Introduction on Software Cost Estimation

Rolf Hintermann

Content of this presentation

- History of software cost estimation
- Overview
- Estimation process
- Conclusion

1940‘s – beginning of the computer era

- Computers
  - A few of them in use
- Applications
  - A few of them in use
  - Small in size
  - Typically one developer
  - „cheap“ to develop

Today

- Computers
  - A lot of them in use
- Applications
  - A lot of them in use
  - Large in size
  - Many developers
  - „expensive“ to develop

Why does software cost so much?

Why does software cost so much?

Compared to what?

Why does software cost so much?

What have we done to make it possible for today‘s software to cost so little?

History of the software cost estimation industry

- Before 1970: rules of thumb, simple algorithms
- Around 1970: Idea of automated tools
- Around 1975: Function points
- 1977: PRICE-S, first commercial tool
- 1979: SLIM, second commercial tool
- 1981: COCOMO

History of the software cost estimation industry

- 1981: paper on function points published world wide
- 1982: deMarco function points
- 1983: Mark II function points
- 1984: revision of IBM‘s function point metric
- 1985: SPQR/20
- 1986: IFPUG founded in Toronto, Canada

History of the software cost estimation industry

- Until today: A lot of new commercial tools have been released

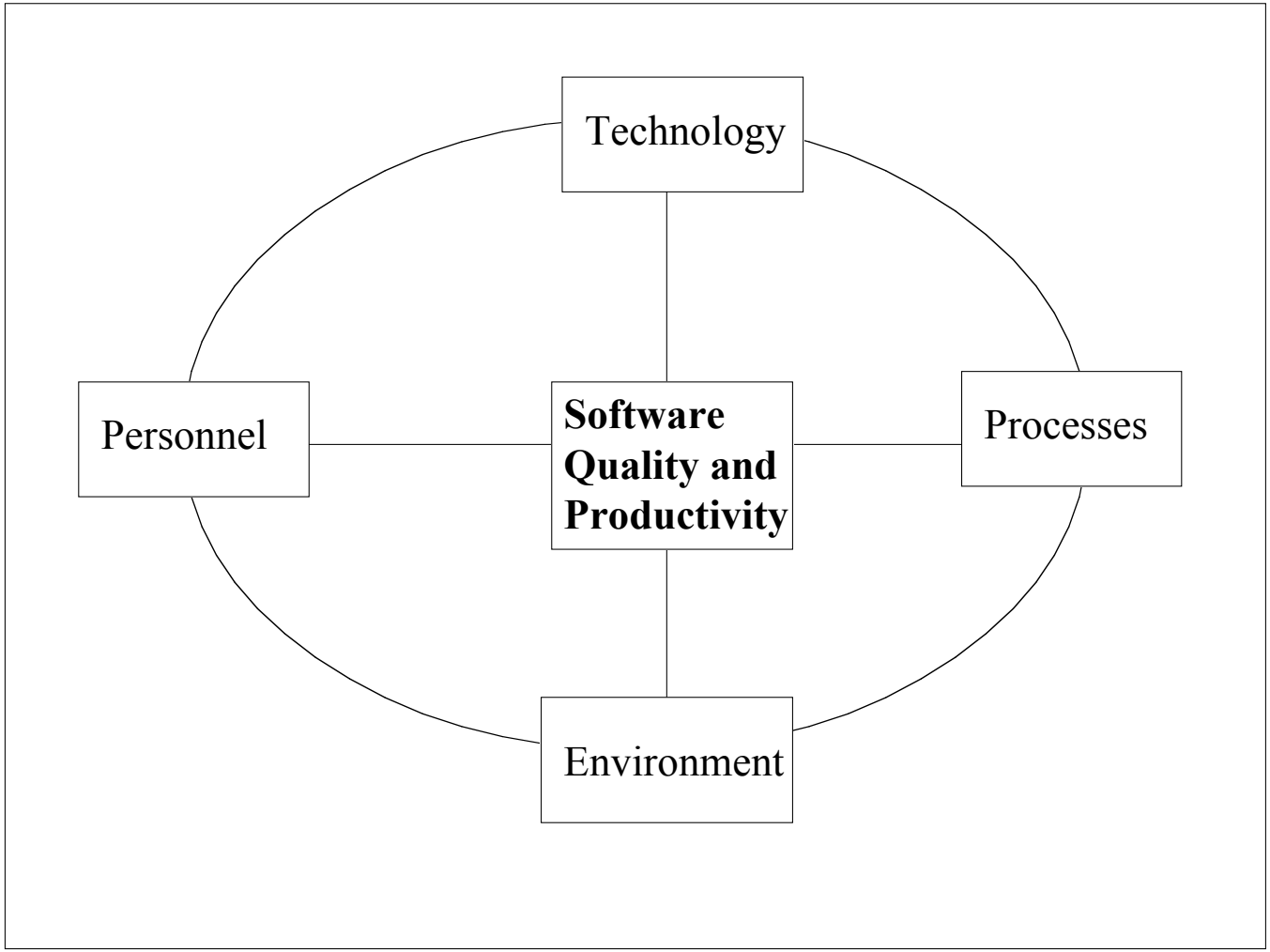
Time and purpose of estimates

- When tendering for a project
- During development process

General problems of software cost estimation

- Changing requirements
- Output of developers
- Underestimation of non-coding work

Key factors


Technology
Personnel
Software Quality and Productivity
Processes
Environment

Objectives of the estimate

- Size of all deliverables
- Staff needed
- Schedule
- Effort
- Costs to develop
- Costs for maintenance and enhancement
- Quality
- Reliability

Estimation steps as proposed by Jones

- Analyze the requirements
- Start with sizing
- Identify the activities to be included
- Estimate software defect potentials and removal methods
- Estimate staffing requirements
- Adjust assumptions based on capabilities and experience
- Estimate effort and schedules
- Estimate development costs
- Estimate maintenance and enhancement costs

Estimation steps as proposed by Boehm

- Establish objectives
- Plan for required data and resources
- Pin down software requirements
- Work out as much detail as feasible
- Use several independent techniques and sources
- Compare and iterate estimates
- Followup

Reasons for underestimating

- Fantasy factor
- Omitting some cost factors

Discussion