Chapter 1. Requirements Engineering Process

Reyes Grangel Seguer

EI1030. Analysis of Inf. Systems / EI1032. Software Analysis / MT1046. Integrated Inf. Systems
Computer Engineering / Computational Mathematics
Dept. of Computer Systems and Languages

January 30, 2019

Outline

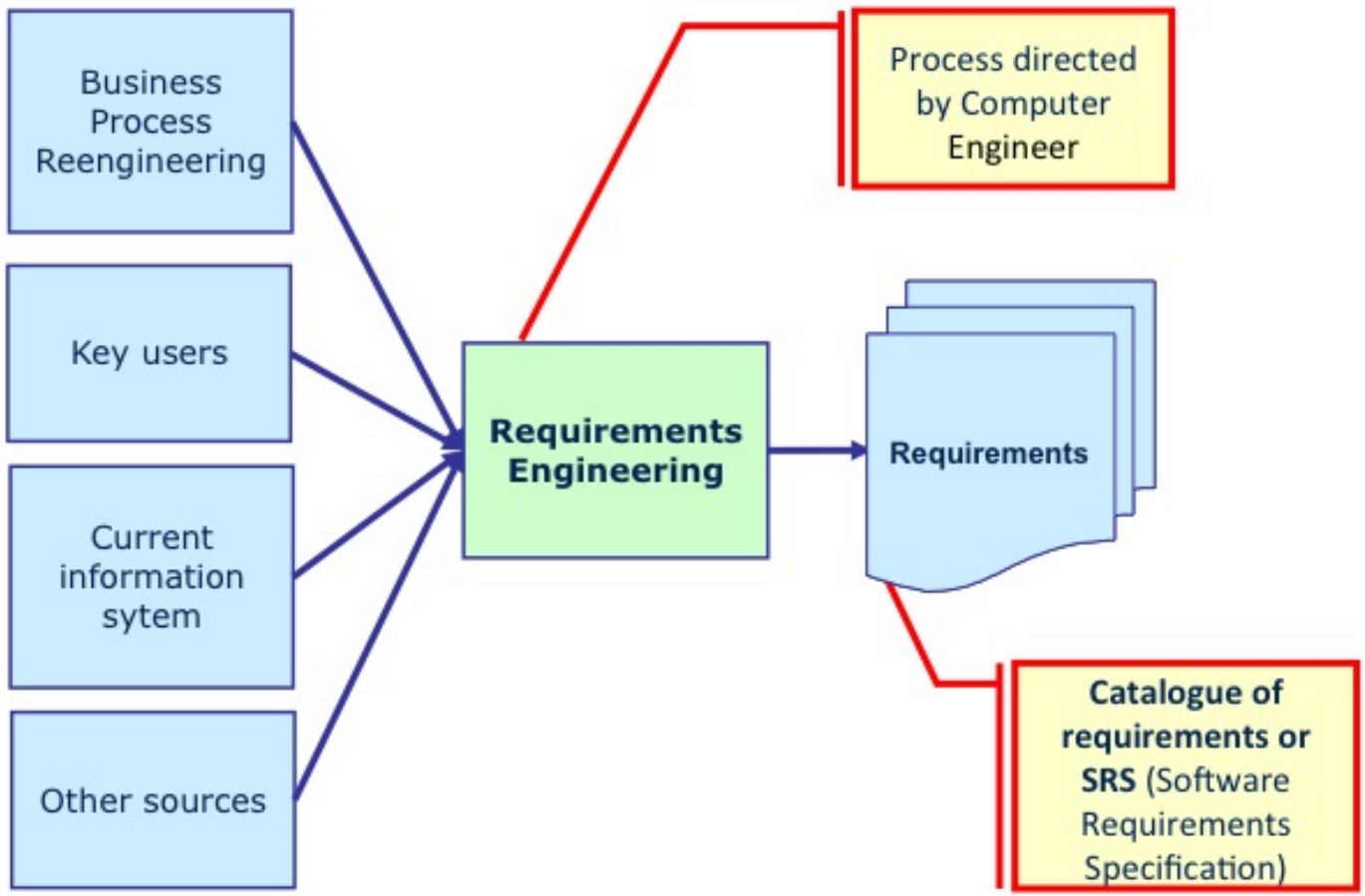
Outline

Concept of Requirement

What is a requirement?

