SENG 471

Software Requirements Engineering

What are Requirements?

[Bra02]

Are These Requirements?

1. The elevator shall only change directions, when stopped at a floor.
2. When a lift is within 20 cm of the sensor's position, the sensor sends a high signal; otherwise a low signal.
3. The new system shall occupy no more than 100 Mbytes of RAM.
4. We would like a new course enrolment system that is easier to use than the old one.

Not about the design!

Requirements – Why, what, how, who

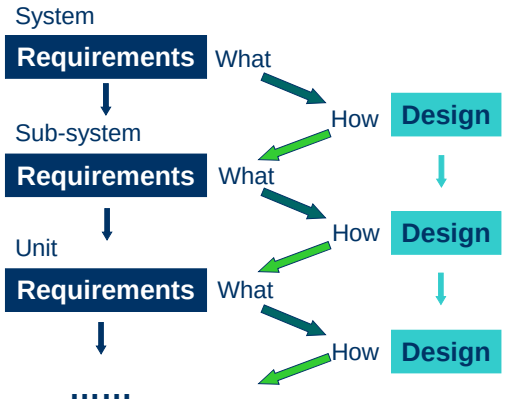
- Enterprise requirements ➔ WHY?
  - Reasons
  - Constraints / conditions
- (System) functional requirements ➔ WHAT?
  - Features
- (System) non-functional requirements ➔ HOW?
  - Constraints
- Client/User requirements ➔ WHO?
  - Problems and constraints

5

Dr. Y. Hu

What vs. How

- Traditionally, specify "what" without "How".
- Missing questions:
  - "Why"+ "Who"

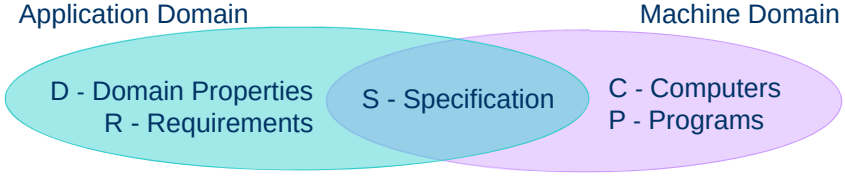


6

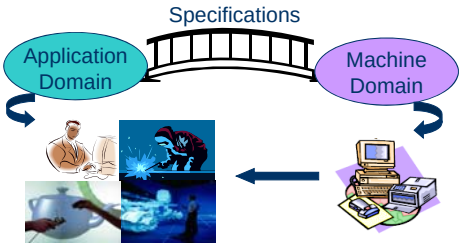
Dr. Y. Hu

* [Zave97], [Gunter00]

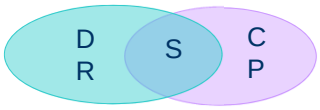
Requirements – The WRSPM Model*



- D ➔ things ……
- R ➔ things ……
- S ➔ behaviors …..



7



Requirements – Verification and validation

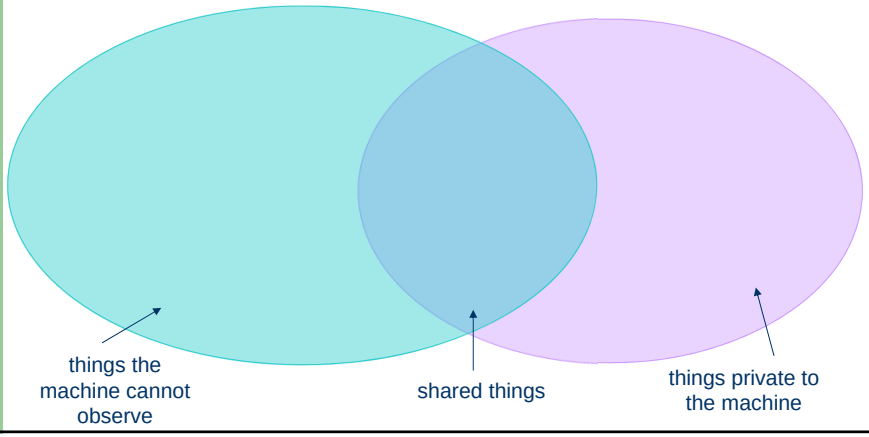
- Verification criteria:
  - S and D entail R ?
  - P on C satisfies S ?
- Validation criteria:
  - Got all the important requirements ?
  - Got all the relevant domain properties ?

