FACHBEREICH INFORMATIK

Lecture

Empirical Model Building and Methods
(Empirische Modellbildung und Methoden)

Prof. Dr. Dr. h.c. Dieter Rombach
Dr. Andreas Jedlitschka

SS 2015

Chapter 2 – Measurement process

AG·SE

© Rombach, Jedlitschka, 2015

Chapter objectives

At the end of this chapter, you should …

- … understand the relevance of measurement within SE context
  - Why? What? How?
- … know the GQM measurement framework and process
  - Be able to define a GQM goal using the GQM goal template and abstraction sheet
- … know the concept of resource, process, and product models
- … have an overview of relevant measures in SE context
  - Be able to propose metrics for software products, process and resources

Outline

Measurement basics

2.1.1 Motivation

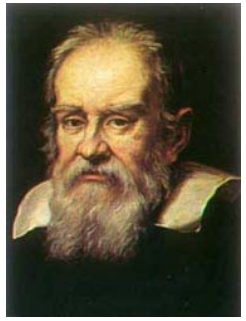
2.1.2 Definitions

2.1.3 Types of Measures

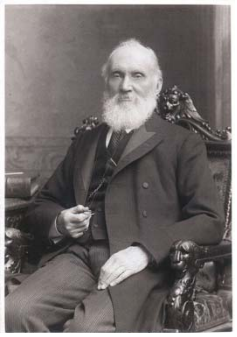
2.1.4 Scales

Motivation

"Measure what is measurable, and make measurable what is not so"



Galileo Galilei



Lord Kelvin

"I often say that *when you can* measure what you are speaking about, and express it in numbers, you *know* something about it; but when you cannot express it in numbers your knowledge is of meager and unsatisfactory kind"

One objective of science is to find ways to measure attributes of entities we are interested in.

Measurement makes concepts more visible and thus more understandable and controllable.

Definitions

Measurement is the process by which numbers or symbols are mapped to attributes of entities in the real world in such a way as to describe them according to clearly defined rules.

Process

effort → person days spent from start to end → 10

Product

size → No. Lines of Code → 700

Resource

experience → >10 projects → high

(ISO 9126, 2001)

Definitions

(SE) Entity

- Each process, activity, product, and resource within software development is an entity.

Attribute

- Any property of an entity.
- Direct and indirect attributes
  - Direct attributes relate directly to the objective of measurement.
  - Indirect attributes relate to the context necessary to interpret the direct attributes.

(SE) Measurement

- Assignment of values (numbers or symbols) to attributes of an entity in a systematic way
- A function from the empirical world of SE (SE entity) into the formal world of mathematical objects

Types of Measures

Basic and aggregate measures

- Basic (or manifest) measures can be obtained directly from the entity
  - mass, volume, length, duration, number of defects, effort
- Aggregate (or latent) measures involve other attributes/measures
  - density, defect density, experience

$$density = \frac{mass}{volume} \qquad defect\_density = \frac{\#defects}{size}$$

Direct and indirect measures

- Direct measures represent and define an attribute and are valid by definition
  - e.g., mean time to software failure for the quality attribute reliability
- Indirect measures are used to predict/characterize e.g., a quality factor
  - e.g., module complexity for predicting the quality attribute reliability

Types of Measures

Objective and subjective measures

- Objective: no judgment in measurement value, it depends only on the entity
  - e.g., LOC, FP, delivery date
- Subjective measures: reflect judgment of measurer, depend on entity and viewpoint
  - e.g., personnel skill, experience
  - e.g., perception on understandability, easy to use, ….
- A measure is classified as objective or subjective depending on its specification, e.g.:
  - personnel performance (objective) → accuracy, completion, attendance, #projects…
  - personnel performance (subjective) → initiative, teamwork, collaboration, professionalism, .

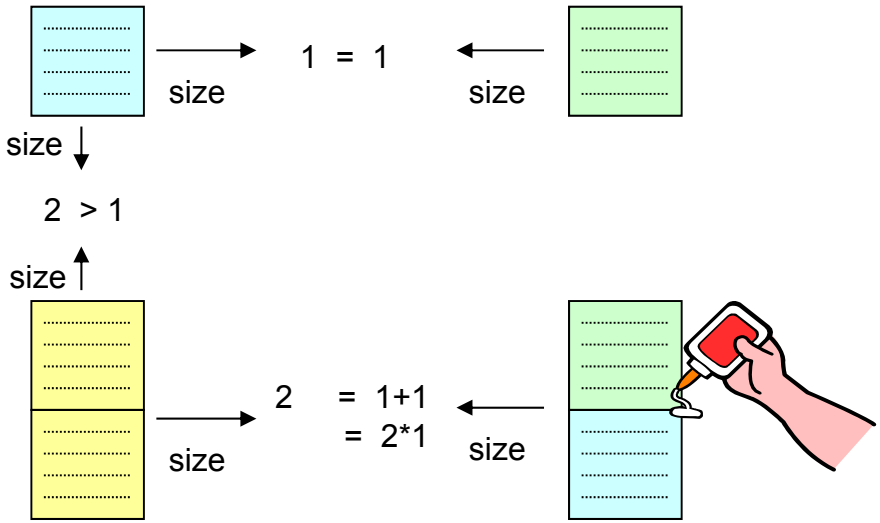
Definitions (Examples)

