Software Project Management

HUSZERL Gábor

huszerl@mit.bme.hu

MŰEGYETEM 1782

BME VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR

Méréstechnika és
Információs Rendszerek
Tanszék

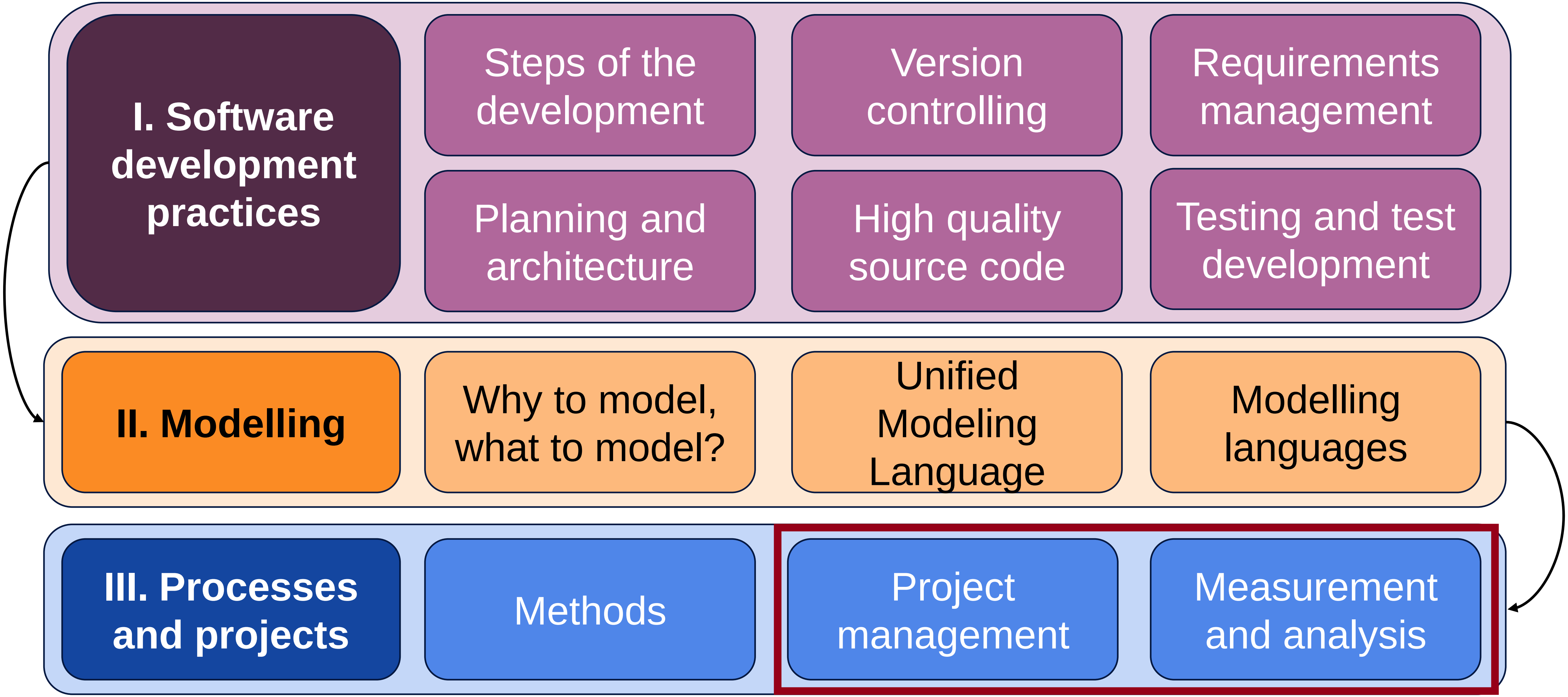
ftsrg

Critical Systems
Research Group

Learning Outcomes

- At the end of the lecture the students are expected to be able to
- (K1) understand the use cases of software metrics,
- (K1) understand the basic concepts of project management,
- (K1) understand the basic concepts of project costs and their estimation.

Further Topics of the Subject



I. Software development practices

- Steps of the development
- Version controlling
- Requirements management
- Planning and architecture
- High quality source code
- Testing and test development

II. Modelling

- Why to model, what to model?
- Unified Modeling Language
- Modelling languages

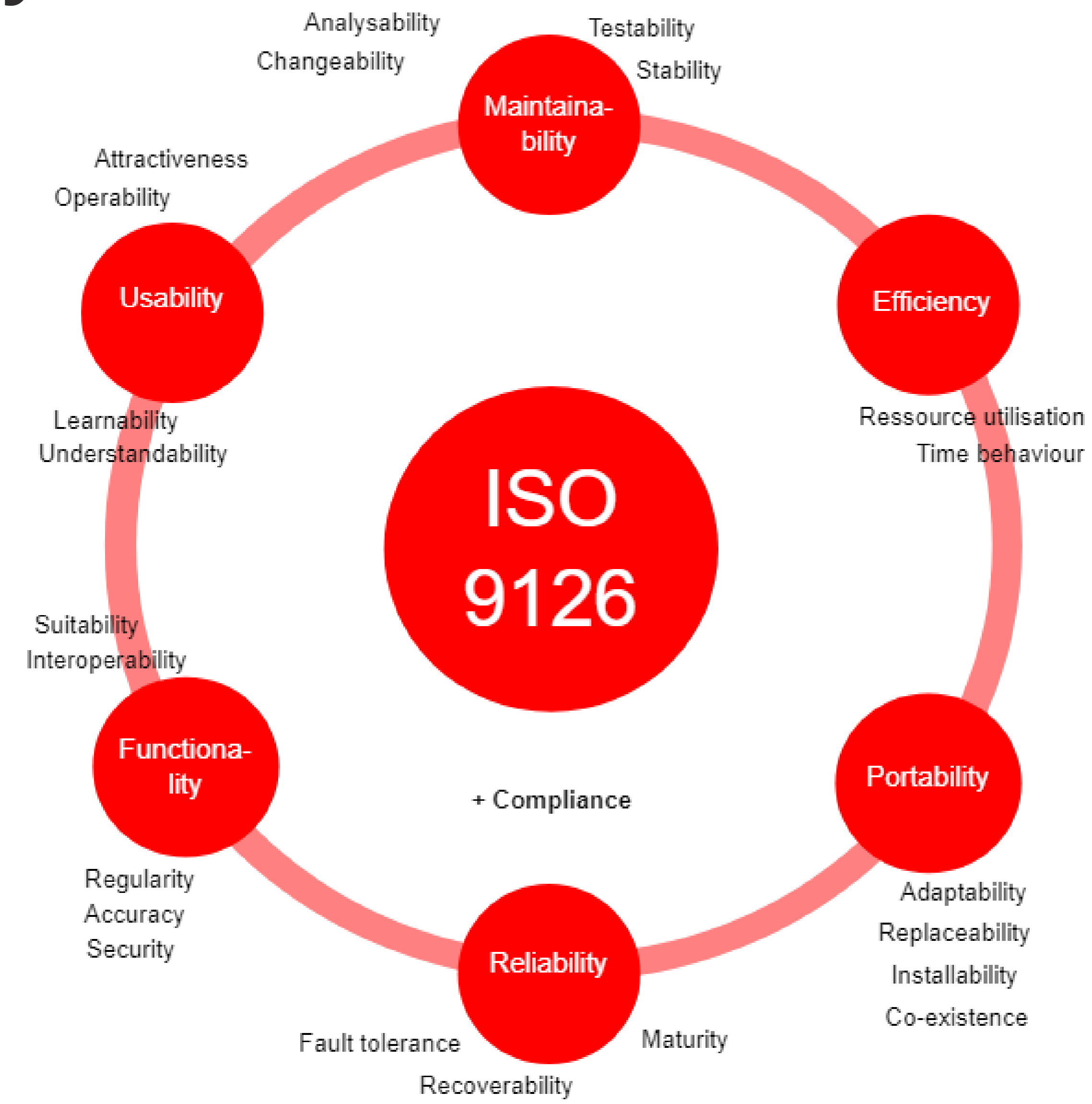
III. Processes and projects

- Methods
- Project management
- Measurement and analysis

What Is a Metric?

What Is Meant by Quality?

- … and how it is measured?



What Is a Metric?

