BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Software Metrics (PA1407)

Lecture 3

Goal Question Metric (GQM) framework

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Goal based measurement

- In goal based measurement the primary focus is not “What measurements should I use?” but rather “What do I need to improve?”
- It is not about having many numbers but rather having access to exactly the information you need to understand, to manage, and to improve your processes, products and business.
- Instead of attempting to develop general-purpose measures, one has to describe an adaptable process that users can use to identify and define measures that provide insights into their own development problems.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Goal based measurement framework

- **Determining what to measure**
  - Identifying and classifying entities to be examined
  - Determining relevant goals

- **Determining how to measure**
  - Identifying and assigning relevant metrics

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Identifying entities

- **Process**
  - A collection of software related activities usually associated with some timescale.
  - Different SE processes: development, maintenance, testing, reuse, configuration and management process etc.
- **Product**
  - Any artifacts, deliverables or documents that result from a process activity.
- **Resource**
  - Entities required by a process activity.

Attribute types

- **Internal attributes**
  - Attributes that can be measured entirely in terms of the process, product or resource itself.
  - They can be measured by examining the product, process or resource on its own, separate from its behavior.
    - Product size is an internal attribute.
- **External attributes**
  - Attributes that can be measured only with respect to how the process, product or resource relates to its environment.
  - Here, the behavior of the process, product or resource is important, rather than the entity itself.
    - Product quality is an external attribute.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Attributes

- Managers and users often want to be able to measure and predict external attributes.
  - E.g. cost-effectiveness of an activity, usability of the developed product etc.
- However, external attributes are more difficult to measure than internal ones, and are measured quite late in the development process.
- At times, internal attributes are used to make inferences/judgments about externals. This can be misleading.
- One goal of measurement research is to identify relationships among internal and external attributes.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

The importance of internal attributes

- Software engineering methods give structure to software products
- The claim is that this structure makes the products easier to understand, analyze and test.
- The structure involves two aspects
  - The development process, since certain products need to be produced at certain stages, and
  - The products themselves, since the products must conform to certain structural principles.
- Product structure is usually characterized by levels of internal attributes such as modularity, coupling, or cohesiveness

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

The importance of internal attributes

- Quality assurance
  - Developers use internal attributes to predict external ones to monitor and control the products during development
    - E.g. relationship between internal design attributes and failures
- Validating composite measures
  - Quality is frequently used to describe internal design or code attributes.
  - However, quality is multidimensional: it does not reflect a single aspect of a particular product.
- Resources
  - Staff, tools (both software and hardware used) and methods.
  - Staff
    - Effort, cost, code or other outputs, productivity
    - What else can be measured?

Components of software measurement: Entities and attributes examples

ENTITIES	ATTRIBUTES	
Products	*Internal*	*External*
Specifications	size, reuse, modularity, redundancy, functionality, syntactic correctness, ...	comprehensibility, maintainability, ...
Designs	size, reuse, modularity, coupling, cohesiveness, functionality, ...	quality, complexity, maintainability, ...
Code	size, reuse, modularity, coupling, functionality, algorithmic complexity, control-flow structuredness, ...	reliability, usability, maintainability, ...
Test data	size, coverage level, ...	quality, ...
...	...	...
Processes		
Constructing specification	time, effort, number of requirements changes, ...	quality, cost, stability, ...
Detailed design	time, effort, number of specification faults found, ...	cost, cost-effectiveness, ...
Testing	time, effort, number of coding faults found, ...	cost, cost-effectiveness, stability, ...
...	...	...
Resources		
Personnel	age, price, ...	productivity, experience, intelligence, ...
Teams	size, communication level, structuredness, ...	productivity, quality, ...
Software	price, size, ...	usability, reliability, ...
Hardware	price, speed, memory size, ...	reliability, ...
Offices	size, temperature, light, ...	comfort, quality, ...
...	...	... ,