Entity	Attribute	Measure
Product	size	#pages, LOC, Function Points (FP), #classes
	reliability	#major defects, probability of failure, mean time between failures
	maintainability	change effort
Process	effectiveness	percentage of defects found, found defect density
	efficiency	defects found per effort
	effort	#hours/days/months per person
	cost	EUR spent
	duration	calendar time between start and end
Resource	experience	{high, medium, low}, #projects, #years
	productivity	LOC/effort; FP/effort

Measurement Scales

Measurements scales refer to ways in which variables or numbers are defined and categorized

- Each scale has certain properties
- These properties determines the appropriateness for use of mathematical operations
- Mathematical operations should make sense in real world



Measurement Scales

Scale	Description	Allowed Operation	Example
Nominal	Numbers or symbols are assigned to represent class memberships. Thus, the scale consists of different classes and no ordering among these classes exist	counting =, ≠	defect classification:= {assignment, checking, algorithm, function, interface, timing} programming language
Ordinal	Used for rank orderings. Size of the interval between different ranks cannot be determined	counting =, ≠, <, >	experience = {high, medium, low} code complexity = {trivial, simple, moderate, complex}
Interval	Intervals between any two consecutive integers represent equal amounts of measured attribute. Thus, order is preserved as well as differences so that we can understand the size of the jump from one class to another	counting =, ≠, <, > +, −	calendar time temperature in degree Celsius
Ratio	Interval scale with an absolute zero point that represents total lack of the measured attribute. Scale must start at zero and increase at equal intervals known as units.	counting =, ≠, <, > +, −, *, /	time intervals, length (LOC), effort cyclomatic complexity temperature in degree Kelvin

Measurement process

2.2.1 Introduction
2.2.2 GQM Framework
2.2.3 GQM Goal definition
2.2.4 GQM Measurement process

Motivation

What are the objectives of measurement activities?

- Increase objectivity
  - reduce subjective influence through the person who measures
- Increase reliability
  - Same result in case of same situation, and different persons who measure
- standardization
  - There is a scale for the metrics and one to compare with
- comparability
  - It should be possible to relate the metric with others
- usefulness
  - The measurement follows practical needs
- validity
  - Results can be traced back to the measurement goal
- economic
  - Cost / Benefit relationship

Introduction

Measurement is not just the collection of data/metrics

- calendar time, total effort, number of defects found in inspections, cyclomatic complexity, machine time, severity of failures, lines of code/staff month, number of failures during system test,…

Measurement must be driven by goals and models for

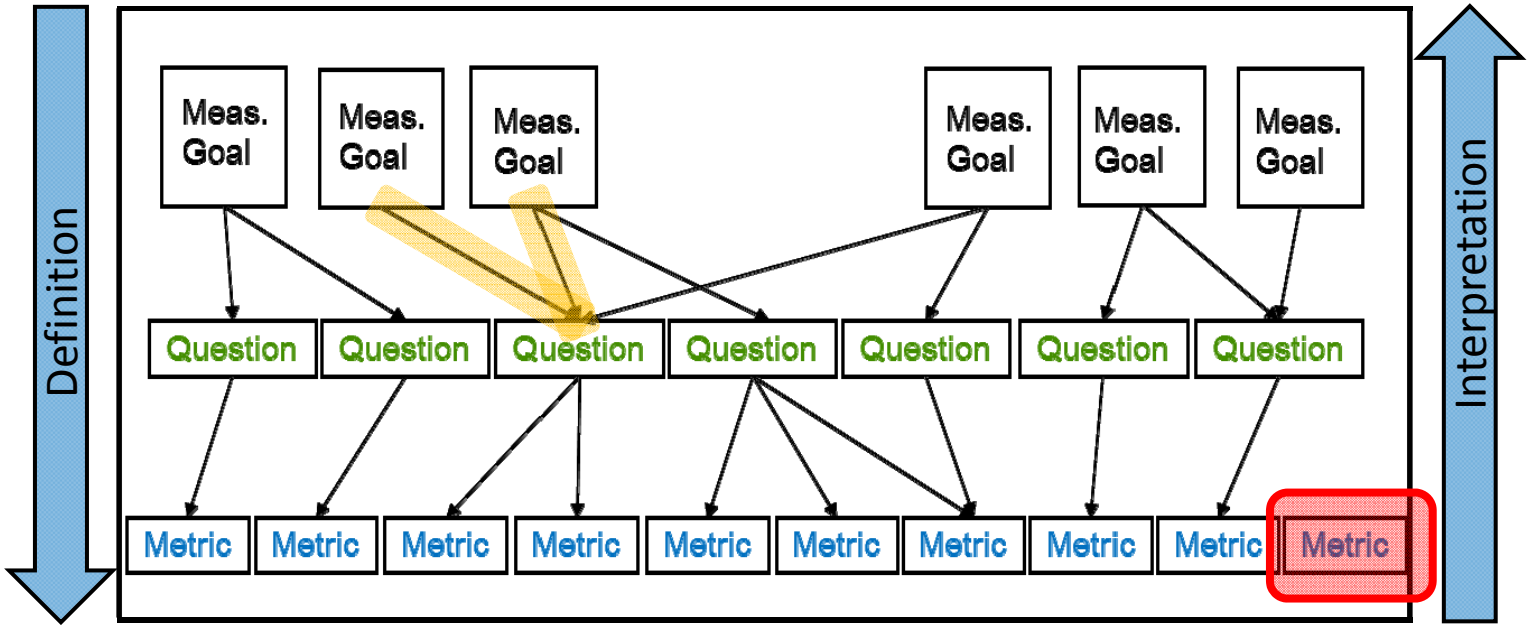
- Processes, products and resources
- Projects
- Organizations