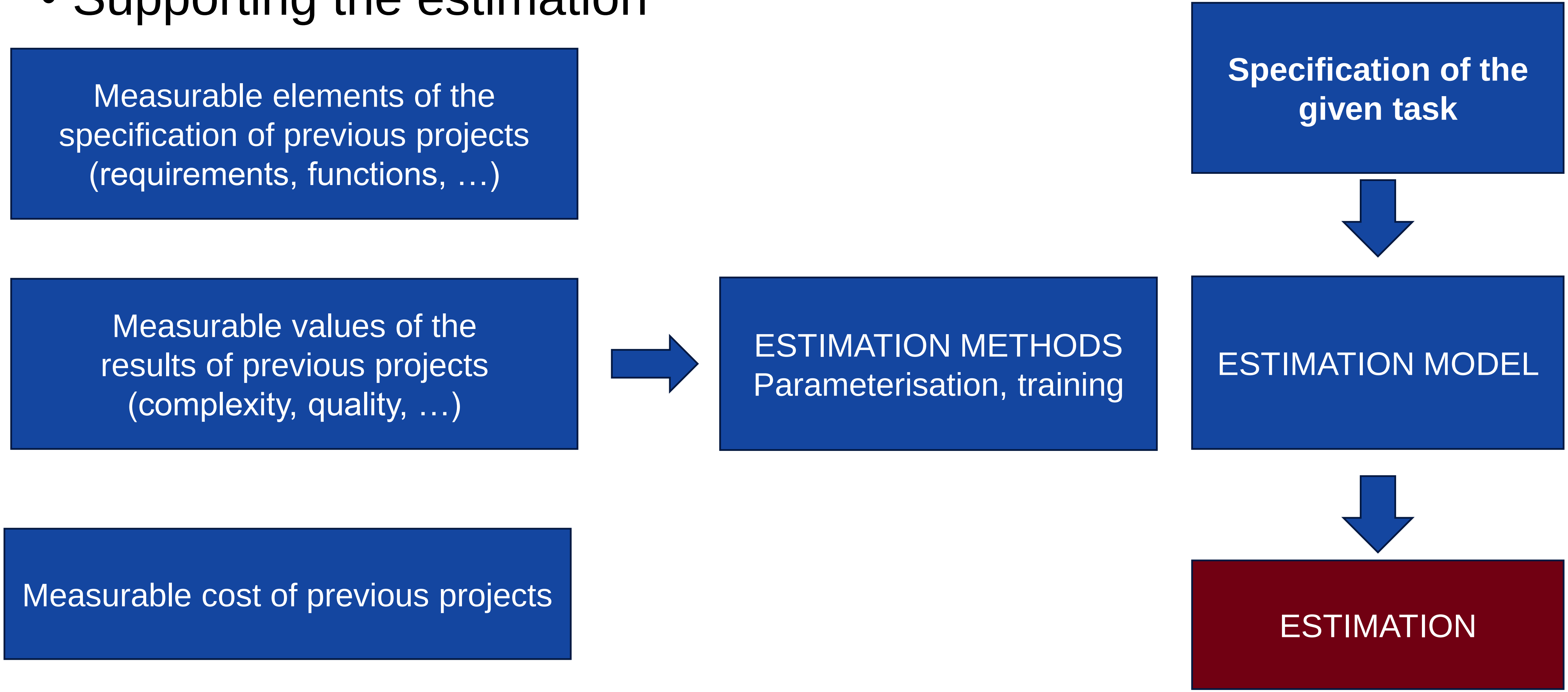
- Objectively measurable, quantifiable quantity
- Must be specified
  - Scale, unit of measurement
  - Measurement method
  - Interpretation
- Measurement and evaluation
  - Mostly static analysis
    (see te 6th lecture about “High Quality Source Code”)

Metrics on Different Levels

- Source code metrics
- Test metrics
- Run-time metrics
- Metrics related to the development process
- Metrics related to the complete operation

Application of Metrics

- Supporting the estimation



Measurable elements of the specification of previous projects (requirements, functions, …)

Measurable values of the results of previous projects (complexity, quality, …)

Measurable cost of previous projects

ESTIMATION METHODS
Parameterisation, training

Specification of the given task

ESTIMATION MODEL

ESTIMATION

Application of Metrics – Evaluation

Quality

- Where an error can be in the software?
- Which parts would be worth of refactoring?
- How maintainable will it be?
- Relative comparison, NOT an absolute measure

Targeted checking

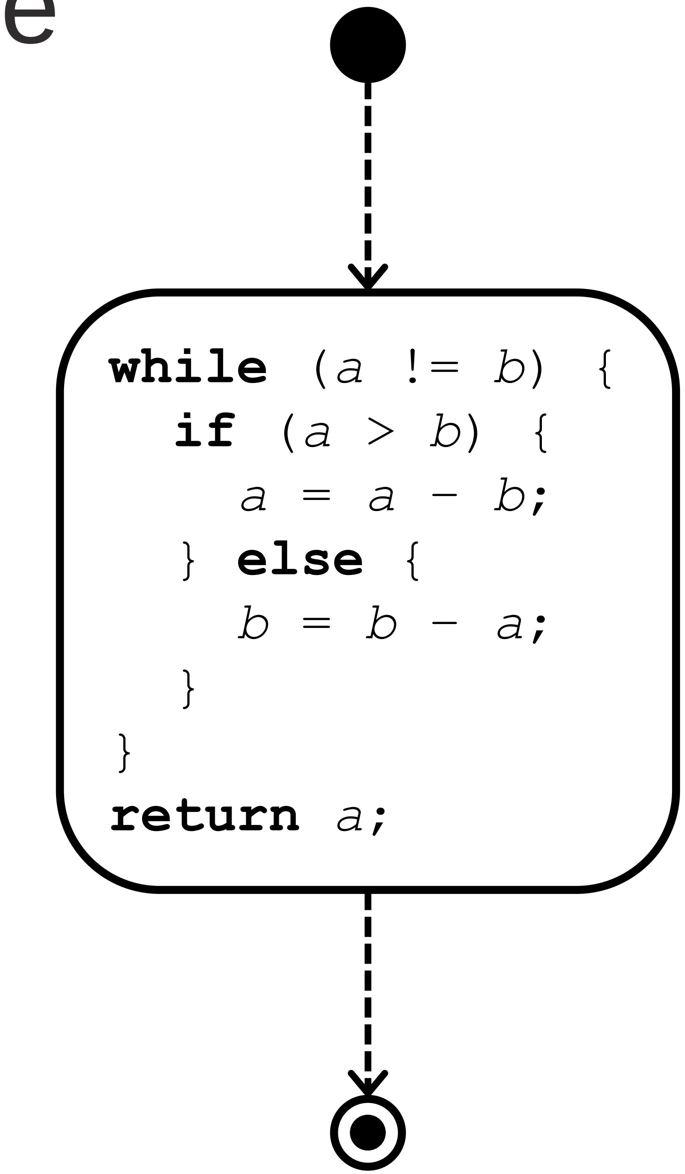
- Based on automatically calculated values
- e.g. support for code verification

Structural Metrics

CFG, cyclomatic complexity

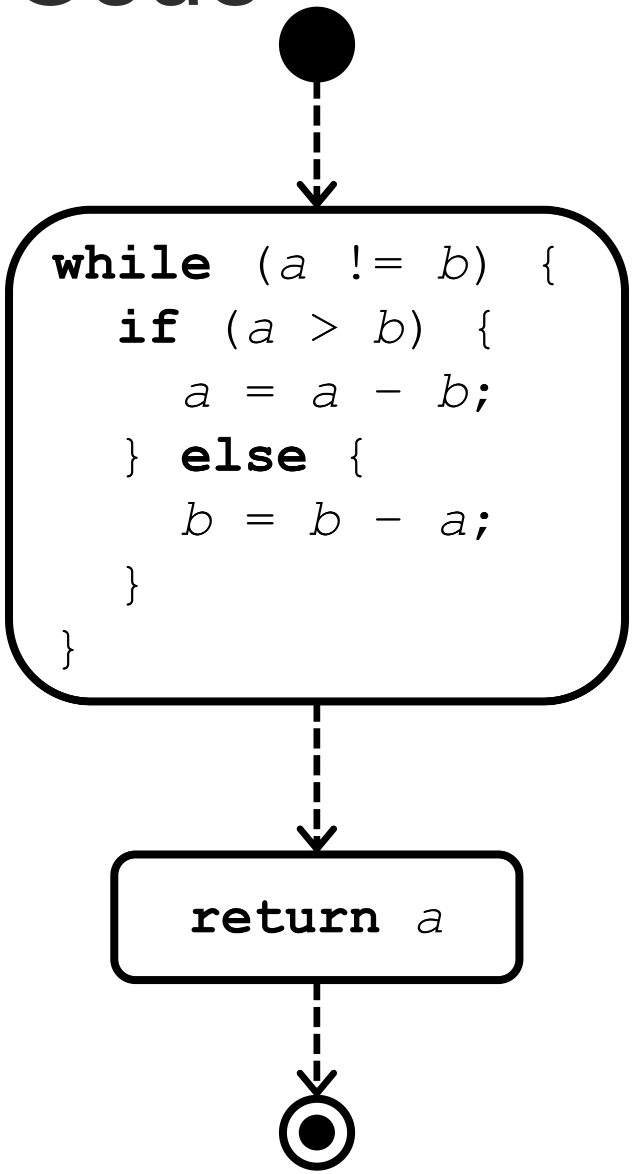
Control Flow of a Code

- Directed graph
  - nodes: instructions
  - edges: dependencies
- See: static code analysis
- Example:



```
while (a != b) {
  if (a > b) {
    a = a - b;
  } else {
    b = b - a;
  }
}
return a;
```