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

GQM approach

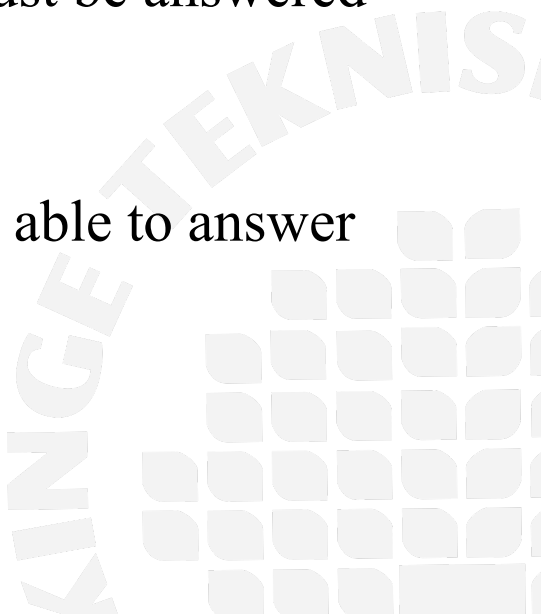
- Goal-Question-Metric (GQM) approach to process metrics provides a framework for deriving measures from organization or business goals.
- Basili's GQM process

Phase	Description
1	Developing a set of corporate, division and project goals
2	Generating questions that define those goals as completely as possible in a quantifiable way
3	Specifying measures (metrics) needed to be collected to answer those questions and to track process and product conformance to the goals
4	Developing mechanisms for data collection
5	Collecting, validating and analyzing the data in real time to provide feedback to projects for corrective action, to assess conformance to the goals and make recommendations for future improvements

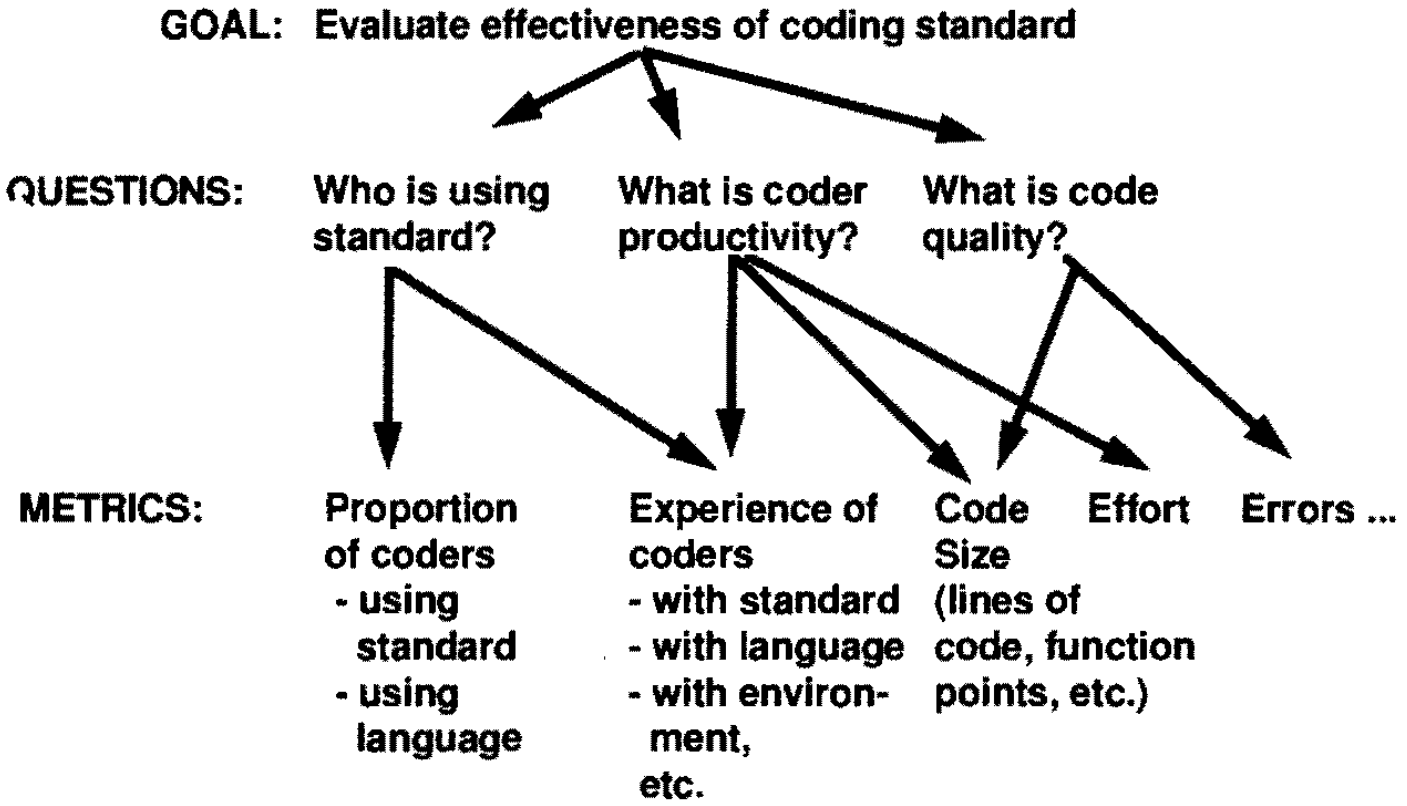
BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

