

# Software Intensive Systems

(Capita Selecta in Software Engineering)

(Wednesday, October 27th 2010 — Artemis/ITEA)

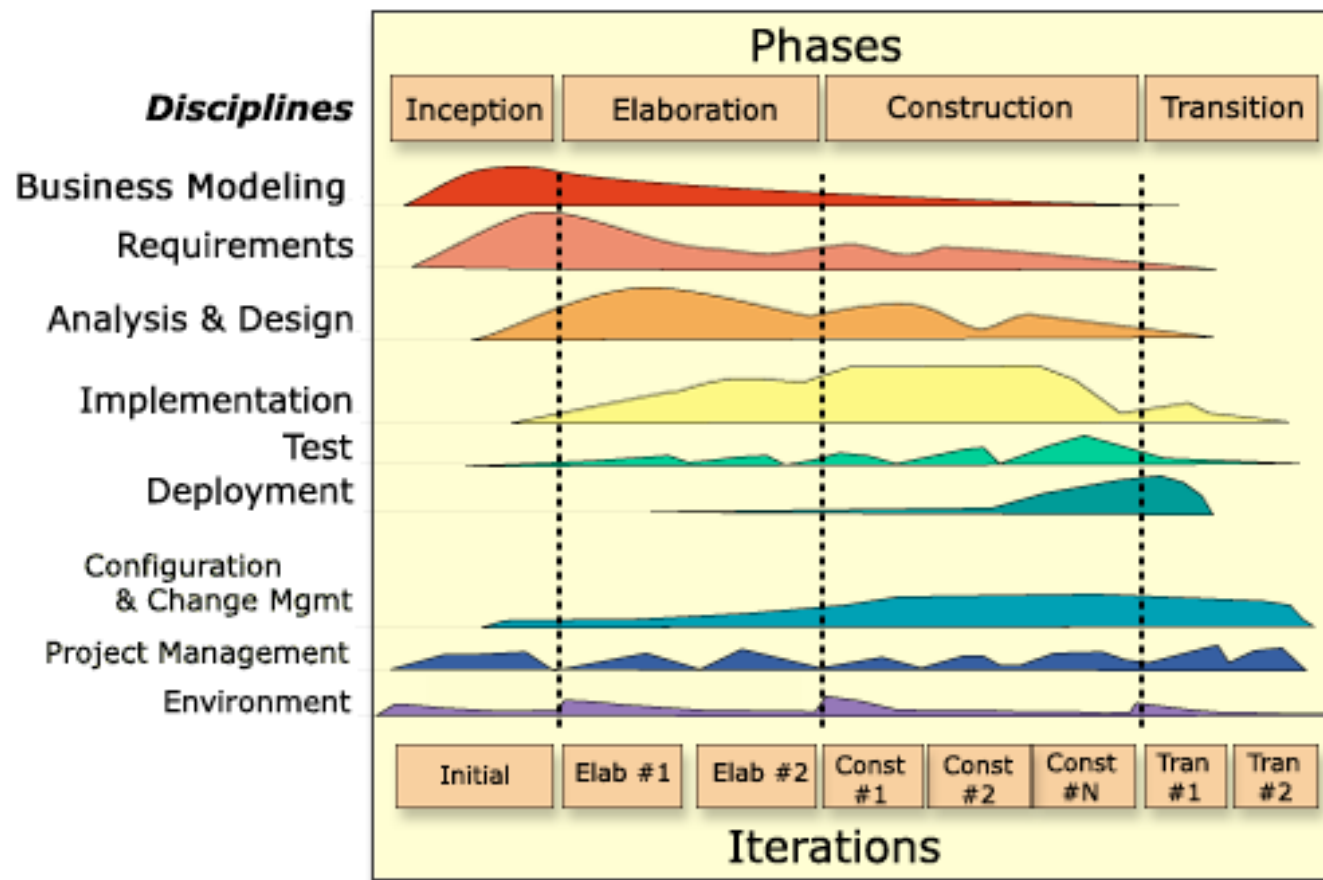
Monday, November 15th, 2010

Monday, November 22nd, 2010

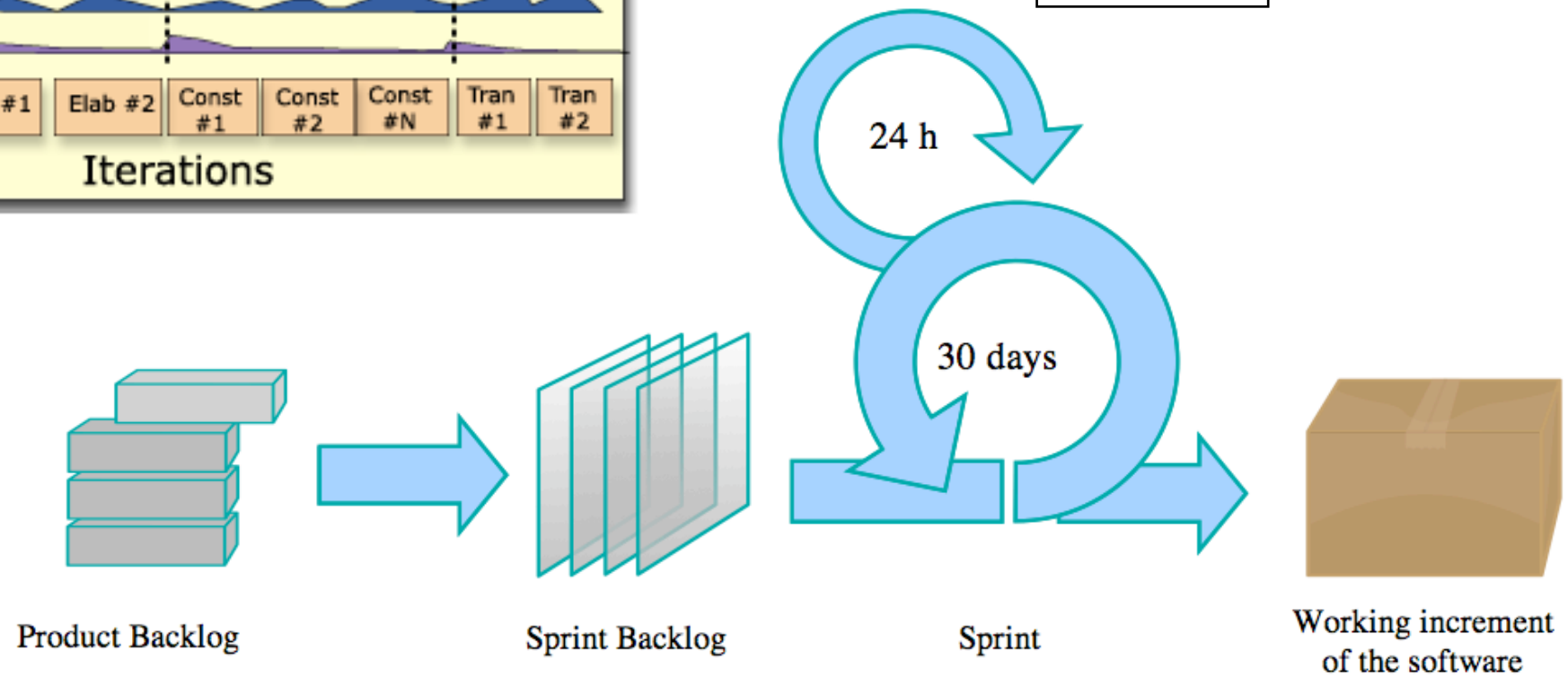


# Mechanistic vs. Organic

## Unified Process



## Scrum



# Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

**Individuals and interactions** over processes and tools  
**Working software** over comprehensive documentation  
**Customer collaboration** over contract negotiation  
**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

© 2001, the above authors this declaration may be freely copied in any form, but only in its entirety through this notice.

<http://agilemanifesto.org/>

# Half-Arsed Agile Manifesto

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through this we have been told to value:

**Individuals and interactions** over processes and tools  
*and we have mandatory processes and tools to control how those individuals (we prefer the term 'resources') interact*

**Working software** over comprehensive documentation  
*as long as that software is comprehensively documented*

**Customer collaboration** over contract negotiation  
*within the boundaries of strict contracts, of course, and subject to rigorous change control*

**Responding to change** over following a plan  
*provided a detailed plan is in place to respond to the change, and it is followed precisely*

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

<http://www.halfarsedagilemanifesto.org/>