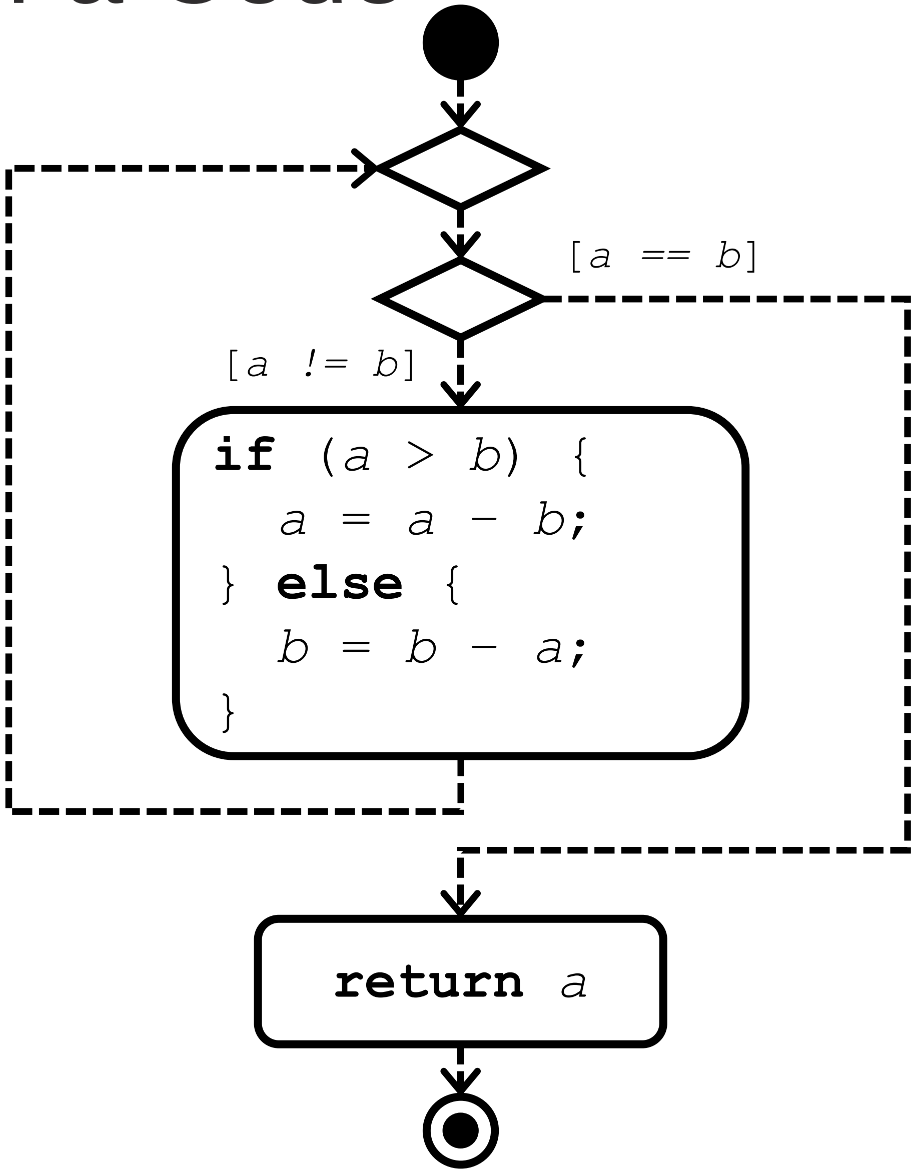
Control Flow of a Code



```
while (a != b) {
  if (a > b) {
    a = a - b;
  } else {
    b = b - a;
  }
}
```

```
return a
```

