INF5180: Software Product- and Process Improvement in Systems Development

Part 08:

Learning from Experience

UNIVERSITETET I OSLO

Dr. Dietmar Pfahl

email: dietmarp@ifi.uio.no

Spring 2010

Topics

- Learning – Basics
- Learning – Skills
- Learning – Organizational Aspects

Terminology

- Data:
  - Symbols organized according to syntactic rules (Syntax)
- Information:
  - Data interpreted in a certain context (Semantics)
- Knowledge:
  - Information, when related to the human mind in order to solve problems; i.e., it is human expertise stored in a person's mind, gained through experience and interaction with a person's environment (Pragmatics)

Terminology

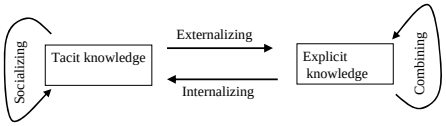
- Experience:
  - The type of knowledge a person acquires by being involved; i.e., it is observation + emotion (with respect to the observed event) + conclusion (derived from the observed event and emotion)
- Know-how:
  - Procedural knowledge
- Skill:
  - Talent and/or ability to perform a task (thus, knowledge is a prerequisite of skill)

Terminology

- Knowledge Worker:
  - Knowledge workers contribute to company success mainly by gathering, organizing, and applying knowledge
- Knowledge Management (KM):
  - KM addresses the following tasks:
    - Acquiring new knowledge
    - Transforming it from tacit into explicit knowledge and back again
    - Systematically storing, disseminating, and evaluating it
    - Treating it as an asset and its infrastructure as a resource
    - Applying knowledge in new situations

Tacit (silent) versus Explicit Knowledge

- Tacit knowledge refers to knowledge that people use every day to perform tasks but which they find hard to express or do not even know about.
- Explicit knowledge is documented knowledge. Others may access and reuse this knowledge.

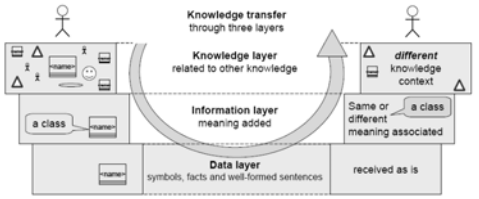


(Nonaka Model)

http://www.cyberartsweb.org/cpace/ht/thonglipfei/nonaka_seci.html

A Layered Model of Knowledge Transfer

Source: Kurt Schneider, Univ. Hannover



UNIVERSITETET I OSLO

Question: What are pre-requisites for knowledge reuse?

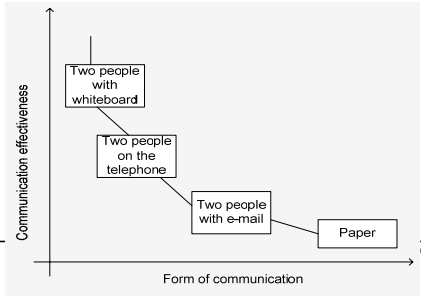
Learning

"I hear and I forget. I see and I remember. I do and I understand."
Chinese Proverb

- Learning and competence development are important elements in process improvement work
- Relations between teacher(s) and learner(s):
  - One-to-one *(mentor arrangements, guru-student-relation in Asia)*
  - One-to-many *(traditional education)*
  - Many-to-many *(professional forum (e.g., conference), experience data bases)*
  - Many-to-one *(inexperienced project member)*
- Learning may activate:
  - Hearing, vision, smell, taste, movements
  - Different parts of the brain (emotions, intellect, creativity)
- Learning may be:
  - Active (participating), passive, "single loop", "double loop",.....
- There exist many different learning theories
- There exist many different learning styles

