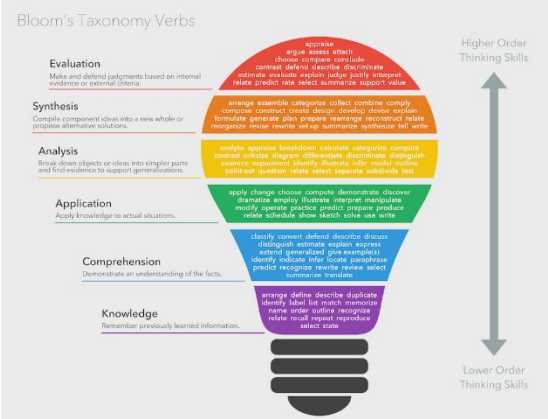
Master Software Technology

Software Project Management 2 – [05] Project Phases: Planning

Learning Goals

Bloom's Taxonomy Verbs
by Fractus Learning,
License: CC-BY-SA 4.0



- ✓ Understand estimation and the planning process
  - ✓ Use network planning techniques, esp. CPM, on examples
- ✓ Work with standard project planning tools on examples
  - ✓ e.g. ProjectLibre as used in the lab

Agenda

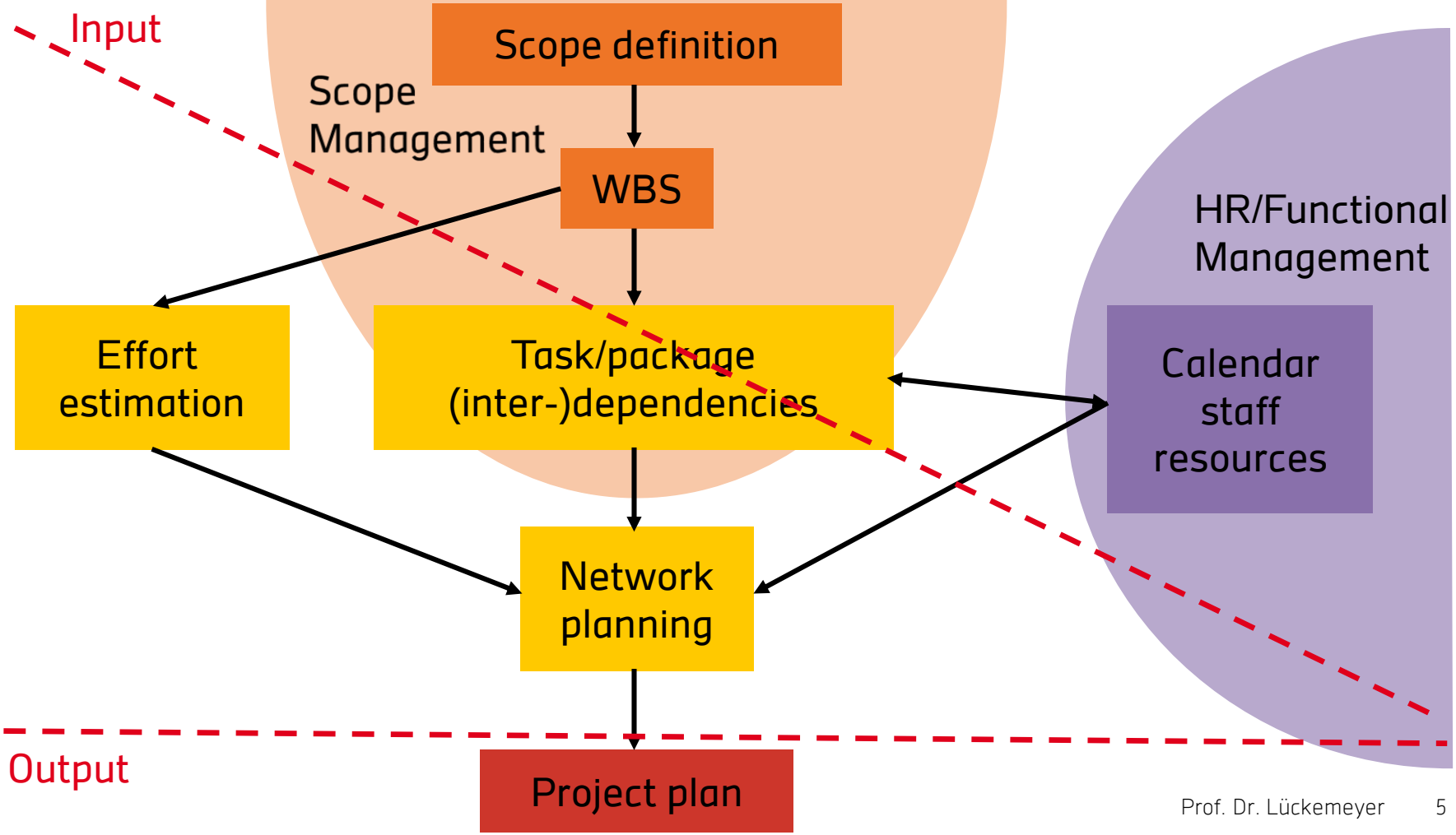
Project Planning
- Estimation
  - requirements w.r.t. estimates
  - ➢ *Are you already familiar with any estimation methods?*
  - distinction between size, effort and cost
  - categorization of different estimation methods
- Network planning techniques -> further topic
- Common planning pitfalls

Conclusions

Project Planning: Objectives

- basis for all future control and management activities during the project
- should be "stable" – avoid fundamental changes of the project organisation as far as possible
- achieving a better understanding of project objectives
- more efficient project execution
- avoid/minimize any redundancies