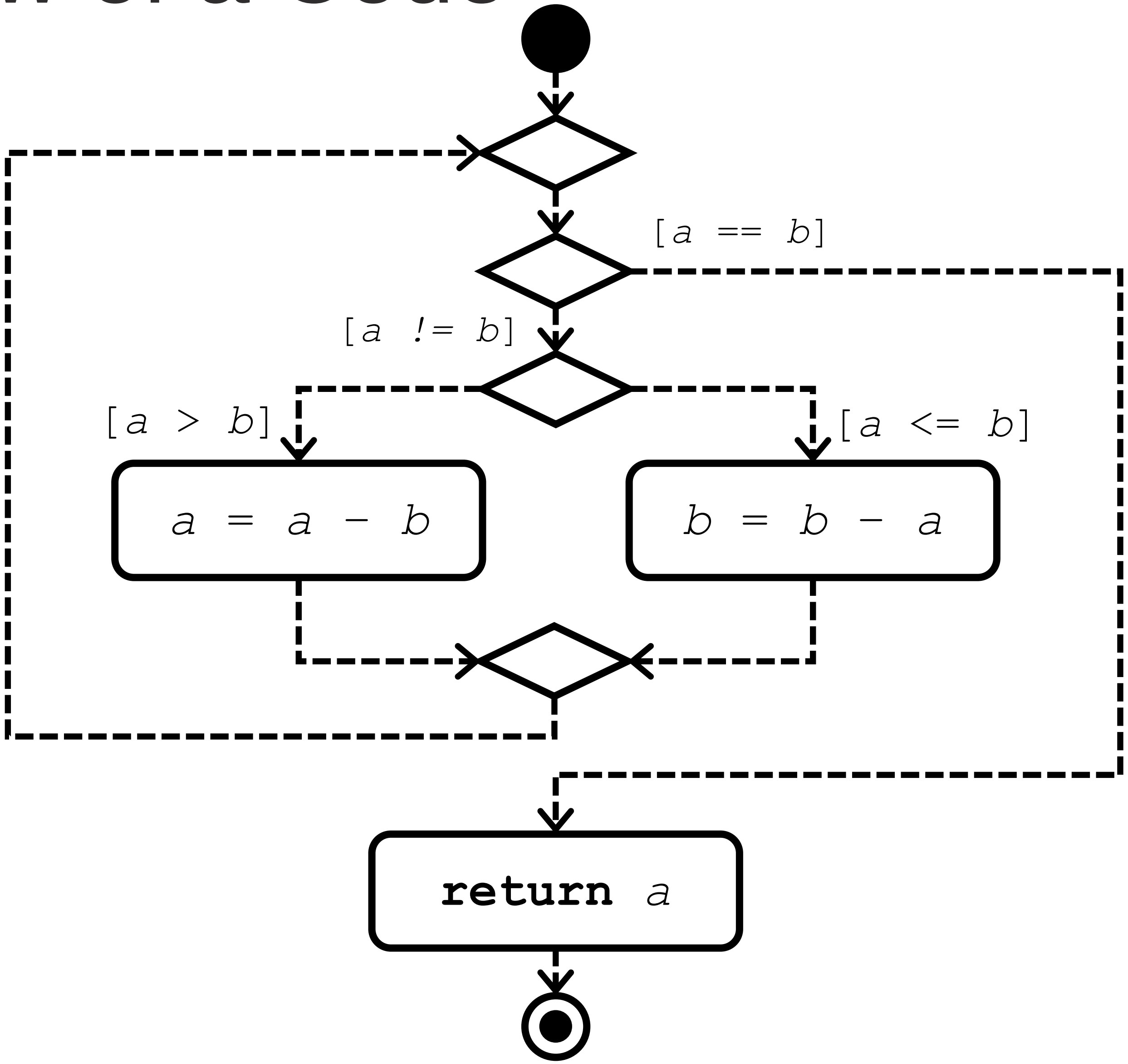
Control Flow of a Code



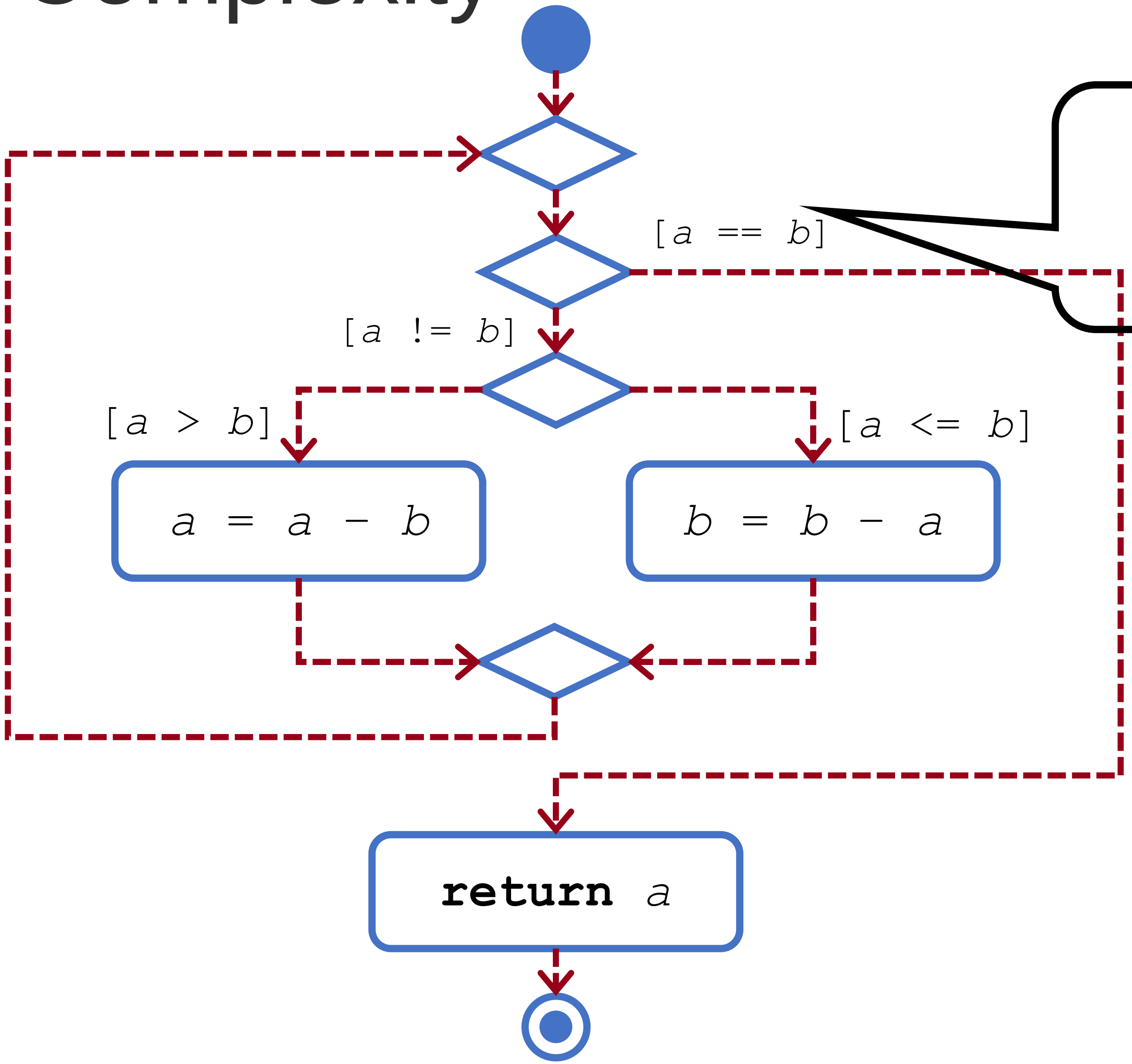
```
if (a > b) {
  a = a - b;
} else {
  b = b - a;
}
```

```
return a
```

Control Flow of a Code



Cyclomatic Complexity



Cyclomatic complexity

$M = E - N + 2$

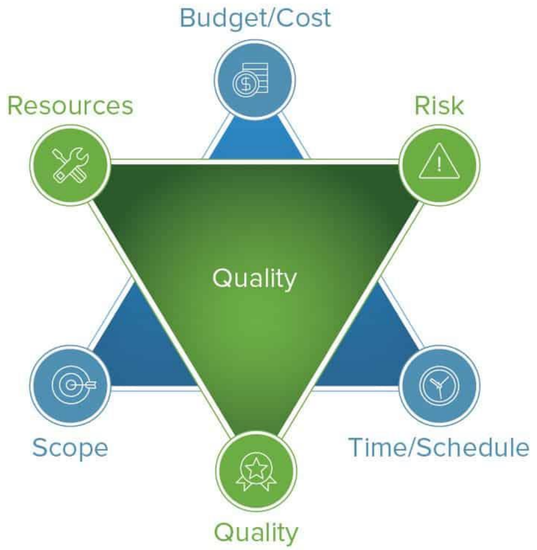
Further Structural Metrics

- Deepest embedded control structure
  - Max nesting
  - Structure, not the execution!
- Knots
  - Number of parallel branches
- Comment-to-Code ratio
  - E.g. requiring a minimum
- Number of code lines (LoC)

How can they be **evaluated**?

- Maximum / average values
- 90. percentile: 90% of all values are smaller or equal to this value
- On the level of functions / modules

Project Management



Scope-Budget-Time

Resources-Risk-Quality

What Is a Project?

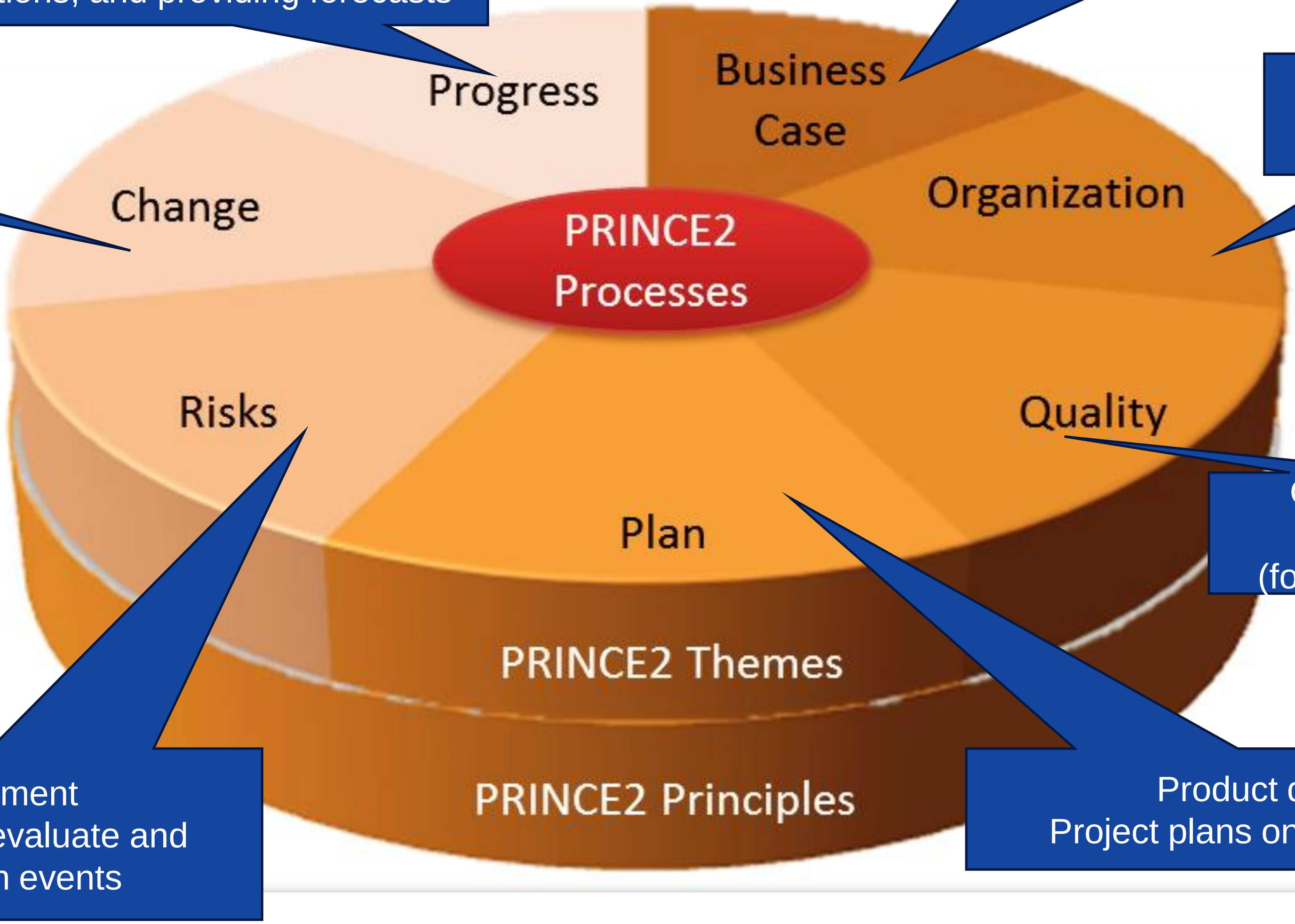
> **Project:** a managed set of interrelated activities and resources, including people, that delivers one or more products or services to a customer or end user. (CMMI- Capability Maturity Model Integration)