# There is no single truth ...

**Mechanistic**



**Organic**



# Program

## November 15th, 2010

### Engineering for High Reliability

- 15:00 - 15:15  
Introduction to Software Intensive Systems (Prof. Serge Demeyer)
- 15:15 - 16:45  
High Availability@KBC (Peter Praet, KBC ICT Services; the ICT branch of the KBC group)
- 16:45 - 18:15  
Developing A Fast Changing Software Product (Mark Plas, MediaGenix; producer of software for managing TV-broadcasting)
- 18:15 - 19:00  
Start up of exam project: Modelling for Automotive (Joachim Denil, KdG).

## November 22nd, 2010

### Engineering for Automotive Systems

- 15:00 - 15:15 -- Introduction to Software Intensive Systems (Prof. Serge Demeyer)
- 15:15 - 16:45 -- Zen and the art of safety engineering (Eric Verhulst, Altreonic; Producer of OpenComRTOS)
- 16:45 - 18:15 -- Developing SW that drives machines (Klaas Gadeyne, Flanders Mechatronics Technology Centre)
- 18:15 - 19:00 -- Preparation of exam project: Modelling for Automotive (Joachim Denil, KdG).

# House Rules

## LECTURES

- 1.5 hour lectures
  - 5 minutes (bio-)break halfway
  - immediate switch over between lectures
- Ask questions \*during\* lectures
- Answer written questions \*after\* lectures

## PROJECT

- Model ABS brake system
  - simulate behaviour
  - estimate processor capacity
- Project assignments (incl report)  
*Friday, April 29, 2011, 18h00*
  - = last day of easter holidays
  - = deadline VeriFast project

# Disclaimer

## Poor Teachers?

- these are *not* teachers

## Poor Cohesion?

- this is *not* a course

## Job advertisement in Academic courses?

- this is an *explicit*  
(sub)goal