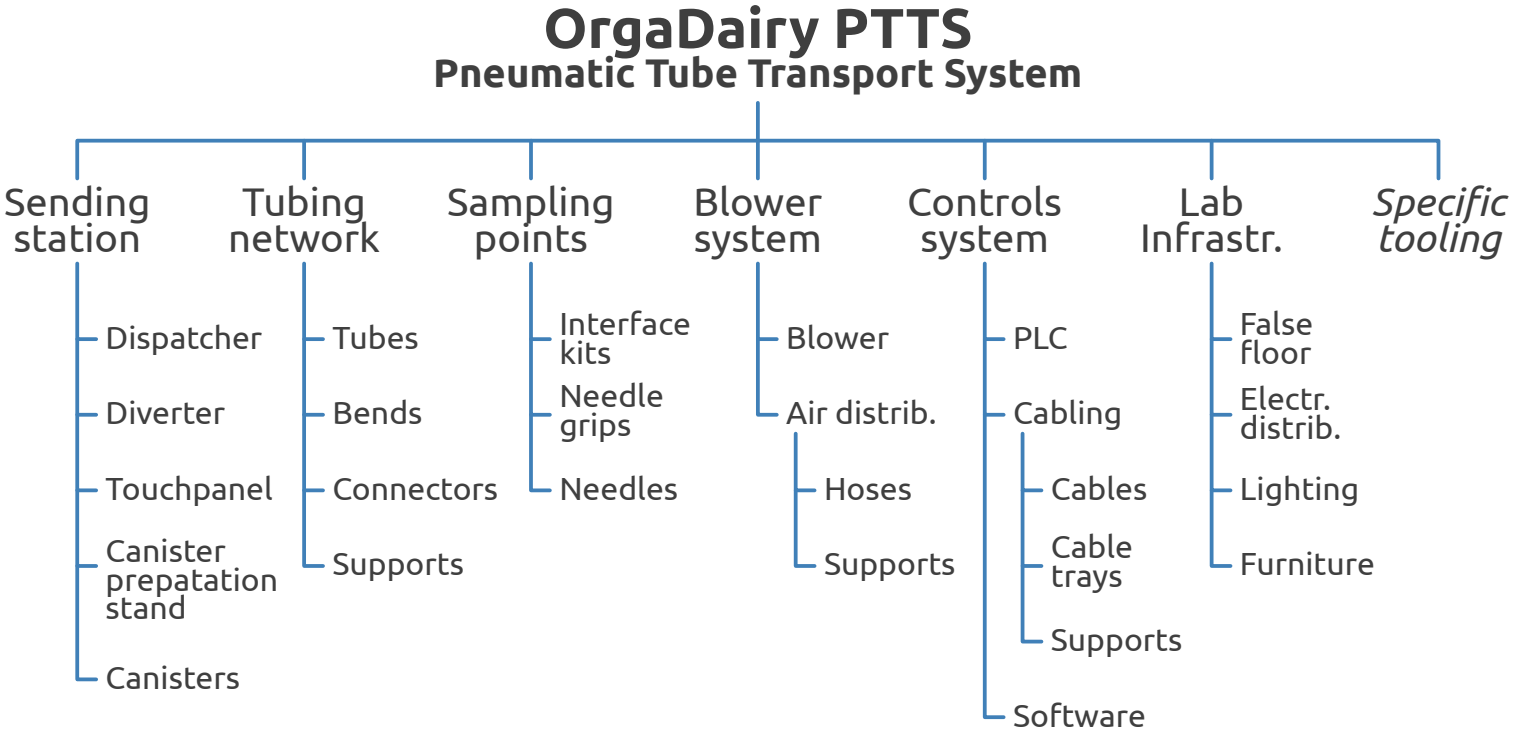

retrieval of the filled canister

The layout


Sampling point
transport tube
needle grip
air
tank
needle
interface kit


sending station
transport tubing
factory laboratory
PLC
air hose
tank no.1
controls cabling
air blower
sampling point
tank no.3
tank no.2
tank room