- It has well defined …
  - Start
  - End
  - Goal
  - Responsible
  - Team
- See also https://www.pmi.org/pmbok-guide-standards/foundational/pmbok

PRINCE2 Method

- A method of effective project management, originally developed for the UK government
- „PRojects IN Controlled Environments“
  - (PROMPT: 1975, PRINCE: 1989, PRINCE2: 1996, PRINCE2 v7: 2023)
- Defines performance targets as project goals
  - Time, cost, quality, scope, benefits, and risk
- Defines 4 levels of management
  - Corporate or Programme Management level, Project board level, Project manager level, and Delivery level

PRINCE2 Method – Themes



Establishing controls for monitoring actual project outcomes, managing deviations, and providing forecasts

Business Case
(continued business justification)

Change management

Communication protocols, management roles

Quality management
Quality assurance
(following specific cases)

Risk management
Anticipate, identify, evaluate and control uncertain events

Product definition
Project plans on different levels

Typical Project Documents

Project Definition Document / Charter (2-3 pages)

- Introducing the project goals
- Project scope
- Priority, risk, responsibilities in the project
- Stakeholders, sponsors

Project Work Plan

- Goals, budget, activities, milestones, …
- Tasks, deadlines, responsible persons, results
- Specific methods for checking the results

Project Handbook

- Project objectives, the approach for achieving the project goal, the key controlling processes
- Project „dictionary”, critical success factors, policies, rules, project mindset
- Aspects beyond development (communication, conflict resolution, escalation procedure, etc.)

Quality Assurance Handbook

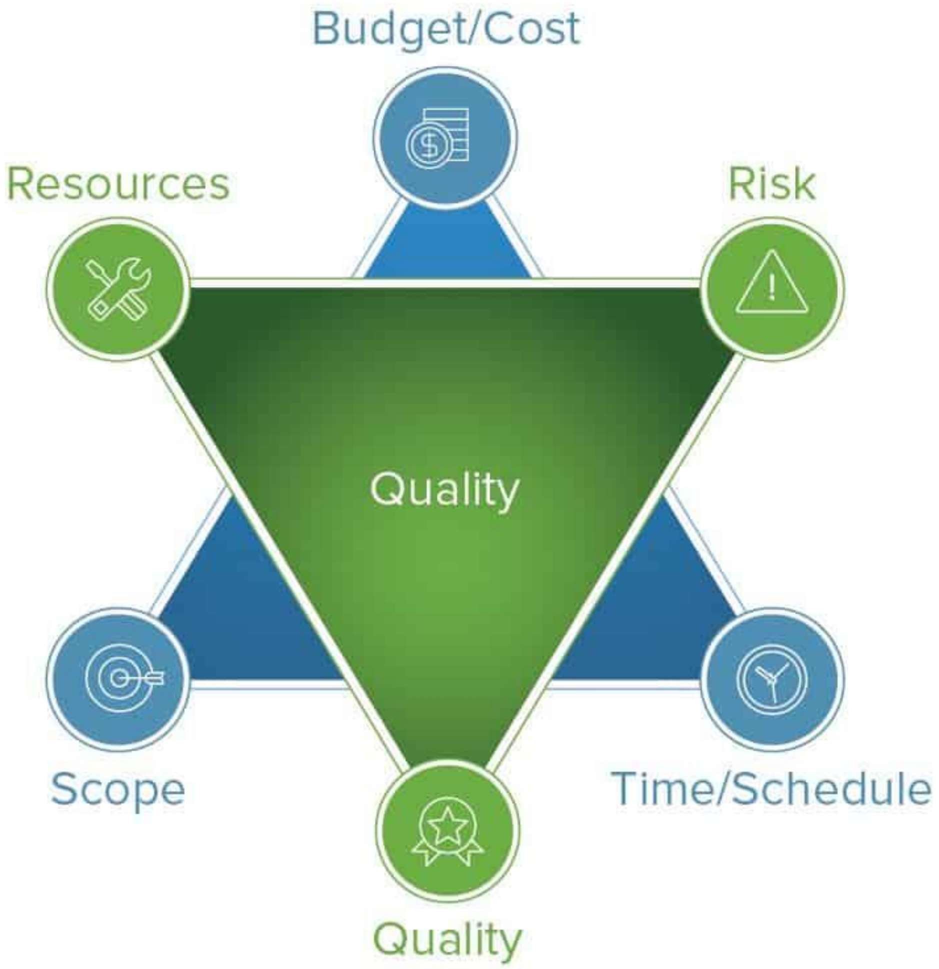
- For the whole project
- Ow to measure what? Who may confirm what? (e.g. four-eyes-principle)
- Checking and correction methods, verification specification
- Products/processes to check

***Final* Project Report**

- What was (not) achieved?
- „Post-mortem”-Analysis
- Development of the project, problems encountered, experiences, …

Aspects and Goals

- What can be reached simultaneously?



See also:

www.smartsheet.com/triple-constraint-triangle-theory

Project Leaders

- Sponsor
  - Responsible for the success of the whole project
  - Responsible for the strategic planning
  - Provides the necessary resources
  - Supports business decision making
  - Establishes/approves success criteria
- Project manager
  - Responsible for the execution of the project
  - Organising and controlling the activities
- Stakeholder
  - Not necessarily actively involved, but the project depends on him
  - Investor, customer, partner, regulatory authority, ...

Work Package

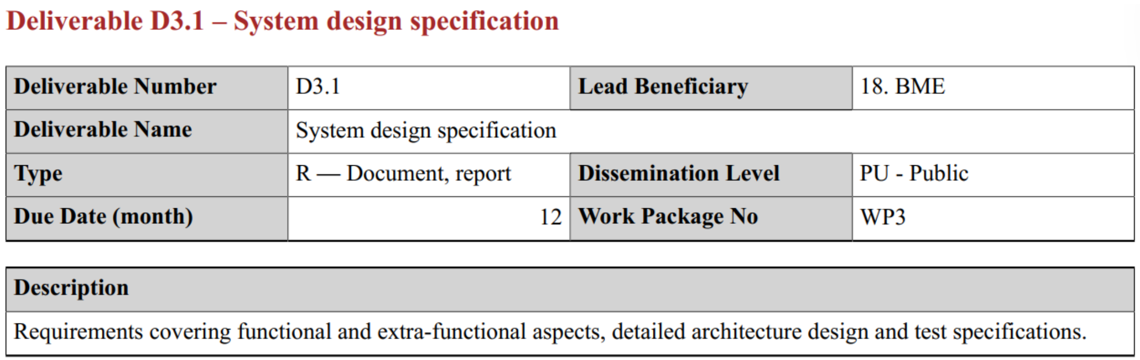
- Related tasks and activities
  - With jointly responsible results and costs
  - Responsible, resources
- Hierarchic decomposition of the project
  - Always necessary
  - WP as a concept, typically used for longer projects with multiple actors
- Outputs:
  - Deliverable: specific results

    “Software to manage student data” (software, but often a document)
  - Milestone: state of the project

    “Pilot version of SW to manage student data available for testing”
    (often submission of a specific document)

Deliverable

- Delivering a specific output
  - System design
  - Software, installation handbook
  - Research report
  - Test evaluation

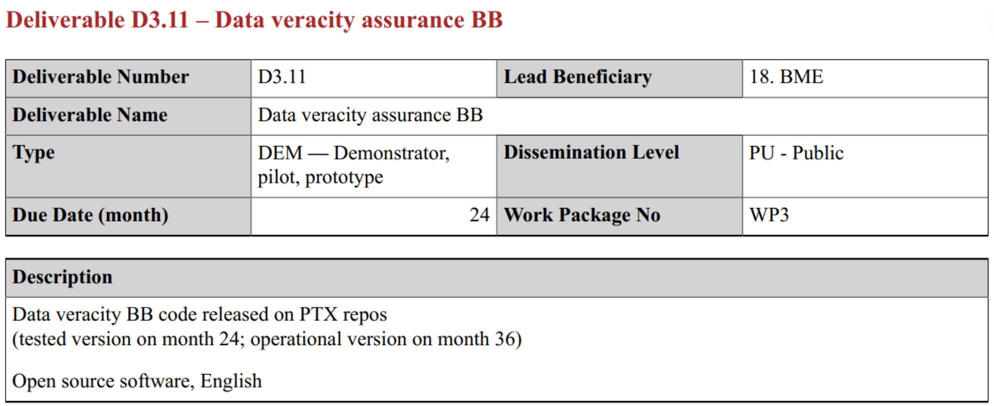


Deliverable D3.1 – System design specification

Deliverable Number	D3.1	**Lead Beneficiary**	18. BME
Deliverable Name	System design specification		
Type	R — Document, report	**Dissemination Level**	PU - Public
Due Date (month)	12	**Work Package No**	WP3

Description
Requirements covering functional and extra-functional aspects, detailed architecture design and test specifications.

