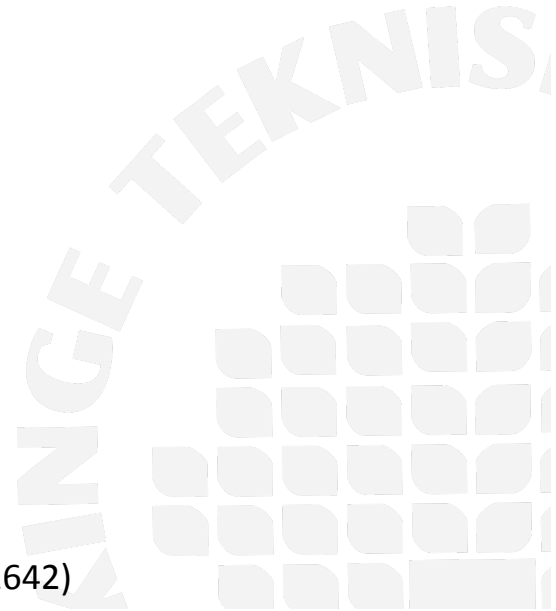
BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Software Metrics (PA1407)

Lecture 1
Course Introduction

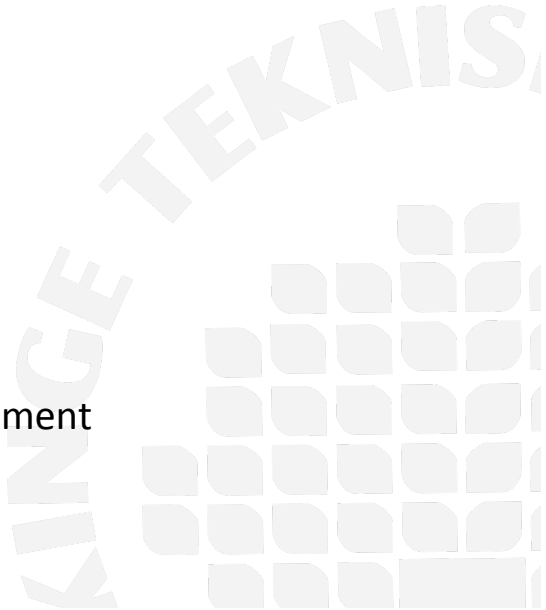


Measure what is measurable, and make measurable what is not so.
–Galileo Galilei (1564–1642)

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Course Responsible

- Muhammad Usman (muu@bth.se)
- Education
  - Currently working as PhD candidate at BTH
  - Licentiate in Software Engineering (2015) from BTH
    - Thesis: Supporting effort estimation in ASD.
  - MPhil – CS in 2006, Pakistan
  - MSc – CS in 2002, Pakistan
- Experience
  - Teaching since 2006
  - Industry 2004-2005
- Research interests
  - Effort estimation, agile software development, global software development
  - Empirical software engineering



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Lecture contents

- Course organization
- Defining measurement
- Measurement in everyday life
- Measurement in SE
- Scope of software metrics