A condition that the system have to accomplish or a feature needed for the system in order to do its functionality in order to achieve the defined objectives and taking into account the restrictions

Origin of the Requirements



1

Example of Requirements

Exemple: definició dels requisits d'un sistema informàtic de gestió de RRHH.

- El sistema ha de permetre crear, modificar i esborrar informació dels empleats[1].
- El sistema ha de permetre crear i modificar els contractes dels treballadors.
- El sistema calcularà a partir de la informació dels contractes l'import de la nòmina per a cada empleat.
- El sistema ha de permetre registrar i modificar les baixes per malaltia.
- El sistema ha de proporcionar funcionalitat per generar els informes mensuals de baixes per malaltia.
- El sistema ha de permetre generar la documentació relativa a les assegurances socials (TC1 i TC2).
- El sistema ha de proporcionar funcionalitat per tal de gestionar els convenis.
- El sistema ha de permetre gestionar el procés de selecció de personal.
- El sistema ha de permetre gestionar la formació dels empleats.
- El sistema ha de permetre gestionar els plans d'incentius.

Requirements Engineering

What is requirements engineering?

It is a discipline of the Software Engineering that supports Computer Engineers to

- Understand better the problem to solve
- Define the computer solution to solve it
- Perform tasks for defining requirements using the convenient techniques

Requirements Engineering Process

1. Statement
2. Elicitation
3. Elaboration
4. Negotiation
5. Specification
6. Validation
7. Management

Statement

- To redefine the general objective of the computer system development
- To define the scope of the future computer system from
  - Functional viewpoint
  - Organisational viewpoint
  - Computational viewpoint
- To identify and specify restrictions
  - Time
  - Economy
  - Human resources
  - Technology
- To define the development and exploitation environment

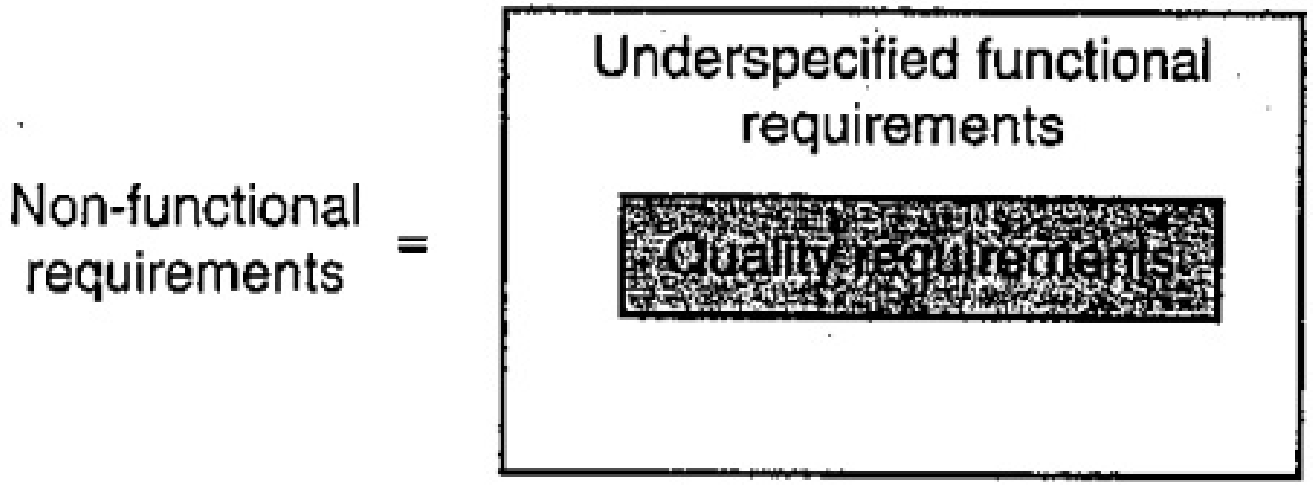
Elicitation

- To identify key users
- To collect information on requirements using different techniques