UNIVERSITETET I OSLO

Learning and Communication



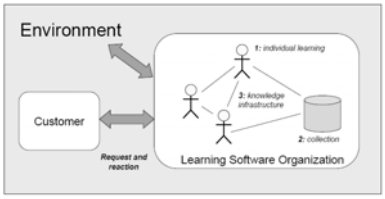
UNIVERSITETET I OSLO

Learning

- Organizations may have different approaches to learning:
  - Develop own knowledge versus infusing extern knowledge
  - Planned (formalized, tested) versus ad-hoc competence development
  - Evolutionary versus revolutionary approach
  - Systematic versus ad-hoc experience transfer
  - Dedicated training courses versus "on-the-job-training"
  - Focus on what is most important in the value chain vs. treating everything as equally important
  - Focus on the individual versus the group
  - Focus on product versus process …

Levels of Learning

- Individual
- Group
- Organization
- (Society)



Source: Kurt Schneider, Univ. Hannover

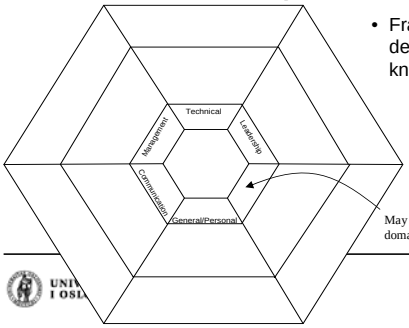
Topics

- Learning – Basics
- Learning – Skills
- Learning – Organizational Aspects

Skill Levels (Hohmann)

- Innocent
  - You are not even aware of the domain..
- Aware
  - You are introduced to the domain
- Apprentice
  - You have some training within the domain and have some experience → the start of *library of cognitive plans*.
- Practitioner
  - You have accomplished relatively complicated tasks within the domain → rather developed *library of cognitive plans*.
- Journeyman
  - You work daily within the domain and start to find your own tailored methods → well developed *library of cognitive plans*.
- Master
  - You master a wide range of tasks within the area and adjust intuitively the method to your needs → you have a wide and sophisticated *library of cognitive plans*.
- Expert
  - You are an distinguished expert and publish/share your knowledge actively → externalizing your *library of cognitive plans*.

Hohmann's Self-Development Framework

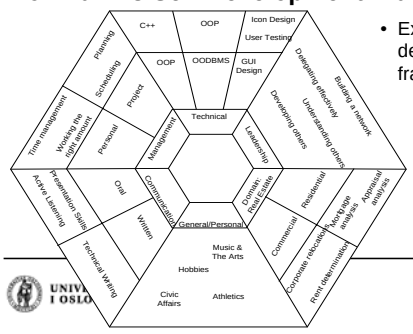


- Framework for development of knowledge

Question: What are typical application domains of software engineering?

May be used for domain competence

Hohmann's Self-Development Framework

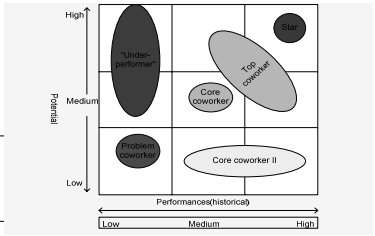


- Example of a self-development framework

Categorizing Knowledge Workers

- Ref Eduviva (www.eduviva.no)

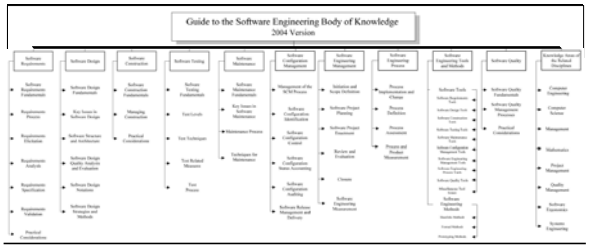
Portfolio analysis of existing human resources may be a useful tool for uncovering gaps in competencies and elaborating training and career plans.