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

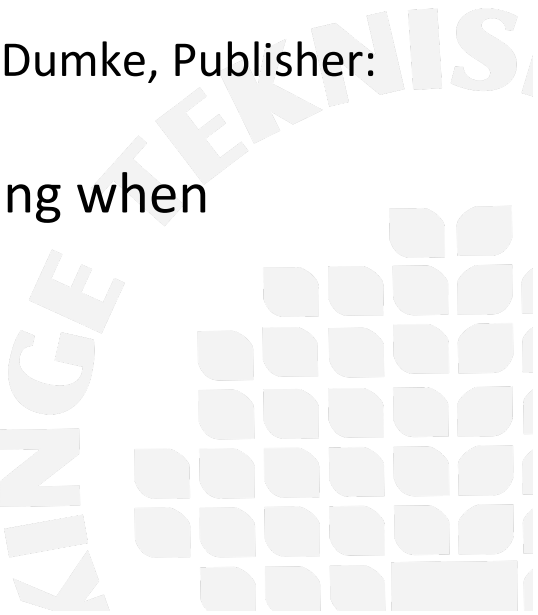
Course organization

- Final exam (4 ECTS) [First opportunity: May 24, 2016]
- Practical exercise (1 ECTS): Review on software maintainability measurement [Due date: April 21, 2016]
- Practical assignment (2 ECTS): Studying maintainability of an OSS [Due date: June 1, 2016]
- Practical experiment(0.5 ECTS): In-class experiment on effort estimation [Due date: May 19, 2016]
- For further details, please refer to course PM.
- The assessments mentioned above are only valid for new students registered in 2016 for the first time. Old students work on the previous assignments.
- Submissions through itslearning only.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Course material

- Lecture notes are based on following two books.
  - 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.
  - T2: Software measurement, Authors: Christof Ebert, Reiner Dumke, Publisher: Springer, 2007, ISBN: 978-3-540-71648-8.
- Plus research articles, which will be shared on itslearning when required.



Lecture Schedule

No	Date	Topics	Reading
1	2016-03-31	Course introduction, defining measurement and the need for it in SE, scope of software metrics, discussion on practical exercise	Chap 1 (T1), Chap 2 (T2)
2	2016-04-07	Basics of measurement: measurement theory, measurement scale types	Chap 2 (T1)
3	2016-04-12	Goal based framework for software measurement, empirical studies in measurement context	Chap 3 and 4 (T1)
4	2016-04-14	Empirical studies in measurement context, software metrics data collection and analysis	Chap 5 and 6 (T1)
5	2016-04-21	Measuring internal product attributes: size and structure, discussion on assignment	Chap 8 and 9 (T1)
6	2016-04-28	Measuring external product attributes, discussion on assignment	Chap 10 (T1)
7	2016-05-03	Effort and cost estimation, estimation in agile software development, velocity estimation	From research papers (will be shared at course page on itslearning before lecture date)
8	2016-05-10	Software process improvement and measurement, introducing a measurement program	Chap 13 (T1); Chap 11 and 6 (T2)
9	2016-05-12	Software measurement program, discussion on the use of psychometrics in SE , progress report on assignment	Chap 6 & From research papers (will be shared at course page on itslearning before lecture date)
10	2016-05-19	Class experiment	A requirements specification document (will be shared before lecture date)

Some additional information

- In all course-related communication, you should use your BTH-student email address.
- You are strongly recommended to read “Writing guide(s)” at http://writingguide.se
- Plagiarism and cheating
  - You have to make sure that you write in your own words and also cite source(s) in the correct manner. Please refer to the following URL for more information on Plagiarism and cheating
  - https://studentportal.bth.se/web/studentportal.nsf/web.xsp/plagiarism_and_cheating

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Measurement in everyday life

- Can you think of scenarios when you use measurement to make a decision?
  - Decisions based on price to select a product and/or service
    - Flight booking
    - Renting a shared accommodation
  - Decisions based on quality of product or service
    - Which internet browser is most appropriate for me?
    - Which internet service provider to use?
  - Height and size measurements for selecting appropriate clothing
  - Measurement in the field of medicine to support diagnosis.
  - Measurement in education
    - Teachers evaluate assessments, while students participate in course evaluations.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Defining 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"*.
- Entity: An object or an event in the real world
- Attribute: Feature or property of an entity.
  - Examples of Entities and attributes
    - Entity = Apartment then Attributes could be its size, rent, noise, light, layout.
    - If entity is a website, then what could be its attributes?
- Numbers and symbols are abstractions that we use to reflect our perceptions o f the real world.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Defining Measurement

- More on entities, attributes and numbers/symbols
  - Often, we talk about entities and their attributes interchangeably. It is important to differentiate them.
  - It is wrong to say that we measure entities or that we measure attributes; in fact, we measure attributes of entities.
    - It is ambiguous to say that we "measure a room" or we "measure a system".
- In defining the numbers and symbols, we try to preserve certain relationships that we see among the entities.
  - Thus, someone who is six feet in height is taller than someone who is five feet in height.
  - This number or symbol can be very useful and important. If we have never met Herman but are told that he is seven feet tall, we can imagine his height in relation to ourselves without our ever having seen him.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Making things measureable