Project Planning: Process



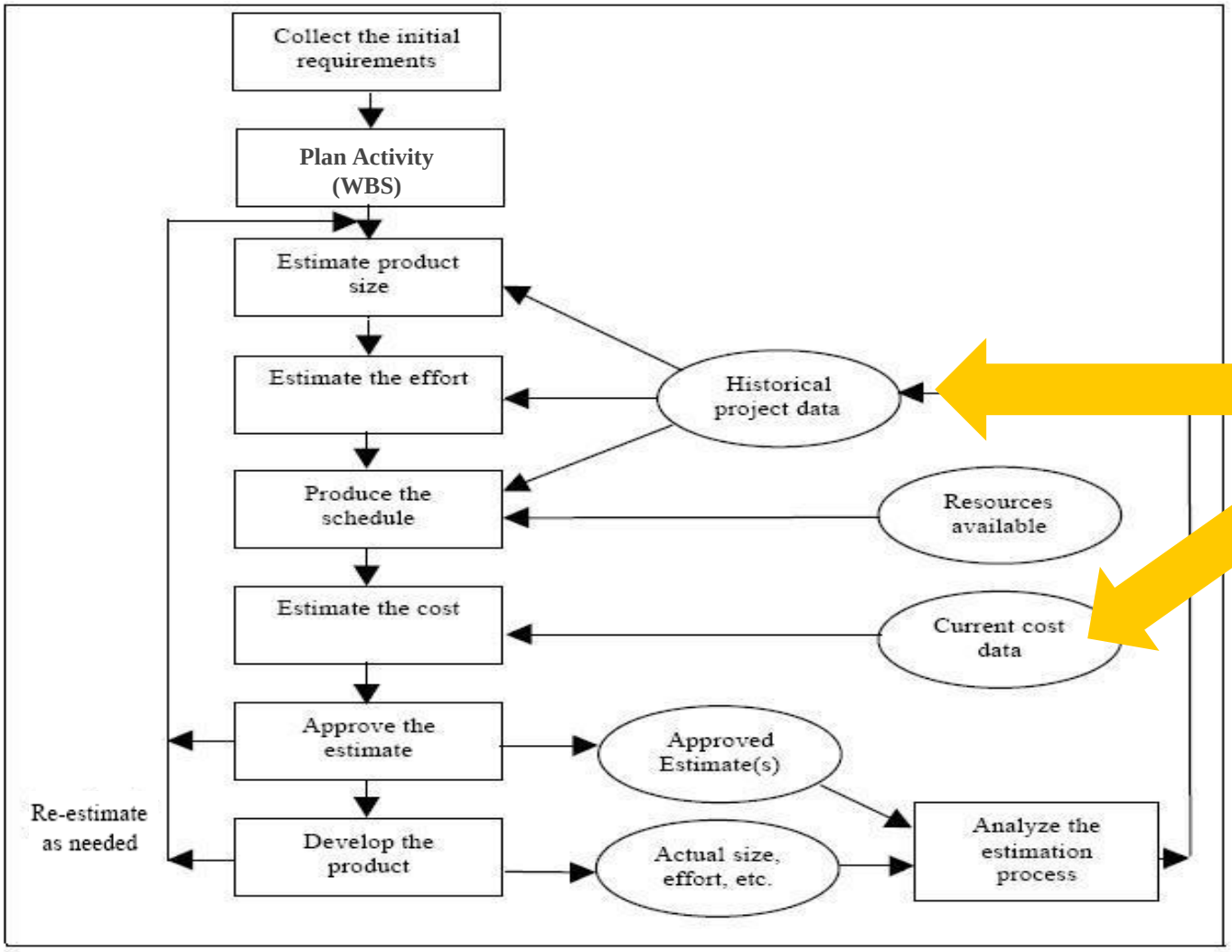
Agile Planning

Plan only the next iteration.

1. First determine relative sizes of stories in Story Points (see Planning Poker method later)
2. Estimate team velocity (story points per iteration)
3. Derive duration
4. Lay out a Release Plan
5. Calibrate plan by doing the planned work
6. Adjust the plan with feedback from measured velocity
7. Regularly revise the plan as you learn more

Source: https://wingman-sw.com/slides/Beyond-Planning-Poker-v1r1.key.pdf

Estimation: Process



Experience advantage

Estimates: Requirements (& Wishes)

traceable
- fundamental requirement to achieve this:
  - systematic approach (i.e., a method that is used)

based on quantitative figures from the past,
i.e., based on experience from previous projects
- How do we cope with "new" projects?

estimates done by those who are responsible to carry out the respective task in the course of the project (experts)
- in case this cannot be achieved: review of any estimates done by others than those who actually carry out the respective task

document all estimates – not just the pure figures
- esp. explanations of any assumptions
- document and keep track of any risks discovered
  - initial input for risk management

Estimates: figures/size, effort and cost

estimating figures and sizes
- Lines of Code LoC,
- number of modules, frames (on screen), database fields, ...
- difficulty, complexity, …

estimating efforts
- effort for work packages (and tasks, sub tasks, …, resp.)
  - in time units, e.g., person days [PD = full time equivalents (FTE)], person months, or person years

estimating costs
- effort x costs per time unit
- plus any additional costs, e.g., travel costs, project specific hardware or software (development environment, database system,), training courses, ...

