MINES Alès

Centre de recherche LGI2P

ISOE

- Systems Engineering- Requirements engineering: principles and good practices

Nicolas Daclin

Ecole Nationale Supérieure des Mines d'Alès
Laboratoire de Génie Informatique et d'Ingénierie de la Production

Presentation

- **Nicolas Daclin, Dr**
  - Associate professor Ecole Nationale Supérieure des Mines d'Alès
  - LGI2P (Laboratoire de Génie Informatique et d'Ingénierie de Production)
  - Team ISOE (Interoperable System and Organisation Engineering)

+33 466 387 039

nicolas.daclin@mines-ales.fr

www.mines-ales.fr / lgi2p.mines-ales.fr

2

References (systems engineering)

- S. Fiorèse, J.-P. Ménadier, *Découvrir et comprendre l'Ingénierie Système*, Collection AFIS, Cépaduès Ed., 2012.
- NASA, *Systems Enginering Handbook*, NASA/SP-2007-6105 Rev1, available online at: ntrs.nasa.gov, december 2007.
- INCOSE, *Systems Engineering Handbook – A guide for system life cycle processes and activities – v. 3.2.1*, INCOSE-TP-2003-002-03.2.1, january 2011.
- AFIS, *Glossaire de base de l'Ingénierie de Systèmes – Version Expérimentale 1.2*, disponible en ligne à : http://homepages.laas.fr/kader/Glossaire_IS.pdf, octobre 2011.

3

References (requirements engineering)

- A. Faisandier, *Ingénierie des systèmes complexes – démarche et méthode d'élaboration des besoins des parties prenantes*, MAP système, avril 2010.
- A. Faisandier, *Ingénierie des systèmes complexes – démarche et méthode d'élaboration des exigences techniques* – MAP système, avril 2010.
- Requirements WG-INCOSE, *Guide for Writing Requirements*, v1.0, INCOSE-TP-2010-006-01, avril 2012
- AFIS, *Guide des bonnes pratiques en ingénierie des exigences*, Collection AFIS, Cépaduès Ed., 2012
- AFIS, *Recommandations pour l'élaboration d'un référentiel d'exigences techniques*, 2001
- AFIS, *Glossaire GT « Ingénierie des exigences »*, fiche n°14, 2013
- IEEE, *Guide de l'IEEE pour la spécification d'exigences de système*, Norme IEEE 1233, 1998
- G.V. Bochmann, *Writing better requirements*, université d'Ottawa, 2010
- N. Sannier, *Eliciting requirements – Specifying requirements*

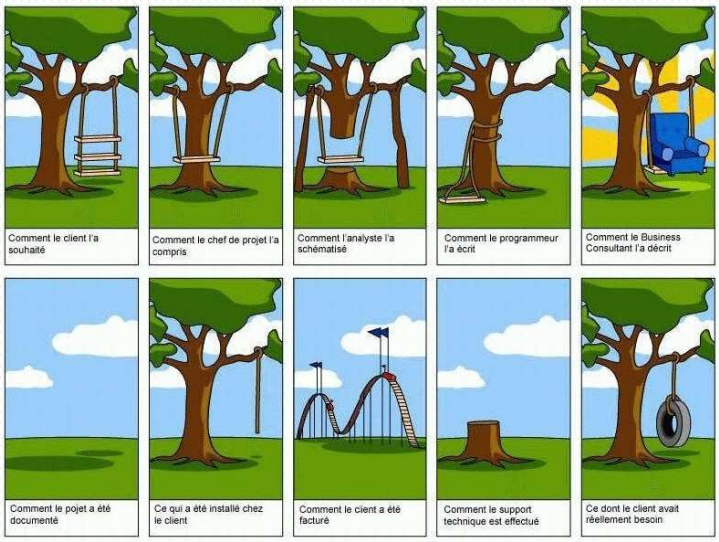
4

References (requirements engineering)

- AFIS, *Guide des bonnes pratiques en expression du besoin*, Collection AFIS, Cépaduès Ed., 2012
- Karl E. Wiegers, *First Thing First: Prioritizing Requirements*, Software Development, September 1999
- Donald Firesmith, *Prioritizing Requirements* , Journal of Object Technology, Vol. 3, No.8, September-October 2004.
- Steve Easterbrook, *Lecture 19 : Requirements Prioritization*, Department of Computer Science, University of Toronto, 2004
- OMG, *OMG Systems Modeling Language (OMG SysML™) Version 1.4*, September 2015
- EIA/IS 632, Processes for Engineering a System, EIA 632, version 1.0, April 1998

5

From stakeholders' expectations to the provided system…

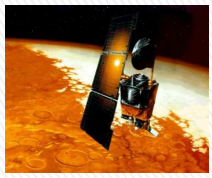


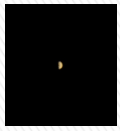
Source: http://www.levasseur.im

6

Requirements Engineering: why?

- **Mars Climate Orbiter (1999)**
  - Initial expectations:
    - To study climate on Mars
    - To study weather on Mars
  - Result:
    - Failure of the mission (destruction
    - 125 millions de $ go up… in smoke
    - 1 picture of Mars
  - Root cause:
    - 1 software produces results in **Imperial units** (inches, feet…), a second expects results in **metric units**.



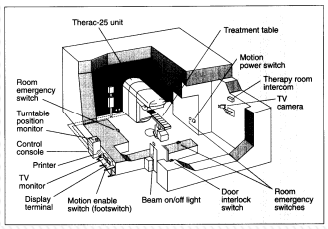


Source: http://www.jpl.nasa.gov

7

Requirements engineering: why?