- Measurement makes concepts more visible and therefore more understandable and controllable
- We are now able to measure attributes that were previously thought un-measurable.
  - Measures of attributes such as human intelligence, air quality, and economic inflation form the basis for important decisions that affect our everyday lives.
- To improve the rigor of measurement in software engineering, we need not restrict the type or range of measurements we can make.
  - When it is not clear how we might measure an attribute, the act of proposing such measures will open a debate that leads to greater understanding.
- There are two kinds of quantifications
  - Measurement: direct quantification
  - Calculation: indirect

Making things measureable – Example from Soccer

EXAMPLE 1.3: There are no universally-recognized measures to identify the best individual soccer players (although number of goals scored is a fairly accurate measure of quality of a striker). Although many fans and players have argued that player quality is an unmeasurable attribute, this issue was addressed prior to the 1994 World Cup games in the USA. To provide an objective (measurable) means of determining the "man of the match," several new measurements were proposed:

> "To help FIFA assess the best players, it will be necessary to add to the pitch markings. At ten meter intervals there will be lines both across and down the pitch. This will allow accurate pass yardage, sideways pass yardage, dribble yardage, and heading yardage to be found for each player."
>
> Translated from "Likely changes to the rules for the 1994 World Cup," *Nouveaux FIFA d'Arbitres* (FIFA Referees News), March 1990.

It was suggested that these measurements be added and weighted with the number of goals scored; tackles, saves or interceptions made; frequency and distance of passes (of various types), dribbles and headers. Notice that the proposed new measure of player quality required a change to the physical environment in which the game is played.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Making things measureable – Software engineering

- In many instances, we want an overall score that combines several measures into a "big picture" in SE
  - We want to be able to tell if a software product is good or bad, based on a set of measures, each of which captures a facet o f "goodness."
  - The composite measures can be controversial.
  - Likewise, controversy erupts when we try to capture qualitative information about some aspect of software engineering.
  - Sometimes it becomes necessary to modify our environment or our practices in order to measure something new or in a new way.
- In many cases, change is difficult for people to accept.
- There are management issues to be considered whenever a measurement program is implemented or changed.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Measurement in SE

- Software pervades our lives
  - From oven controls to airbags, from banking transactions to air traffic control, and from sophisticated power plants to sophisticated weapons, our lives and the quality of life depend on software.
- SE has done an admirable job, but there is always room for further improvement.
  - The literature is rife with project failures, and also stories about software that has put lives and businesses at risk.
  - There has been progress in proposing new techniques, tools etc. However, these alone are not enough.
- Underpinning the scientific process is measurement.

Measurement in SE

- Measurement has been considered a luxury in software engineering. For most development projects:
  - We fail to set measurable targets for our software products
  - We fail to understand and quantify the component costs of software projects.
  - We do not quantify or predict the quality of the products we produce.
  - We allow anecdotal evidence to convince us to try yet another revolutionary new development technology, without doing a carefully controlled study to determine if the technology is efficient and effective.

