ICS 500: Research Methods and Experiment Design in Computing

Lecture

Measurements

The slides are prepared by: Dr. Mahmood Niazi, Dr. Malak Baslyman, Dr. Hamoud Aljamaan & Dr. Mohammad Alshayeb

Lecture Objectives

- ✓ Introduction to measurements
- ✓ Measurements scale types
- ✓ Measurements Classification
- ✓ Measurements in SWE/CS
- ✓ GQM template

"You cannot control what you cannot measure."

--*Tom DeMarco*

“What is not measurable make measurable”

--Galileo Galilei

Measurement in Everyday Life

- Measurement lies at the heart of many systems that govern our lives.
  - Economic, radar systems, medical systems, atmospheric systems, etc.

- Each of us uses it in everyday life.
  - Price, height, size, distance ,speed, etc.

- Measurement helps us to understand our world, interact with our surroundings and improve our lives.

What is Measurement?

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

- An entity is an object (such as a person or a room) or an event (such as a journey or the testing phase of a software project) in the real world.

- An attribute is a feature or property of an entity.
  - Examples: the area or color (of a room), the cost (of a journey), or the elapsed time (of the testing phase).

- Attributes often defined by using numbers or symbols:
  - Height: inches, feet, centimeters
  - Price: dollars, pounds sterling, Euros
  - Clothing: small, medium, large

Objectives of Measurement

- Measurement helps us to *understand* what is happening during development and maintenance.
  - Assess the current situation, establishing baselines that help us to set goals for future behavior.
- Measurement allows us to *control* what is happening on our projects.
  - Using our baselines, goals and understanding of relationships, we predict what is likely to happen and make changes to processes and products that help us to meet our goals.
- Measurement encourages us to *improve* our processes and products.
  - For instance, we may increase the number or type of design reviews we do, based on measures of specification quality and predictions of likely design quality.

Measurement Accuracy

- The accuracy of a measure depends on the measuring **instrument** as well as on the definition of the measurement and measurement **scale**
  - A scale is a way of mapping attributes to measurement values
  - For example, length can be measured accurately as long as the ruler is accurate and used properly.

- Even when the measuring devices are reliable and used properly, there is **margin for error in measuring** the best understood physical attributes.
  - When measuring person height, should we allow shoes to be worn?

Measurement Accuracy *(cont'd)*

- When is a scale acceptable for the purpose to which it is put?
  - Height on cm or inches vs. miles or kilometers?

- Once we obtain measurements, we want to **analyze** them and draw **conclusions** about the entities from which they were derived.
  - What kind of manipulations can we apply to the results of measurement?
  - For example, why is it acceptable to say that Ahmed is twice as tall as Khalid, but not acceptable to say that it is twice as hot today as it was yesterday?

- **Rescale**
  - As a measure of an attribute can be measured in different scales, we sometimes want to transform the measure into another scale.

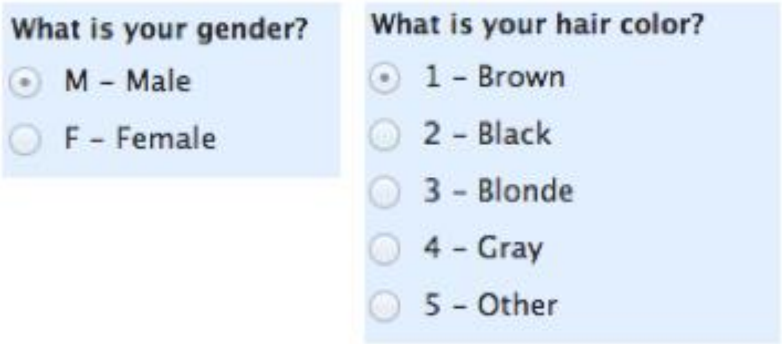
Measurements Scale Types

- Nominal
- Ordinal
- Interval
- Ratio
- Absolute*

* Fenton, N. and J. Bieman, “Software Metrics: A Rigorous & Practical Approach”, 3rd Edition, CRC Press, 2014.

Nominal Measurements Scale

- **Nominal**
  - The nominal scale is the least powerful of the scale types. It only maps the attribute of the entity into a name or symbol.
  - The nominal scale differentiates between items or subjects based only on their names or categories and other qualitative classifications they belong to
  - Nominal scales could simply be called "labels"
  - Examples of a nominal scale are gender, language, ethnicity etc



Ordinal Measurements Scale

- **Ordinal**
  - The ordinal scale ranks the entities after an ordering criterion and is therefore more powerful than the nominal scale.
  - An ordinal variable, is one where the order matters but not the difference between values
  - Examples of ordering criteria are “greater than”, “better than”, and “more complex”.
  - Other examples of an ordinal scale are grades and software complexity.