Elaboration

- To improve, refine and analyse the collected information
- To organise the requirements according to [Pohl 2010]
  - **Functional requirement**s: services that the system should to provide
    - **Data perspective**: to document which kind of information to introduce, to extract and to save
    - **Functional perspective**: to document the main functions of the system
    - **Behavioural perspective**: to document the overall behaviour of the system and its different states
  - **Quality requirements**: a quality property of the entire system or a system component, service, or function
  - **Constraints**: an organisational or technological requirement that restricts the way in which the system shall be developed
    - Affecting the system
    - Affecting the development process

Elaboration: non-functional requirements?



Elaboration: quality requirements

- Important to users
  - Availability
  - Efficiency
  - Flexibility
  - Integrity
  - Interoperability
  - Reliability
  - Robustness
  - Usability
- Important to developers
  - Maintainability
  - Portability
  - Reusability
  - Testability

Negotiation

- To obtain the list of requirements classified by types
- To prioritise requirements
- To estimate costs and risks
- To solve conflicts

Specification

- To document requirements
  - To show only external behaviours of the system, the future system vision for the users
  - To describe exhaustively the conditions that needs to accomplish the system to achieve the defined objectives and scope, and taking into account restrictions

Specification: features of requirements statement

- **Unambiguous**: it doesn't allow different interpretations
- **Verifiable**: you can 'measure', check compliance
- **Consistency**: a requirement cannot have any conflict with any other


DON'T FEEL BAD BECAUSE YOU'RE AWKWARD, RATBERT.
E-Mail: SCOTTADAMS@AOL.COM
SAdams
THERE ARE PEOPLE LEADING HAPPY LIVES WHO ARE NOT ONLY AWKWARD BUT THEY'RE ALSO HOMELY AND DULL!
© 1993 United Feature Syndicate, Inc.
DO I HAVE TO LEARN ANY COMPUTER SKILLS?
HEY!
IT SEEMS LIKE A REQUIREMENT, BUT IT'S NOT.
10-28

Specification: result

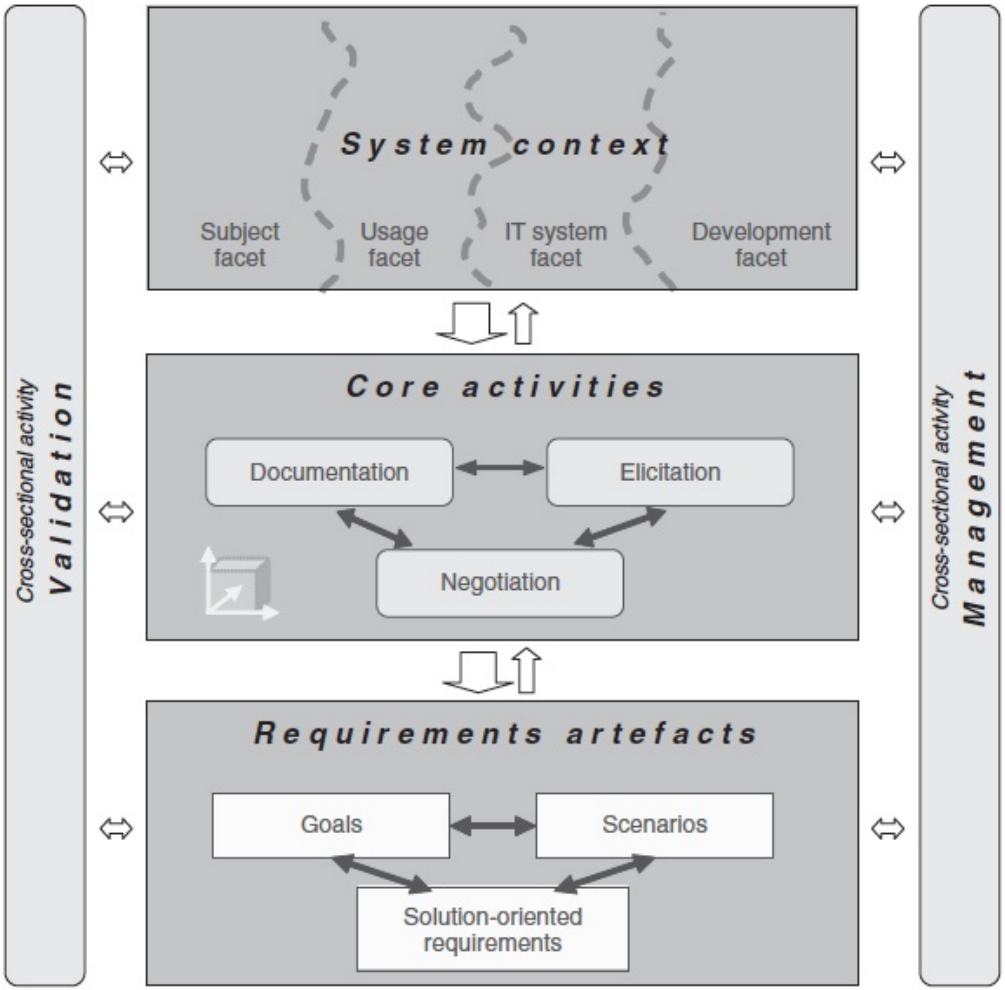
- Catalogue of requirements or SRS (Software Requirements Specification)
  - It is not a design document
  - Must include an index or table of contents, introduction and glossary
  - It should be easily modified and complete
  - It will be present throughout the life cycle of software