GQM approach

- Goal: List major goals of development or maintenance project.
- Question: Derive from each goal the questions that must be answered to determine if the goals are being met.
- Metrics: Decide what must be measured in order to be able to answer the questions adequately.



GQM tree example



Example 1: AT&T goals, questions and metrics

Goal	Questions	Metrics
Plan	How much does the inspection process cost?	Average effort per KLOC Percentage of reinspections
	How much calendar time does the inspection process take?	Average effort per KLOC Total KLOC inspected
Monitor and control	What is the quality of the inspected software?	Average faults detected per KLOC Average inspection rate Average preparation rate
	To what degree did the staff conform to the procedures?	Average inspection rate Average preparation rate Average lines of code inspected Percentage of reinspections
	What is the status of the inspection process?	Total KLOC inspected
Improve	How effective is the inspection process?	Defect removal efficiency Average faults detected per KLOC Average inspection rate Average preparation rate Average lines of code inspected
	What is the productivity of the inspection process?	Average effort per fault detected Average inspection rate Average preparation rate Average lines of code inspected

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Example 2

- Goal: Improve staff performance
- Question 1
  - At present, how productive is our team?
- Entity
  - Team
- Attributes
  - Team velocity
  - Quality of the produced code or delivered modules
  - Team's expertise in programming languages

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Example 2 (cont'd)

- Goal: Improve staff performance
- Question 1
  - How healthy is the work environment?
- Entity
  - Work environment
- Attributes
  - Workspace (room/desk size, ergonomics)
  - Extra-curricular activities
  - Work time to break time ratio
  - Incentives/bonuses/increments

Templates for goal definition

- **Purpose:** To (characterize, evaluate, predict, motivate, etc.) the (process, product, model, metric, etc.) in order to (understand, assess, manage, engineer, learn, improve, etc.) it.

  Example: To *evaluate* the *maintenance process* in order to *improve* it.

- **Perspective:** Examine the (cost, effectiveness, correctness, defects, changes, product measures, etc.) from the viewpoint of the (developer, manager, customer, etc.)

  Example: Examine the *cost* from the viewpoint of the *manager*.

- **Environment:** The environment consists of the following: process factors, people factors, problem factors, methods, tools, constraints, etc.

  Example: The maintenance staff are poorly motivated programmers who have limited access to tools.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Goal template example