Categorizing SE Skills

- IEEE, together with some partners (Rational, SAP, Boeing etc...), have specified *Software Engineering Body of Knowledge* which is meant to be a standard for training and certification of software developers. See www.swebok.org.
- This standard has become:
  - Rather lucid and seems to be complete regarding what constitute Software Engineering. It may be used as a definition of this term.
  - Very extensive (like most other large consortiums)
  - Possible to use at a high level of abstraction
  - Well synchronized with the maturity model CMM(I)

SWEBOK Knowledge Categories www.swebok.org



Competence Profile - Roles

	Skill areas	Product Manager	Project Responsible	Project Manager	Developer	Configuration Manager	Test Responsible	Release Responsible	Change Responsible
Software Engineering									
	Software Requirements	4	3	3	2	1	3	3	2
	Software Design	2	2	3	3	2	2	2	2
	Software Construction	2	2	2	2	3	2	4	2
	Software Testing	2	2	2	3	1	4	3	3
	Software Maintenance	3	2	3	3	3	2	3	4
	Software Configuration Management	2	2	2	3	4	2	4	3
	Software Engineering Management	4	3	3	1	1	1	2	2
	Software Engineering Processes	3	3	3	2	1	1	2	1
	Software Engineering Tools and Metho	1	2	2	2	2	2	2	2
	Software Quality	4	3	3	2	2	3	3	2
Specific areas	...								
Social areas	...								
Domain specific areas ...									

Pfahl

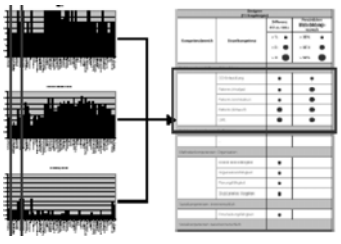
Individual Gap Analysis

Skills GAP analysis					
Name: <name>		Current status	Personal wish	GAP	Action
Software Engineering					
	Software Requirements	2	3	1	Attend 3-day course
	Software Design	3	4	1	Attend course, go to 2 conferences
	Software Construction	2	2	0	
	Software Testing	2	2	0	
	Software Maintenance	1	1	0	
	Software Configuration Management	3	2	-1	
	Software Engineering Management	2	2	0	
	Software Engineering Processes	2	2	0	
	Software Engineering Tools and Metho	3	3	0	
	Software Quality	1	3	2	Attend course, participate in QA-audits
Social	...				
Specific areas	...				
Domain specific areas ...					

UNIVERSITETET I OSLO

Gap Analysis – Example: QUALISEM-People

- Required Skills
- Current Skills
- Training preference



UNIVERSITETET I OSLO

QUALISEM-People – Steps

1. Selection of an adequate set of standard profiles, specific roles and employees within the company.
2. Tailoring of the standard profiles in order to meet customer needs and to fit in with the specific company context.
3. Definition of the target profiles based on a role-based questionnaire in which either the employees or company managers rate desired performance levels in relation to the specific skill competences. In completing the questionnaire it is also important to take into account the future needs of an organization or department, as well as new methods that may be applied.
4. Assessment and documentation of the actual competences are developed on the basis of a role-based questionnaire in which the employees rate their performance level in relation to the specific competence areas of their role.
5. Elicitation of qualification preferences based upon the questionnaire ratings of the employees.
6. Comparison and aggregation of the data from stages 3 and 4 resulting in a skills gaps analysis. Aggregation of data relating to qualification preferences from stage 5. Balancing of the skill gaps and qualification preferences.
7. Stakeholder workshop - the objective of which is to prioritize the skill gaps and identify the preferred ways in which to provide training for them.

