Lecture : 03

Software Process Models

Today’s Objective

1) Software Process Models Definitions

2) perspective vs descriptive process model

3) The Process Cube

4) Standard Process and Tailoring

5) Case Study

6) Software Life Cycles

7) Properties of Process Meta Models

8) Multi-View Process Modeling Notation

Model

- ❑ Representation of a real world process, device, or concept.
- ❑ An abstract representation of an existing entity or an entity to be created, where entity denotes any part of reality or any other conceivable set of elements or phenomena, including other models. With respect to a model, the entity is called the original.

Types of Model

- Descriptive: describe how something is done in reality.

- Prescriptive: tell us how something should be done

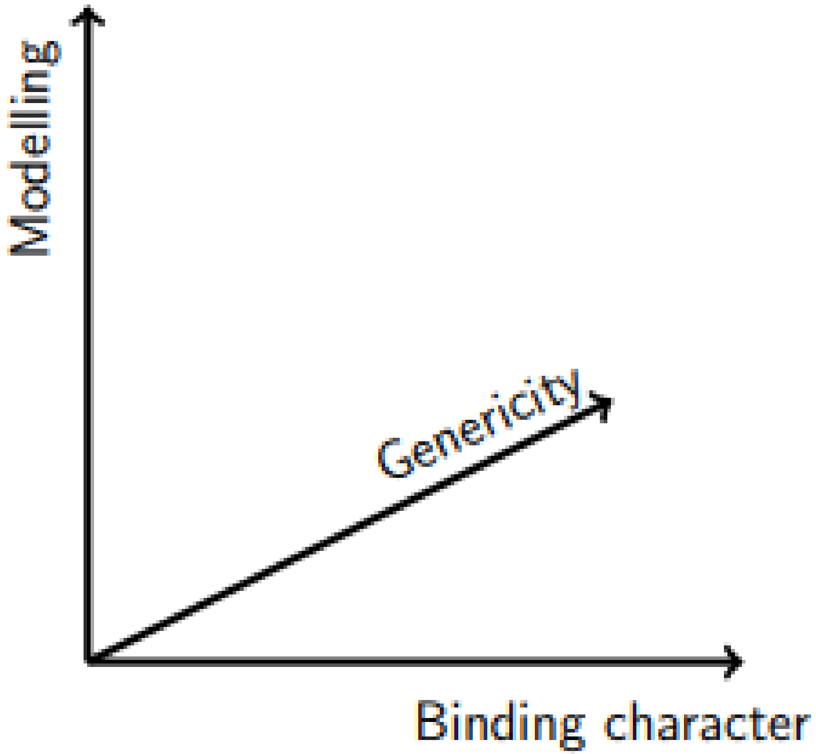
Stachowiak’s General Model Theory

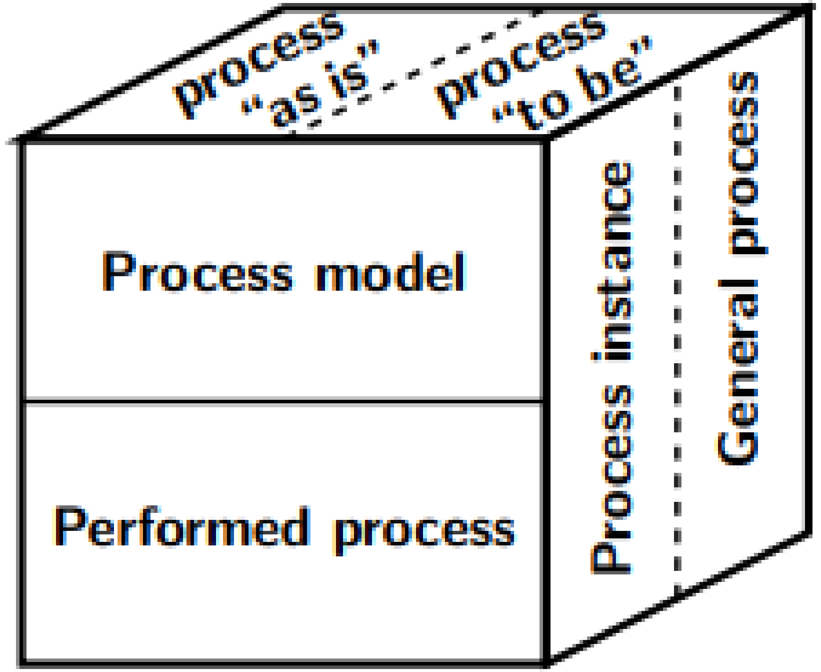
❑ Mapping Property

❑ Reduction Property

❑ Pragmatic Property

Process Cube

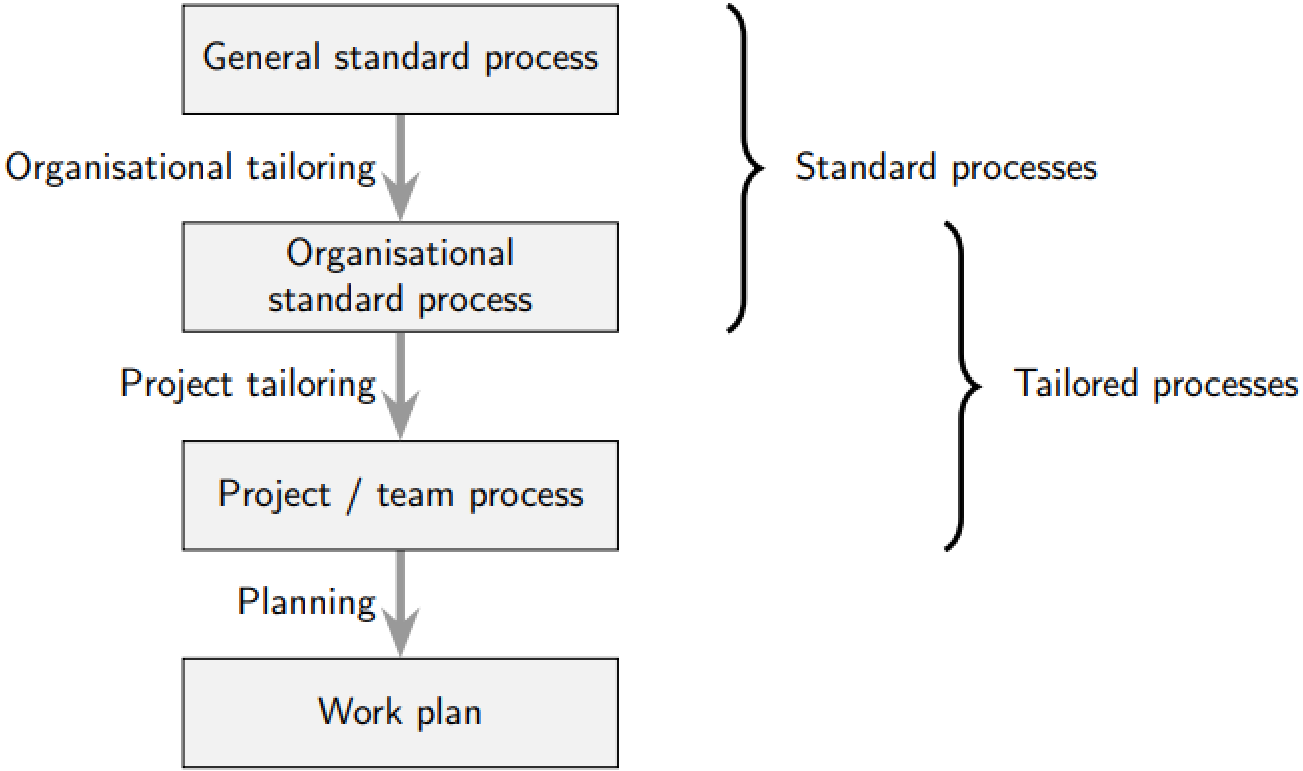




Standard Processes and Tailoring

- ❑ Standard : A set of definitions of the Processes used to guide processes in an organization.

- ❑ Tailoring : Making, altering or adapting a process description for a particular end.



Prescriptive Process Models

- ❑ Lifecycle Models : focus on ‘what’ (in what sequence)

- ❑ Engineering Models : focus on both ‘what’ and ‘how’

Lifecycle Models

❑ A (software/system) lifecycle model is a description of the sequence of activities

carried out in an SE project, and the relative order of these activities.

❑ Normally, a lifecycle model covers the entire lifetime of a product. From birth of a commercial idea to final de-installation of last release

❑ Note that we can sometimes combine lifecycle models, e.g. waterfall inside evolutionary – onboard shuttle software.

❑ We can also change lifecycle model between releases as a product matures,

e.g. rapid prototyping → waterfall

Waterfall Model