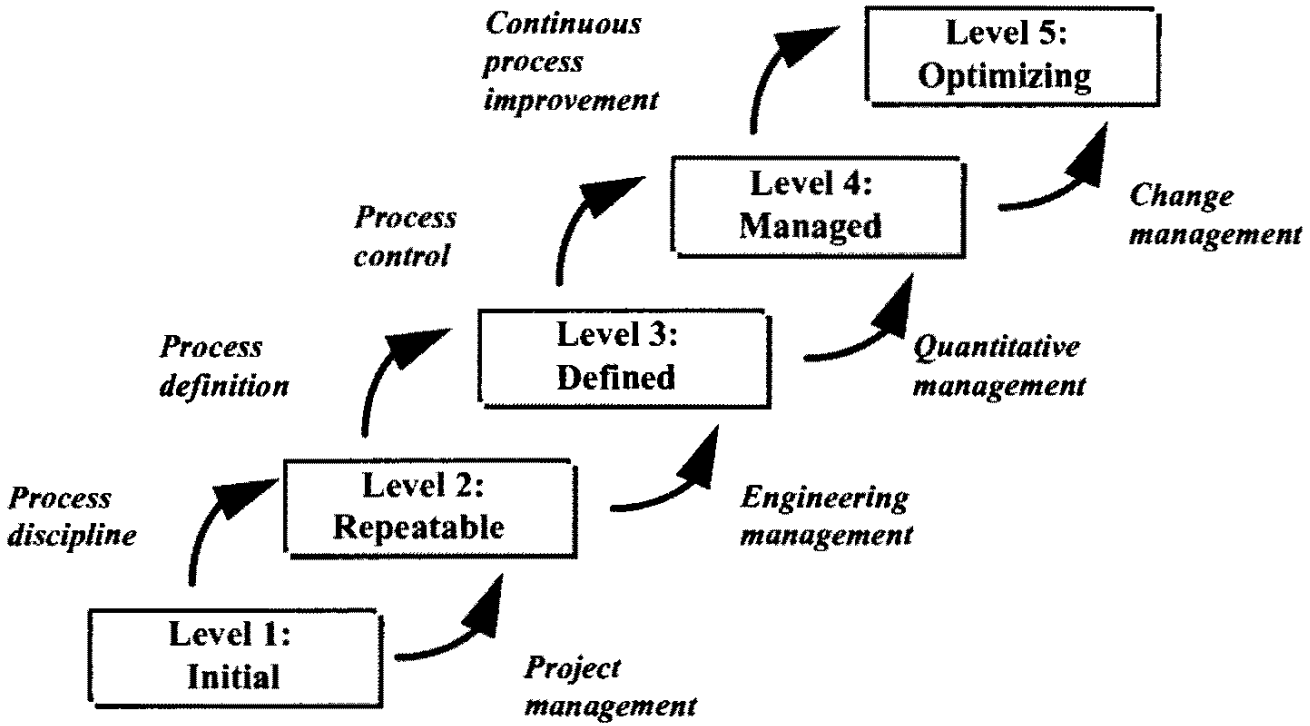
- **Purpose**
  - Evaluate the impact of various CASE tools on the productivity of the development team.
- **Perspective**
  - Examine the effectiveness of using various CASE tools to help in the development of our product from the point of view of the developers and testers.
- **Environment and Constraints**
  - Payroll applications programming in C++
  - 100 software developers with 5 or more years experience in C++
  - Do not maintain a reusable module database.
  - Examine projects completed and sold during last five years.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Measurement and process improvement

- Measurement offers visibility into the ways in which the processes, products, resources, methods, and technologies of software development relate to one another.
- Measurement is useful for
  - Understanding
  - Establishing a baseline
  - Assessing and predicting
- Measurement is useful only in the larger context of assessment and improvement
  - Some organizations are more mature than others, and have defined processes offering more visibility as compared to others.

SEI's levels of process maturity



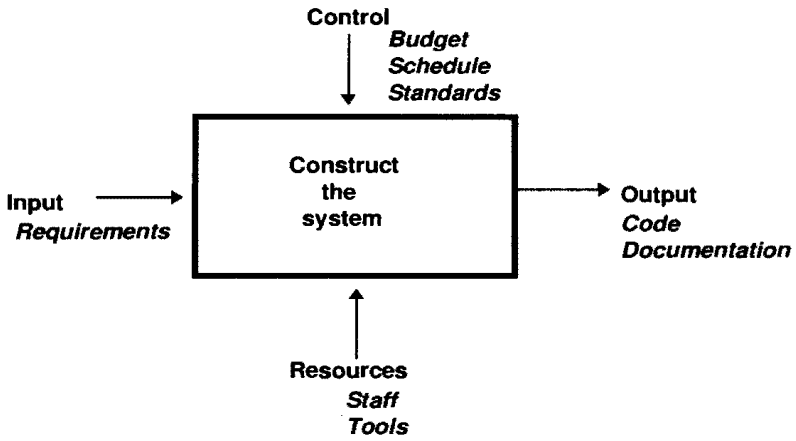
BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