- Basis of evaluating/accepting the project
  - Deadline
  - Responsible person



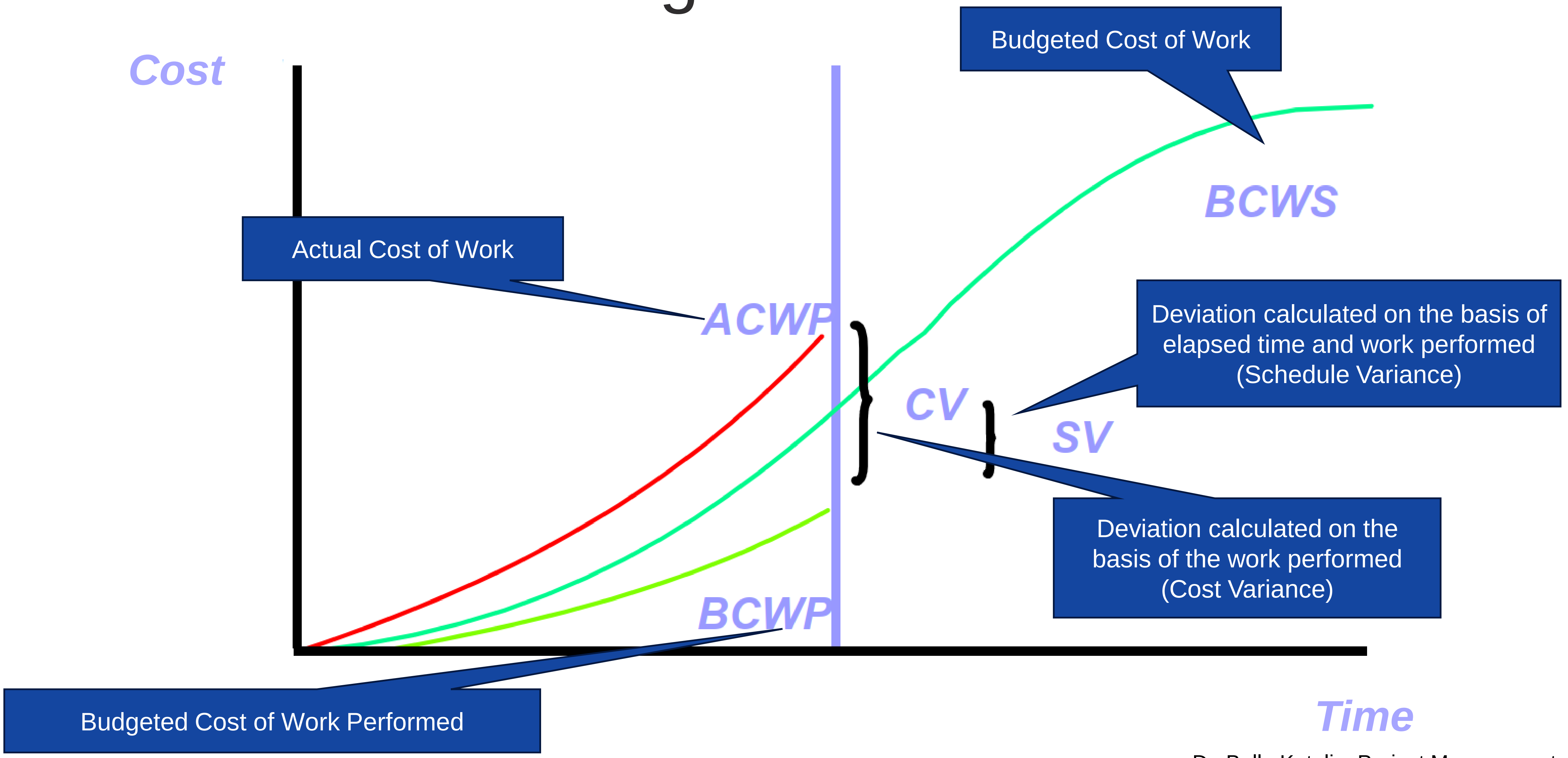
Deliverable D3.11 – Data veracity assurance BB

Deliverable Number	D3.11	**Lead Beneficiary**	18. BME
Deliverable Name	Data veracity assurance BB		
Type	DEM — Demonstrator, pilot, prototype	**Dissemination Level**	PU - Public
Due Date (month)	24	**Work Package No**	WP3

Description
Data veracity BB code released on PTX repos (tested version on month 24; operational version on month 36) Open source software, English

Following the Costs

Performance Management Chart



Deviation calculated on the basis of elapsed time and work performed (Schedule Variance)

Deviation calculated on the basis of the work performed (Cost Variance)

Budgeted Cost of Work Performed

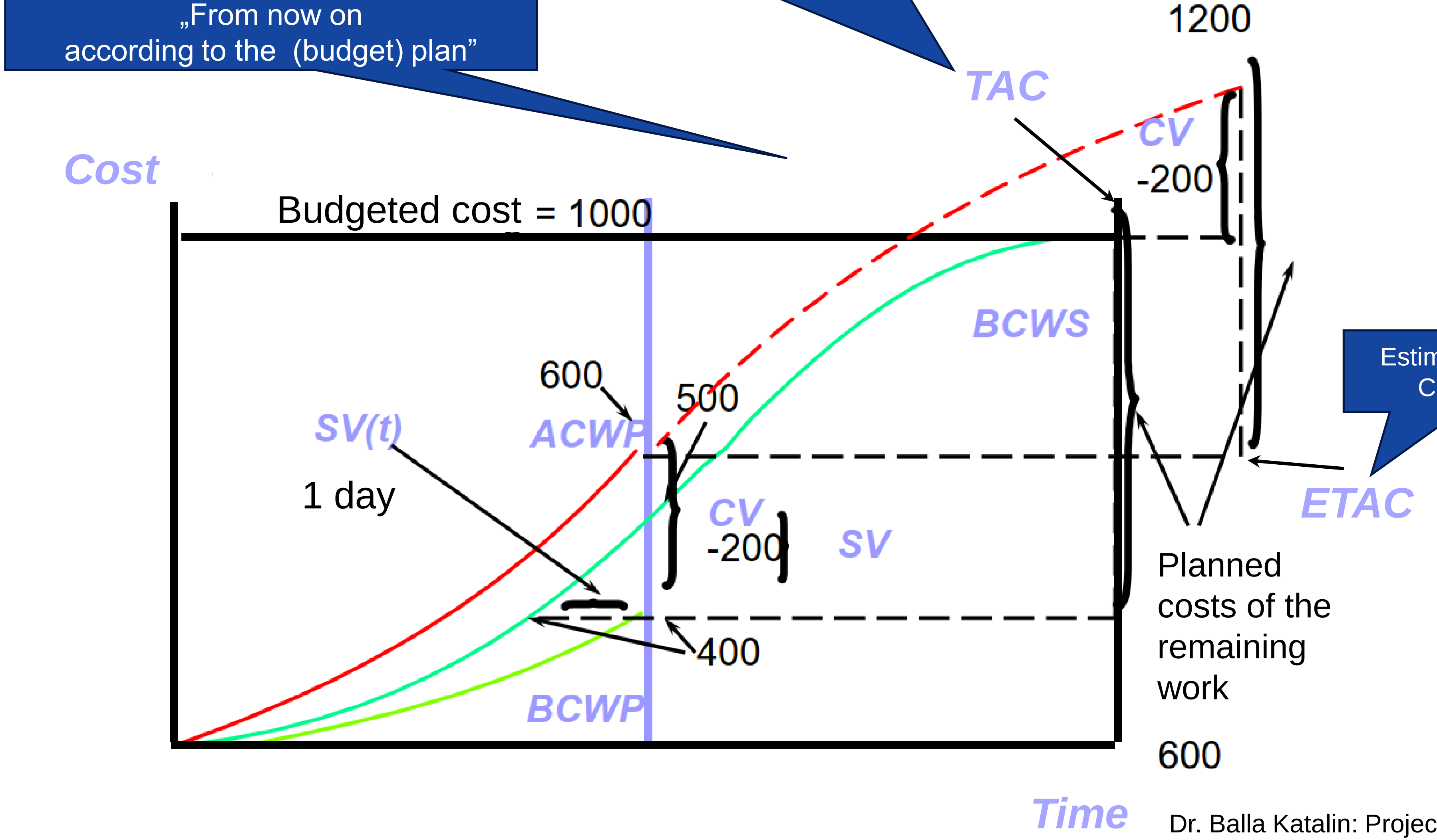
Dr. Balla Katalin: Project Management

Estimation: by Current Reference Point

„From now on according to the (budget) plan”