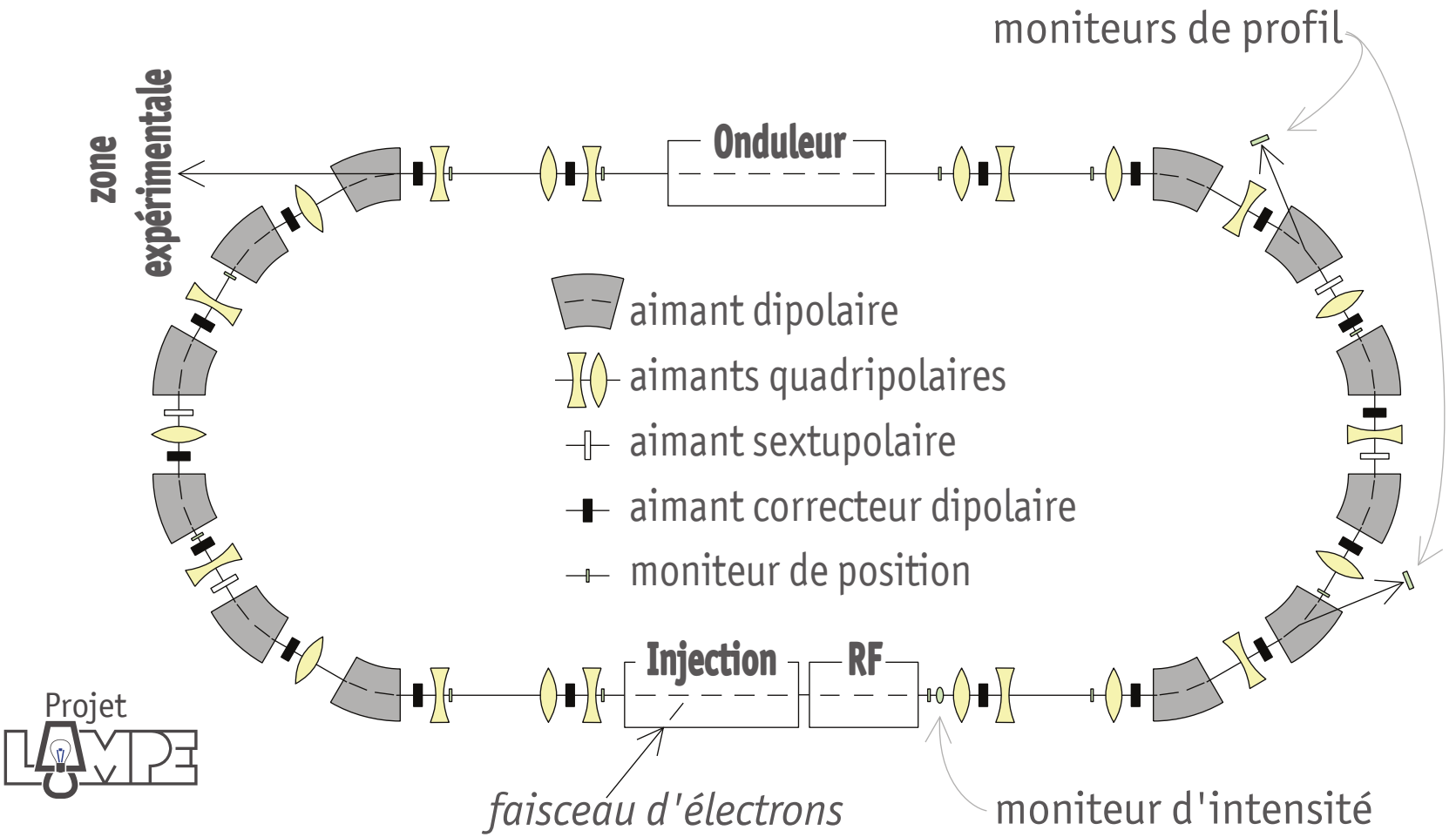
*Product Breakdown Structure (**PBS**)*



Coordination Planning & Scheduling