Standard Skill Profiles for ICT Roles – Categories

Knowledge (Cognitive Competence)

- Declarative and tacit knowledge (breadth, kind)
  - *Application of knowledge*
- Understanding
  - *Comprehension*

Skills (Functional Competence)

- Range and Selectivity
  - Ability to *select* from a certain range of skills (and tools, methods, procedures…)
- Decision Taking based on:
  - *Analysis*
  - *Evaluation*
  - *Synthesis*

Wider Competences (Pers. Competences)

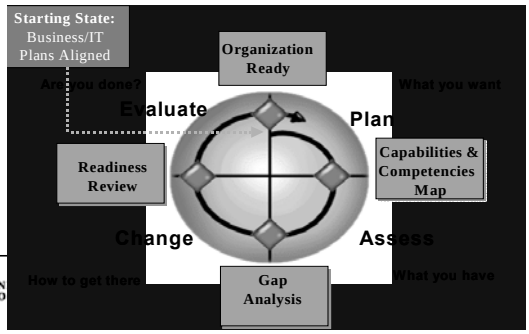
- Autonomy/Responsibility
  - *Autonomy*
  - *Responsibility*
  - *Context* (Ability to operate within context)
- Learning Competence
  - *Learning to learn*
- Social Competence
  - *Communication*
  - *Cooperation* (including Role)
- Professional Competence
  - *Problem Solving*
  - Training (and briefing) others (*Transfer of Knowledge*)

Microsoft Readiness Framework (MRF)

- In the context of Microsoft Readiness Framework, ***readiness*** means the state achieved by either an individual or an organization as a result of activities geared toward "getting ready for" new technology, including "getting ready to" plan, build, manage or operate that technology.

- Organizational readiness
  - Leadership
  - Culture
  - Process
  - **Skills**
  - Hardware
  - Software
- Individual readiness
  - **Knowledge**
  - **Skills**

MRF Process



Microsoft Skills

"Skills, like hardware, are assets. They have value, require an investment, and depreciate over time. Like any asset, they need to be constantly examined and re-evaluated. A skills management system is vital to an organization's ability to develop its human capital. A skills management system has three primary components:

Competency Management – organizations must think about development of their intellectual capital from a competency management perspective. What skills are important to the organization? What job roles are important?

Assessment Management – how do companies assess whether their employees have the required knowledge and skills? If companies cannot make this assessment, they cannot measure the return on investment in hiring, training, and career development.

Learning Management System – an organization can measure the difference between competencies and assessment or, assess where the competencies don't meet the requirements and then implement a personalized learning plan that will provide development (closing of the skills gaps) for each employee."

Microsoft Skill Levels

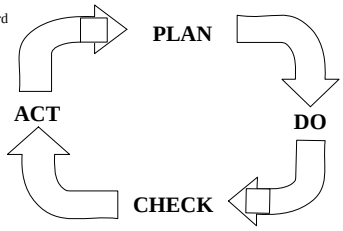
Skill Level Rating	Simple Description	Description
0	No Experience	Not applicable.
1	Familiar	Familiarity: Skill in formative stages, individual has limited knowledge. Not able to function independently in this area.
2	Intermediate	Working knowledge: Good understanding of skill area, is able to apply it with reasonable effectiveness. Functions fairly independently in this area, but periodically seeks guidance from others.
3	Experienced	Strong working knowledge: Strong understanding of skill area, is able to apply it very effectively in position. Seldom needs others' assistance in this area.
4	Expert	Expert: Has highly detailed, thorough understanding of this area and is able to apply it with tremendous effectiveness in this position. Often sought out for advice when others are unable to solve a problem related to this skill area.

Topics

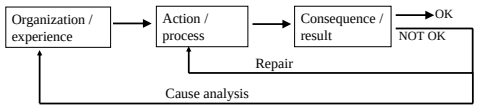
- Learning – Basics
- Learning – Skills
- Learning – Organizational Aspects

Process Improvement Processes – PDCA

PDCA, Deming/Sheward



Learning in (Single- and Double Loops)