- **Therac 25 (1985 - 1987)**
  - Initial expectations:
    - Tumor treatment with x-ray
  - Result:
    - Overexposure (~100 x intended dose)
    - 3 serious wounded person
    - 3 deaths
  - Root causes:
    - **No specifications**
    - No document
    - No test



Source: http://users.csc.calpoly.edu/~jdalbey/SWE/Papers/THERAC25.html

8

Requirements engineering: why?

- **The Øresund bridge (1996 – 2000)**
  - Initial expectations:
    - Bridge connecting Sweden and Denmark
    - Railway and roadway
  - Result:
    - Planning (time, cost, quality, + 4000 schemas)
    - **Requirements engineering**
      - Lifespan: 100 years
      - Speed train/vehicle : 200km/h, 120 km/h
      - Environment : wind (61 m/s), wave (2,5 m) temperature (+/- 27° C)
      - Evaluation between requirements and solutions
      - **Application of SE principles: « an award-winning bridge! »**

Source: INCOSE Systems Engineering Handbook v. 3.2.2, 2011

Requirements engineering problematic

- **Requirements engineering is often neglected because…**
  - Actors do not know **what** to do
    - How to write a requirement?
    - What is the process?
  - Actors do not understand **why** requirements are necessary
    - They do not understand the impact
    - Requirements are just obviousness
    - It is just another document to write…
  - Actors prefer **do something else**
    - No time
    - We will see… during operational phase
    - They don't see any reward

Source: G.V. Bochmann, *Writing better requirements*, université d'Ottawa, 2010
Willis J., *Systems Engineering and the forgotten '-illities*, 2011

Importance of the requirements engineering

- From the Standish Group:

Requirements engineering is the main cause of project failure (cancellation) or difficulties (drift from original objectives)…

but…

Requirements engineering is the main cause of success!

Source: www.standishgroup.com, the Chaos report

11

Requirements engineering

« Organized collaborative and multidisciplinary activity – based on the science and experience – allowing to collect, specify, refine, qualify requirements related to a system to place in its environment and to handle the evolutions of requirements to provide an optimized solution regarding to identified expectations. »

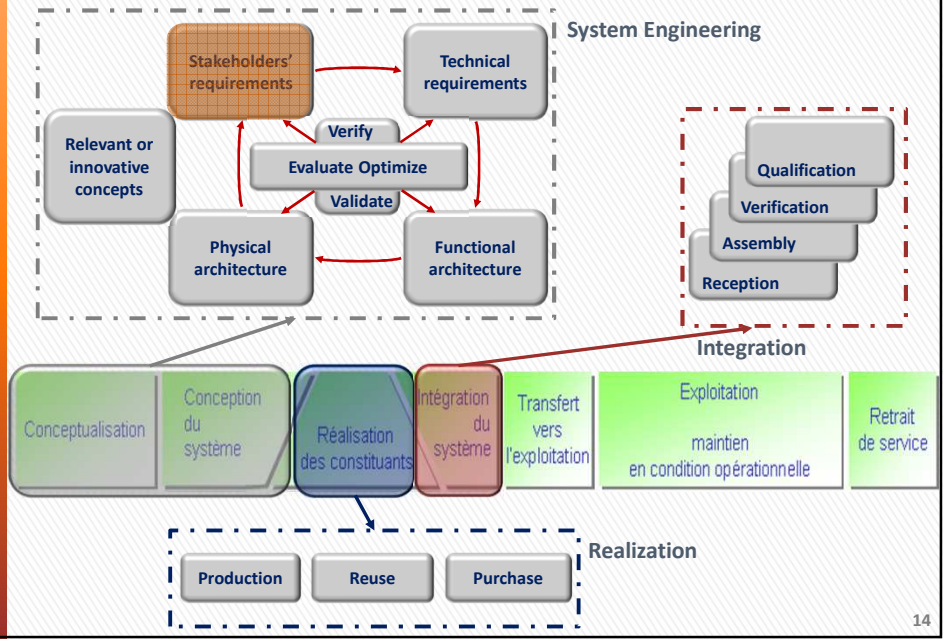
12

Stakeholders' requirements process

1. Principles
2. Process

13

Position of the stakeholders' requirements process



14

Issues

What is our first perception of the quality of a product or a service?

"Its ability to satisfy stated or implied needs"

1. Need to collect expectations
2. Need to express expectation

- **Two essentials principles:**
  - **There is no need if there is no problem to solve! It is required, first, to clarify the problem.**
  - **Think the expectations, not the solution! Usually, we define the expectations through the solutions we know…**

15

Definition

Need **:**

- **Necessity or desire expressed by a user or any stakeholder and interested by the use or the exploitation of the system.**
- **Expressed in the language of the acquirer (purchaser).**
- **(initial) expectation, stakeholder requirement, user requirement, initial requirement, originating requirement.**

- **A need stems from a lack, a dissatisfaction, an expectation.**
- **The answer to a need is a set of action.**
- **The need is valid when the stakeholders acknowledge its usefulness and it is possible to justify it.**