1 Identifying the project **activities** → analytical approach

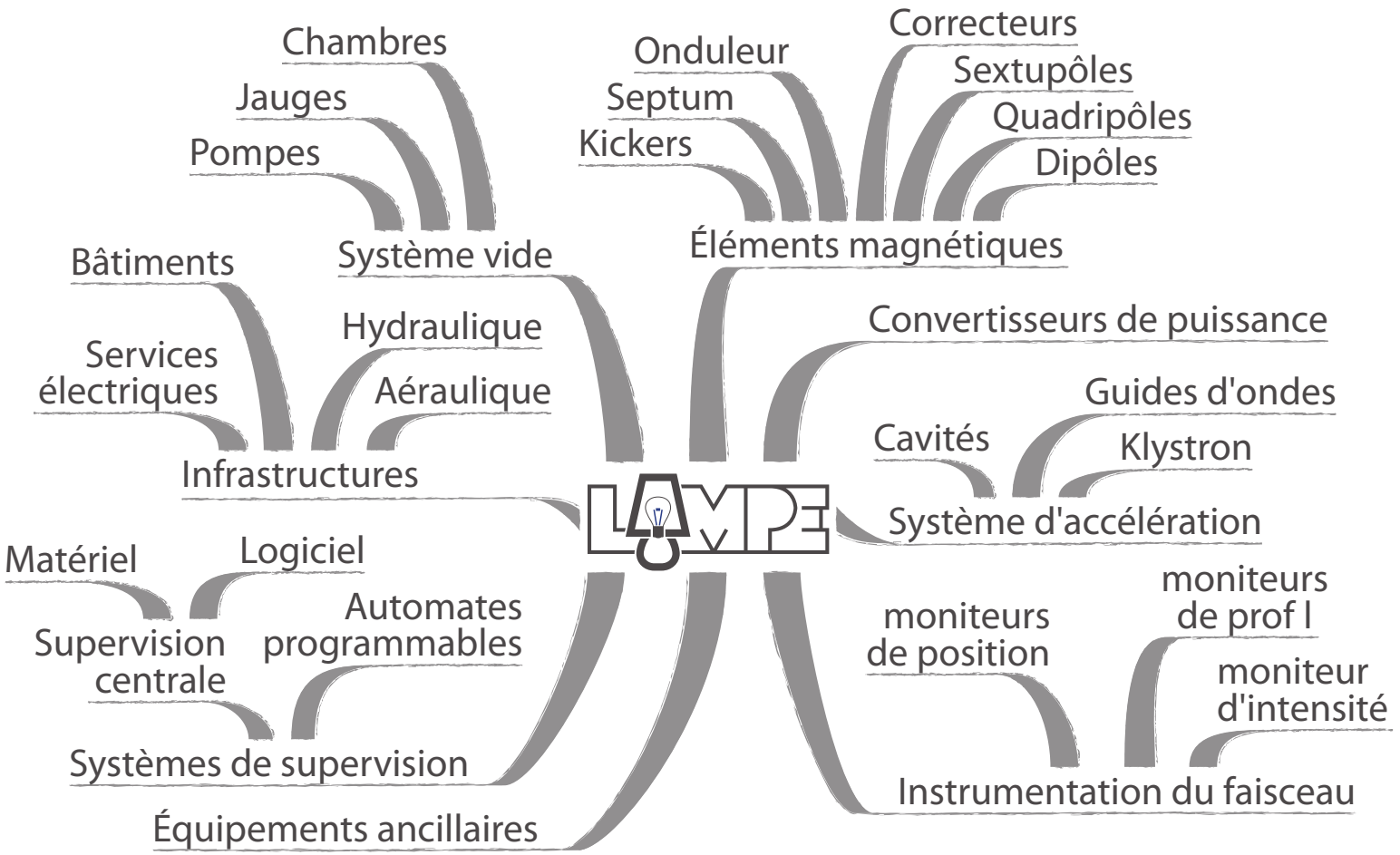
1.1 Describing the final **deliverable**(**s**)