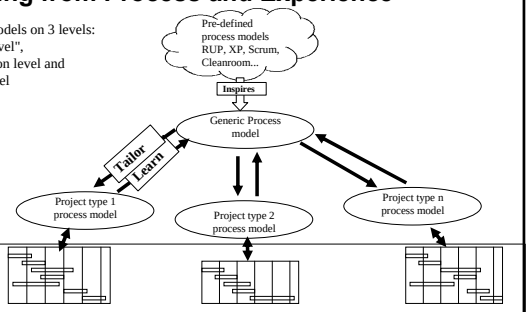


- We get a better product when we correct a fault, but if we don't eliminate the root-cause of the defect there is always a risk that the same fault is injected over and over again.
- By seeking the cause of the fault, we are able to
  - Remove systematic faults once and for all
  - Get the opportunity to improve the process which caused the fault(s)
  - Nurture innovation

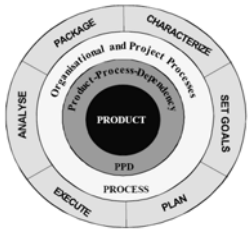
Organizational learning: A theory of action perspective
Chris Argyris & Donald A. Schön, 1978

Learning from Process and Experience

Process models on 3 levels: "Global level", organization level and project level



PROFES



The PROFES improvement cycle

Phases and steps of the PROFES improvement methodology

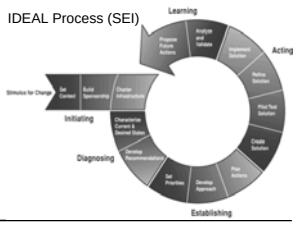
UNIVERSITETET I OSLO

Continuous Improvement (a cyclic process)

General procedure:

- Initiate an improvement program
- Define goals
- Analyze the current situation
- Plan changes
- Implement changes
- Check the effect of the changes
- Adjust goals, analyze the current situation, ... etc, etc

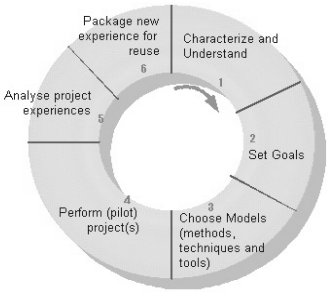
IDEAL Process (SEI)



UNIVERSITETET I OSLO

Process Improvement Processes - QIP

QIP – Quality Improvement Paradigm, Victor Basili, University of Maryland



UNIVERSITETET I OSLO

1. Characterize

- Characterize the organization. Identify characteristics and locate the organization in the improvement circle.
- Identify particular problematic areas. Are the customers satisfied? Are the collaborators satisfied? Are there any signs of weakness in the process?
- Look ahead. Do you think the characteristics will change during the next period? Are we headed for new markets that will demand other requirements than we are familiar with?
- Exploit knowledge that resulted from the last run of the Quality Improvement Paradigm (if not the first)
- NOTE: This step should be carried out very carefully in the first run of the QIP. It is recommended to make semi-structured interviews (e.g., via CMM assessments) of various roles in order to identify pros and cons. Analyze, if available, data from past projects.

UNIVERSITETET I OSLO

2. Set Goals

- Set goals for the next period. What goals are defined with respect to market and strategy? What superior improvement goals are identified?
- The goals should be broken down into concrete improvement goals. Define the goals so that it is possible to verify them later on.
- Use the knowledge about what is important for the organization- in particular the experience/knowledge from the last run of the QIP (unless it is the first run).

UNIVERSITETET I OSLO

3. Choose Models

- Decide what efforts are necessary in order to reach the goals
- Prioritize
- Make explicit what new methods/models/knowledge will be tested
- If necessary, choose the tool support to be used
- Make a risk analysis in order to make the right decision about aiming widely or only implement the effort in the pilot project
- Complete the improvement plan

4. Perform Projects

- This involves the testing of the new models/methods and tools in a certain number of projects (or pilot-projects)
- For each (pilot) project
  - Characterize the project
  - Agree what process model to use in the project
  - Make a measurement plan
  - Coordinate the process, the measurement plan and the project plan
  - Collect data, analyze and provide feedback during the project. **Learning meetings!**
  - At the end of the project, analyze the results and document the experiences