How do you feel today?

- 1 – Very Unhappy
- 2 – Unhappy
- 3 – OK
- 4 – Happy
- 5 – Very Happy

How satisfied are you with our service?

- 1 – Very Unsatisfied
- 2 – Somewhat Unsatisfied
- 3 – Neutral
- 4 – Somewhat Satisfied
- 5 – Very Satisfied

Interval Measurements Scale

- **Interval**
  - Interval scales are numeric scales in which we know not only the order, but also the exact differences between the values.

  - This scale, orders the values in the same way as the ordinal scale but there is a notion of “relative distance” between two entities (i.e., where the difference between two values is meaningful).

  - Examples of an interval scale are temperature measured in Celsius
    - With an interval scale, you know not only whether different values are bigger or smaller, you also know *how much* bigger or smaller they are. For example, suppose it is 60 degrees Fahrenheit on Monday and 70 degrees on Tuesday. You know not only that it was hotter on Tuesday, you also know that it was 10 degrees hotter.

Ratio Measurements Scale

- **Ratio**
  - If there exists a **meaningful zero value** and the **ratio between two measures is meaningful**, a ratio scale can be used.
  - It is an interval scale with the additional property that its zero position indicates the absence of the quantity being measured.
  - Example
    - Money is measured on a ratio scale because, in addition to having the properties of an interval scale, it has a true zero point: if you have zero money, this implies the absence of money. Since money has a true zero point, it makes sense to say that someone with 50 cents has twice as much money as someone with 25 cents
    - Time is ratio since 0 time is meaningful.
    - Height and Weight

Absolute Measurements Scale

- **Absolute**
  - Most restrictive in terms of admissible transformations.
  - For any two measures, there is only one transformation
  - Four characteristics:
    - Measurement is made simply by counting the number of elements in the entity set.
    - The attribute always take the form "number of occurrences of x in the entity".
    - There is only one possible measurement mapping, namely the actual count.
    - All arithmetic analysis of the resulting count are meaningful.
  - Example
    - The number of failures observed during integration testing can be measured only in one way: by counting the number of failures observed.
    - The number of people working on a software project can be measured only in one way: by counting the number of people.

Measurements Scales Example

Primary Scales of Measurement

Scale					
Nominal	Numbers Assigned to Runners	7	8	3	Finish
Ordinal	Rank Order of Winners	Third place	Second place	First place	Finish
Interval	Performance Rating on a 0 to 10 Scale	8.2	9.1	9.6	
Ratio	Time to Finish, in Seconds	15.2	14.1	13.4	

Summary of Measurement Scales

Scale Type	Characterization	Example (generic)	Example (SE)
Nominal	**Divides the set of objects into categories, with no particular ordering among them**	**Labeling, classification**	**Name of programming language, name of defect type**
Ordinal	**Divides the set of entities into categories that are ordered**	**Preference, ranking, difficulty**	**Ranking of failures (as measure of failure severity)**
Interval	**Comparing the differences between values is meaningful**	**Calendar time, temperature (Fahrenheit, Celsius)**	**Beginning and end date of activities (as measures of time distance)**
Ratio	**There is a meaningful "zero" value, and ratios between values are meaningful**	**Length, weight, time intervals, absolute temperature (Kelvin)**	**Lines of code (as measure of attribute "Program length/size")**

Measurements Scales Operations

Provides:	Nominal	Ordinal	Interval	Ratio
The "order" of values is known		✔	✔	✔
"Counts," aka "Frequency of Distribution"	✔	✔	✔	✔
Mode	✔	✔	✔	✔
Median		✔	✔	✔
Mean			✔	✔
Can quantify the difference between each value			✔	✔
Can add or subtract values			✔	✔
Can multiple and divide values				✔
Has "true zero"				✔

Qualitative and Quantitative Measurements

- Measurement scales are related to **qualitative** and **quantitative** research, and it relates to which **statistics** can be used on the measures.

- Qualitative research is concerned with measurement
  - Nominal and ordinal scales

- Quantitative research treats measurement on
  - Interval and ratio scales.

Measurements Classification

- Measures can also be classified in two ways:
  - (1) if the measure is direct or indirect
  - (2) if the measure is objective or subjective.