Coordination Planning & Scheduling

1 Identifying the project **activities** → analytical approach

1.1 Describing the final **deliverable**(**s**)



5.3

Product Requirements

openSE

Product Requirements

- User requirements are expressed in the language of the user
- Too much space is left for subjective interpretation
- The achievement of product requirements shall be measurable
- Product requirements are expressed in engineer's language

4 Translating the **user requirements** into **target product requirements** (~ target specifications setting)

- **4.1** Based on the IUR's, preparing a list of **metrics** → one to one mapping (House of Quality, **QFD**)
- **4.2** Collecting **competitive benchmarking** information
- **4.3** Setting ideal and marginally acceptable **target values**
- **4.4** Translate target values into **target product requirement** statements → "formal shall-statements"

4.1 Preparing a list of **metrics**

mountain bike suspension fork

ID	IUR	Imp.
1	The suspension shall reduce vibration to the hands	3
2	The suspension shall allow easy traversal of slow, difficult terrain	2
3	The suspension shall enable high-speed descents on bumpy trails	5
4	The suspension shall allow sensitivity adjustment	3
5	The suspension shall preserve the steering characteristics of the bike	4
6	The suspension shall remain rigid during hard cornering	4
7	The suspension shall be lightweight	4
8	The suspension shall provide stiff mounting points for the brakes	2
9	The suspension shall fit a wide variety of bikes, wheels, and tires	5
10	The suspension shall be easy to install	1
11	The suspension shall work with fenders	1
12	The suspension shall instill pride	5
13	The suspension shall be affordable for an amateur enthusiast	5
14	The suspension shall be not contaminated by water	5
15	The suspension shall be not contaminated by grunge	5
16	The suspension can be easily accessed for maintenance	3
17	The suspension shall allow easy replacement of worn parts	1
18	The suspension can be maintained with readily available tools	3
19	The suspension shall last a long time	5
20	The suspension shall be safe in a crash	5

4.1 Preparing a list of **metrics**

mountain bike suspension fork

ID	Metric	Imp.	Unit
1, 3	Attenuation from dropout to handlebar at 10Hz	3	dB
2, 6	Spring preload	3	N
1, 3	Maximum value from the Monster	5	g
1, 3	Minimum descent time on test track	5	s
4	Damping coefficient adjustment range	3	N-s/m
5	Maximum travel (26-in. wheel)	3	mm
5	Rake offset	3	mm
6	Lateral stiffness at the tip	3	kN/m
7	Total mass	4	kg
8	Lateral stiffness at brake pivots	2	kN/m
9	Headset sizes	5	in.
9	Steertube length	5	mm
9	Wheel sizes	5	List

4.1 Preparing a list of **metrics**

mountain bike suspension fork

ID	Metric	Imp.	Unit
9	Maximum tire width	5	in.
10	Time to assemble to frame	1	s
11	Fender compatibility	1	List
12	Instils pride	5	Subj.
13	Unit manufacturing cost	5	US$
14	Time in spray chamber without water entry	5	s
15	Cycles in mud chamber without contamination	5	k-cycles
16, 17	Time to disassemble/assemble for maintenance	3	s
17, 18	Special tools required for maintenance	3	List
19	UV test duration to degrade rubber parts	5	hr
19	Monster cycles to failure	5	cycles
20	Japan Industrial Standards test	5	binary
20	Bending strength (frontal loading)	5	kN