Estimation Methods

(1) analogy
   a) multiplication methods: linear correlation between 2 figures, e.g., LoC – person days
   b) percentage method: see next slide
(2) expert methods
   a) Delphi Method
   b) informal expert estimations 
   c) three point estimates: see subsequent slide
(3) further advanced methods
   a) Constructive Cost Model CoCoMo: brief intro on subsequent slide
   b) Function Point

See student paper/presentation for more…

Estimation Methods: (1) Analogy: percentage method

Assigning efforts proportional to project phases based on previous projects

Estimates for any new project: „forecast"

- based on estimating – at least – one phase of that project or
- based on any phases that have already been accomplished

Example

project phase	"historical" percentage	new project [person days]
Project management	5%	8,33
design	15%	25,00
detailed spec´s	30%	50,00
implementation and test	30%	50,00
installation & on-site test	20%	33,33
total	**100%**	**166,67**

Estimation Methods: (2) Expert methods: three point method

take into account any "fuzziness" during estimation
three estimates
- optimistic: ideal, lowest estimate
- „realistic": most probable „average" value
  (but not – necessarily – the arithmetic average!)
- pessimistic: worst case, highest estimate

How can these estimates be further used?
- separate calculations for all cases
  - huge effort ☹
- weighted average according to PERT:
  (opt + 4 real + pess)/6
- go-/no-go decision for tendering
  - Especially consider the pessimistic estimation with respect to potential risk
- ...

Estimation Methods: (3) Advanced Method: Constructive Cost Model

"COCOMO II is a model that allows one to estimate the cost, effort, and schedule when planning a new software development activity. It consists of three submodels, each one offering increased fidelity the further along one is in the project planning and design process. Listed in increasing fidelity, these sub models are called the Applications Composition, Early Design, and Post-architecture models."

http://sunset.usc.edu/csse/research/COCOMOII/cocomo_main.html

created in 1981, Barry Boehm:
http://sunset.usc.edu/Research_Group/barry.html
apparently, no/very few updates since 2000
brief characteristics

- based on regression analysis
- starting point: analyzing 63 software projects
- open model → can incorporate almost any effects by additional parameters ☺ ☹

Agile Estimation: Planning Poker

Each player (developer) has a hand of planning poker/blank note cards.
- Cards contain numbers fine grained low, coarse grained large numbers (e.g 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, ?, infinity)

Customer (a.k.a. Product Owner) reads a story.
Process similar to Delphi Method: until estimates converge
– Developers discuss to make sure they understand the story, not how they would build it.
– Each secretly chooses their estimate for the relative effort (no times).
– All expose their estimate simultaneously.
– Discuss extremes, re-deal if needed.

No need to discuss how, if all give the same estimate
→ Allows faster estimation.

Source: https://wingman-sw.com/slides/Beyond-Planning-Poker-v1r1.key.pdf

Estimation: common problems

- too optimistic
- inherently insecure,
  esp. when starting a project
- Benefit from experience, but projects bear novelty by definition
- team members with different, individual background
  - different estimates!
  - Which reflect different actual activity completion times!
    - Importance of staff management!
    - Value the differences!

Project Planning: Steps

deriving activities from the work break down structure
additionally taking into account:

- dependencies among steps (tasks)
- efforts (input from the previous estimation step)
- assigning resources, esp. staff
- using milestones and deliverables from the WBS

calculating starting and finishing dates

- for the complete project
- for individual steps (tasks)
- iterative calculation
  - multiple iterations, if necessary

Project Planning: Dependency Types

must vs. "may be":
- mandatory:
  - based on dependencies among activities
  - e.g.: <your example>
- random:
  - should be taken into account, e.g., <give an example>

internal (i.e., *within* the project) and external relationships:
- internal: everything related to the project
- external
  - relationships to external activities, e.g.: <give an example>

Project Planning: Relationship Types

end-begin relationship
- ✓ most common case
- – predecessor completed before successor starts
- – e.g.: design – implementation

end-end-relationship
- – predecessor completed before successor completed
- – e.g.: test completed before QA completed

beginning-beginning relationship
- – predecessor starts before successor starts
- – e.g.: implementation has already started before writing documentation starts

beginning-end relationship
- – rarely used
- – definition of maximal distance between activities

Project Planning: Staff Assignment

Where are we in the overall process?
- known so far: effort estimations for all activities
- question now: who actually does the job?

➢assigning resources to tasks

taking into account:
- staff qualification
- "common sense"/experience
  - putting more people in a late project makes it even later
  - basic reason: overhead for coordination and introduction
- productive times: depends on company and activity
  - 15+- days (between 13 and 17) chargeable days per month
  - vacation, training courses, any non-project activities, reading and answering emails, ...

Project Planning: Responsibility Planning with RACI Charts

lead
- final responsibility – single (!) person
- bear in mind: others help/contribute, of course!
  - see contributor below

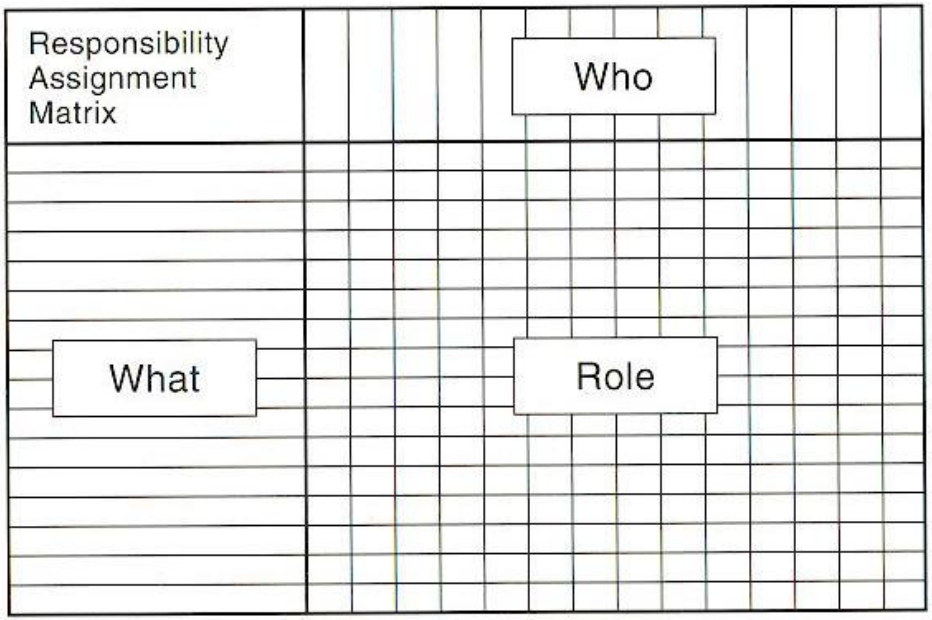
approval
- approves the item as complete

secondary
- backup for the lead person

contributor
- contributing to achieving the item

reviewer
- reviewer duties

(none)
- no participation expected



Project Planning: Resource Leveling

resources evenly distributed

try to avoid any considerable fluctuations

– substantial effort for orientation for any staff member when starting a new task!