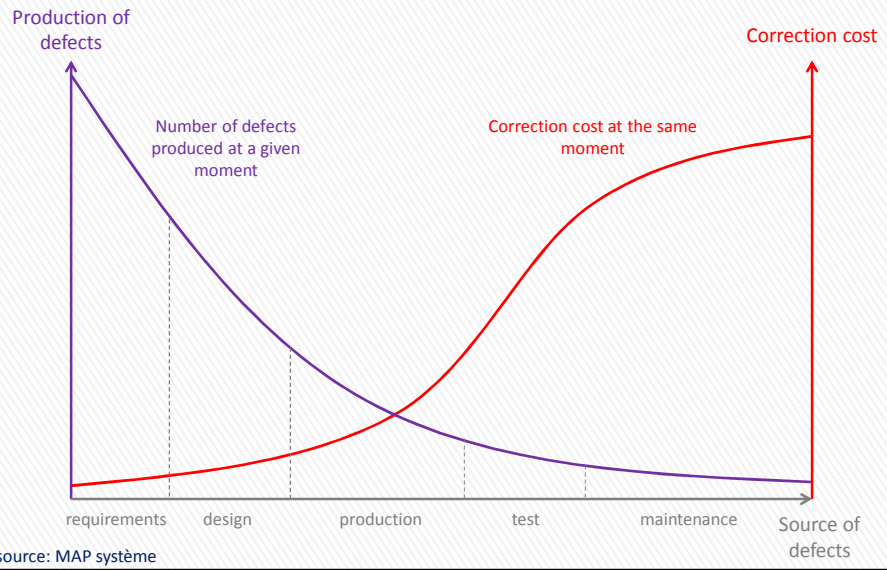
16

Usefulness of the definition of needs

- **The definition of the needs allows to have a common understanding of the problem to solve.**
- **For the acquirer:**
  - To define clearly its **expectations**.
  - To be **understood by the prime contractor**.
- **For the prime contractor:**
  - To understand clearly **what to be done and why**.
  - To be **understood by the acquirer**.

17

Origin and costs to extract defects



source: MAP système

18

Expression of needs: a complex operation

- **Relational** aspect with multiple stakeholders to express the needs and constraints and to find a consensus.
- **Analysis** and **realignment** aspects to identify inconsistencies between the needs.
- **Background analysis** with the consideration of the environment in which the system will operate.
- **Mission analysis**, **validation** and **simulation** of scenarii.
- **Impact analysis** (system on environment and *vice versa*).
- **Market research**, **survey**.
- **Feasibility** study.

19

Expression of needs: obstacles

- **Identification**: the need is not well identified, defined or expressed.
- **Misunderstanding**: misunderstanding of the customer-supplier relationship.
- **Separation**: no clear separation between the problem and the solution.
- **Communication**: bad communication between the stakeholders.
- **Negative perception**: negative perception of the problem to solve.

20

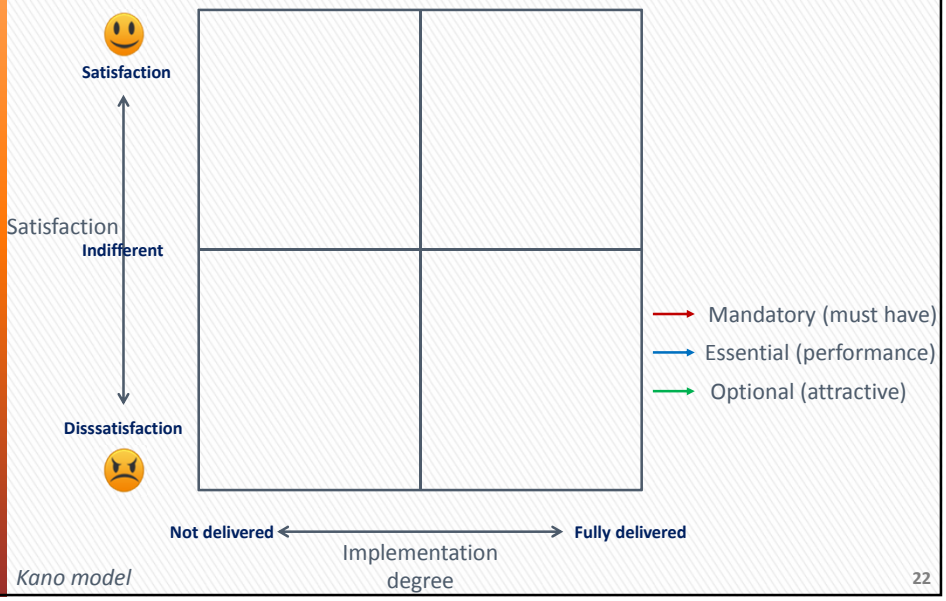
Expression of needs : attributes

- **Attributes are characteristics to complete the needs to manage them:**
  - **Identifier**: alphanumeric symbol to identify a need among other.
  - **Importance**: importance for the stakeholders (essential, mandatory, optional).
  - **Criticality**: importance for the safety or the reliability of the system.
  - **Flexibility**:
    - **imperative**: dissatisfaction leads to the inability to perform the mission or unacceptable risks for the system or the environment.
    - **expected**: consideration of additional means to get the satisfaction (e.g. cost – extra cost for a given expectation).

Identificateur	Expression	Importance	Criticality(0,1,2,3)
B.S.1	The system protects against the radiation	mandatory	3

21

Back to… the ~~future~~ importance and the consequences



Kano model

22

Scales on requirements

- **Scales allow to stakeholders to give importance to the requirements according to different attributes (safety, time to implement, cost…)**
- **Scales can be quantitative (e.g. 0, 1, 2…) or qualitative (e.g. high, average, low) but each level must be perfectly defined:**
  - 0 – no risks for the safety if the function is not delivered.
  - Mandatory – The product is not acceptable if the function is not delivered.
- **Attributes and associated scales allow to prioritize the requirements:**
  - Implementation order according to the impact on the schedule of the project.
  - Impact of a functionality for the end-user on the cost of the system.

23

Classification : typology

- **Expected services (functions): actions to do to perform the mission.**
- **Performances: set of performances related to expected services.**
  Interfaces: flow exchanged and physical connection between the system and the context.
- **Operational: conditions under which the service is delivered:**
  - Safety,
  - Ergonomic,
  - Human factors,
  - Dependability,
  - Environment…
- **Constraints: limitation coming from prime contractor, dimensions, regulation…**

- **The classification can allow a better traceability and management of needs. However it is better to have all expectations pell-mell rather than omit expectations…**

24

Definition of needs: principles to collect

- **Some principles for the collection:**
  - **Organized listening of the stakeholders** (identification of the persons, meetings).
  - **Only facts must be collect**, no opinion, no interpretation.
  - **Reformulation with hypothesis** on the facts and validation of the hypothesis with the stakeholders.
  - **Focus** on the fundamental.
  - **Group work** .