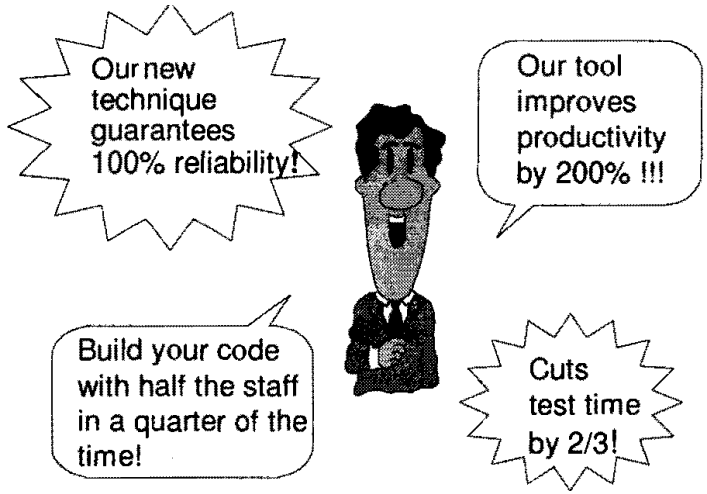
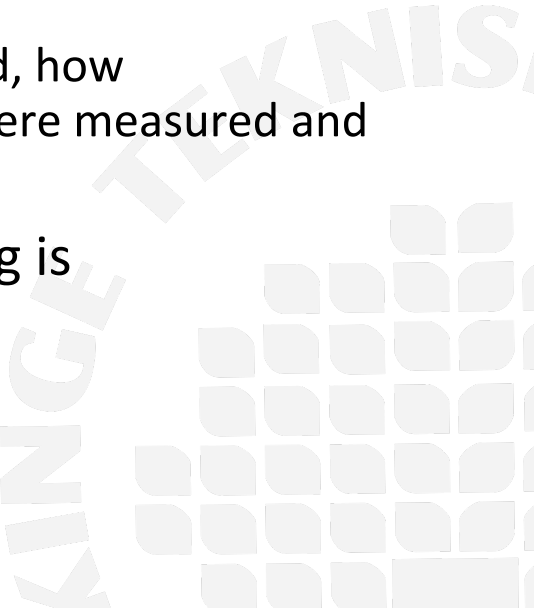


Figure 1.1: Measurement for promotion

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Measurement in SE

- When measurements are made, they are often done infrequently, inconsistently, and incompletely.
  - For example, a developer may claim that there are on average 55 faults in every I000 lines of software code.
  - But we are not always told how these results were obtained, how experiments were designed and executed, which entities were measured and how, and what were the realistic error margins.
- Thus, the lack of measurement in software engineering is compounded by the lack of a rigorous approach.



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Objectives for software measurement

- Measurement is needed at least for assessing the status of your projects, products, processes, and resources.
  - "You cannot control what you cannot measure." (DeMarco, 1982)
- Every measurement action must be motivated by a particular goal or need that is clearly defined and easily understandable.
- The measurement objectives must be specific, tied to what the managers, developers and users need to know.
- It is the goals that tell us how the measurement information will be used once it is collected.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Objectives for software measurement

- Managers
  - Process cost
  - Staff productivity
  - Code quality
  - User satisfaction
  - How can we improve
- Engineers
  - Are requirements stable?
  - Have we found all faults?
  - Have we met process/product goals
  - Future?

Measurement for understanding, control and improvement

- Measurement supports three basic activities
  - Understanding
  - Control
  - Improvement
- Users of the data should always be aware of the limited accuracy of prediction and of the margin of error in the measurements.
  - You can neither predict nor control what you cannot measure.