try to flatten the histogram (workload <= 100%)

– Usually leads to prolongations → creative re-planning needed

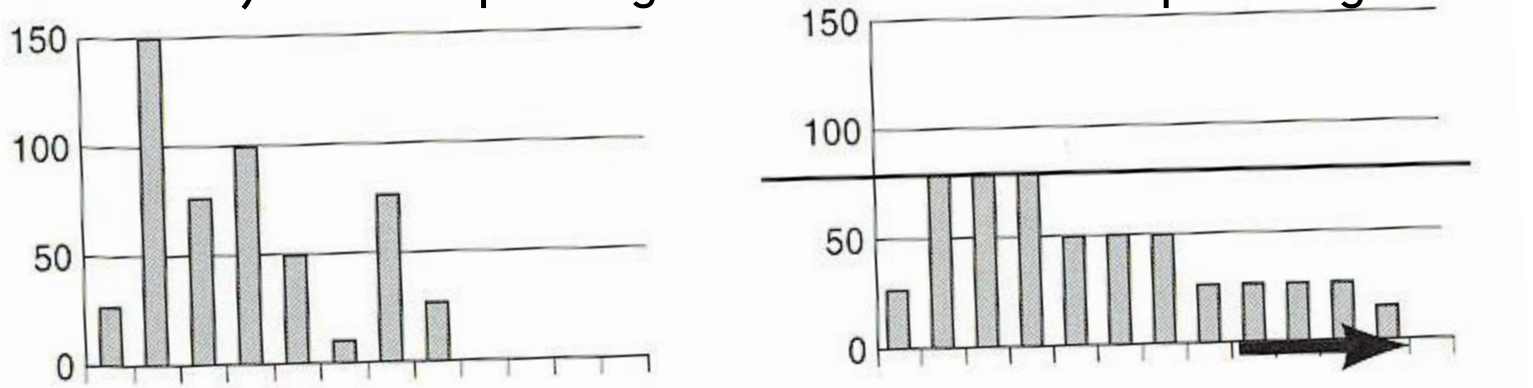


FIGURE 12–7
Resource Leveling

Project Planning: Network Planning Methods

➢rather a *very brief* introduction to Operations Research
survey of methods

- stochastic structures
  - more than one way through the network is possible
  - example: <you name it?!>
- deterministic structure
  - *all* paths through the network are used during the project
  - in the sequel, we will concentrate on deterministic networks ☺

networks comprise

- tasks: require time for execution
- events: state, does not require any time (e.g., milestones!)
- relationships: dependencies

Network Planning Example: Critical Path Method (CPM)

Graphical method portraying relationship of project activities

History

- Developed by DuPont (1950's)
- Goal: plan and control maintenance of chemical plants
- credited with reducing length of maintenance shutdown by 40%

Activity: any discrete part or task of a project which takes resources and time to complete

Activities exhibit precedence relations

- some must be completed before others can start

Activities with precedence relations form a project network

CPM finds the longest path through a project network

- Its length determines the minimum duration of the project if all dependencies on the path are strict

CPM: Terminology

Critical Path: the chain of activities along which the delay of any activity will delay the project

Early Start Time (EST): the earliest that an activity could possibly start, given precedence relations

Late Start Time (LST): the latest that an activity could possibly start without delaying the project

Early Finish Time (EFT): the earliest that an activity could possibly finish

Late Finish Time (LFT): the latest that an activity could possibly finish without delaying the project

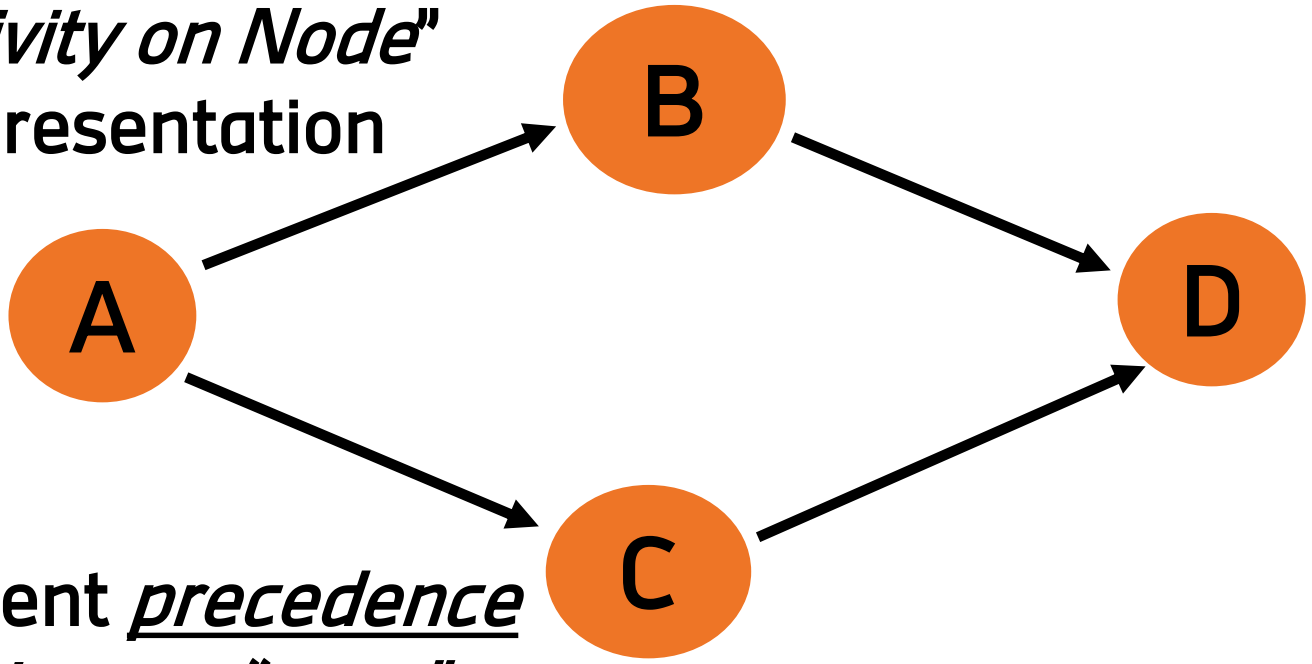
Activity Slack: the amount of "play" in the timing of the activity; **slack (i.e. time buffer) = LST-EST = LFT-EFT**