5. Analyze

- Collect the results (analysis) from all the (pilot) projects
- Conduct the analysis on behalf of the organization
- Document the results

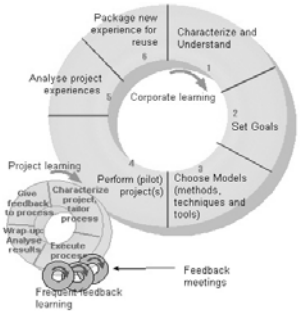
6. Package Experience

- This is about how to make a documented good process available for future projects.
- Document (package) the results from the projects so that the experiences may be reused (i.e., are understandable and transferable).
- For instance, this may involve updating a process model and linking quality models to it.

UNIVERSITETET I OSLO

QIP – Focus on Learning

QIP – Also included: project level and feedback meetings. Learning in three levels!



UNIVERSITETET I OSLO

Learning Meetings

- Conduct learning meetings (or reflection meetings)
  - Schedule the meetings at the end of main activities (milestones, iterations, etc.)
  - Gather the project group at the occurrence of particular events ("de-briefings")
- In the learning meetings discuss the following:
  - What was supposed to happen (the plan)?
  - What happened actually?
  - Why were there deviations?
  - What did we learn?
  - How can we prevent this to happen again?

UNIVERSITETET I OSLO

Evaluation Meetings

- Use Post Mortem Analysis (PMA) as described in "*Postmortem reviews: purpose and approaches in software engineering*" [file *post-mortems.pdf* in reading materials P08]
- Evaluation meeting:
  - What can be considered to be successful parts of the project and should be repeated?
  - What went OK, but could have been done better?
  - Which faults were made that should be avoided in the future?
  - Identify the causes to both good and bad experiences

Bring the Experience Back to the Process

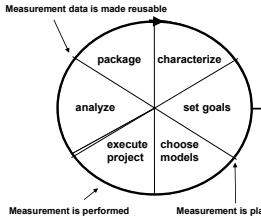
- Do this closely together with the PMA
- If you have a well-defined standard process, the experiences should lead to changes.
- Discuss suggestions for how to change the standard process with the organization itself – then carry out the changes that were decided!
- Institutionalizing: do not underestimate the job of changing the way other people work!

"It is easier to dissolve an organization than to change it"

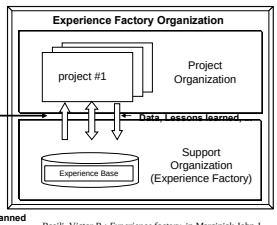
Tom Peters

QIP and Learning Organizations

QIP
(Basili, Rombach, 1988)

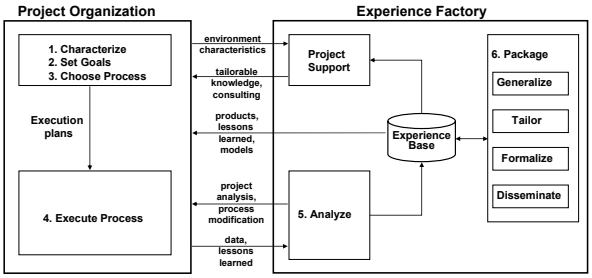


Experience Factory Organization
(Basili, Rombach, 1988)



Basili, Victor R.: *Experience factory*, in Marciniak John J. (ed.) Encyclopedia of Software Engineering, New York: John Wiley & Sons Inc. pp. 469 - 476, 1994