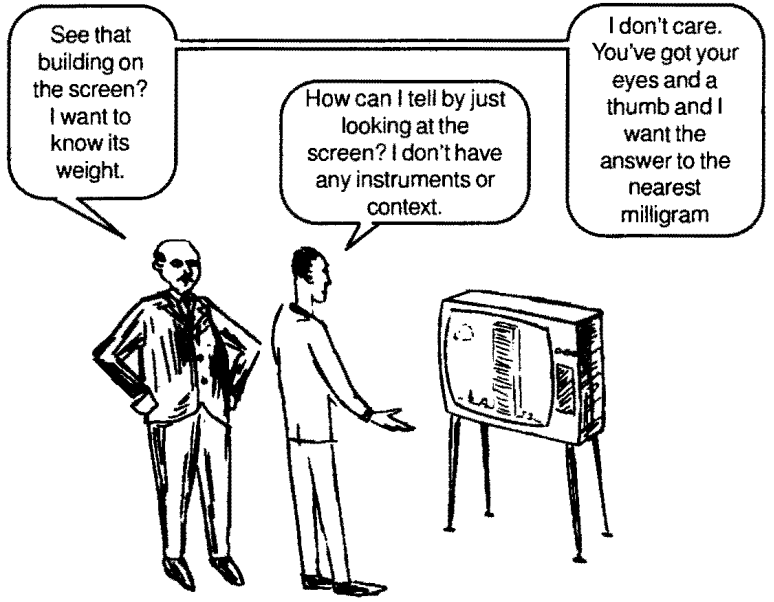
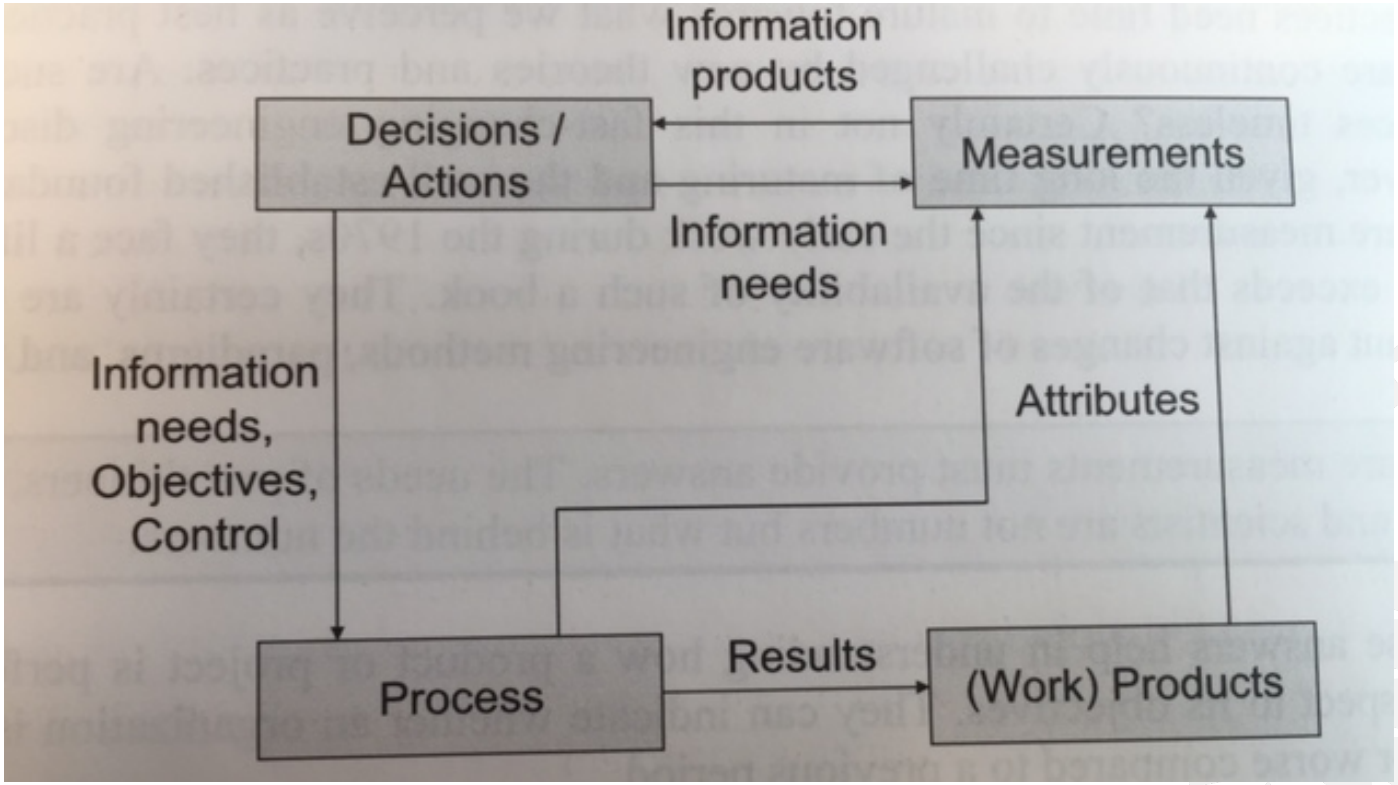