CPM: Example

Activity	Predecessor	Duration
A	(Start)	4
B	A	3
C	A	5
D	B, C	2

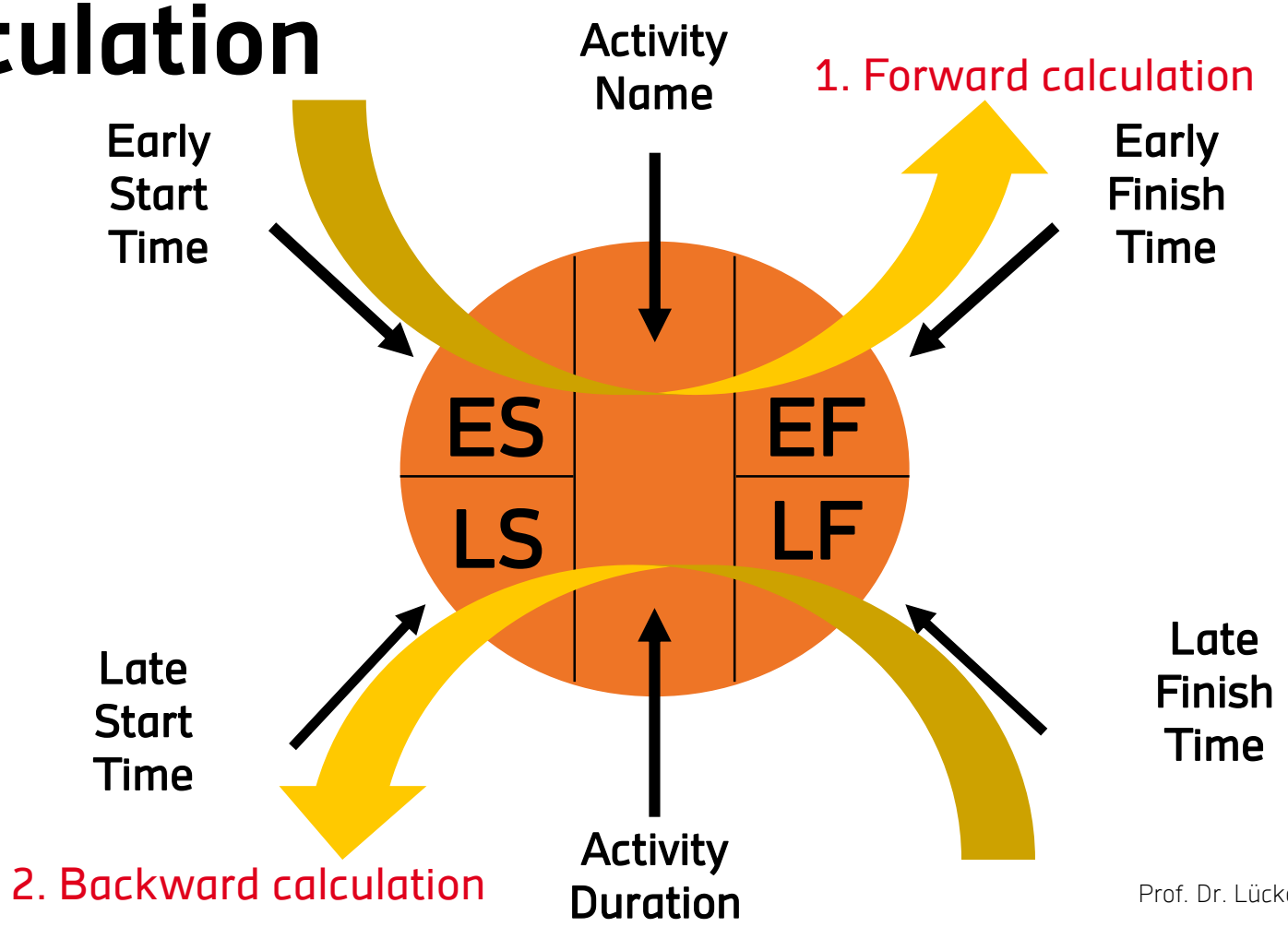
Project Network

"*Activity on Node*" representation