Validation and Management

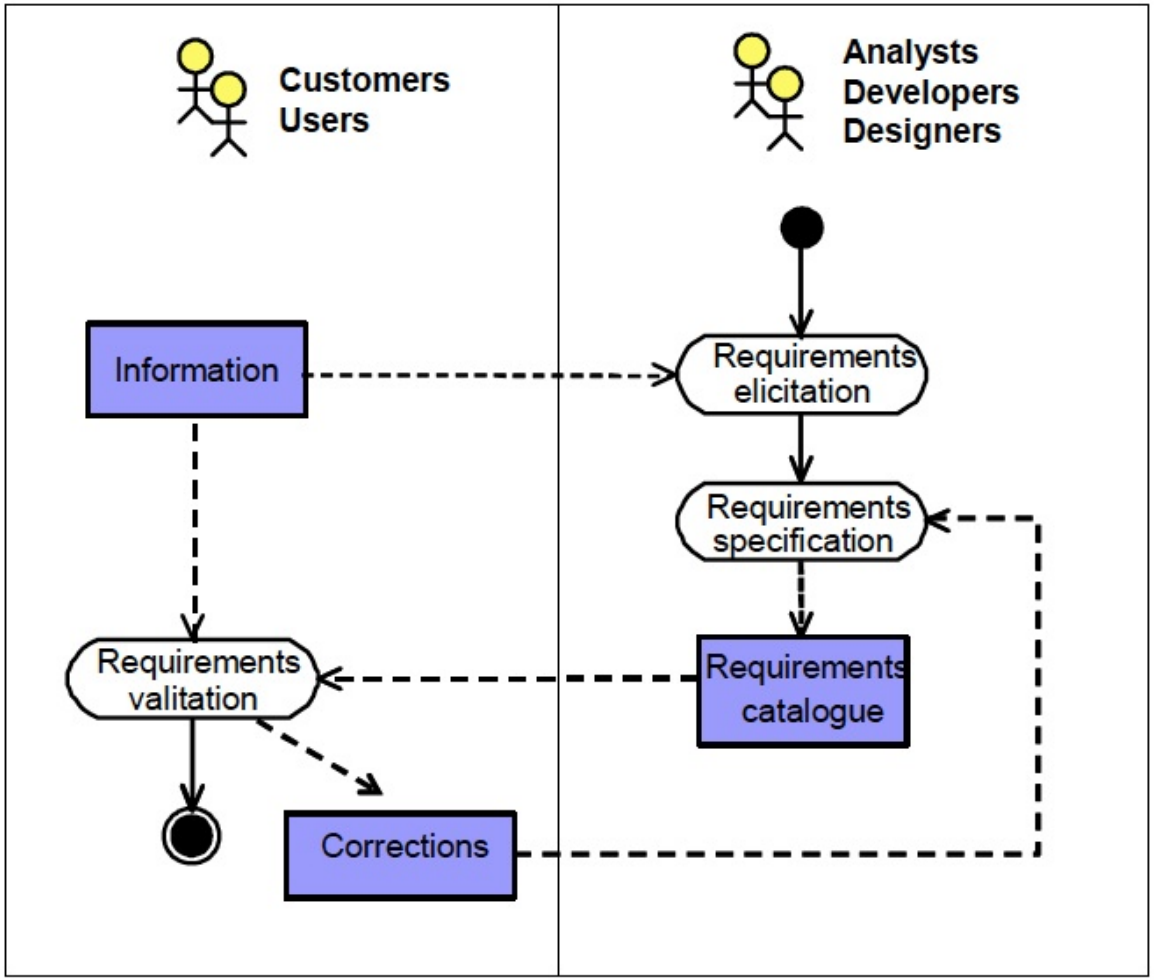
- To joint review with key users
- To modify SRS as a useful document throughout the life cycle of software