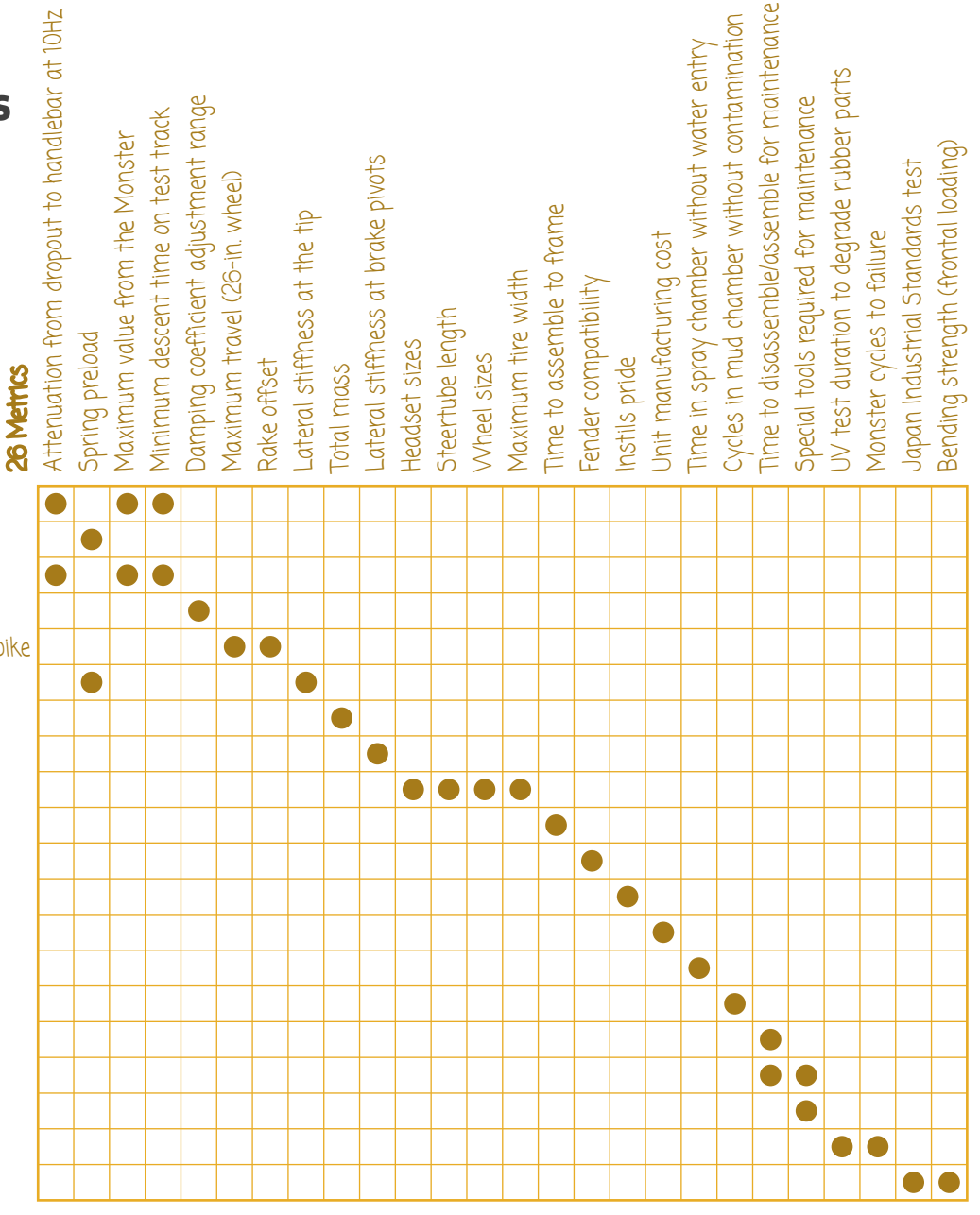
4.1 Preparing a list of **metrics**

IUR's × Metrics Matrix

mountain bike suspension fork

20 IUR's

The suspension shall reduce vibration to the hands
The suspension shall allow easy traversal of slow, difficult terrain
The suspension shall enable high-speed descents on bumpy trails
The suspension shall allow sensitivity adjustment
The suspension shall preserve the steering characteristics of the bike
The suspension shall remain rigid during hard cornering
The suspension shall be lightweight
The suspension shall provide stiff mounting points for the brakes
The suspension shall fit a wide variety of bikes, wheels, and tires
The suspension shall be easy to install
The suspension shall work with fenders
The suspension shall instill pride
The suspension shall be affordable for an amateur enthusiast
The suspension shall be not contaminated by water
The suspension shall be not contaminated by grunge
The suspension can be easily accessed for maintenance
The suspension shall allow easy replacement of worn parts
The suspension can be maintained with readily available tools
The suspension shall last a long time
The suspension shall be safe in a crash