Figure 1.2: Software measurement – resource estimation

Simple information measurement model



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

The scope of software metrics

- Cost and effort estimation
- Productivity measures and models
- Data collection
- Quality models and measures
- Reliability models
- Performance evaluation and models
- Structural and complexity metrics
- Capability-maturity assessment
- Management by metrics
- Evaluation of methods and tools

Software quality model

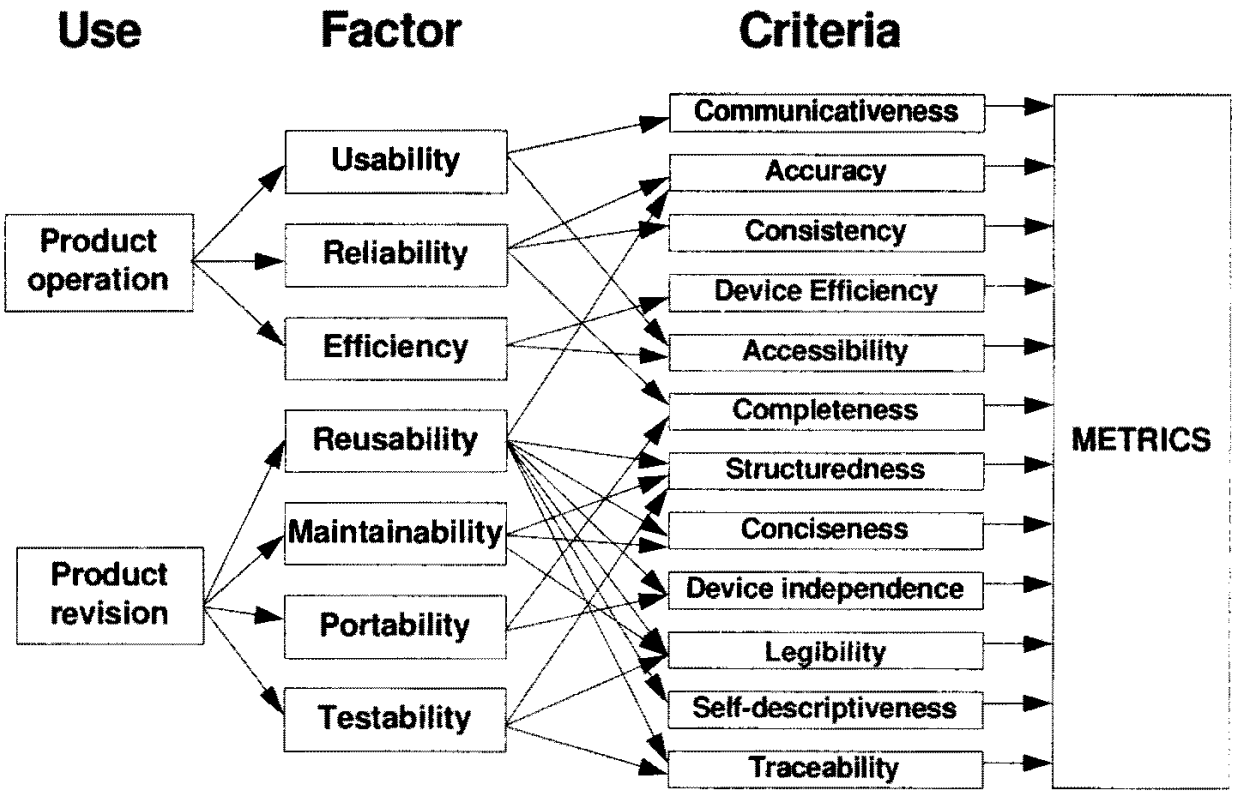


Figure 1.5: Software quality model

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Acknowledgement

- 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.
  - T2: Software measurement, Authors: Christof Ebert, Reiner Dumke, Publisher: Springer, 2007, ISBN: 978-3-540-71648-8.