The University of Western Australia
School of Computer Science and Software Engineering

CITS5502 Software Processes

Lecture 3

The Concept and Use of a Software Meta-Process

Key concepts

- Hierarchy of levels of control in software development
- Meta-CASE
  - Tools to produce CASE tools
- Need to understand the influence and interaction between the levels
- Meta-Risk – the risk that the risk management plan is incorrect, inadequate or has omitted some important threat
- The PIR included within the PPET (Personal Process Estimation Tool) served as a meta-process (2^{nd} order change) for your estimation tool
- The Metrel Rules

Meta-Process and Meta-CASE

- The Concept of Meta-Process

Meta-process = a process for improving processes

- CASE = Computer-aided software engineering
- *CASE tools* refers to the tools that we use in SE to remove some of the drudgery in producing software.
- A *Meta-CASE tool* therefore refers to a tool
  - that manages our CASE tools, or
  - that improves / customizes our CASE tools

  for a particular industry, e.g., working in AI.

Risk Management Plan

- There is a whole range of factors that can be included in our risk analysis for a project, e.g.,
  - What is the likelihood that the project would be behind schedule?
  - How likely is it that we would go over budget?

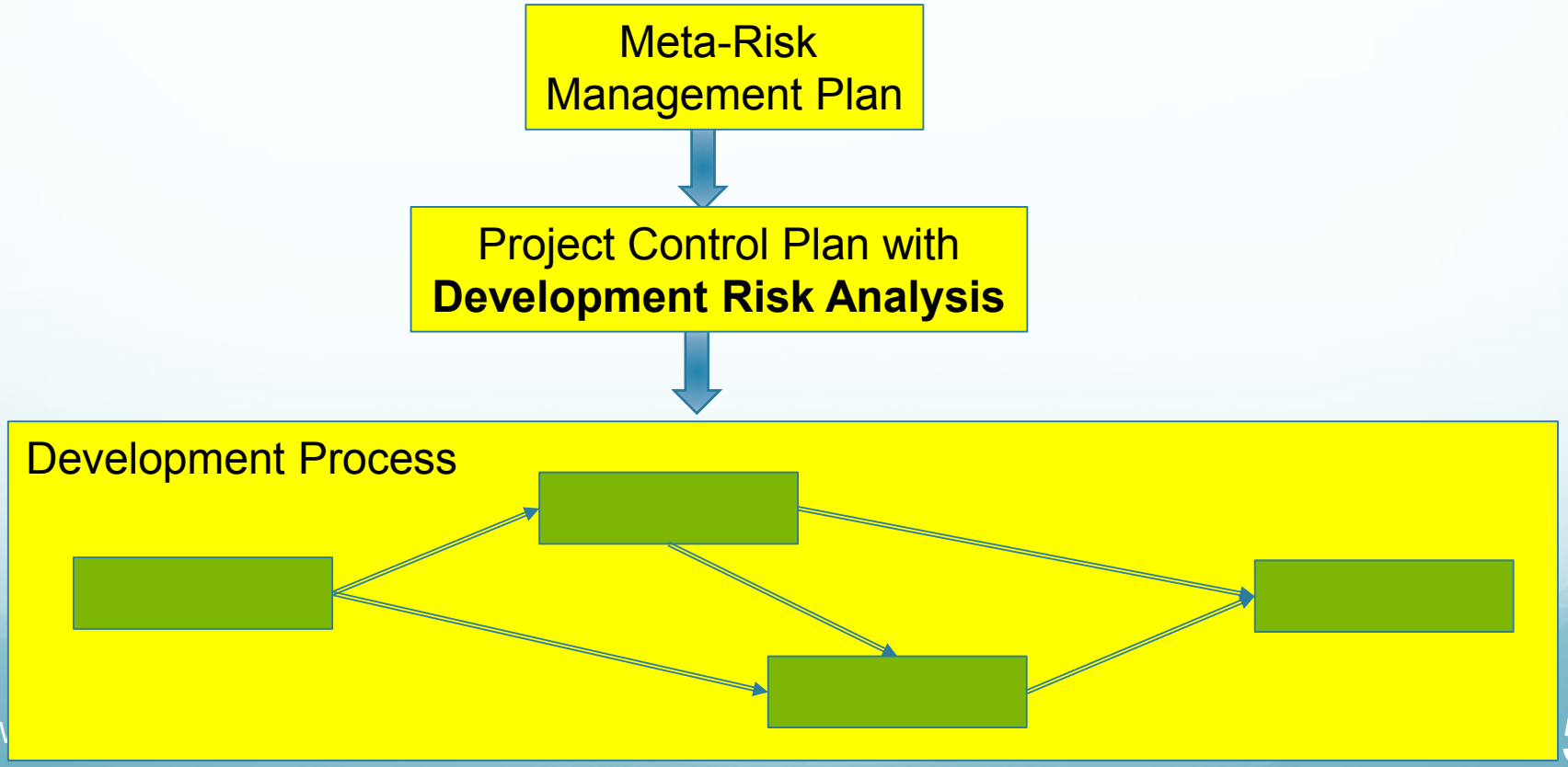
  ➔we need a risk management plan to cover all these risks