Direct and Indirect Measures

- **Direct**
  - A direct measurement of an attribute is directly measurable and does not involve measurements on other attributes.
  - Examples of direct measures are
    - Lines of code
    - Number of defects found in test
- **Indirect**
  - An indirect measurement involves the measurement of other attributes. The indirect measure is derived from the other measures.
  - Examples of indirect measures are
    - Defect density (number of defects divided by the number of lines of code)
    - Programmer's productivity (lines of code divided by the programmer's effort)

Objective and Subjective Measures

- An **objective** measure is a measure where there is no judgment in the measurement value (by subjects) and is therefore only dependent on the object that is being measured.
  - In other words, the performance measure is not subject to personal opinion or interpretation of results, and it is a clear objective measure.
- An objective measure can be measured several times and by different researchers, and the same value can be obtained.
  - Examples of objective measures are
    - Lines of code (LOC)
    - Delivery date.

Objective and Subjective Measures *(cont'd)*

- A **subjective** measure is the opposite of the objective measure.
  - The person making the measurement contributes by making some sort of judgment.
  - The measure depends on both the object and the viewpoint from which they are taken.

- A subjective measure can be different if the object is measured again. A subjective measure is mostly of nominal or ordinal scale type.
  - Subjective measures are always subject to potential bias
  - Examples of subjective measures are
    - Personnel skill
    - Usability
    - Judging performance

Measurements in SWE/CS

- The objects that are of interest in software engineering/ computer science can be divided into three different classes:
  - **Process**
    - The process describes which activities that are needed to produce the software.
  - **Product**
    - The products are the artifacts, deliverables or documents that results from a process activity.
  - **Resources**
    - Resources are the objects, such as personnel, hardware, or software, needed for a process activity.

- The 4Ps: Process, product, project and people.

Measurements in SWE/CS *(cont'd)*

- Many of the best software developers measure characteristics of the software to get some sense of
  - whether the requirements are consistent and complete
  - whether the design is of high quality
  - whether the code is ready to be tested.

- Effective project managers measure attributes of process and product to be able to tell when the software will be ready for delivery and whether the budget will be exceeded.

- Informed customers measure aspects of the final product to determine if it meets the requirements and is of sufficient quality.

- Maintainers must be able to assess the current product to see what should be upgraded and improved.

Measurements in SWE/CS *(cont'd)*

- In each of the classes we also make a distinction between internal and external attributes.
  - An internal attribute is an attribute that can be measured purely in terms of the object.
  - The external attributes can only be measured with respect to how the object relates to other objects

Table 3.1 Examples of measures in software engineering

Class	Examples of objects	Type of attribute	Example of measures
Process	Testing	Internal	Effort
		External	Cost
Product	Code	Internal	Size
		External	Reliability
Resource	Personnel	Internal	Age
		External	Productivity

Relative and Absolute Measures

- **Relative** measurement is measuring something compared to another thing or estimating things proportionally to one another.
  - For example, knowing that a store is a farther distance than the school
  - knowing that one cup has more water than another

- **Absolute** measurement is measuring things in known amounts with Standard Units. Without units, absolute measurements are meaningless.
  - For example: a pencil is 5 inches, or a glass of water is 500 milliliters

Meaningfulness in Measurement

- After making measurements, key question is "can we deduce meaningful statements about the entities being measured?"

- Harder to answer than it first appears – consider the following statements:
  - The number of errors during the integration testing of program *X* was at least 100.
    - Seems to make sense
  - The cost of fixing each error in program *X* is at least 100.
    - Doesn't make sense – the number of errors may be specified without reference to a particular scale, but the cost of fixing an error cannot be.

Meaningfulness in Measurement

- **Meaningful**
  - With the measures of the attribute, we make statements about the object or the relation between different objects. The statements are true even if the measures are rescaled.
    - If we measure the lengths of objects A and B to 1 m and 2 m respectively, we can make the statement that B is twice as long as A.
    - This statement is true even if we rescale the measures to centimeters or inches
- **Meaningless**
  - The opposite
    - We measure the temperature in room A and room B to 10C and 20C, and make the statement that room B is twice as warm as room A.
    - If we rescale the temperatures to the Fahrenheit scale, we get the temperatures 50F and 68F. The statement is no longer true and is therefore meaningless.

Examples

- John is twice as tall as Fred
  - This statement implies that measures are at least on the ratio scale.
  - The statement is meaningful because no matter which measure of height we use (inches, feet, etc.), the truth or falsity of the statement remains consistent.