Original Time at Completion

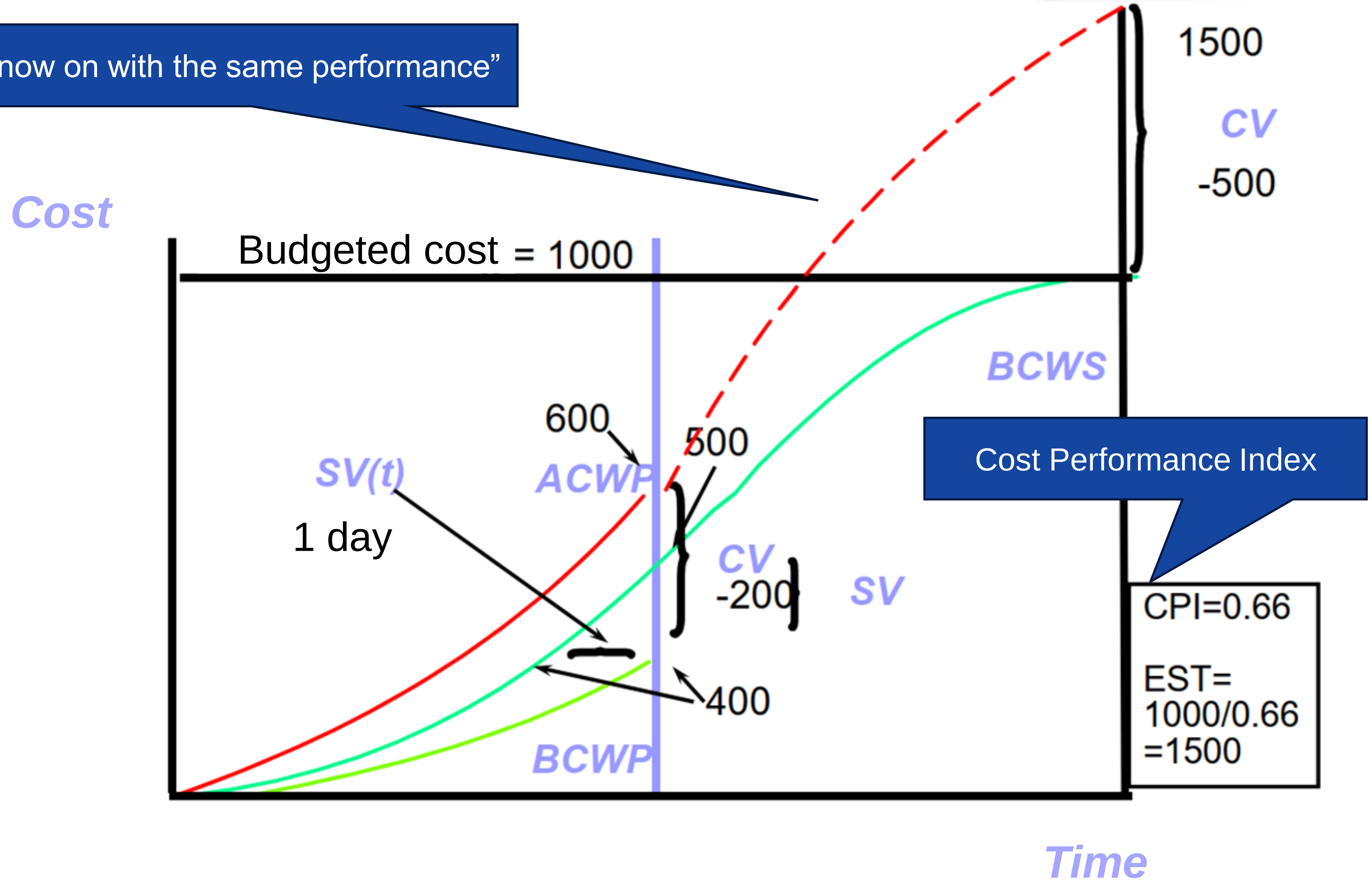
Estimated Time at Completion



Dr. Balla Katalin: Project Management

Estimation: by Current Performance

„ From now on with the same performance”