8

Dr. Y. Hu

Requirements – Examples

Example 1: Access to a network

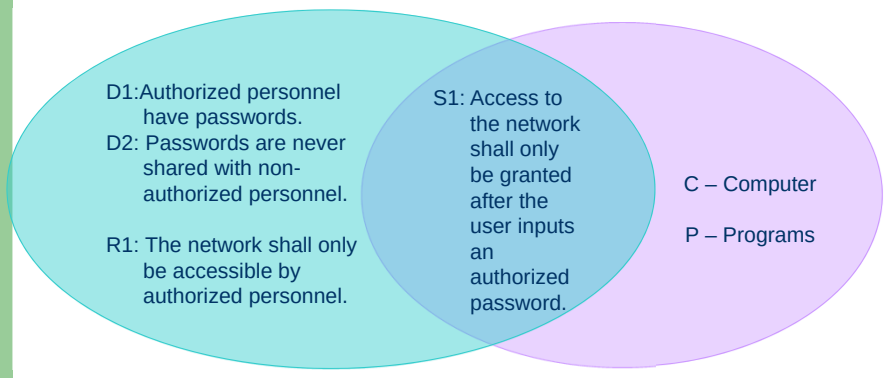


9

Verification: S and D entail R? P on C satisfies S?
Validation: Have we discovered everything needed?

Requirements – Examples

Example 1: Access to a network



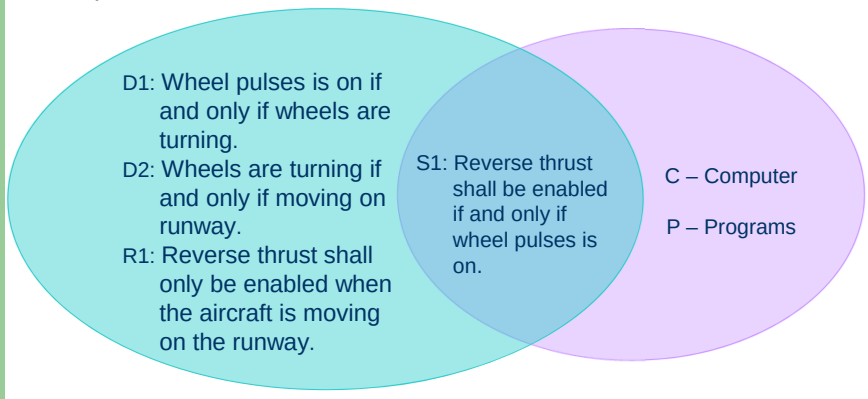
10

Dr. Y. Hu

Verification:
Validation:

Requirements – Examples

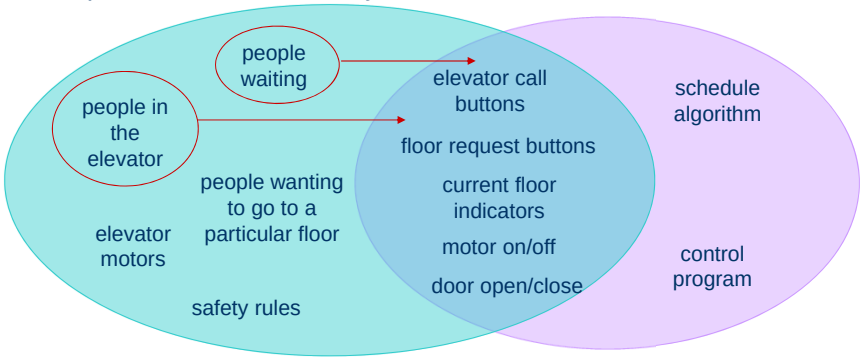
Example 2: Aircraft control



11

Requirements – Moving the boundaries

Example 3: Elevator control system



15

Recap

- Requirements Engineering is about describing problems
  - It is useful to separate the problem from the solution
  - Problems evolve continuously
- Key distinctions:
  - Application Domains vs. Machine Domains
  - Verification vs. Validation