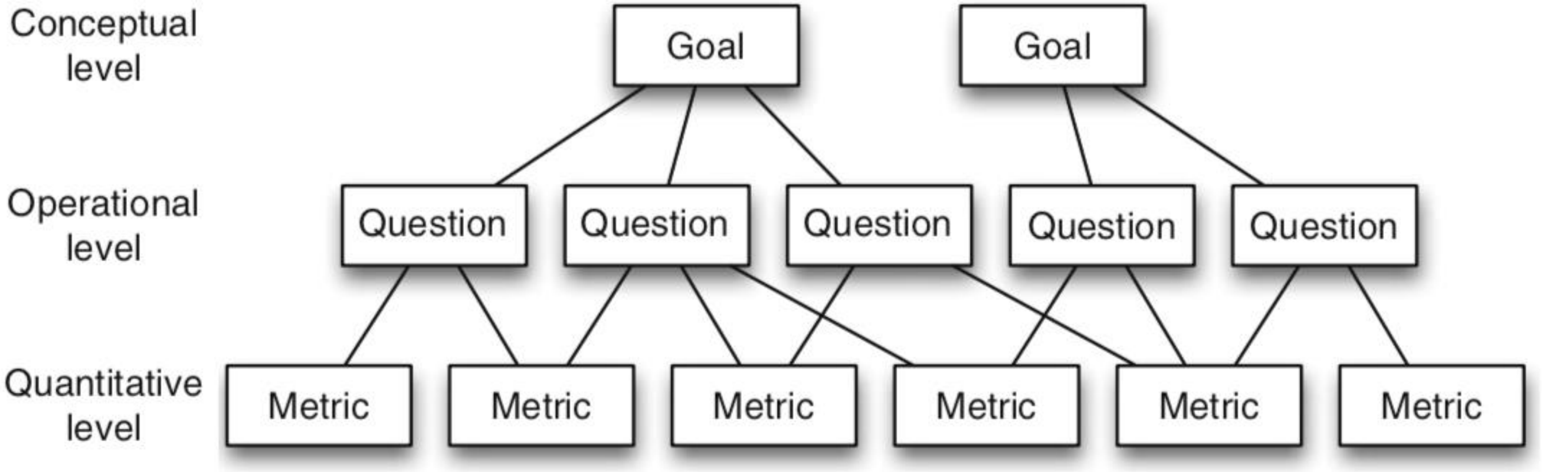
- Temperature in Tokyo today is twice that in London.
  - This statement implies a ratio scale but is not meaningful.
  - If we consider centigrade scale and Tokyo is 40 °C and London is 20 °C, the statement is true.
  - However, considering the Fahrenheit scale, Tokyo is 104 °F and London is 68 °F. The statement is not true.

- *Statement's meaningfulness involving measurement is quite distinct from the notion of the statement's truth.*