Product Requirements

4.1 Based on the IUR's, preparing a list of **metrics**

- Metrics should be **complete** → one metric per IUR
- Metrics should be **quantifiable**
- Metrics should be **pratical** → measurable with usual measurement means
- A few metrics can be non quantificable (identified as '**Subj.**' in the list)
- Metrics may also include **popular comparison criteria**

Product Requirements

4.2 Collecting **competitive benchmarking** information

4.2.1 Identifying **benchmark products**

i.e. competitors' products, existing products offering similar features, prototypes

4.2.2 Mesuring the benchmark products w.r.t. metrics

4.2.3 Assessing the benchmark products w.r.t. IUR's

4.2.2 Mesuring the benchmark products w.r.t. metrics (1/2)

#	Metric	Imp.	Unit						
1	Attenuation from dropout to handlebar at 10Hz	3	dB	8	15	10	15	9	13
2	Spring preload	3	N	550	760	500	770	480	680
3	Maximum value from the Monster	5	g	3.6	3.2	3.7	3.3	3.7	3.4
4	Minimum descent time on test track	5	s	13	11.3	12.6	11.2	13.2	11.0
5	Damping coefficient adjustment range	3	N-s/m	0	0	0	200	0	0
6	Maximum travel (26-in. wheel)	3	mm	28	48	43	46	33	38
7	Rake offset	3	mm	41.5	39	38	38	43.2	39
8	Lateral stiffness at the tip	3	kN/m	59	110	85	84	65	130
9	Total mass	4	kg	1.409	1.385	1.409	1.364	1.222	1.100
10	Lateral stiffness at brake pivots	2	kN/m	295	550	425	425	325	650
11	Headset sizes	5	in.	1.000 1.125	1.000 1.125 1.250	1.000 1.125	1.000 1.125 1.250	1.000 1.125	N/A
12	Steertube length	5	mm	150 180 210 230 255	140 165 190 215	150 170 190 210	150 170 190 210 230	150 190 210 220	N/A
13	Wheel sizes	5	List	26 in.	26 in.	26 in.	26 in. 700 mm	26 in.	26 in.

4.2.2 Mesuring the benchmark products w.r.t. metrics (2/2)






#	Metric	Imp.	Unit						
14	Maximum tire width	5	in.	1.5	1.75	1.5	1.75	1.5	1.5
15	Time to assemble to frame	1	s	35	35	45	45	35	85
16	Fender compatibility	1	List	Zefaf	None	None	None	None	All
17	Instils pride	5	Subj.	1	4	3	5	3	5
18	Unit manufacturing cost	5	US$	65	105	85	115	80	100
19	Time in spray chamber without water entry	5	s	1300	2900	>3600	>3600	2300	>3600
20	Cycles in mud chamber without contamination	5	k-cycles	15	19	15	25	18	35
21	Time to disassemble/assemble for maintenance	3	s	160	245	215	245	200	425
22	Special tools required for maintenance	3	List	Hex.	Hex.	Hex.	Hex.	Long hex.	Hex., pin wrench
23	UV test duration to degrade rubber parts	5	hr	400+	250	400+	400+	400+	250
24	Monster cycles to failure	5	cycles	500k+	500k+	500k+	480k	500k+	330k
25	Japan Industrial Standards test	5	binary	Pass	Pass	Pass	Pass	Pass	Pass
26	Bending strength (frontal loading)	5	kN	5.5	8.9	7.5	7.5	6.2	10.2