Measurement frameworks help you to define the appropriate metrics and interpret the results

- Goal/Question/Metric Paradigm (GQM) NASA, Basili, Caldiera, Rombach ’94
- Practical Software Measurement (PSM) sponsored by DoD (MCGARRY John et al. 2001)
- Quality Function Deployment (QFD), 1966 in Japan by Yoji Akao,

Goal orientation

- Internal and external stakeholders have their own goals
- Well defined goals enable business success
  - Questions guide and focus metric selection and interpretation
  - Each metric supports goals

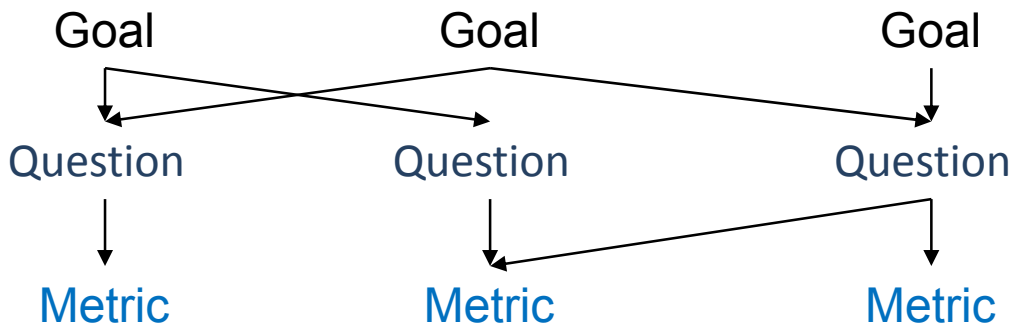


GQM Framework

Goal/Question/Metric Paradigm (V. Basili et al.)

- Provides a mechanism for defining and interpreting operational, measurable goals
- It uses four parameters:
  - a model of an **object of study**, e.g., a process, product, or any other experience-based model
  - a model of **one or more foci**, e.g., models that view the object of study for particular characteristics (e.g., quality)
  - a **point of view**, e.g., the perspective of the stakeholder needing the information
  - a **purpose**, e.g., how the results will be used
- to generate a GQM model
  - relative to a particular **context** (environment)

GQM Framework



A goal links two models:

- a model of the object of interest and a model of the focus
- to allow developing an integrated GQM model

Example

- Goal: Analyze the final product to characterize it with respect to the various defect classes from the point of view of the organization
- Question: What is the defect distribution by phase of entry?
- Metric: # requirements defects, number of design defects, ...

GQM Framework

Business Goal

- Understand problem areas in the software business

Measurement Goal

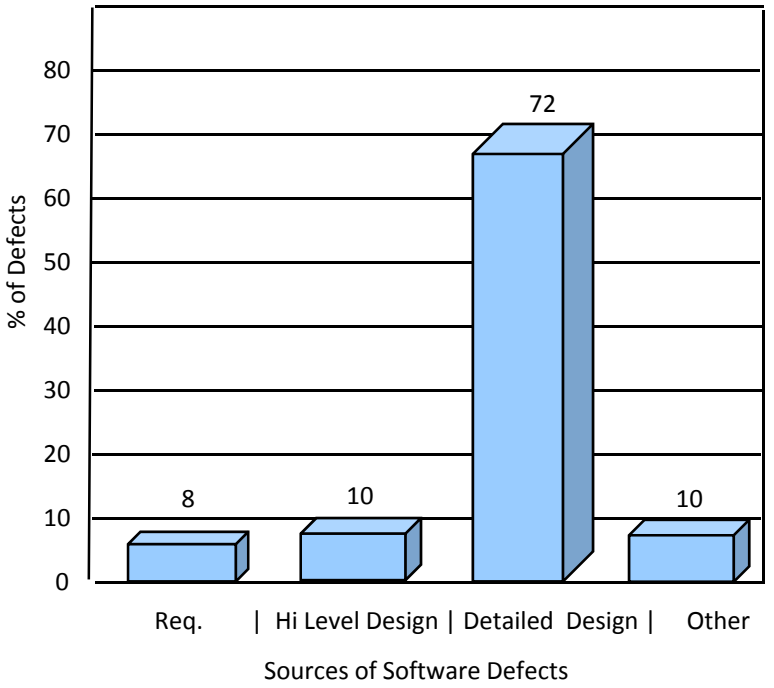
- Analyze the final product to characterize it with respect to the various defect classes from the point of view of the organization

Question

- What is the defect distribution by type of defect?

Metrics

- Number of requirements defects,
- Number of design defects, ...



Defects := Errors/Faults/Failures (IEEE)

GQM Goal Definition

Goals may be defined for any object, for a variety of reasons, with respect to various models of "anything", from various points of view, relative to a particular environment.

Goal Template

- Analyze some (object of study: process, product, other experience model)
- for the purpose (purpose: characterize, evaluate, predict, improve)
- with respect to (focus: cost, correctness, defect removal, changes, reliability, usability, time, ...)
- from the point of view of (stakeholder: user, customer, manager, developer, corporation,...)
- in the following (context: problem factors, people factors, resource factors, process factors,...)