SEI's levels of process maturity

- Level 1: Ad hoc
  - Ill-defined inputs; transition from inputs to outputs undefined and uncontrolled; Similar projects may vary widely in their productivity and quality.
  - Visibility is nil and comprehensive measurement difficult
  - Baseline measurements are needed

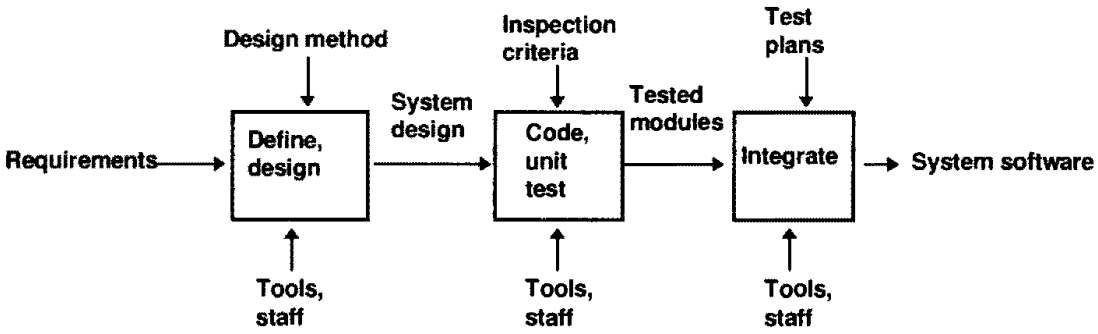
SEI's levels of process maturity

- Level 2: Repeatable
  - Identified inputs, outputs, constraints and the resources
  - Process is repeatable in the same sense that a subroutine is repeatable:
    - i.e. proper inputs produce proper outputs, but there is no visibility into how the outputs are produced.



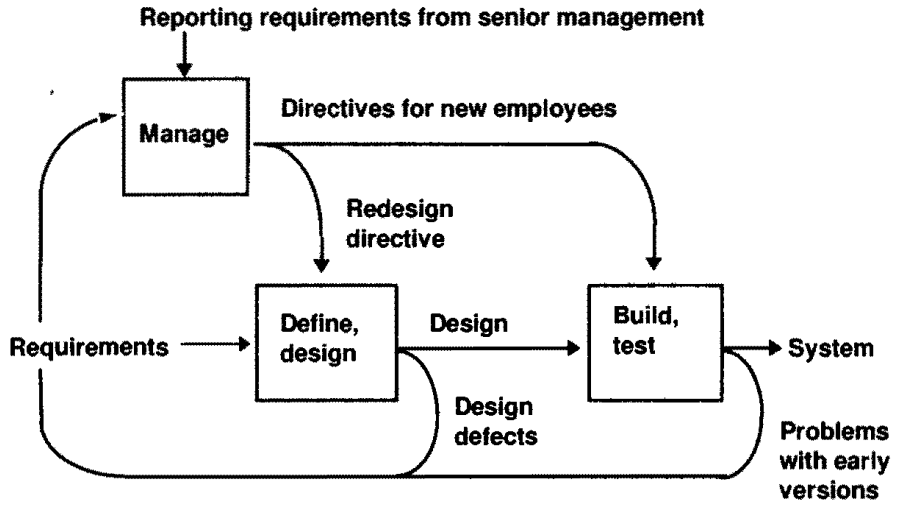
SEI's levels of process maturity

- Level 3: Defined
  - A defined process provides visibility into the "construct the system" box.
    - At level 3, intermediate activities are defined, and their inputs and outputs are known and understood.
    - The input to and output from the intermediate activities can be examined, measured, and assessed, since these intermediate products are well-defined and visible.



SEI's levels of process maturity

- Level 4: Managed
  - Adds management oversight to a defined process
  - Use feedback from early project activities to set priorities for current activities and later project activities.