4.2.3 Assessing the benchmark products w.r.t. IUR's






Interpreted User Requirements	Imp.						
The Susp. shall reduce vibration to the hands	3	•	••••	••	•••••	••	•••
The Susp. shall allow easy traversal of slow, difficult terrain	2	••	••••	•••	•••••	•••	•••••
The Susp. shall enable high-speed descents on bumpy trails	5	•	•••••	••	•••••	••	•••
The Susp. shall allow sensitivity adjustment	3	•	••••	••	•••••	••	•••
The Susp. shall preserve the steering characteristics of the bike	4	••••	••	•	••	•••••	•••••
The Susp. shall remain rigid during hard cornering	4	•	•••	•	•••••	•	•••••
The Susp. shall be lightweight	4	•	•••	•	•••	••••	•••••
The Susp. shall provide stiff mounting points for the brakes	2	•	••••	•••	•••	•••••	••
The Susp. shall fit a wide variety of bikes, wheels, and tires	5	••••	•••••	•••	•••••	•••	•
The Susp. shall be easy to install	1	••••	•••••	••••	••••	•••••	•
The Susp. shall work with fenders	1	•••	•	•	•	•	•••••
The Susp. shall instill pride	5	•	••••	•••	•••••	•••	•••••
The Susp. shall be affordable for an amateur enthusiast	5	•••••	•	•••	•	•••	••
The Susp. shall be not contaminated by water	5	•	•••	••••	••••	••	•••••
The Susp. shall be not contaminated by grunge	5	•	•••	•	••••	••	•••••
The Susp. can be easily accessed for maintenance	3	••••	•••••	••••	••••	•••••	•
The Susp. shall allow easy replacement of worn parts	1	••••	•••••	••••	••••	•••••	•
The Susp. can be maintained with readily available tools	3	•••••	•••••	•••••	•••••	••	•
The Susp. shall last a long time	5	•••••	•••••	•••••	•••	•••••	•
The Susp. shall be safe in a crash	5	•••••	•••••	•••••	•••••	•••••	•••••

Product Requirements

4.3 Setting ideal and marginally acceptable **target values**

- Five ways to express a value in the metrics:
  at least *X*, at most *X*, between *X* and *Y*, exactly *X*, discrete values

Metric #1:
Attenuation from drop out to handlebar at 10 Hz > 13 dB
Metric #2:
Spring preload > 700 N

4.4 Translate target values into **target product requirement** statements

- In the form of a formal *"shall-statement"*:
  "the product [shall | should | can | may] do, be, etc..."

Product Reqt. #1:
The fork shall have an attenuation from drop out to handlebar at 10 Hz that is at least 13 dB
Product Reqt. #2:
The fork should have a spring preload of at least 700 N

Service Requirements

4.3 Setting ideal and marginally acceptable **target values**

- Five ways to express values in metrics:
  at least *X*, at most *X*, between *X* and *Y*, exactly *X*, discrete values

Metric #1:
Attenuation from drop out to handlebar at 10 Hz > 13 dB

Metric #2:
Spring preload > 700 N

Metric #3:
Number of travel requests processed per day > 10

Metric #4:
ERP - Travel-IT DB synchronization < 10 min

4.4 Translate target values into **target requirement** statements

- In the form of a formal *"shall-statement"*:
  "the product/service [shall | should | can | may] do, be, etc..."

Product Reqt. #1:
The fork shall have an attenuation from drop out to handlebar at 10 Hz that is at least 13 dB

Product Reqt. #2:
The fork should have a spring preload of at least 700 N

Service Reqt. #3:
The travel arrangers shall process at least 10 travel requests per day

Service Reqt. #4:
The Travel-IT DB shall be synchonized with the central ERP at most every 10 minutes

5.4

Requirements Register

Requirements Engineering

A nine-step process → **incremental** innovations / **market-pull** products

1. Identifying the **stakeholders** (end users, key users, customers, etc.)
2. Collecting the **user requirements** (~ needs gathering)
3. Searching for **solutions** (~ concept generation)
4. Translating the **user requirements** into **target product requirements** (~ target specifications setting)
5. Benchmarking the **solutions** (~ concept selection and testing)
6. Setting the **final product requirements** (~ final specifications setting)
7. Developing the **solution**, the product, the service or the facility (i.e. going through the **DESIGN** and **BUILD** phases)
8. Verifying the product w.r.t. the **product requirements**
9. Validating the end product w.r.t. the **user requirements**