GQM parameters

GQM example: PBR experiment

Goal definition

Analyze PBR and Ad-hoc reading techniques
for the purpose of their evaluation
with respect to their effectiveness
from the viewpoint of the researcher
in the context of the Software Engineering (SE) lecture at
the University of Kaiserslautern (UKL).

Effectiveness

- For individuals
- For teams
- detecting different defect classes
- detecting differing defects

(Ciolkowski et al., 1997)

GQM example: PBR experiment

Refinement of goals

Goal 1
Analyze PBR and Ad-hoc reading techniques for the purpose of their evaluation with respect to their **effectiveness for individuals** ...

Goal 2
Analyze PBR and Ad-hoc reading techniques for the purpose of their evaluation with respect to their **effectiveness for teams** ...

Goal 3
Analyze PBR perspectives for the purpose of their evaluation with respect to their **effectiveness to detect different defect classes** ...

Goal 4
Analyze PBR and Ad-hoc reading techniques for the purpose of their evaluation with respect to **detect differing defects**

(Ciolkowski et al., 1997)

GQM example: PBR experiment

Goal 1

Analyze PBR and Ad-hoc reading techniques for the purpose of their evaluation with respect to their **effectiveness for individuals** ...

Quality Focus – effectiveness for individuals

- Q1. Which is the mean defect detection rate of the individuals (DDR)?
  - DDR = number of defects found by individual / total number of defects in the document
    - M1.1 NRDEFFOUND = Number of defects found by each subject
    - M1.2 NRDEFDOC = Total number of defects in each document

(Ciolkowski et al., 1997)

GQM example: PBR experiment

Goal 1

Analyze PBR and Ad-hoc **reading techniques** for the purpose of their evaluation with respect to their effectiveness for individuals ...

(Ciolkowski et al., 1997)

Process – reading techniques

- Q2. Which document was used?
  - M2. Document used by subject
- Q3. Which technique was applied?
  - M3. Reading technique used by subject
- Q4. Which perspective was applied?
  - M4. Perspective used by subject
- Q5. How much experience did the subjects have with inspections?
  - M5. Experience of subject
- Q6. How well did the subjects know the English language?
  - M6. English knowledge of subject
- Q7. How well did the subjects understand the technique?
  - M7. Understanding of technique of subject
- Q8. How well were the subjects motivated?
  - M8. Motivation of subject
- Q9. How well did the subjects follow the reading scenario?
  - M9. Following of guidelines of subject
- Q10. Did the subjects have enough time to complete their work?
  - M10.1. Enough time for subject
  - M10.2. Percentage of document finished

© Rombach, Jedlitschka, 2015

GQM abstraction sheet[1] (Example[2])

Object	Purpose	Quality Focus	View Point	Context
PBR and Ad-Hoc reading	Evaluation	**Effectiveness for individuals**	Researcher	SE-lecture at TUKL

Quality Focus: Effectiveness for individuals

- DDR = number of defects found by individual / total number of defects in the document
- M1.1 NRDEFFOUND = Number of defects found by each subject
- M1.2 NRDEFDOC = Total number of defects in each document

Variation Factors:

- M2. Document used by subject (DOC)
- M5. Experience of subject (EXP)

(Baseline) Hypotheses:

H1: Individuals applying PBR perform better than individuals using Ad-hoc reading with respect to their DDR.

H2: Individuals applying each PBR perspective respectively perform better than individuals applying Ad-hoc reading (no perspective) with respect to their DDR.

Impact of variation factors:

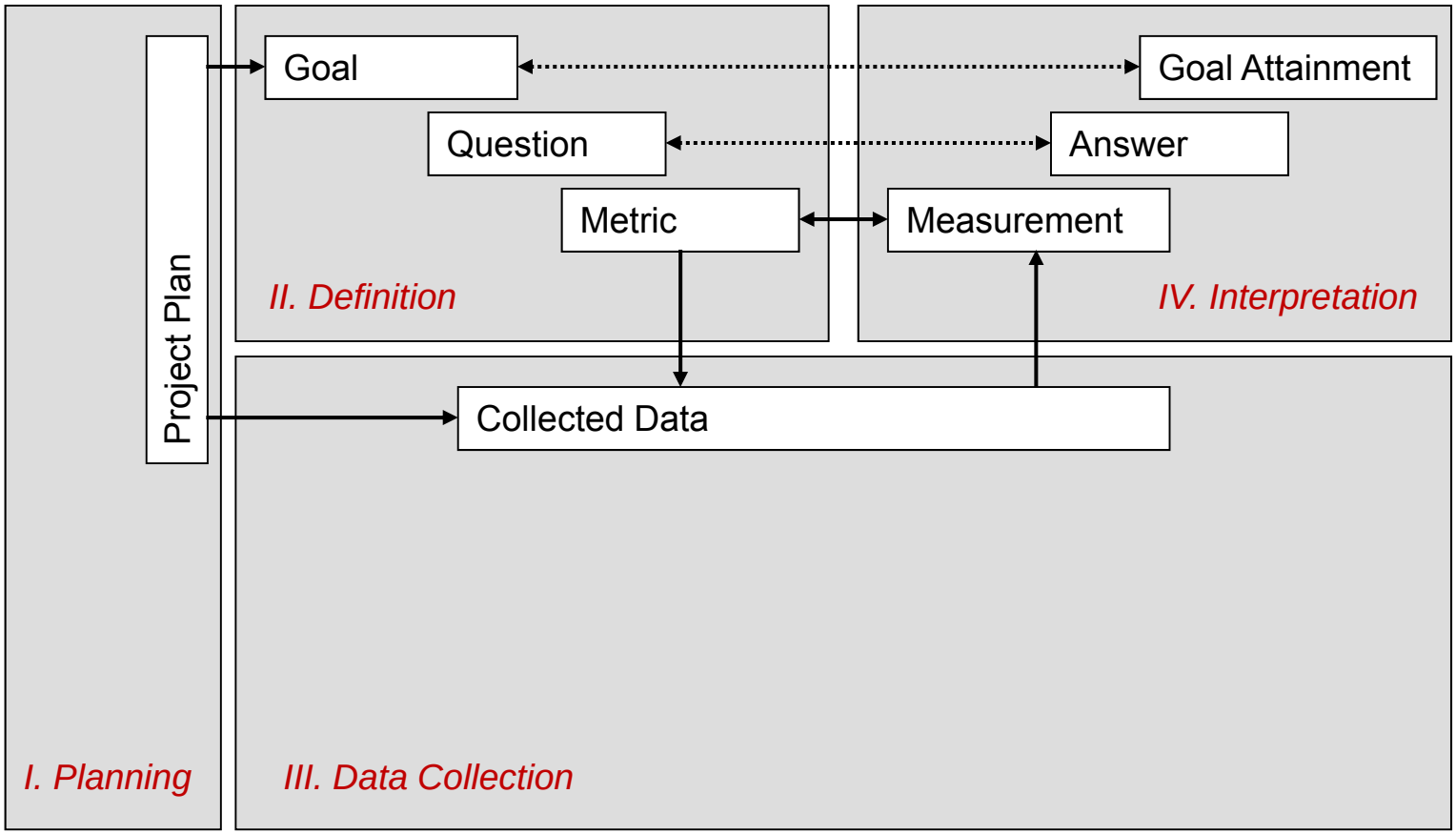
H3: There is no difference between individuals reading the different DOC with respect to their mean defect detection rate.

H4: There is no difference between subjects applying PBR on one DOC and those applying Ad-hoc on the other document and vice versa with respect to their mean defect detection rate. (Interaction effect)

H5: The EXP of the subjects has no influence on their mean defect detection rate.

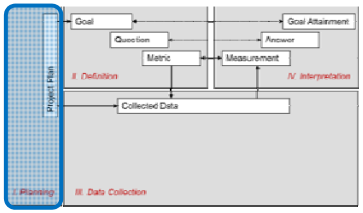
[1] Solingen und Berghout, 1999
[2] Ciolkowski et al., 1997

GQM measurement process



(corrected from R. van Solingen and E. Berghout, 1999)

A2. GQM Measurement approach



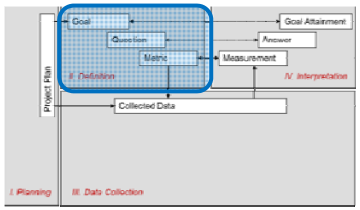
I. Planning

- Activities
  - Establish GQM Team
  - Select an improvement area
  - Select application project and establish project team
  - Create project plan
  - Training and promotion

- GQM team should:
  - Be independent from project teams (cf. experience factory)
  - Have “no interest” in measurement results
  - Possess sufficient background knowledge on the objects of measurement
  - Keep in mind that the project team ‘owns’ the improvement program, because a project team is most knowledgeable on a project.
  - Be improvement oriented, which includes that it is willing to improve itself too
  - Be enthusiastic, in order to motivate the project team

A2. GQM Measurement approach

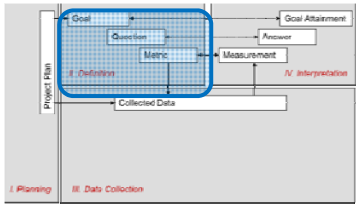
II. Definition



- Activities
  1. Define measurement goals
  2. Review or produce software process models
  3. Conduct GQM interviews
  4. Define questions and hypotheses
  5. Review questions and hypotheses
  6. Define metrics
  7. Check metrics on consistency and completeness
  8. Produce GQM plan
  9. Produce measurement plan
  10. Produce analysis plan
  11. Review plans

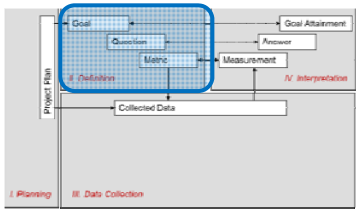
A2. GQM Measurement approach

II. Definition



- Questions
  1. What are the strategic goals of your organization?
  2. What forces have an impact on your strategic goals?
  3. How can you improve your performance?
  4. What are your major concerns (problems)?
  5. What are your improvement goals?
  6. How can you reach your improvement goals?
  7. What are possible measurement goals, and what are their priorities?

A2. GQM Measurement approach



II. Definition

- Abstraction sheets helps to define goals by characterizing
  - Quality focus
    - what are possible metrics to measure an object of a goal, according to the project members?
  - Baseline hypothesis
    - what is project members' current knowledge and expectations w.r.t. these metrics?
  - Variation factors
    - which (environmental) factors does a project member expect to be of influence on the metrics?
  - Impact on baseline hypothesis
    - how could these variation factors influence the actual measurements?
    - what kind of dependencies between metrics and influencing factors are assumed?