The smartphone measures the temperature → **Useful/useless?**

The smartphone must do shots → **Pictures, videos, both?**

- **Some tools:**
  - Definition of a common language, interview, consensus, functional analysis…

25

Verification and validation of needs

- **Verification**
  - **Maturity/ accuracy**: the expression of the need is conform to the stakeholders' expectations .
  - **Exhaustiveness**: All stakeholders' expectations are expressed.
  - **Feasibility**: operational concepts can be identified to estimate the feasibility to solve the problem.
  - **Translatable**: the needs can be translated into requirements.
  - **Consistency**: there are no inconsistencies between the needs.
  - **Relevance**: the expression of the need allow to define a relevant answer to the problem.
- **Validation**
  - **Why these needs?**
  - **What causes make the needs disappear?**

26

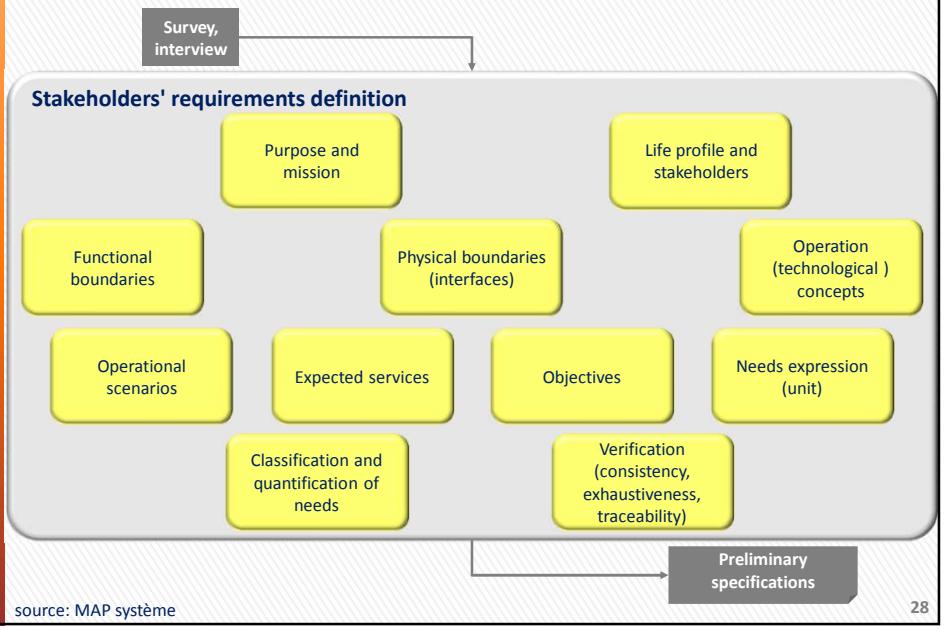
Stakeholders' requirements process

- **A complex process**

- **Expected results:**
  - All expectations, desires which represent the stakeholders' requirements (customers, end-users, operators).
  - All contexts of intended use.
  - A basis to establish the technical requirements.
  - A basis to establish the operational validation of the delivered system.

27

Stakeholders' requirements process (2)



source: MAP système

28

Classic outline of preliminary specifications

1. **Introduction**
   - *a) Object*
   - *b) Documents (references)*
   - *c) Terminology*
2. **The system**
   - *a) Purpose, mission, objectives*
   - *b) Functional and Physical context*
   - *c) Stakeholders*
3. **Stakeholders' requirements**
   - *a) Operational mode and scenarios*
   - *b) Needs*
     - Services, performances, autonomy, lifespan, interaction & physical connections with the context, human factors, ergonomic, dependability, safety, transport, storage
   - *c) Constraints*
     - Realization, commissioning, decommissioning, logistic, maintenance, production, regulations, costs, deadlines, validation

29

Risks

- **Wrong identification of the system (upper/subsystem)**
  - **Stop the development and identify the system**
- **Wrong perimeter (added/omitted elements)**
  - **Wrong delivered product, rejection by users**
- **No (or lack of) modes and operational scenarios**
  - **Contentious validation, delay (development, commissioning)**
- **Incompleteness of the needs, wrong identification of stakeholders**
  - **Delay (development), contentious validation**
- **Wrong classification of needs**
  - **Waste of time (prime contractor) , cost to develop more expensive**