- ❑ First formally described by Winston Royce in 1970.
- ❑ Sequential creation of products on different levels of abstraction and integration in reverse direction.
- ❑ Advantage: simple version and configuration management

Disadvantages:

- ❑ Hard to achieve
- ❑ Need requirements to be stable. Unexpected changes are high risks
- ❑ Gaining experience and learning during the project is difficult
- ❑ Documentation is heavyweight
- ❑ Not suitable for large projects and long cycle times

Iterative Enhancement Model

- ❑ Basili and Turner in 1975
- ❑ Implement part of a complete system and then add functionality in a number of iterations. (each iteration may be completed in waterfall style)
- ❑ Advantages: supports efficient learning
- ❑ Core of the final product is available very early
- ❑ Involves customer very early
- ❑ Challenges: good and comprehensive version management and configuration management is required
- ❑ Integration may become difficult
- ❑ Many modern SP Models a re based on this (scrum, UP, XP)
- ❑ Mostly used when a new technology is being used

Prototyping Model

- Only selected properties of the final system is taken into consideration.
- May be built in order to evaluate the user interface of the system and therefore ignore important properties. (such as performance and security)
- A prototype is an experimental model either functional or non functional of the system or part of the system
- 4 parts (functional selection, construction, evaluation, future use)
- Exploratory & experimental prototypes
- Evolutionary & throw away prototypes
- Mostly suitable for online web based systems

Spiral Model

- ❑ Barry Boehm in 1986
- ❑ Represents a risk driven approach i.e. assessment of the risk determines the next phase.
- ❑ Combines aspects of (waterfall, incremental and prototype)
- ❑ Good for large mission critical systems
- ❑ Disadvantages: costly, requires high expertise , not suitable for smaller projects