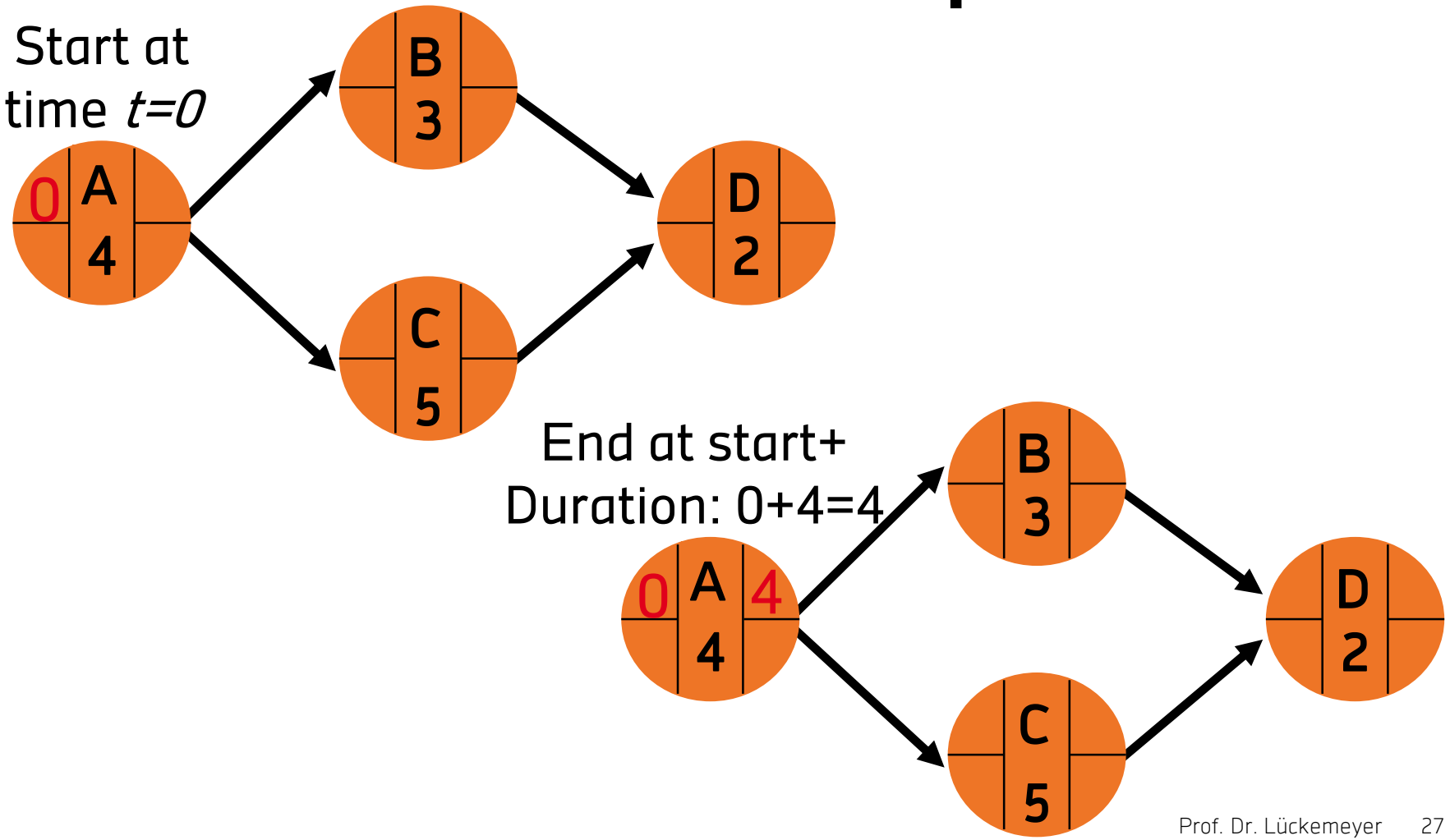


Represent *precedence relations* as "*arcs*"

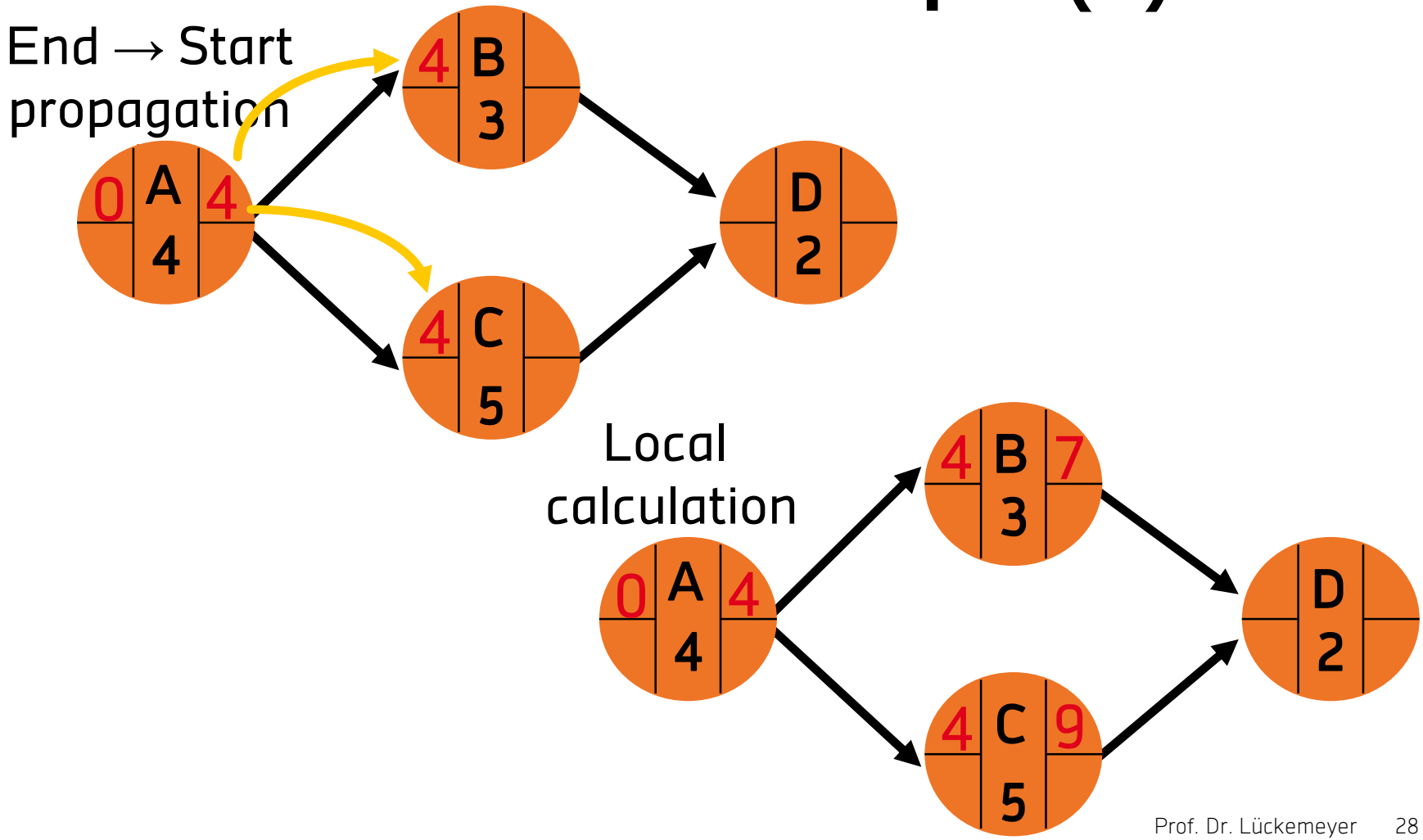
CPM: Activity Start/Finish Time Calculation



