POLYTECH
PARIS-SUD

2025-26

Cycle Ingénieur – 2ème année
Département Informatique

Verification and Validation

Overall Motivation

Burkhart Wolff
Département Informatique
Université Paris-Saclay / LMF
usr.lmf.cnrs.fr/~wolff/teaching.html

Why is it important to get software right?

? ? ?

Why is it important to get software right?

- Since information technology becomes more and more pervasive, the risks become more important
  - Reliability, Safety and Security becomes more critical :
    - transport systems (Cars, Métros, TGV), aviation controls, aerospace, ...
    - critical industriel processes, nuclear power plants, weapons, …
    - medical technologies: tele-surgery, radiation control…
    - critical telecommunication infrastuctures and networks,
    - electronic commerce

This should be the most important reason, but actually, it isn't.

Why is it important to get software right?

- The more likely reason is:

  it is so expensive if you don't !!! (It's the economy, stupid !)

  50 % of the overall costs were spent for test and verification in large software projects ... So, if the development of MS Vista cost 8 billion $ ...

- Another reason is:

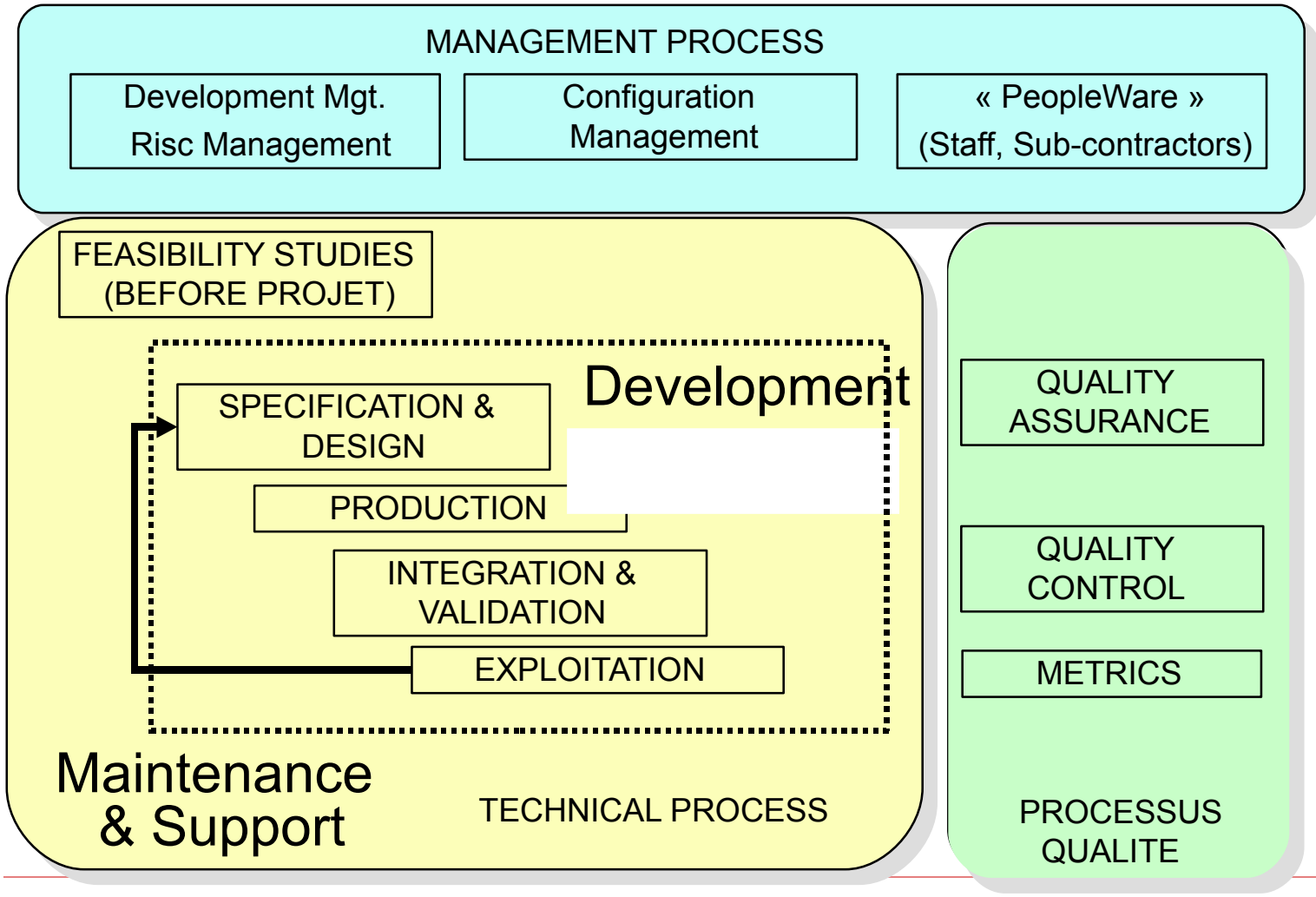
  We want to build more complex systems, and validation and verification techniques are a limiting factor!

