

Objectives for Today

Build an understanding of...

- Importance of estimations
- Different estimation approaches (initial situation, expectations, top-down versus bottom-up...)
- Advantages and disadvantages of different approaches
- Common pitfalls



Importance of Estimations

- During the planning phase of a project, a first guess about cost and time is necessary
 - Estimations are often the basis for the decision to start a project
 - Estimations are the foundation for project planning and for further actions
- Estimating is one of the core tasks of project management, but still considered as black magic !



Challenges

- Incomplete knowledge about:
 - Project scope and changes
 - Prospective resources and staffing
 - Technical and organizational environment
 - Infrastructure
 - Feasibility of functional requirements
- Comparability of projects in case of new or changing technologies, staff, methodologies
- Learning curve problem
- Different expectations towards project manager.



Problems with Estimations

- Estimation results (effort and time) are almost always too high (for political / human reasons) and have to be adjusted in a structured and careful manner
- Reviews by experts always necessary
- New technologies can make new parameters necessary
- Depending on the situation, multiple methods are to be used in combination.



Guiding Principles

- Documentation of assumptions about
 - Estimation methodology
 - Project scope, staffing, technology
- Definition of estimation accuracy
- Increasing accuracy with project phases
 - Example: Better estimation for implementation phase after object design is finished
- Reviews by experienced colleagues



Components of an Estimation

This lecture

- **Cost**
 - Personnel (in person days or valued in personnel cost)
 - **Person day**: Effort of one person per working day
 - Material (PCs, software, tools etc.)
 - Extra costs (travel expenses)
- **Development Time**
 - Project duration
 - Dependencies
- **Infrastructure**
 - Rooms, technical infrastructure, especially in offshore scenarios

Lecture on Scheduling.



Estimating Development Time

Development time often estimated by formula

$$\text{Duration} = \text{Effort} / \text{People}$$

Problem with formula, because:

- A larger project team increases communication complexity which usually reduces productivity
- Therefore it is not possible to reduce **duration** arbitrarily by adding more **people** to a project
- In the lectures on organization and scheduling we take a more detailed look at this issue.



Estimating Personnel Cost

- Personnel type: Team leader, application domain expert, analyst, designer, programmer, tester...
- **Cost rate**: Cost per person per day
- 2 alternatives for cost rate:
 - Single cost rate for all types (no differentiation necessary)
 - Assign different cost rates to different personnel types based on experience, qualification and skills
- **Personnel cost**: person days x cost rate.



Estimating Effort

- Most difficult part during project planning
 - Many planning tasks (especially project schedule) depend on determination of effort
- Basic principle:
 - Select an estimation model (or build one first)
 - Evaluate known information: size and project data, resources, software process, system components
 - Feed this information as parametric input data into the model
 - Model converts the input into estimates: effort, schedule, performance, cycle time.



Basic Use of Estimation Models



Examples:

Data Input

Size & Project Data

System Model

Software Process

Estimate

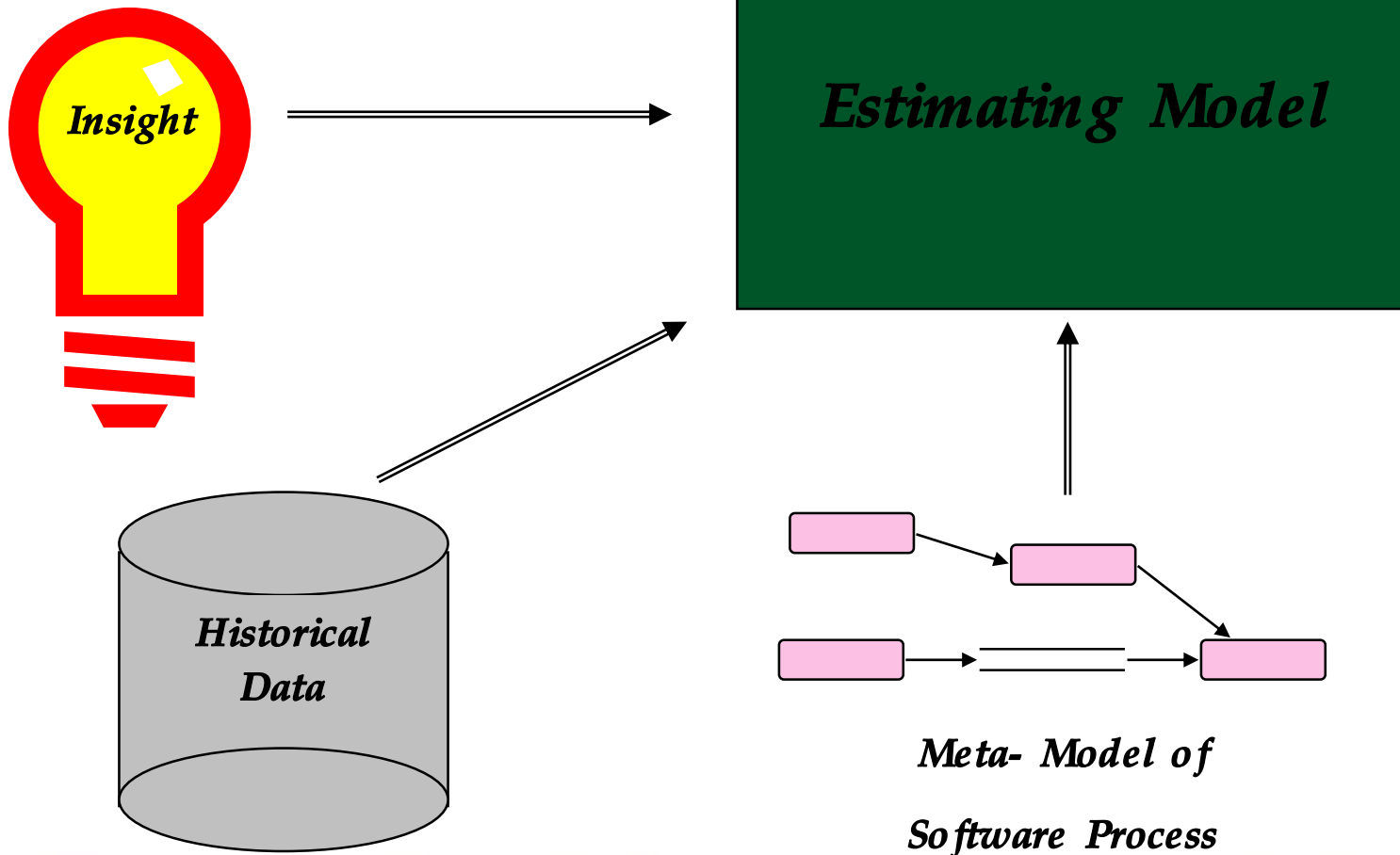
Effort & Schedule

Performance

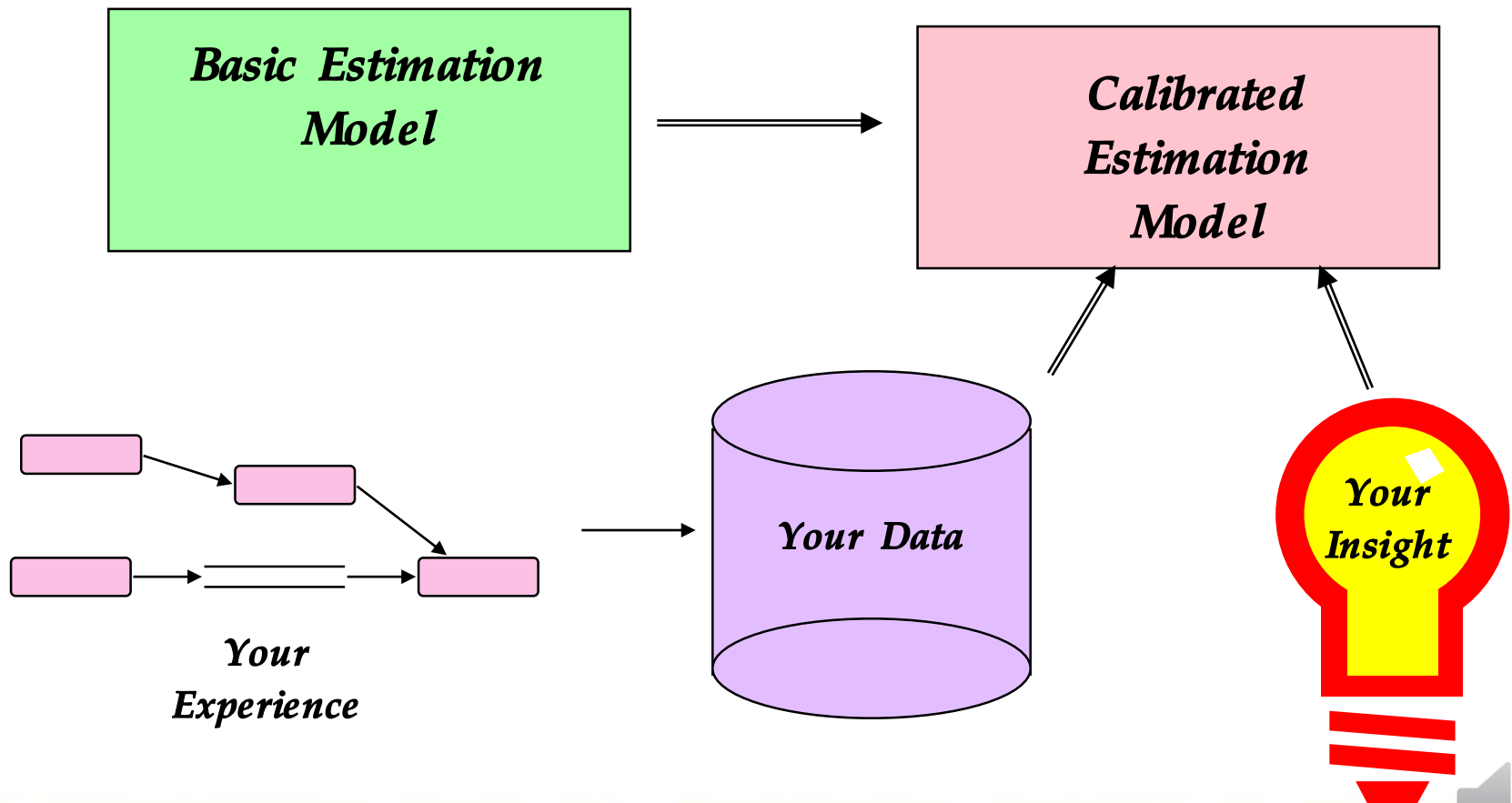
Cycle Time



How do you Build an Estimating Model?



Calibrating an Estimation Model



Top-Down and Bottom-Up Estimation

- Two common approaches for estimations
 - Top-Down Approach
 - Estimate effort for the whole project
 - Breakdown to different project phases and work products
 - Bottom-Up Approach
 - Start with effort estimates for tasks on the lowest possible level
 - Aggregate the estimates until top activities are reached.



Top-Down versus Bottom-Up (cont'd)

- Top-Down Approach
 - Normally used in the planning phase when little information is available how to solve the problem
 - Based on experiences from similar projects
 - Not appropriate for project controlling (too high-level)
 - Risk add-ons usual
- Bottom-Up Approach
 - Normally used after activities are broken down the task level and estimates for the tasks are available
 - Result can be used for project controlling (detailed level)
 - Smaller risk add-ons
- Often a mixed approach with recurring estimation cycles is used.



Estimation Techniques

- Expert estimates
- Lines of code
- Function point analysis
- COCOMO I
- COCOMO II



Expert Estimates

= Guess from experienced people

- No better than the participants
- Suitable for atypical projects
- Result justification difficult
- Important when no detailed estimation can be done (due to lacking information about scope)



Lines of Code

- Traditional way for estimating application size
- Advantage: Easy to do
- Disadvantages:
 - Focus on developer's point of view
 - No standard definition for "Line of Code"
 - "You get what you measure": If the number of lines of code is the primary measure of productivity, programmers ignore opportunities of reuse
 - Multi-language environments: Hard to compare mixed language projects with single language projects

"The use of lines of code metrics for productivity should be regarded as professional malpractice"
(Caspers Jones)