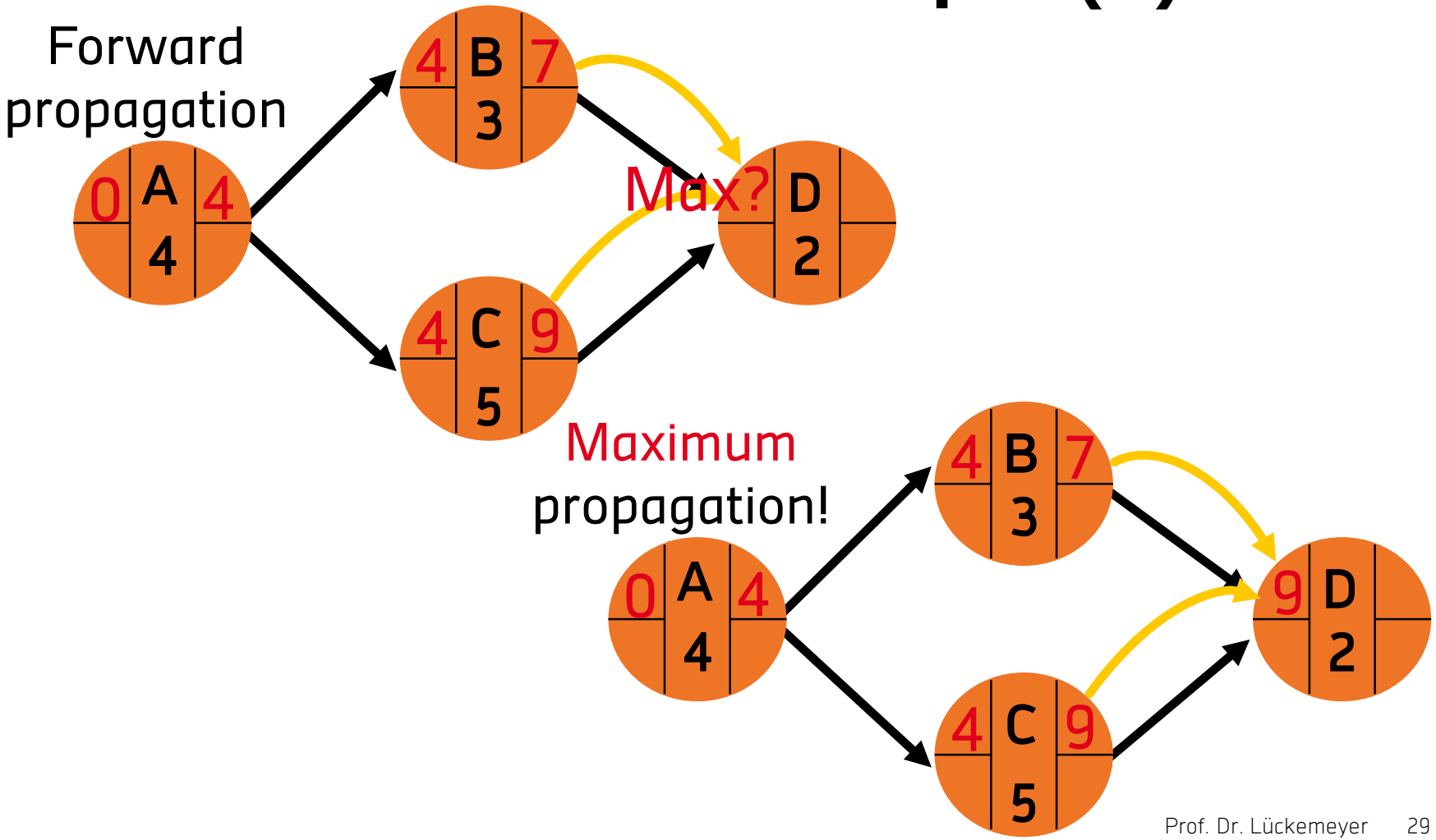
CPM: Calculation Example



CPM: Calculation Example (2)