- But…

  What if our risk management plan is wrong? E.g. missing out some important risk factors? Or missing out that some risk factors are correlated?

Meta-Risk Management Plan

- We need a Meta-risk management plan (top yellow box) to overlook our project control plan (middle yellow box), which overlooks our development process.



PIR

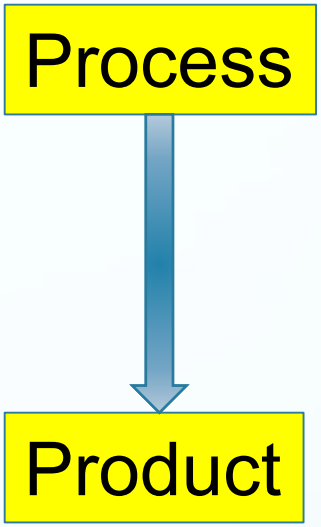
- PIR = **Post-implementation review**.
- Sometimes also called **post-project review** or **post-mortem**.
- PIR is one way of looking at improvement of the process, by
  - conducting a full review at the end of the project to find out what went right and what went wrong, and
  - summarizing the findings in a PIR report

  Note: many organizations claimed that they did PIR but they actually only did a review on the product (not the process)

PIR (cont.)

- Many organizations consider PIR adds overhead to their process and overlook the importance of PIR.

  However, a 2% (say) overhead is likely to bring in more than 2% improvement to the next project, as *Process* has a strong influence on the *Product*.



The Metrel Rules

- **For any valid product metric, its derivative w.r.t. time is a valid process metric.**
- **For any valid process metric, its derivative w.r.t. time is a valid metric for the organization.**

  Woodings and Bundell name these the **Metrel** (Metric Relationship) **Rules**.

- If M is a valid *product metric*
  then $\frac{\partial M}{\partial t}$ is a valid *process metric*
  e.g., M could be a measure on the number of defects in the system, then $\frac{\partial M}{\partial t}$ would be a measure on the rate of change of this number over time, i.e., how fast can the defects be removed? (here we assume that the defects are independent of each other.)

The Metrel Rules (cont.)

- Let M be the number of defects.

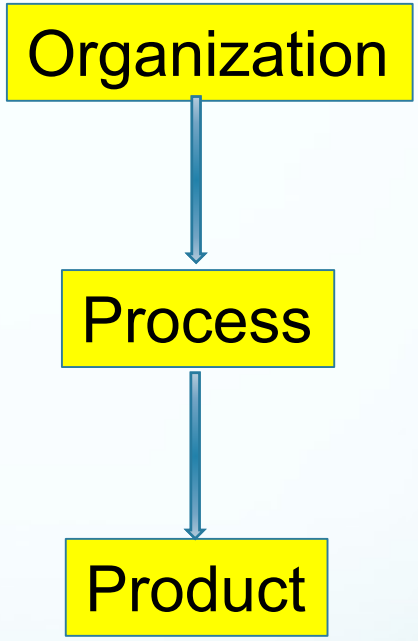
  We can have two different measures for $\frac{\partial M}{\partial t}$, one for the "Code" and one for "Test":