  We simply can't do it without !

How can software be «built systematically»?

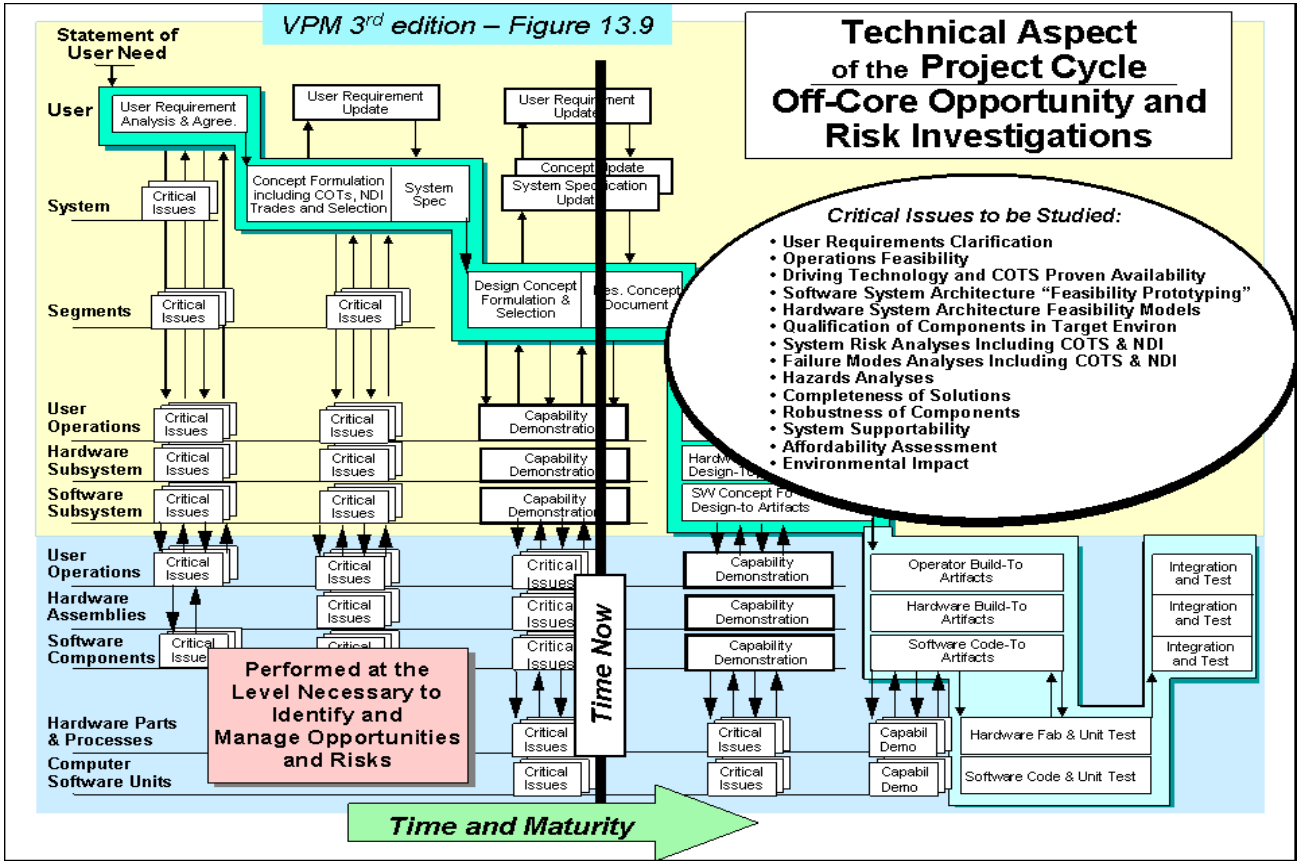
- Organise a development into formally described development process !
  - ... with identified phases,
  - ... staff (and organisation and cost-plans)
  - ... defined deliverables (i.e. documents, codes, ...)
  - ... procedures (and tools !) to validate the quality of the deliverables (reviews, static checks)
  - ... procedures to version and configure deliverables (in particular code)
  - Compare: THE SWEBOOK
    IEEE Computer Society an international standard ISO/IEC TR 19759:2005

A "Software Engineering Process" (example)

