

# Software Process Models

**Yvonne Dittrich**

IT-University in Copenhagen  
Design and Use of IT

© Dittrich September 05

1

## today's menu

- about the history of software processes:
  - waterfall, iterative, and agile process models
- how do we use plans
  - and other models of human behavior
- different perspectives on software development
- the unified process

© Dittrich September 05

2

## after today, you should ...

- have an idea about different ways to organise and picture a development process.
- understand how the perspective on software development influences the process model.
- that the process model matters.
- have started to plan and model the process a fictive software development process.

© Dittrich September 05

3

## about the history of software processes

© Dittrich September 05

4



## the agile manifesto

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.

## how do we use plans - and other models of human behavior

- descriptive vs. prescriptive
- plans and situated action
- predictive vs adaptive

## perspectives on software development

'Development generally proceeds from concepts through requirements, etc. with one description being developed by some intellectual or automated process from the preceding description of descriptions.

...

In an ideal world the transformations would yield a sequence of descriptions, resulting in executable programs which satisfied their requirements and the initial concepts. In practice errors and infelicities are discovered during development (and maintenance) which cause iteration, i.e. repetition of the current transformation or rework of earlier descriptions.'

© SERB 15/4

## Software Processes are Software too

We now believe that a software environment is best viewed as a vehicle for the specification of process programs, and for their compilation and interpretation.

In such an environment tools would be treated as operators (...) whose jobs were defined in terms of the need to create and transform software objects (...)

...

Humans would participate in executing such software processes by serving as the execution devices for subprocesses which were not elaborated to sufficient detail to enable interpretation by either tool or the host computing system execution environment

© Osterweil

## "Programming as Theory Building"

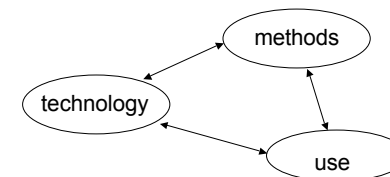
- "The programmer having a theory of the program can explain how the solution relates to the affairs of the world that it helps to handle."
- That means, he is not just able to program in a certain way, but to explain it, to answer queries about it, to argue about it and so forth.
- Such a theory can not be communicated just through texts or diagrams.

© Naur

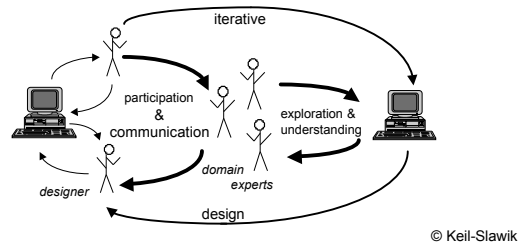
## software development as design

'By Design we mean a specific type of insight-building process that is geared to producing feasible and desirable results within a particular domain.'

© Christiane Floyd



## software development as a learning process



© Dittrich September 05

17

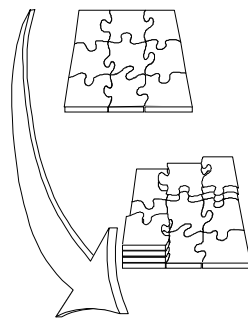
## models and documents in this learning process

- ...are seen as a medium for learning.
- ...should support the shared theory building.
- ...are revised and changed throughout the process.
- ...are written from a special perspective, having a reader and a special purpose in mind.

© Dittrich September 05

18

## Software development from a document centered perspective



- The product consists of a set of related documents.
- They are developed in parallel, and according to the actual stage of the project.
- One of these documents is the program in its current version.

© Dittrich September 05

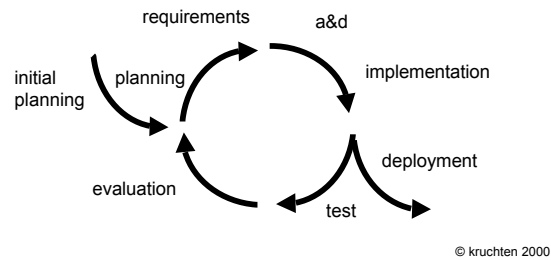
19

## The (rational) unified process

© Dittrich September 05

20

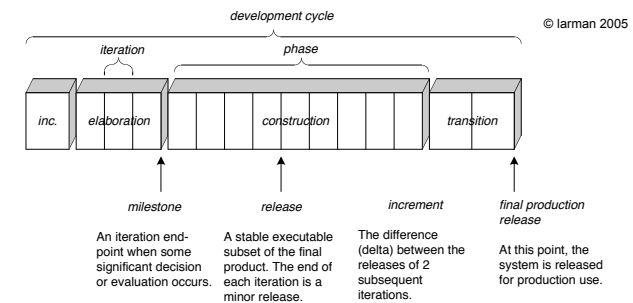
## The RUP way of evolutionary development



© Dittrich September 05

21

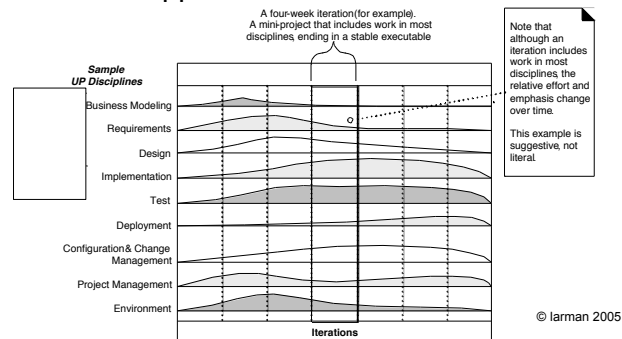
## the development cycle consisting of phases that consist of iterations



© Dittrich September 05

22

## what happens in the iterations



© Dittrich September 05

23

## milestones

- are points in time operating as gating functions based on clear criteria
- provide points at which we can decide to proceed, abort or change course
- progress will be measured in the number of use cases completed, features completed, test cases passed, aso.

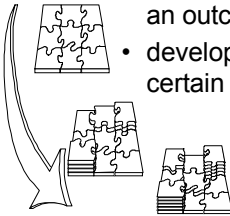
**Sometimes such 'milestones' are also called reference lines.**

© Dittrich September 05

24

## phases

- end with a milestone
- focus on a number of documents as an outcome
- develop a number of documents to a certain state



## disciplines or workflows

- each of the up disciplines is implemented as a rup workflow
- a workflow consists of workflow details which combine
  - workers (role)
  - activities
  - support documents and tools
  - artefacts to be used and elaborated
- concrete people are assigned a role resp. act as a worker

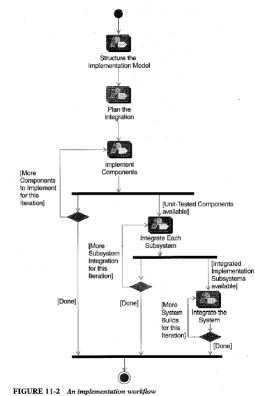


FIGURE 11-2 An implementation workflow

## rup as a product

- provides templates, handles dependencies
- provides a set of tools
- provides a project environment containing the concrete artefacts, the process description, the templates and the tools.