Requirements Register

ProR eclipse
consentor
IBM DOORS
Innoslate

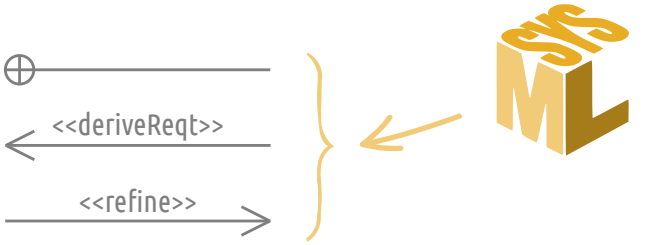
It is a structured list of requirements

Microsoft Excel spreadsheet

- Rqt. **ID** and a short description
- So-called "***shall-statement***"
- Category or **type**, e.g. raw need/IUR or P/S Reqts and **subtype**
- **Compliance** to solutions, and for each solution:
  - Compliant (C)
  - Partialy compliant (PC)
  - Not compliant (¬C or NC)
  - Compliance not applicable (NA)
  - Compliance to be defined (TBD)
- **Deviation** request(s) and decision(s)

Requirements Register (cont'd)

- **Relationships** between requirements:
  - **Containement** Split of a composite reqt.
  - **Derivation** Reqt. of lower level in hierarchy
  - **Refinement**



- **Qualification** method:
  - **Tests** (T), destructive on samples or not destructive
  - **Analyses** (A), calculations, etc.
  - **Inspections** (I), incl. visual inspections
  - **Reviews** (R), design reviews, etc.
- **Qualification** procedure(s), report(s) and status
- **Nonconformance** report(s) and decision(s)
- Editorial quality control: comments, traceability information, requirement status (draft, V&V, etc.)

Verification for PRs
Validation for IURs

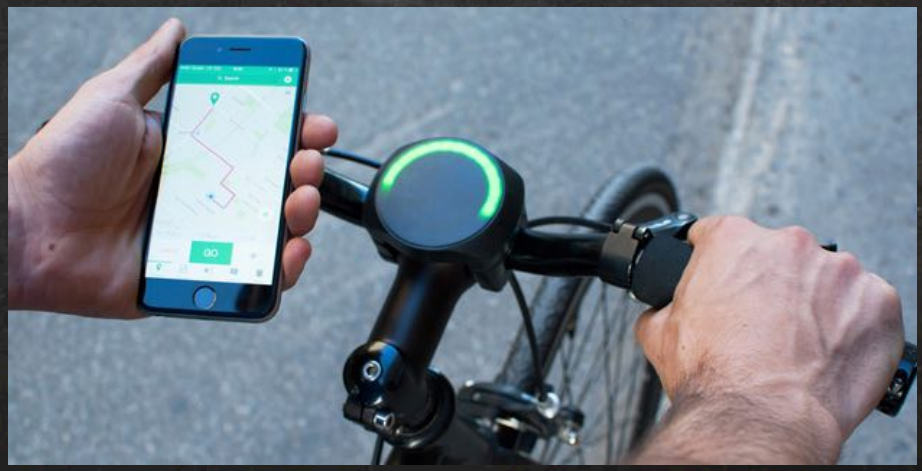
For reqt. statements

smarthalo

Your task: reengineer the **product requirement register**

A GPS for urban bikes coupled to a smartphone

- Collect five **raw needs**
- Translate them into **interpreted user reqts.**
- Find the **metrics**
- Set the **target values**
- Formulate the **target/ final product reqts.**



Project Management

What are we going to see together?

Lifecycle

Processes

- Launching a project
- Defining requirements
- Planning & scheduling
- Managing risks
- Handling issues
- Ensuring quality
- Costing
- Reporting progress
- Finalizing a project

openSE

Roles

- Initiators
- Project Board
- Project Manager
- Project Team
- Study Team
- Study Leader
- Project Stakeholders

Artefacts

- Project Proposal
- Project Roadmap
- Project Mngt Plan
- Issue Record(s)
- WBS & RACI
- Coord. Schedule
- Cost Estimate
- Budget
- PBS
- Reqts. Register
- Risk Register
- Progress Report(s)
- Close-out Report

cern.ch/openSE

cern.ch/quality

cern.ch/go/8rMF

Pierre.Bonnal@cern.ch +41 22 767 5710 +41 75 411 2072