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

SEI's levels of process maturity

- Level 5: Optimizing
  - An optimizing process is the ultimate level of process maturity
  - Measures from activities are used to improve the process, possibly by removing and adding process activities, and changing the process structure dynamically in response to measurement feedback.
  - The process change can affect the organization and project as well as the process

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

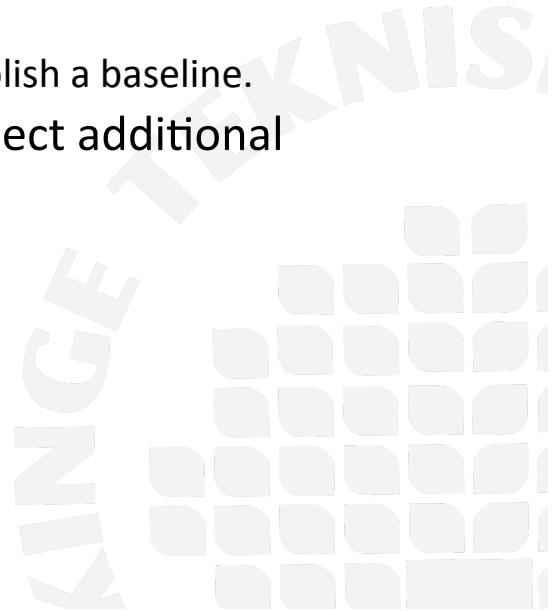
Combining GQM with Process Maturity

- Since process maturity suggests that one can only measure what is visible, the incorporation of process maturity with GQM helps identify what measures will be most useful to the organisation at that particular moment in time.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

GQM with Process Maturity Example (1/2)

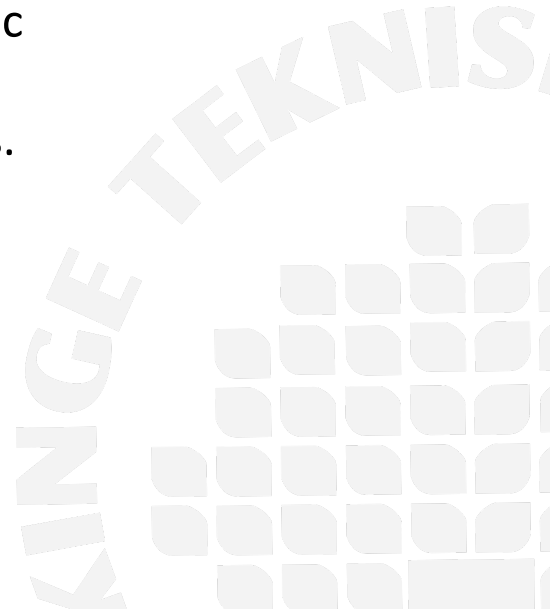
- You want to answer the question "Is the set of requirements maintainable?"
  - At level I, the project is likely to have ill-defined requirements.
    - Measuring requirements' characteristics is difficult at level 1.
    - You may choose to count requirements and changes in them to establish a baseline.
  - At level 2, the requirements are well-defined and you can collect additional information
    - Like type of each requirement (interface or performance etc.)



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

GQM/Process Maturity Example (2/2)

- At level 3, you can collect a richer type of measurement:
  - Measuring the traceability of each requirement to the corresponding design, code, and test components, and noting the effects of each change on the related components.
- Thus, the goal and question analysis is the same, but the metric recommendations vary with maturity.
- The more mature your process, the richer your measurements.



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

References

- Lecture notes are prepared from following sources:
  - T1: Software Metrics - A Rigorous & Practical Approach, 2nd edition, Authors: N. E. Fenton, S. L. Pfleeger, Publishers: International Thomson Computer Press, 1996, ISBN: 1-85032-275-9.
- Basili's GQM references:
  - V.R. Basili and D. Weiss (1984), *A Methodology for Collecting Valid Software Engineering Data*, IEEE Trans. Software Engineering, vol. 10, pp.728-738.
  - V.R. Basili and H.D. Rombach (1988), *The Tame Project: Towards Improvement-Oriented Software Environments*, IEEE Trans. Software Engineering, vol.14, pp.758-773.