source: MAP système

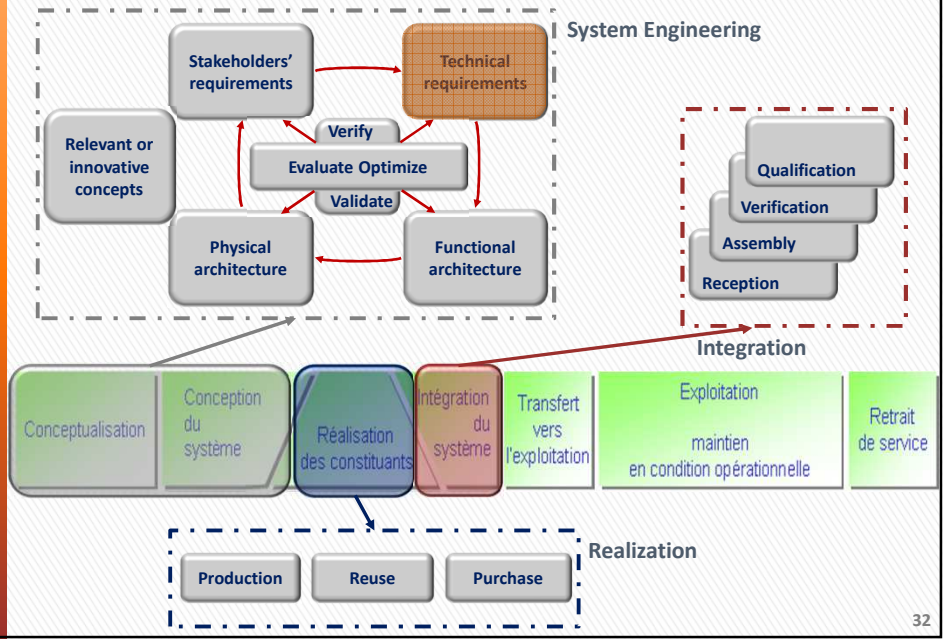
30

Technical requirements' process

1. Principles
2. Management
3. Process

31

Position of the technical requirements process

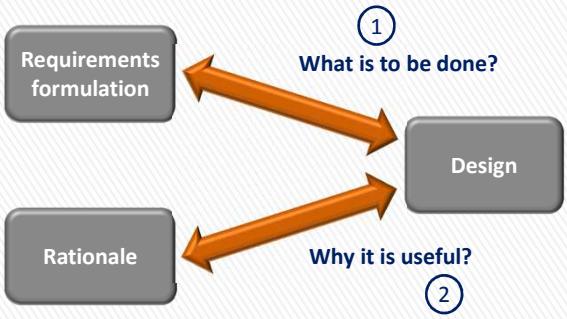


32

Roles of technical requirements

- **Two roles**

1. To identify the work to perform
2. To be used for the rationale and validation



33

Definition

***Requirement*:**
- **A statement that prescribes a function, an aptitude, a characteristic or a limitation to be met by a product or a process under given environmental conditions.**
- **Expressed in the language of the prime contractor.**

- **Characteristics of a requirement:**
  - **Uniqueness**: it is related to only one concern
  - **Accuracy**: its expression is well structured (rigorous)
  - **Unambiguous**: there is only one reading of the requirement
  - **Verifiable**: it can be assessed
  - **Achievable**: it can be satisfied

- **Characteristics of a set of requirements:**
  - **Consistency**: There are no contradictions between requirements
  - **Completeness**: no omission, the whole problem is considered

34

Characteristics: some clarification about…

- … a requirement:
  - Necessary
  - Implementation independent
  - Unambiguous
  - Complete
  - Singular
  - Feasible
  - Verifiable
  - Correct
  - (Conforming)
- … the set of requirements:
  - Complete
  - Consistent
  - Feasible
  - Bounded
  - Structured

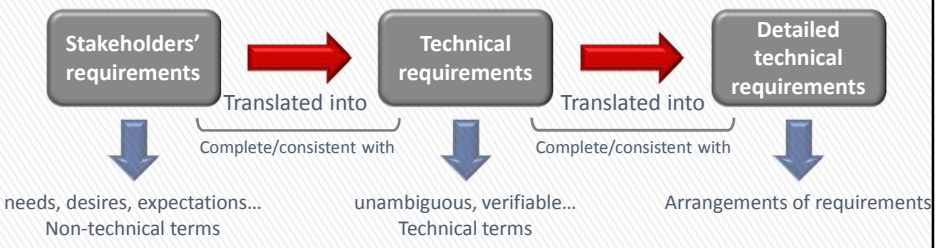
35

Usefulness to define technical requirements

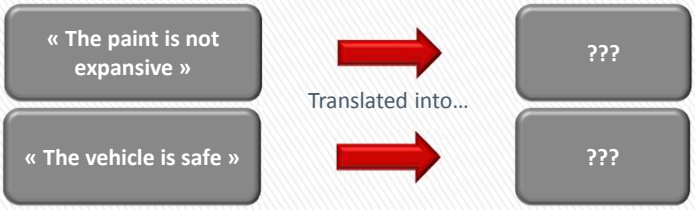
- To ensure the quality of the communication between different technical communities.
  - Numerous actors can lead to a misunderstanding…
- To verify - all along the development cycle - that stakeholders ' requirements are met
  - This avoids a wrong reading from designers and to select a solution which does not meet expectations.
- To consider all the stakeholders' requirements.
  - This avoids drift (cost and delay), the requirements are a contractual commitment.

36

From stakeholders' requirements to technical requirements



- **Example:**



Source: MAP système

37

Languages to express requirements

The language to express the requirements must be accurate, readable and allow to create

Natural Language (NL)

Advantages: no training required, readability, richness, extensibility

Drawbacks: ambiguous, no automation, weak consistency, lack of accuracy

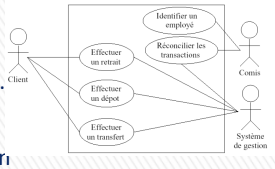
- **Example:**
  - The system shall wash clothes

Semi-formal language

Advantages : graphical syntax, easy to learn and manipulate

Drawbacks : still a risk of ambiguities

- **Example:**
  - UML, URN, GRL, KAOS (method + language), SYSML…



Formal language

Advantages : great accuracy, consistency, automation

Drawbacks : weak readability, difficult to learn

- **Example:**
  - E<> t.Working and r.Active and T > t.timeMin and T < t.timeMax

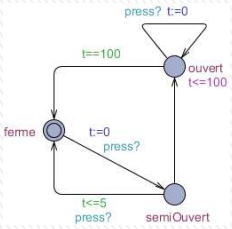
38

Formal language

- **Its use (e.g. Mathematical symbols) is reserved for those who know how to handle it and understand it..**
- **Any equation/formula must be formulated, each term must be clarified, each unity must be expressed according to the selected units system.**
- **Formal languages are often associated with tools.**