The Experience Factory Organization

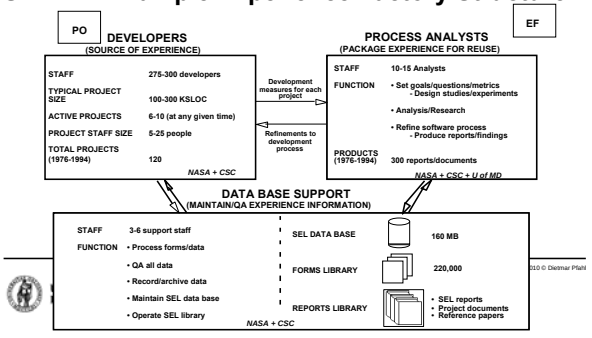


The Experience Factory Organization – A Different Paradigm

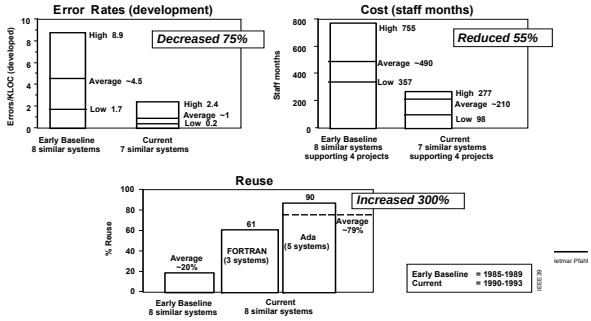
Project Organization Problem Solving	Experience Factory Experience Packaging
Decomposition of a problem into simpler ones	Unification of different solutions and re-definition of the problem
Instantiation	Generalization, Formalization
Design/Implementation process	Analysis/Synthesis process
Validation and Verification	Experimentation
Product Delivery within Schedule and Cost	**Experience / Recommendations Delivery to Project**

UNIVERSITETET I OSLO

SEL: An Example Experience Factory Structure



Using Baselines to Show Improvement: 1987 vs. 1991



Using Baselines to Show Improvement: 1987 vs. 1991 vs. 1995

Continuous Improvement in the SEL

Decreased **Development Defect rates** by
75% (87 - 91) **37%** (91 - 95)
Reduced **Cost** by
55% (87 - 91) **42%** (91 - 95)
Improved **Reuse** by
300% (87 - 91) **8%** (91 - 95)
Increased **Functionality** five-fold (76 - 92)

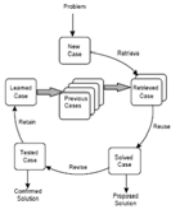
CSC officially assessed as CMM level 5 and ISO certified (1998), starting with SEL organizational elements and activities

Fraunhofer Center for Experimental Software Engineering - 1998

CeBASE Center for Empirically-based Software Engineering - 2000

UNIVERSITETET I OSLO

How to find suitable Models in the EB?



The CBR cycle (adapted from Aamodt & Plaza 1994)

Case-Based Reasoning (CBR):

- Involves (a) matching the current problem against ones that have already been encountered in the past and (b) adapting the solutions of the past problems to the current context.
- It can be represented as a cyclical process that is divided into the four following sub-processes as depicted in the Figure (Aamodt & Plaza 1994):
  - retrieve the most similar cases from the case base
  - reuse the case to solve the problem
  - revise the proposed solution – if necessary
  - retain the solution for future problem solving

UNIVERSITETET I OSLO

Effort Estimation Model – Example (1)

Case-Based Reasoning (CBR) Example: $Effort = f(System\_Size)$

Attributes	New Case	Retrieved Case 1	Retrieved Case 2
Project Category	Real Time	Real Time	Simulator
Language	C++	C++	C++
Team Size	10	10	9
System Size	150	200	175
Effort	**?**	1000	950
Similarity		**90%**	**~50%**

Effort Estimation Model – Example (2)

Case-Based Reasoning (CBR) Example:

Attributes	New Case	Retrieved Case 1	Retrieved Case 2
Project Category	Real Time	Real Time	Simulator
Language	C++	C++	C++
Team Size	10	10	9
System Size	150	200	175
Effort	**?**	1000	950
Similarity		**90%**	**~50%**