A2. GQM Measurement approach

II. Definition (Example of abstraction sheet, Goal 2)

Object	Purpose	Quality Focus	View Point	Context
PBR and Ad-Hoc reading	Evaluation	Effectiveness for teams	Researcher	SE)lecture at TUKL

Quality Focus: Effectiveness for teams

- DDR = number of defects found by team / number of defects in the document. A team is a group of at most three individuals.
  - M1.1: Number of defects found per team (NRDEFTEAM)
  - M1.2: Number of defects in document (NRDEFDOC)

Variation Factors:
…

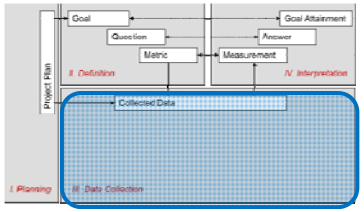
(Baseline) Hypotheses:
H6: Nominal PBR teams detect more defects than nominal Ad-hoc teams, i.e. they have a higher mean defect detection rate.

H7: (New) The defect detection rate of nominal teams is lower than the defect detection rate of the real teams.

Impact of variation factors:
…

A2. GQM Measurement approach

III. Data Collection

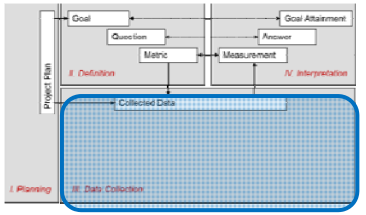


- For each metric, data collection procedures should specify
  - Which person should collect the data?
  - When should the data be collected?
  - How can the data be collected most efficiently and effectively?
  - To whom should the collected data be delivered?

- Data can be collected via
  - Manual forms, electronic forms, e-mail forms, spreadsheet forms
  - Data base driven, web page driven
  - Automated tools

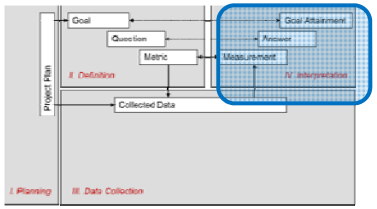
A2. GQM Measurement approach

III. Data Collection



- Activities
  - Data Collection Start up
  - Trial period to test forms
  - Kick-off Session
  - Checking forms for correctness and completeness
  - Storing form data
  - Building a measurement support system
    - Spreadsheets, statistical tools, database applications,
    - Presentation tools
  - Aggregate the data in analysis sheets
    - Raw data, e.g., data, fault id
    - Processed data, e.g., date, # of faults
    - Graphs and tables (offers the characterization data)

A2. GQM Measurement approach



IV. Interpretation

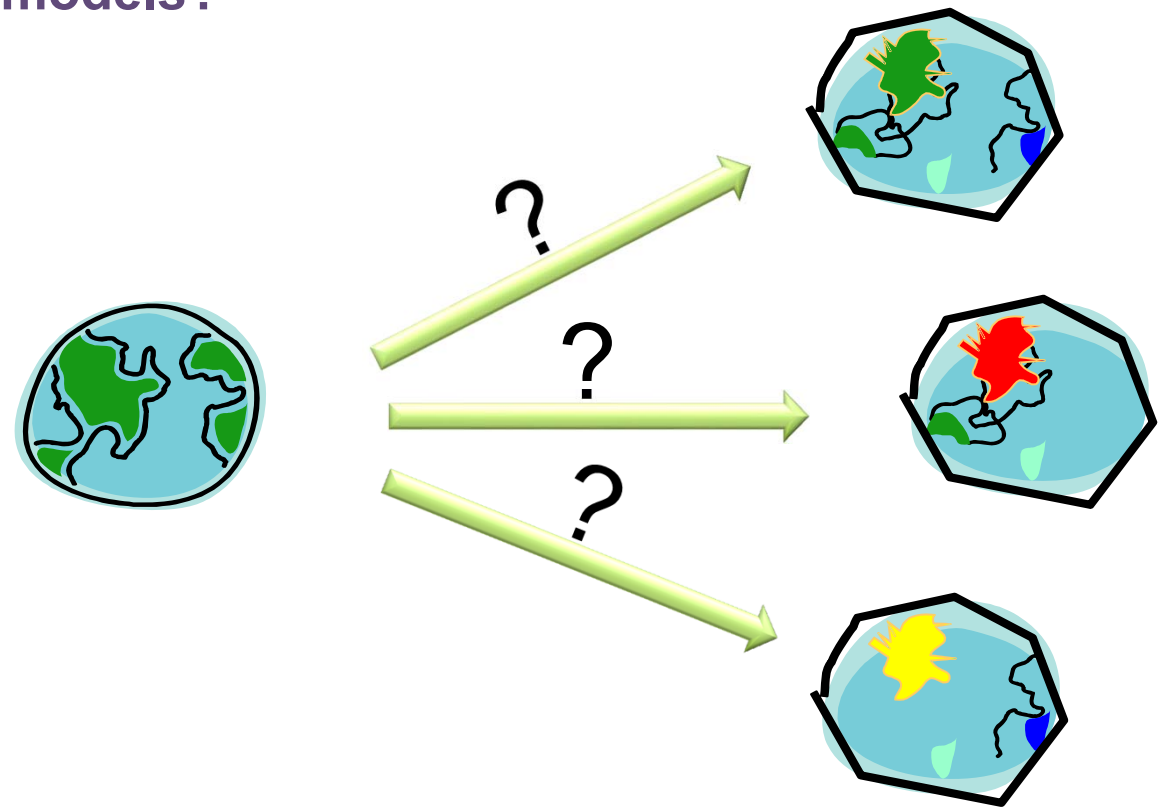
- Activities
  - Data analysis and interpretation according to GQM model
  - Planning, performing and reporting feedback sessions