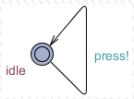
Requirement expressed in NL:
"The portal shall remain open beyond 100 time units is true one day"

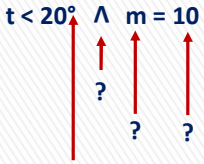
Requirement expressed in Temporal Logic:
E<> Portal.ouvert && t>100



Portal

Button



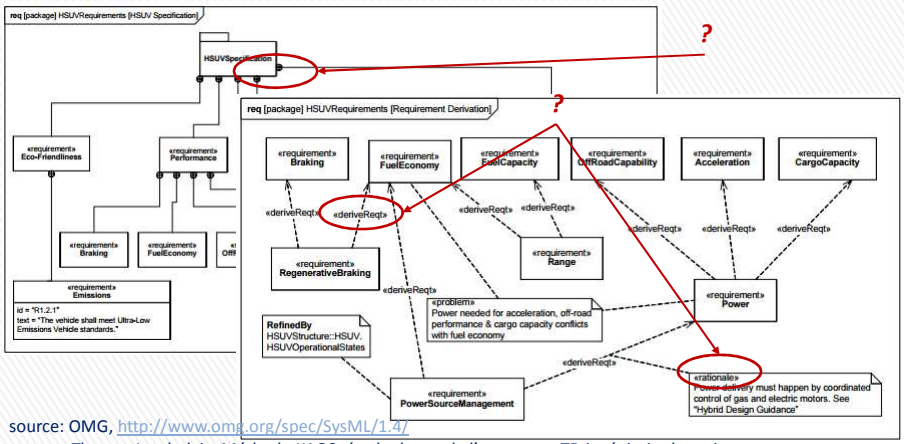


Temperature (C, K, F…)?
Angle?

39

Semi-formal language

- **Its learning and use is easy because it relies on graphical formalisms.**
- **All shapes, symbols, colors must be clearly defined to avoid any hazardous interpretations.**



source: OMG, http://www.omg.org/spec/SysML/1.4/

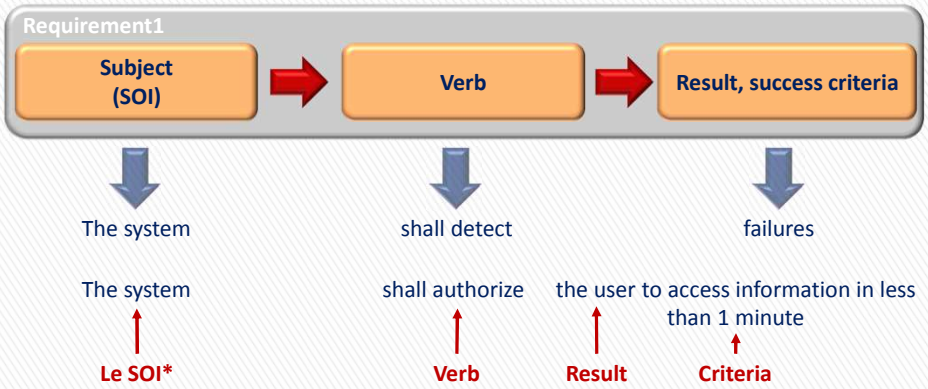
Thomas Lambolais, Méthode KAOS: étude de cas de l'ascenseur, TD ingénierie des exigences

40

Natural Language

- **A technical requirement is the translation of an expectation**

Its expression is a sentence that must be simple.



* System of Interest

41

Natural Language

- **A Requirement is written in the present tense.**
- **Writing to avoid:**
  - **Negations**
  - Vague terms (**buzzwords**)
    - *a lot, few, user-friendly, easily…*
  - **Conjunctions**
    - *and, or, with, also…*
  - **Escape clauses**
    - *If, but, when, except, unless, although…*
  - **Speculations**
    - *normally, generally, often…*
  - **Suggestions**
    - *may, might, should could, perhaps, probably…*
  - **Wishful thinking**
    - *100% reliable, 0 bugs, handle all failures…*

Normally, the system should detect the failures unless it is outside its working envelope.

42

Controlled Natural language (CNL)

- **The CNL is a sufficient and limited language to express requirements.**
- **The relies on predefined structures and is enough generic to be applied in any domain.**
- **Example: the boilerplates**
- **Propose some requirements relying on the following boilerplates**

The <*system*> shall <*function*> to <*value*> <*unit*>

In <*mode*>, the <*system*> <*function*>

43

Requirements expression: mnemonics

- **A requirement must be MUST:**

Measurable
Useful
Simple
Traceable

- **A requirement must be SMART:**

Specific
Measurable
Achievable
Relevant
Traceable (time-bounded)

44

Examples

- The vehicle shall reach 200 km/h.
- The cost of the paint is less than 1000€.
- A warning light shall inform the pilot in case of failure.
- The system shall record the failures in real-time.
- The probability of interruption of the electronic system functions leading to the loss of a mission is less than 5x 10-6 per flight hour.
- The turning cycle shall allows a parallel parking.

45

Defects to avoid in a requirement

- **Noise/Perfectionism**: the requirement includes unnecessary items.
- **Vague**: global description of the requirement.
- **Dependency**: the requirement includes solutions.
- **Redundancy**: the requirements come several times.
- **Under-specification**: the requirement is implicit, vague, omitted.
- **Over-specification**: the requirement goes beyond the initial expectation.
- **Contradiction**: several requirements include contradictory features.
- **Ambiguity**: the requirement can be understood in different manner.
- **Unverified Reference**: the requirement refers to other document whose the relevance and/or the accuracy is not verified.
- **Option**: the requirement is “open” or incomplete.