Possibilities to predict effort:
- adapted effort based on 1 project
- average effort of 2 projects
- weighted average effort of 2 projects

Possible adaptation rule:

$$Predicted\_Effort = \frac{1}{2}\left(\frac{150}{200} * 1000 + \frac{150}{175} * 950\right) \approx 782$$

Effort Estimation Model – Example (3)

Case-Based Reasoning (CBR) Example:

Attributes	New Case	Retrieved Case 1	Retrieved Case 2
Project Category	Real Time	Real Time	Simulator
Language	C++	C++	C++
Team Size	10	10	9
System Size	150	200	175
Effort	**?**	1000	950
Similarity		**90%**	**~50%**

Possibilities to predict effort:
- adapted effort based on 1 project
- average effort of 2 projects
- weighted average effort of 2 projects

Possible adaptation rule:

$$Predicted\_Effort = \frac{150}{200} * 1000 * \frac{9}{14} + \frac{150}{175} * 950 * \frac{5}{14} \approx 773$$

Effort Estimation Model – Example (3)

Case-Based Reasoning (CBR) Example:

Distance Measure (Euclidean Distance) → Similarity = 1 – Distance

$$distance(P_i, P_j) = \sqrt{\frac{\sum_{k=1}^{n} \delta(P_{ik}, P_{jk})}{n}} \qquad \delta(P_{ik}, P_{jk}) = \begin{cases} \left(\frac{|P_{ik} - P_{jk}|}{max_k - min_k} \right)^2 & \text{if } k \text{ continuous} \\ 0, \text{if } k \text{ categorical AND } P_{ik} = P_{jk} \\ 1, \text{if } k \text{ categorical AND } P_{ik} \neq P_{jk} \end{cases}$$

P_k	$P_{new,k}$	$P_{1,k}$	$\delta(P_{new,k}, P_{1,k})$	
Project Category	Real Time	Real Time	0	
Language	C++	C++	0	
Team Size	10	10	0	
System Size	150	200	0.04	$=(50/250)^2$

⇒ ***distance***$(P_{new}, P_1) = 0.1$

UNIVERSITETET I OSLO

Effort Estimation Model – Example (4)

Case-Based Reasoning (CBR) Example:

Distance Measure (Euclidean Distance) → Similarity = 1 – Distance

$$distance(P_i, P_j) = \sqrt{\frac{\sum_{k=1}^{n} \delta(P_{ik}, P_{jk})}{n}} \qquad \delta(P_{ik}, P_{jk}) = \begin{cases} \left(\frac{|P_{ik} - P_{jk}|}{max_k - min_k} \right)^2 & \text{if } k \text{ continuous} \\ 0, \text{if } k \text{ categorical AND } P_{ik} = P_{jk} \\ 1, \text{if } k \text{ categorical AND } P_{ik} \neq P_{jk} \end{cases}$$

P_k	$P_{new,k}$	$P_{2,k}$	$\delta(P_{new,k}, P_{2,k})$	
Project Category	Real Time	Simulator	1	
Language	C++	C++	0	
Team Size	10	9	0.01	$=(1/10)^2$
System Size	150	200	0.01	$=(25/250)^2$

⇒ ***distance***$(P_{new}, P_2) \approx 0.5$

UNIVERSITETET I OSLO

EF Discussion

- What are potential obstacles for a functioning EF?
- What could be done to overcome the obstacles?

UNIVERSITETET I OSLO

Research done by SINTEF/NTNU

- Cf. Literature Part 08 – Guest Lecture slides
- Focus: Knowledge Management in mid-sized companies
- 3 KM tools investigated in case studies:
  - Electronic project Guide: Description of common processes and work roles in project work, with templates, checklists and examples.
  - Well of Experience: A knowledge repository ("collective yellow stickers"). Contains everything from bugfixes to telephone numbers.
  - Skills Manager: An overview of the skill levels of all employees on about 250 different skills that are considered important for the company.