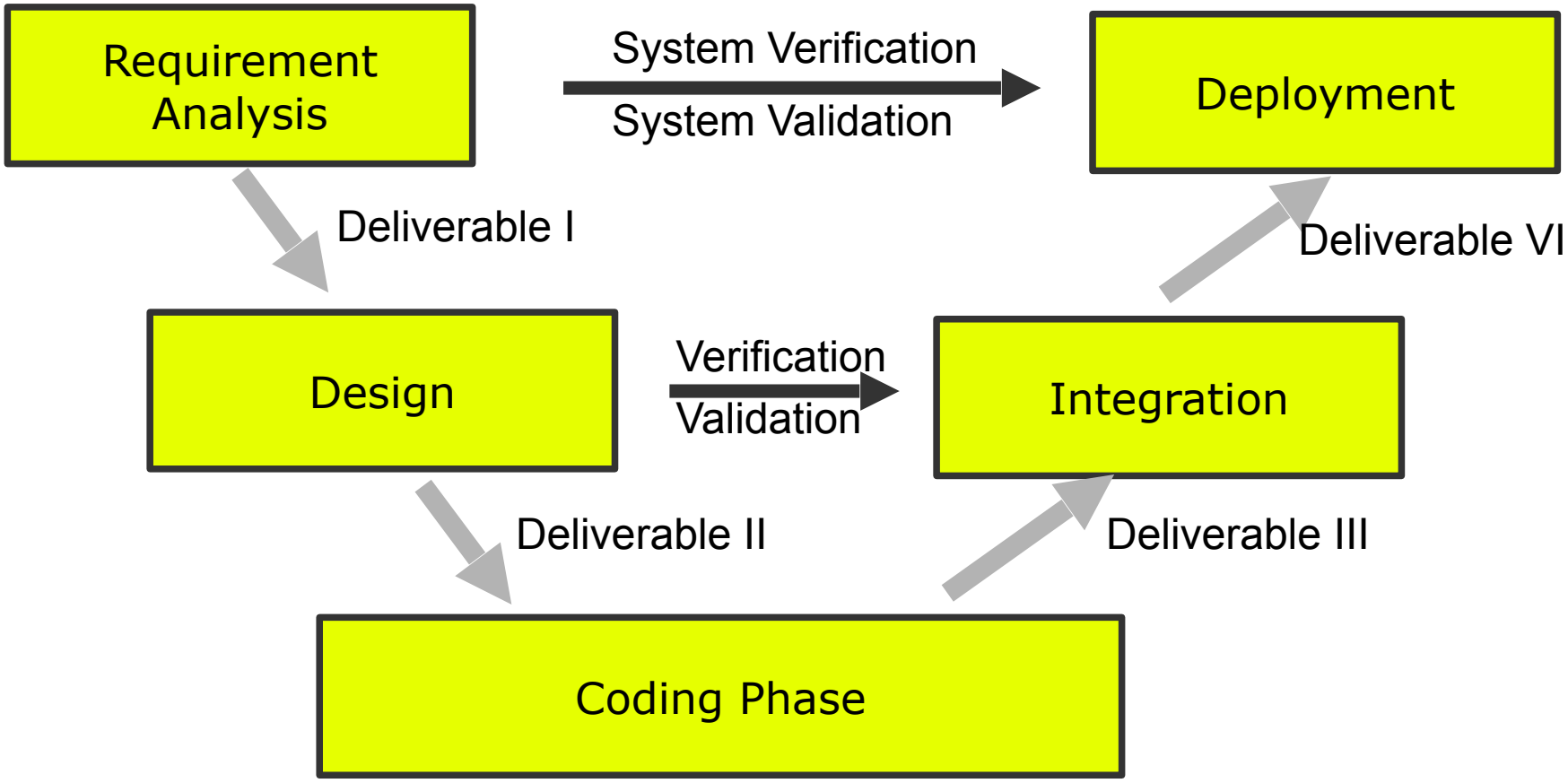


A "Software Engineering Process" (example)

- Another Example: The VPM3-Model (Daimler)



A V-model-like Development Process



Validation and Verification : A Clarification

- Validation :
  - Does the system meet the clients requirements ?
  - Will the performance be sufficient ?
  - Will the usability be sufficient ?

  Do we build the right system ?

- Verification:
  - Does the system meet the specification ?
  - Does it correspond to a (mathematical, formal) model ?

  Do we build the system right ? Is it « correct » ?

Why is so difficult to get software right?

each of these phases in the software engineering process is complicated, but making all this work together is challenging.

In this course, we study the techniques that make sure that a component does, what it was planned to do.