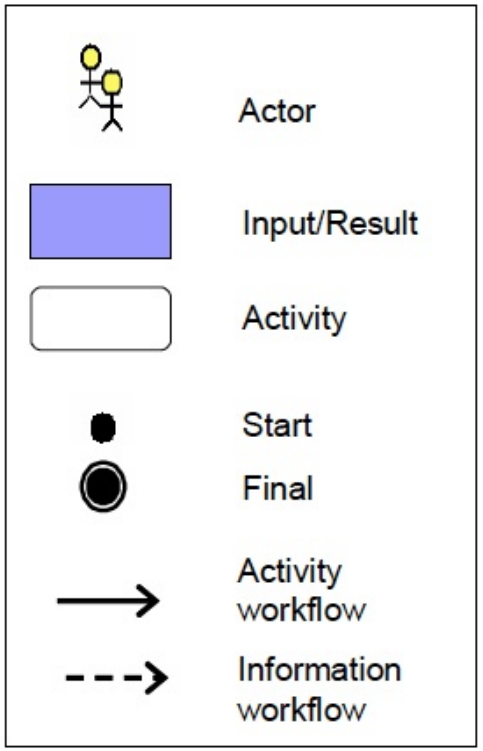
Outline

Requirements Engineering Process [Pohl 2010]



Requirements Engineering Process [Escalona 2004]





Outline

Objective

- To show only external behaviour of the system, the view that future users have on the system
- To show what should perform the system without specifying the form or the way in which it is carried out (it is not a design document, but may take into account the needs of generic hardware)
- To be a reference document throughout the life cycle of software, so it should be easily modifiable