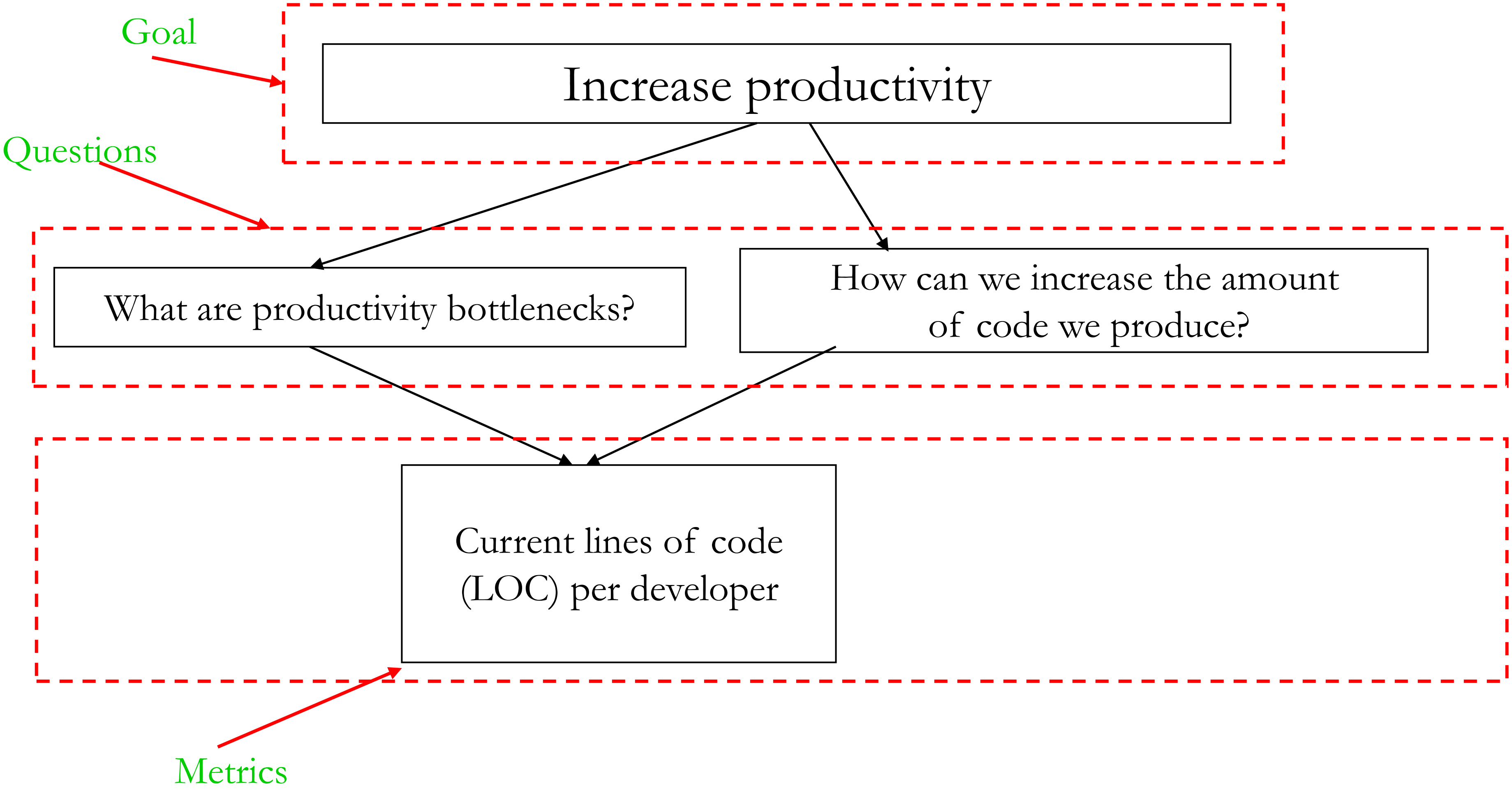
Determining what to measure

- The **Goal-Question-Metric** (GQM) Paradigm approach provides a framework involving three steps:
  1. List the major **goals** of the development or maintenance project.
  2. Derive from each goal the **questions** that must be answered to determine if the goal is being met.
  3. Decide what must be **measured** in order to be able to answer each question.