46

Example

- **Which problems?**
  - The control will be effective in all cases.
  - Quality of the sensors: adaptability and infrared measure
  - The actuator could be controlled automatically with a supra-neutron relay connected in shunt on the alternator starter.
  - All unnecessary information are not displayed.
  - The fixation of the system uses the R-00-XXX-125 process.
  - The age of the pilot is 35.
  - The interface is user-friendly.
  - The data are saved as much as possible.
  - The application has no bugs.

47

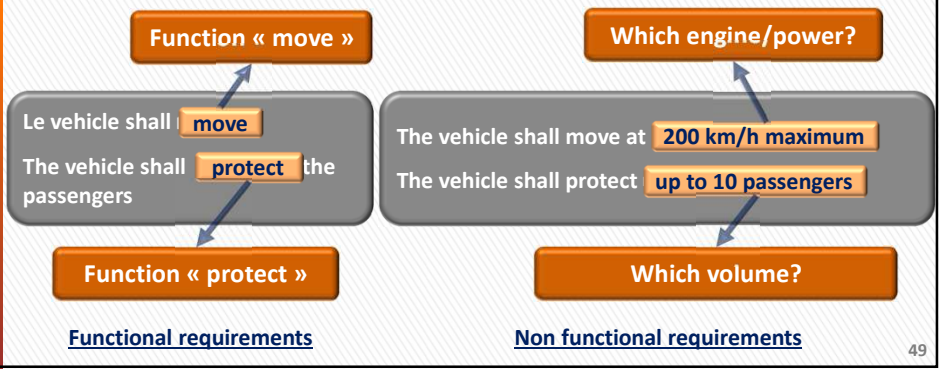
Expression of requirements: attributes

- **Attributes are characteristics to complete the requirements to manage them:**
  - **Identifier**: alphanumeric symbol to identify a requirement among other
  - **Criticality**: importance for the safety or the reliability of the system.
  - **Priority**: indication to the designer to order the development.
  - **Weight**: state of the requirement (identified, analyzed, verified, allocated, satisfied).
  - **History**: landmark to identify the modifications on the requirement (date, author, modification, deletion, justification of the amendment).

48

A first classification of requirements

- **The classification allows the designers to identify what they have to do.**
  - **Functional requirements**: requirements which describe what the system has to do (function)
  - **Non functional requirement (NFR)**: requirements not related to a function. A NFR expresses a constraint, a performance, an ability… A NFR influences the choice of a technical solution.



49

Other classification of requirements

- **The classification allows the designers to identify what they have to do:**
  - Functional
  - Performance
  - Constraint
  - Interface
  - Scenarios, modes et operational
  - Justification and validation

- **The classification may vary depending on the domain:**
  - Environment
  - Support
  - …

50

Classification: meaning

Type	Meaning
	Transformation to perform (what the system has to do)
	Definition of the field of use to achieve the transformation (ex. quantification of the functional requirements)
	Identification of the constraints on components, standards…
	Interfaces (internal and external)
	Life situation, operational conditions
	Methods and means to validate the system

51

Classification for the justification

Type	Justification
	The system perform all functions
	The functions reach all performances
	Components and physical architecture respect the limitations
	Interfaces are fulfilled
	All life situation are identified
	All justification and validation situations are identified

52

Functional requirements

- **The functional requirements express what the system has to do in terms of action, behavior and expected result**

- **These requirements are only related to the transformations (or actions) to be done by the system and in agreements with its operational mission.**

- **Examples:**
  - The system X shall move.
  - The system X shall collect data.
  - The system X shall memorize collected data.

53

Performance requirements

- **These requirements are associated to a function (or a device) and define a measurable criteria of the considered function.**

- **These requirements are often quantitative (evaluation is easier).**
- **The performances are used to make a choice among several solutions.**

- **Examples:**
  - The system shall be operational in less than 2 minutes.
  - The system shall monitor airspace within a radius of 30 km.
  - The system shall be waterproof until 100 m.

54

Constraints

- **(1) a restriction, limit or regulation on a product, project or process.**
- **(2) type of requirement or design feature that can not be traded off.**

- **The constraints are sizing the system: renewal of components, imposed solution.**

- **Examples:**
  - The system shall respect the environmental standards in effect.
  - The volume of the system is 1 m^3.
  - The system shall ensure all maintenance operations of its equipments.

55

Interface requirements

- **An interface requirement defines the conditions of interactions between elements.**

- **These requirements include the interfaces between the SOI and its environment (external) and the interfaces between the elements of the SOI (internal).**
- **The interfaces can be functional and physical.**

- **Examples:**
  - The system X shall exchange data with the system Y
  - The tank caps of the system X are compatibles with the A612 nozzles.
  - The system X shall receipt order from the system Y.

56

Scenarios, mode and operational requirements

- **The scenarios and modes describe the expected behavior of the system in all phases of its life cycle.**
- **An operational requirement expresses the conditions of execution and operation of the system.**

- **Operational requirements include:**
  - Ergonomics
  - Human factors
  - Dependability
  - Logistic
  - Environment

- **Examples:**
  - The availability rate of the system X is 80%.
  - The system X is air transportable.
  - 1 operator is enough to monitor the system X.

57

Justification and validation requirements

- **Justification and validation requirements allow to the designers to know all justification and validation situations which will be met.**

- **These requirements express the justification to give to the acquirer to be sure the system meet its expectations.**
- **These requirements express the method to be sure the system meet the expectations.**
- **These requirements consider the level at which the justification is applied (component, sub system, system).**

- **Examples:**
  - The achievement of performance requirements is demonstrated by test.
  - The transmission equipment is tested as defined by the standard XX-250-EF.
  - The achievement of functional requirements is demonstrated for all defined scenarios.