Content

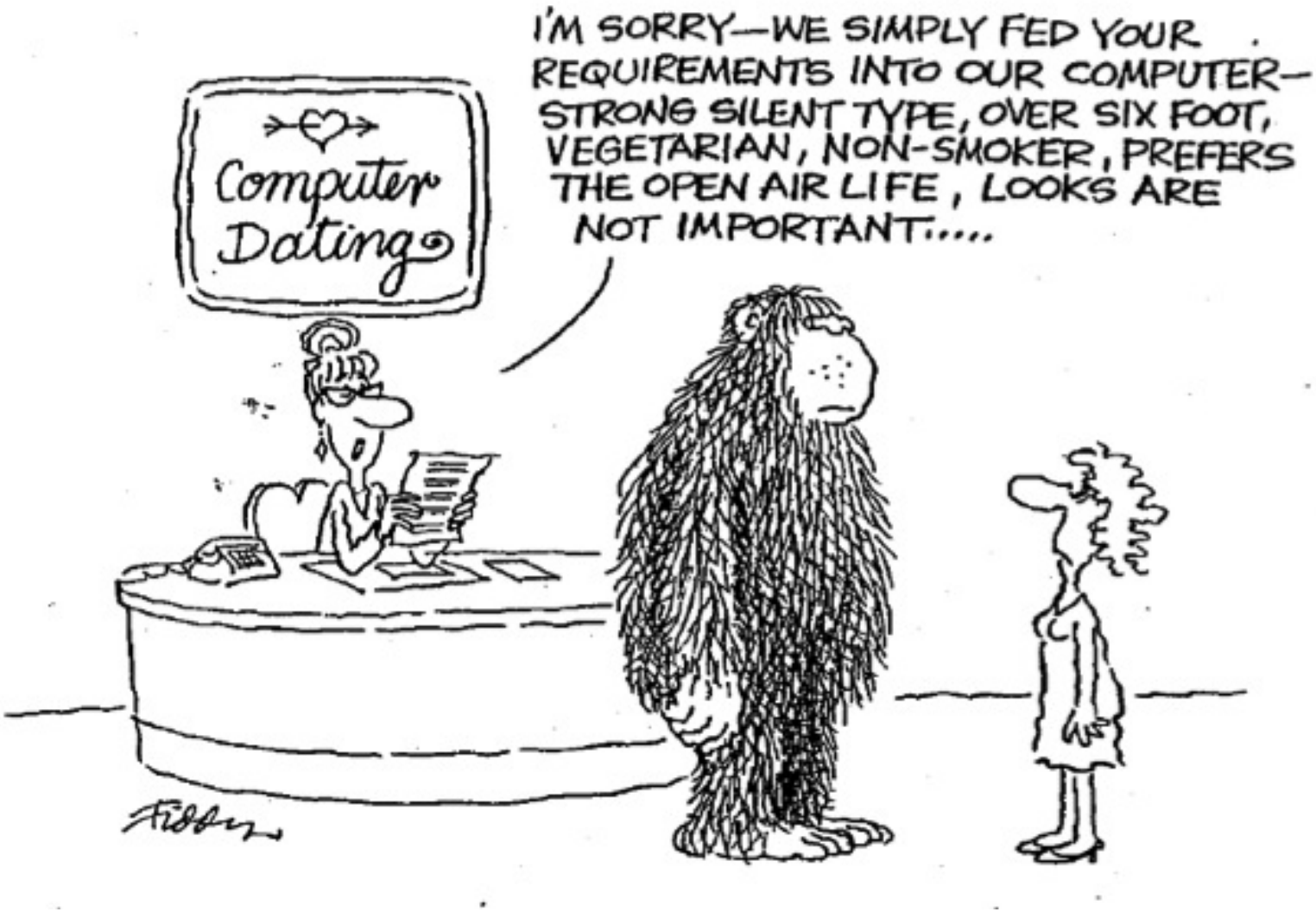
- Possible restrictions on the operation of the system that are needed to consider for the design
- A table of contents, glossary of used terms, and possible changes in anticipation of requirements initially defined

To be used as a ...

- Reference when performing system maintenance
- Reference document for analysts and developers throughout the software life cycle
- Contract to establish the product that is expected to develop, so your content must have the approval of the users and managers of the involved areas

Organisation

1. Index
2. Introduction
3. Review the overall objective of the project
4. Definition of the scope: functional, organisational and computational
5. Definition of the development and exploitation environment
6. List classified by type of the requirements
   - Functional requirements: data, functional and behavioural perspective
   - Quality requirements
   - Constraints
7. Templates for the requirements specification
8. Glossary


I'M SORRY—WE SIMPLY FED YOUR REQUIREMENTS INTO OUR COMPUTER— STRONG SILENT TYPE, OVER SIX FOOT, VEGETARIAN, NON-SMOKER, PREFERS THE OPEN AIR LIFE, LOOKS ARE NOT IMPORTANT.....
Computer Dating