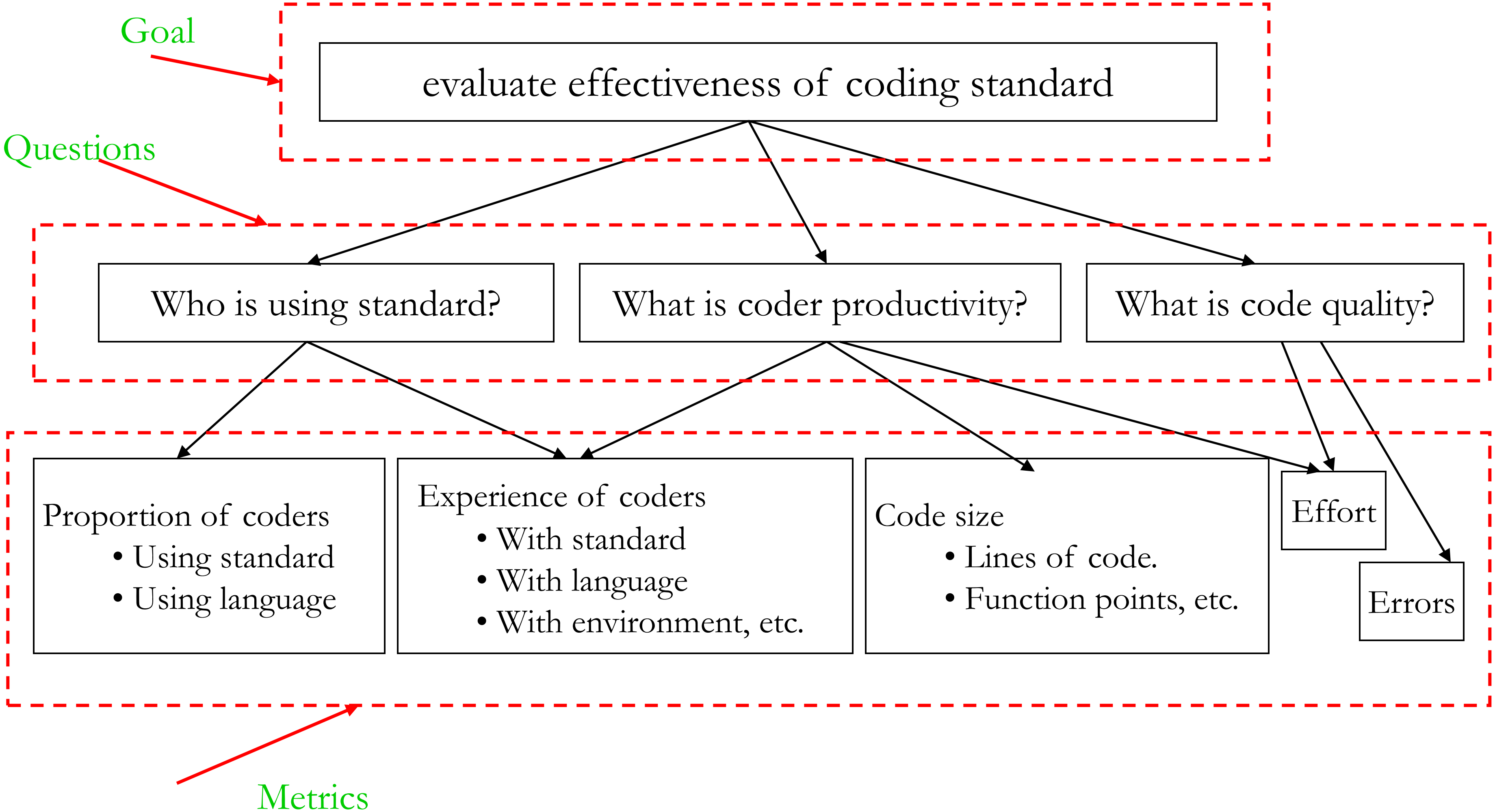
Determining what to measure



GQM example


Goal
Increase productivity
Questions
What are productivity bottlenecks?
How can we increase the amount of code we produce?
Current lines of code (LOC) per developer
Metrics

GQM example 2



GQM tree on assessing inspection process

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

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 (developer, manager, customer, etc.)
  - *Example*: Examine the cost from the view point of the manager.

- Environment/Context: 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.

GQM example 3

- Experiment Scope definition

Application of TreeNet in Predicting Object-Oriented Software Maintainability: A Comparative Study

4.1. Goal

Using GQM template [2] for goal definition, the goal of this empirical study is defined as follows: *Evaluate* TreeNet *for the purpose of* predicting object-oriented software maintainability *with respect to* its prediction accuracy against the five compared models (MARS, MLR, SVR, ANN, and RT) *from the point of view of* researchers and practitioners *in the context of* two object-oriented software datasets.

Measurement validation

- Informally, we say that a measure is “valid” if it accurately characterizes the attribute it claims to measure.

- If we cannot validate the measures, then we cannot be sure that the decisions we make based on those measures will have the effects we expect.

- Validating a software measure is the process of ensuring that the measure is a proper numerical characterization of the claimed attribute by showing that the representation condition is satisfied.

Software Measurement validation

- Measures or measurement systems are used to assess an existing entity by numerically characterizing one or more of its attributes.

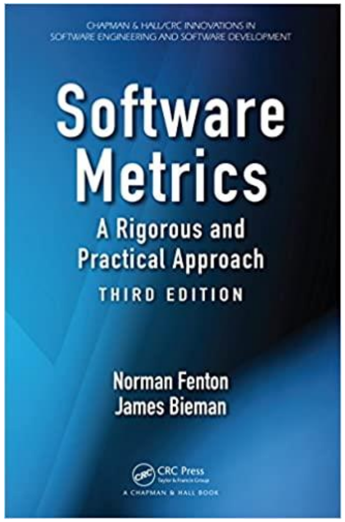
- Prediction systems are used to predict some attribute of a future entity, involving a mathematical model with associated prediction procedures.

Summary

- Introduction to measurements
- Measurements scale types.
- Measurements Classification
- Measurements in SWE/CS
- GQM

Resources

Chapter 3



Chapter 2
Chapter 3