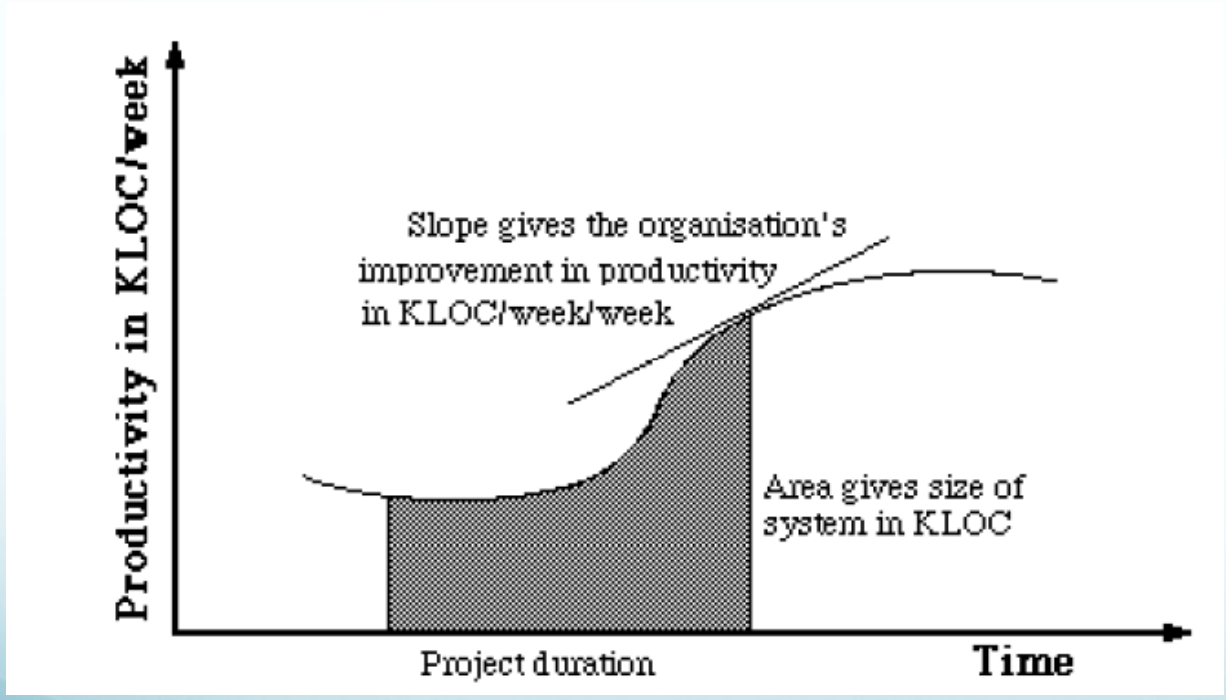
  $\frac{\partial M}{\partial t}(\text{Code})$ is the **defect proneness**, i.e. the rate defects are introduced to the code during code development process;

  $\frac{\partial M}{\partial t}(\text{Test})$ is the **defect removal efficiency**, i.e., the rate defects are removed in the Test process.

The Metrel Rules (cont.)

- The *Organization* has major influence on the *Process*
- Let M be a measure on the *Product*, then $\frac{\partial^2 M}{\partial t^2}$ is a measure on the *Organization*





Three metrics portrayed as variation over time (Woodings & Bundell, 2001)

Taxonomy of Metrics

(extracted from Woodings and Bundell, 2001)

- Measures of Customer Satisfaction
  - Proportion of requirements met
  - Amount of usage of new system
- Product Metrics
  - System size
  - Complexity of design at various levels of resolution, e.g., control flow
  - Quality attributes

Taxonomy of Metrics (cont.)

(extracted from Woodings and Bundell, 2001)

- Process Metrics
  - Risk Assessment – given as probability × cost of impact
  - Open question clearance rate
  - Defect occurrence rates and propagation
- Organization Metrics
  - Profitability (counting the dollars)
  - Staff job satisfaction (shown by staff turnover, sick leave)
  - Organizational Capability measured against, e.g., previous performance and industry average performance.
- Driver Metrics
  - Management Commitment to improvement
  - External pressures for change (e.g., % of competitors with better measurements)

Recommended Reading

- Sommerville *Software Engineering,* 7th Ed., briefly mentions Meta-CASE in Section 13.4 “Language Processing Systems”
- Woodings, T. L. and Bundell, G. A. “A Framework for Software Project Metrics”, Proceedings of the 12th ESCOM Conference on Software Control and Metrics, London, Apr 2001.