- Feedback Session
  - Prepare feedback session
    - Slides and handouts on answers to the questions and goals
  - Organize and hold session
    - Run by a project team member
    - Evaluate action points from earlier sessions
    - Interpret measurement data
    - Draw conclusions
    - Take notes
  - Report Measurement Results (Feedback session report)

Models

2.3.1 Introduction
2.3.2 Resource models
2.3.3 Process models
2.3.4 Product models

Why models?



What is the purpose of models?

The selection of the most appropriate model is crucial for interpreting, understanding, predicting the reality.

Why do we need models in SE?

- Measurement and assessment of software often fails because of the inability to choose appropriate models and measures.

- The choice of appropriate measures depends on
  - Project goals
  - Models
  - Context of interest

What should be modelled?

- Experience shows that the following aspects have to be modelled in a measurable manner:
  - Object of interest
    - Resources
    - Process
    - Product
  - Relevant environmental context
  - (Quality) aspects of interest.

- Models should be selected considering the
  - Purpose
  - Perspective

Introduction

What kind of models?

- Descriptive
- Explanatory
- Predictive
- Prescriptive

Characteristics of a good model

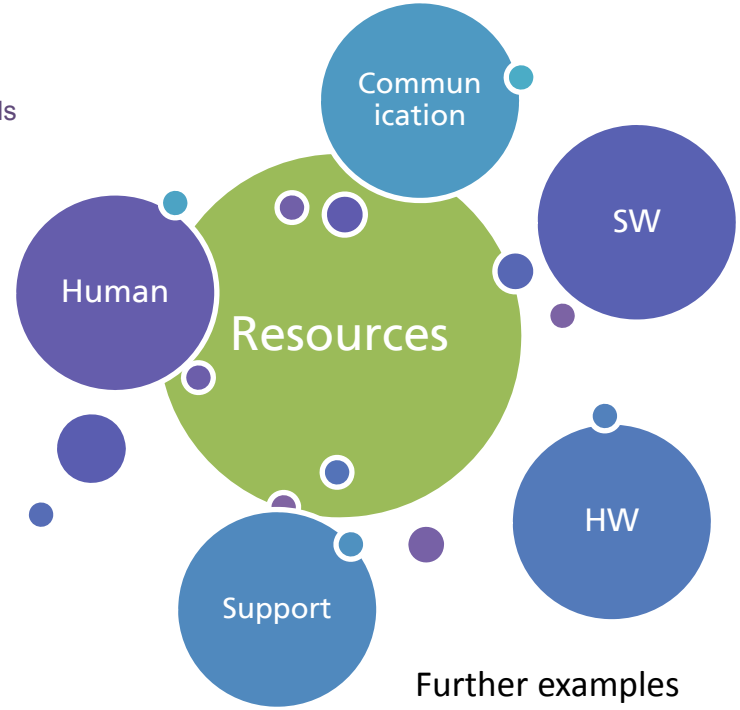
- Explain the reality, e.g., our behavior and the development environment
- Model parameters
  - are calculable from known data
  - describe and can be calibrated for a specific environment
  - include redundancy checks and risk analysis factors

SE Models

- Resource, Process, Product

Resource models

What?



Further examples
- Supplies
- Materials
- Facility
- ….

Metrics

- Human effort: staff-hours, weeks, months, years
- Calendar time: calendar hours, days, weeks, months, date to date
- Computer time: calendar time, execution time

They may be associated with

- Processes: phases, tasks, activities
- Products: documents, software components
- Others project characteristics: from date to date

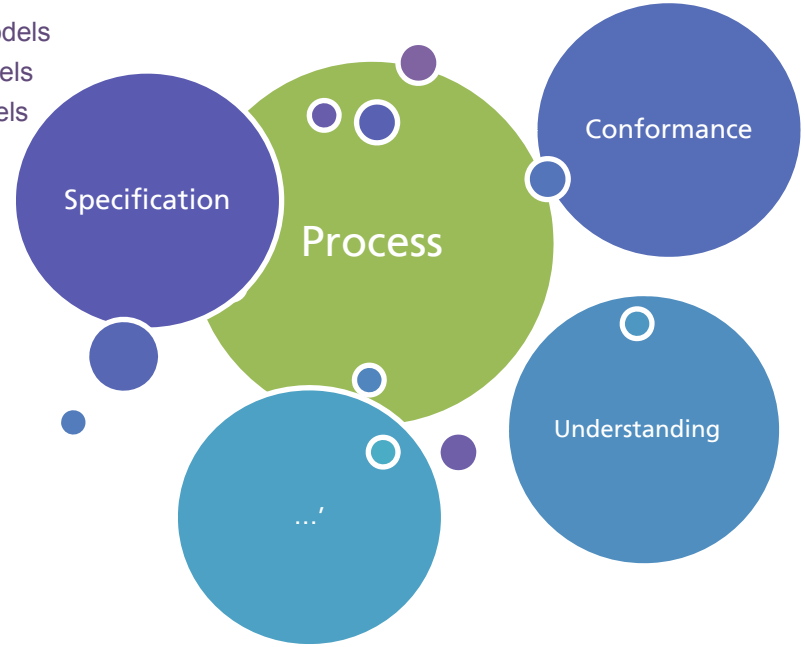
Resource models

Why?