58

Classification: example

- **To which class belong the following requirements?**
  - The system X supplies the system Y in fuel.
  - The X system is operational in fog.
  - The X system shall use an existing industrial chassis.
  - The cost to develop the system X is minimized.
  - The X system is air transportable..
  - The transmission system is compatible RX-32.
  - In standby mode, the system shall diagnosis the status of its equipments.
  - The functioning in the fog is demonstrated by test in real condition.
  - The system X shall exchange information with the control center.
  - The system X shall perform a task in less than 24 hours.

- **As for the stakeholders' requirements, the classification allows a better traceability and reading of the requirements.**
- **The positioning of a requirement can be discussed, however it is better to place a requirement in a class rather than omit it…**

59

Verification and validation of technical requirements

- **Verification**
  - **Unambiguous**: there is only one reading of the requirement.
  - **Completeness**: The designer has all information to work.
  - **Verifiable**: each requirement can be verified.
  - **Consistency**: there are no conflicts between requirements.
  - **Editable**: the set of requirement (document) is easily editable
  - **Identifiable**: the requirements are clearly identified to make their referencing easier in the document of technical requirements.
- **Validation**
  - **The requirements are the correct translation of the stakeholders' expectations.**

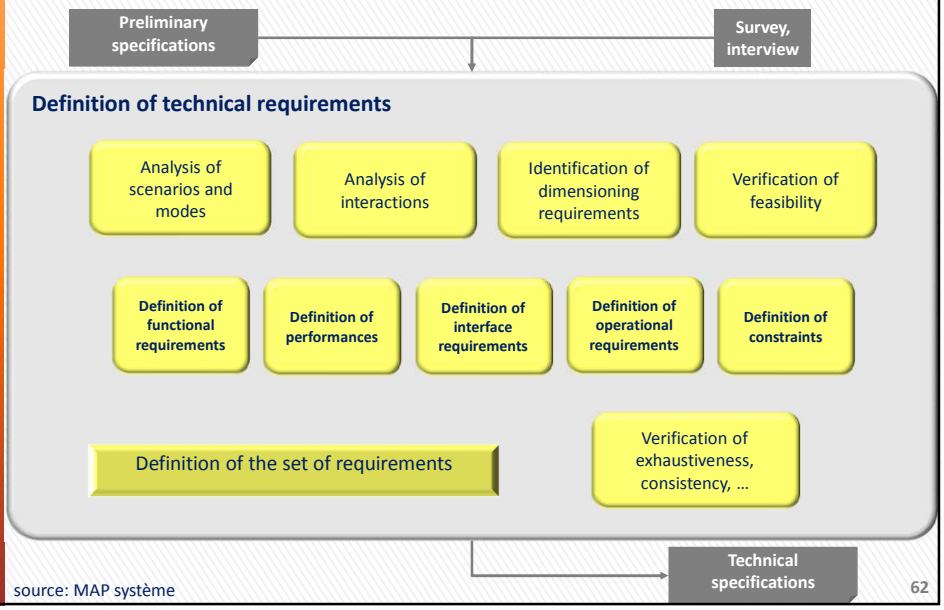
60

Technical requirements' process

- **Aims to translate stakeholders' requirements into technical requirements which guide the prime contractor to design the system (1 initial expectation leads to, at least, 1 requirement) .**
- **Expected results:**
  - A set of technical requirements
  - A full description of the problem to solve
  - A basis to establish the architectures (functional and organic)
  - A basis to validate the solution

61

Technical requirements' process (2)



source: MAP système

62

Classic outline of technical requirements

1. **Introduction**
   - *a) Object*
   - *b) Documents (references)*
   - *c) Terminology*
2. **The system**
   - *a) Purpose, mission, objectives*
   - *b) Functional and Physical context*
   - *c) Stakeholders*
3. **Technical requirements**
   - *a) Functional requirements*
   - *b) Performance requirements*
   - *c) Interface requirements*
     - o Functional interface, physical interface
   - *d) Operational requirements*
     - o Modes and scenarios, ergonomics, human factors, dependability, safety, environment, means
   - *e) Constraints*
     - o Physical, design, realization, commissioning, decommissioning, support, maintenance, cost , time…
   - f) Validation requirements

63

Risks

- **Not ensuring that stakeholders' expectations are mature**
  - ➡ **The prime contractor plays the role of the acquirer, non-optimal solution**
- **Lack of modes and scenarios**
  - ➡ **Contentious validation, delay (development, commissioning)**
- **Incompleteness, imprecision**
  - ➡ **Design without considering requirements**
  - ➡ **Contentious validation, delay (development, commissioning)**
- **Wrong classification**
  - ➡ **Requirements not used**
  - ➡ **Waste of time (prime contractor) , cost to develop more expensive**
- **No traceability with initial requirements**
  - ➡ **Difficulties to manage customer changes**

source: MAP système

64

Conclusion

- **The requirements engineering is an upstream phase that influences strongly the development cycle (downstream phases)**
- **The requirements engineering plays a major in the success or failure of a project but still remains too often neglected.**
- **The omission/ bad identification of the requirements can lead to:**
  - The non implementation of a function
  - The implementation of a useless function
  - The omission of a component
  - The bad design (e.g. size) of a component
- **And:**
  - The release of a non adapted system that can be rejected by the customer

65

Glossary

- **Systems engineering: an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem: operations, cost and schedule, performance, training and support, test, manufacturing, and disposal. SE considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. (INCOSE)**
- **Stakeholder: a party having a right, share or claim in a system or in its possession of characteristics that meet that party's needs and expectations (ISO 15288).**
- **Supplier, prime contractor: an organization or an individual that enters into an agreement with the acquirer for the supply of a product or service.**
- **Acquirer, purchaser: the stakeholder that acquires or procures a product or service from a supplier.**

66

Thank you!!

nicolas.daclin@mines-ales.fr

67