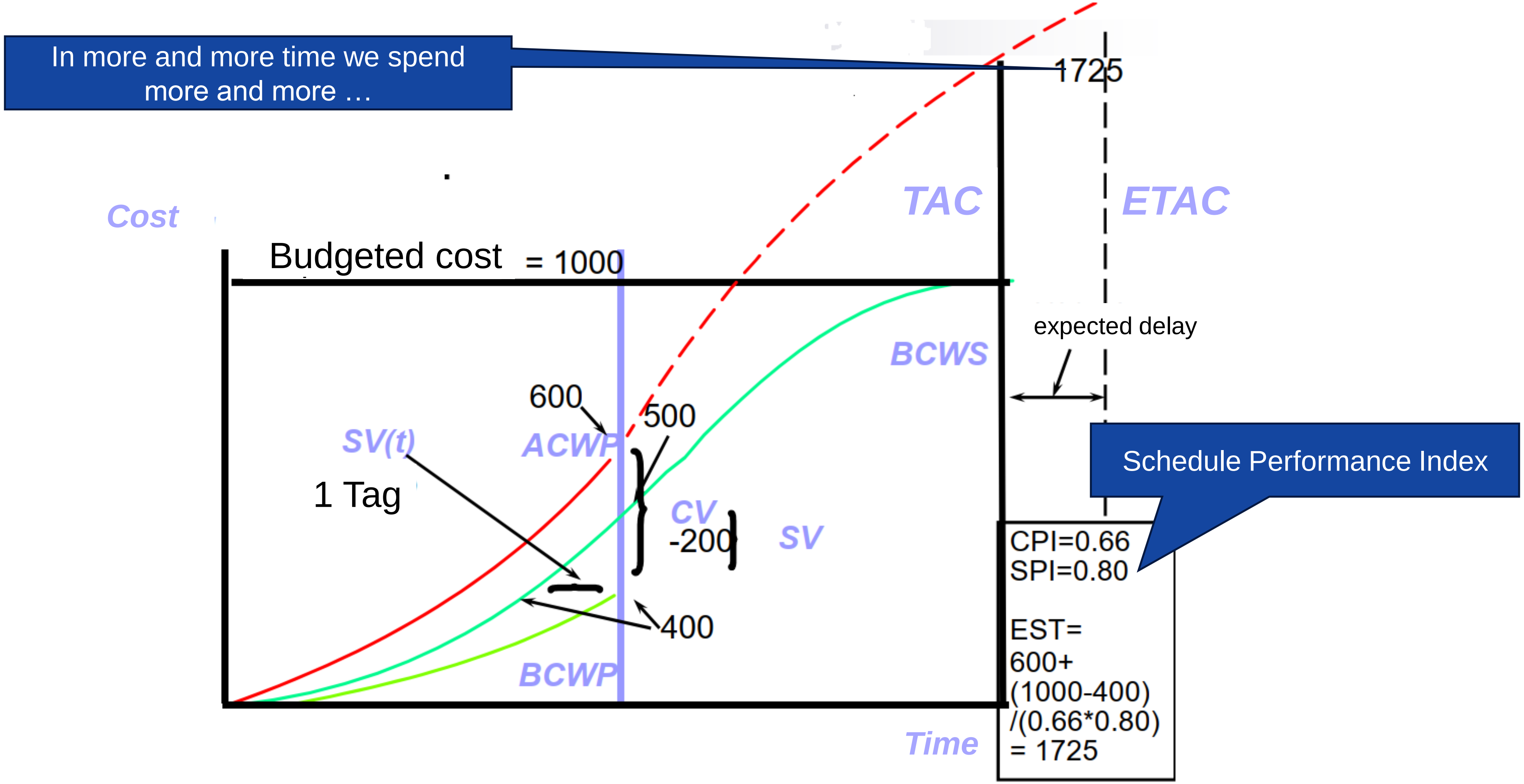
Cost Performance Index

CPI=0.66

EST=
1000/0.66
=1500

Dr. Balla Katalin: Project Management

Estimation: Considering the Time Factor



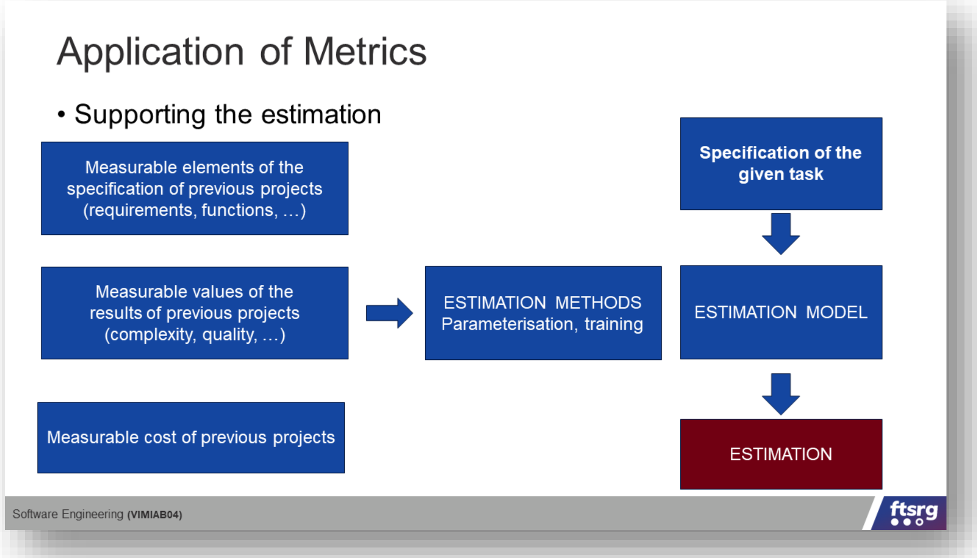
CPI=0.66
SPI=0.80

EST=
600+
(1000-400)
/(0.66*0.80)
= 1725

Dr. Balla Katalin: Project Management

Summary

Summary



Application of Metrics

- Supporting the estimation

Measurable elements of the specification of previous projects (requirements, functions, …)

Measurable values of the results of previous projects (complexity, quality, …)

Measurable cost of previous projects

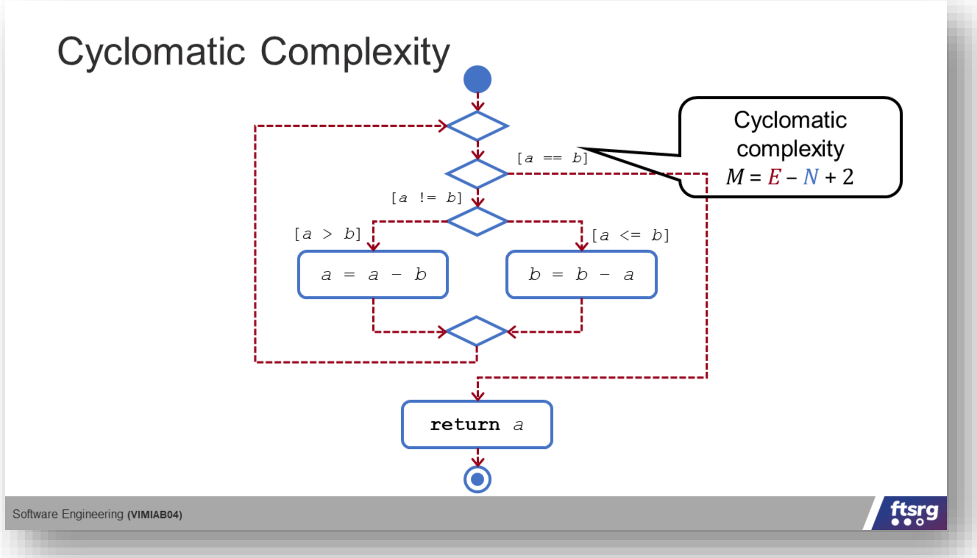
ESTIMATION METHODS Parameterisation, training

Specification of the given task

ESTIMATION MODEL

ESTIMATION

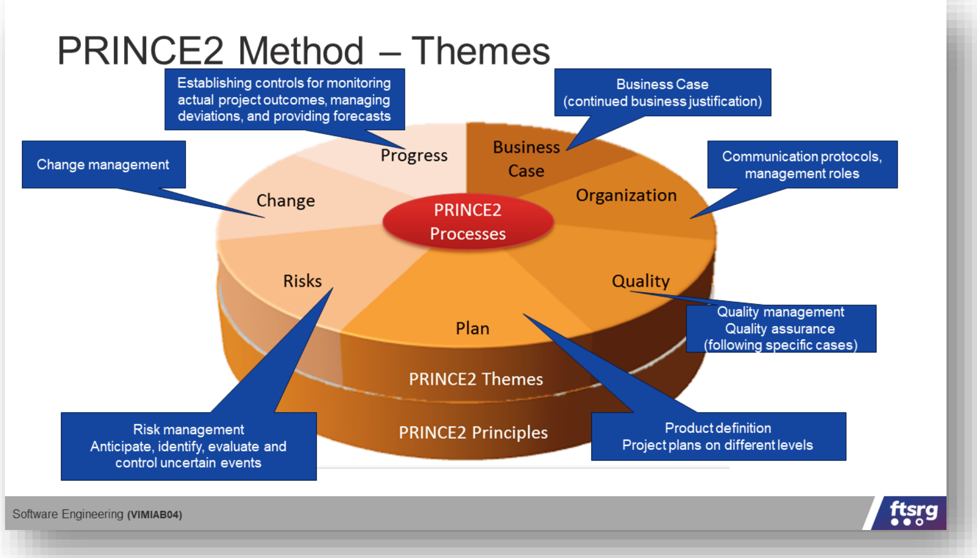
Software Engineering **(VIMIAB04)**



Cyclomatic Complexity

Cyclomatic complexity $M = E - N + 2$

Software Engineering **(VIMIAB04)**



PRINCE2 Method – Themes

Establishing controls for monitoring actual project outcomes, managing deviations, and providing forecasts

Business Case (continued business justification)

Change management

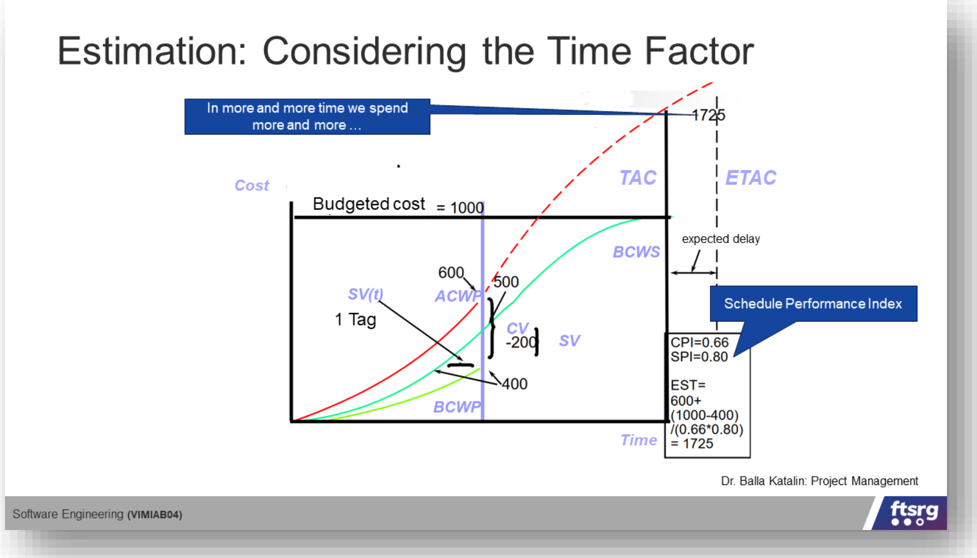
Communication protocols, management roles

Quality management
Quality assurance
(following specific cases)

Risk management
Anticipate, identify, evaluate and control uncertain events

Product definition
Project plans on different levels

Software Engineering **(VIMIAB04)**



Estimation: Considering the Time Factor

In more and more time we spend more and more …

Schedule Performance Index

CPI=0.66
SPI=0.80

EST=
600+
(1000-400)
/(0.66*0.80)
= 1725

Dr. Balla Katalin: Project Management

Software Engineering **(VIMIAB04)**