- Initial prediction of, e.g., effort (cost), staffing, computer use, …
- Description of development patterns
  - What is going on?
  - How do different parameters change the pattern?
  - What can we learn about future developments?
- Prediction of the next phase from the current phase
- Support the evaluation of techniques, methodology and engineering

Process models

What?



Metrics

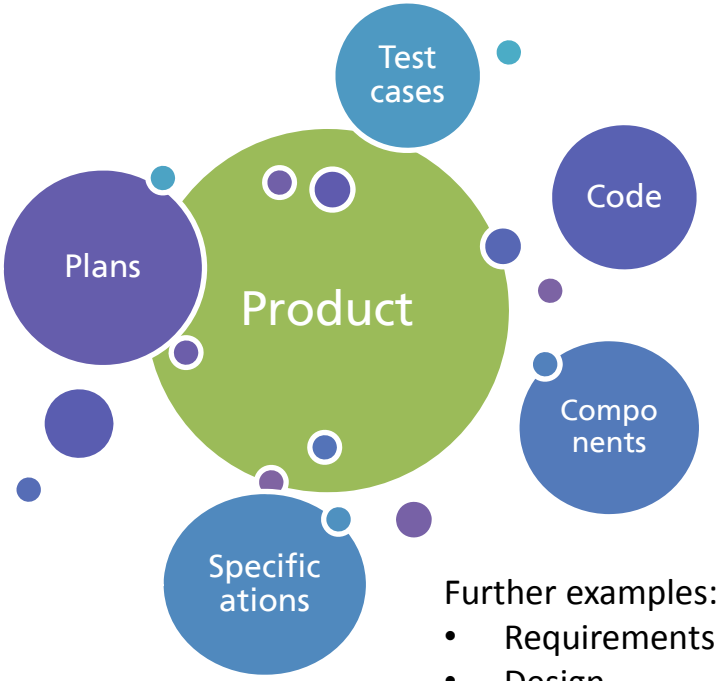
- Effort, Cost
- Process conformance
- Maturity
- Level of documentation

Why?

- Guiding developers
- Evaluation processes, methodologies, techniques, …
- Improvement
- Gaining insight into the product
- Tailoring and evolving the processes over time

Product models

What?



Further examples:

- Requirements
- Design
- Architectural models
- ….

Metrics

- Logical: application domain, function
- Static: size (FP, LOC, #modules), structure
- Dynamic: MTTF, test coverage
- Use/Context related: design method

Why?

- Evaluate the process or the product
- Estimate the cost of quality of the product
- Monitor the stability or quality of the product over time

Definitions (Examples)

Entity	Attribute	Measure
Product	size	#pages, LOC, Function Points (FP), #classes
	reliability	#major defects, probability of failure, mean time between failures
	maintainability	change effort
Process	effectiveness	percentage of defects found, found defect density
	efficiency	defects found per effort
	effort	#hours/days/months per person
	cost	EUR spent
	duration	calendar time between start and end
Resource	experience	{high, medium, low}, #projects, #years
	productivity	LOC/effort; FP/effort

References

- V. Basili, Software Quality Assurance and Measurement: A Worldwide perspective, Applying the Goal/question/Metric Paradigm in the Experience factory, Chapter 2, pp 21-44, International Thomson Computer Press, ITP An International Thomson Publishing Company, 1995
- V. Basili and D. Weiss, A Methodology for Collecting Valid Software Engineering Data, IEEE Transactions on Software Engineering, pp. 728-738, November 1984.
- V. Basili, Models and Metrics for Software Management and Engineering, Proceedings of ASME Century International Computer Technology Conference (invited paper), August 1980.
- J. Bailey and V. Basili, A Meta-Model for Software Development Resource Expenditures, Proceedings of the Fifth International Conference on Software Engineering, pp. 107-116, March 1981.
- M. Ciolkowski, C, Differding, O. Laitenberger, J. Münch, Empirical Investigation of Perspective-based Reading: A Replicated Experiment, ISERN-Report-97-13
- D. Weiss and V. Basili, Evaluating Software Development by Analysis of Changes: The Data from the Software Engineering Laboratory, IEEE Transactions on Software Engineering, pp. 157-168. February 1985.
- V. Basili and H.D. Rombach, The TAME Project: Towards Improvement-Oriented Software Environments, IEEE Transactions on Software Eng., vol. 14, #6, June 1988.
- V. Basili, G. Caldiera, and H.D. Rombach, Goal Question Metric Paradigm, Encyclopedia of Software Engineering - 2 Volume Set, pp. 528-532, Copyright by John Wiley & Sons, Inc., 1994.
- R. van Solingen and E. Berghout, The Goal/Question/Metric Method, McGraw Hill, 1999
- Barry Boehm. Software Engineering Economics. Englewood Cliffs, NJ:Prentice-Hall, 1981. ISBN 0-13-822122-7