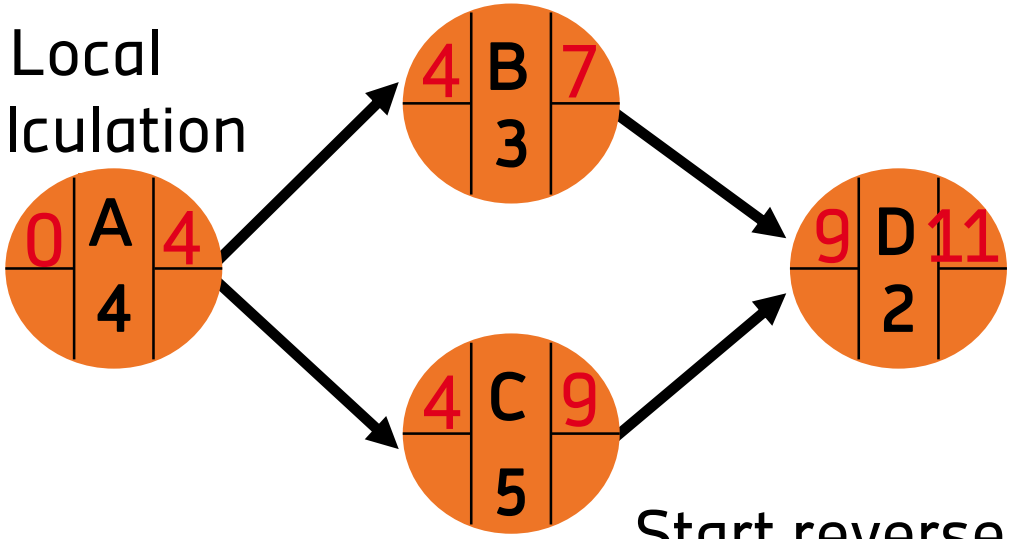


CPM: Calculation Example (3)



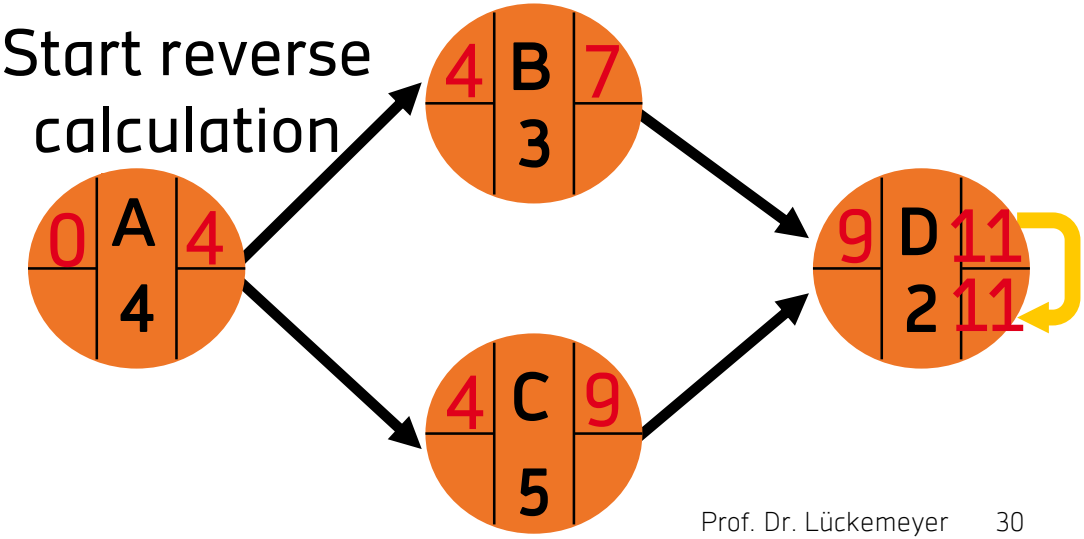
CPM: Calculation Example (4)

Local calculation

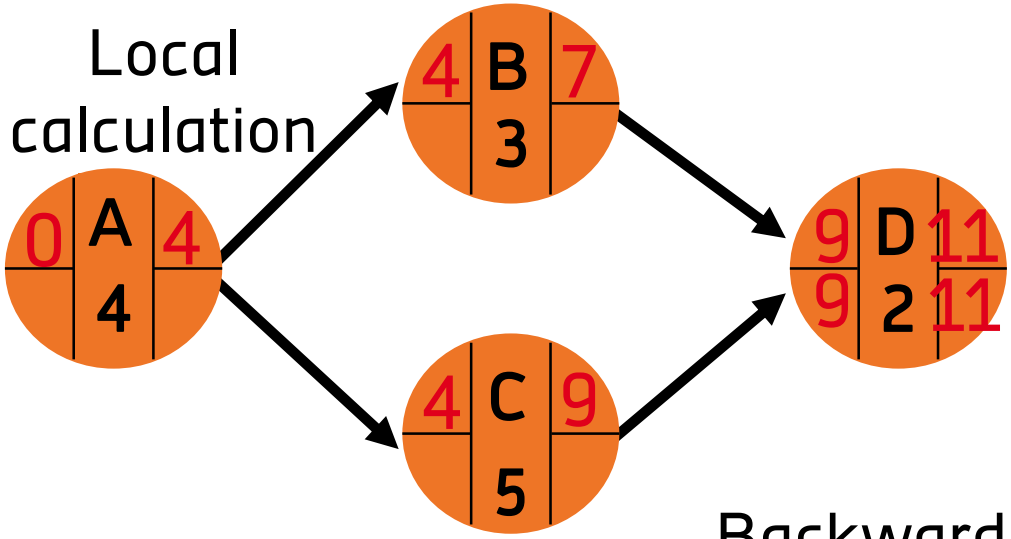


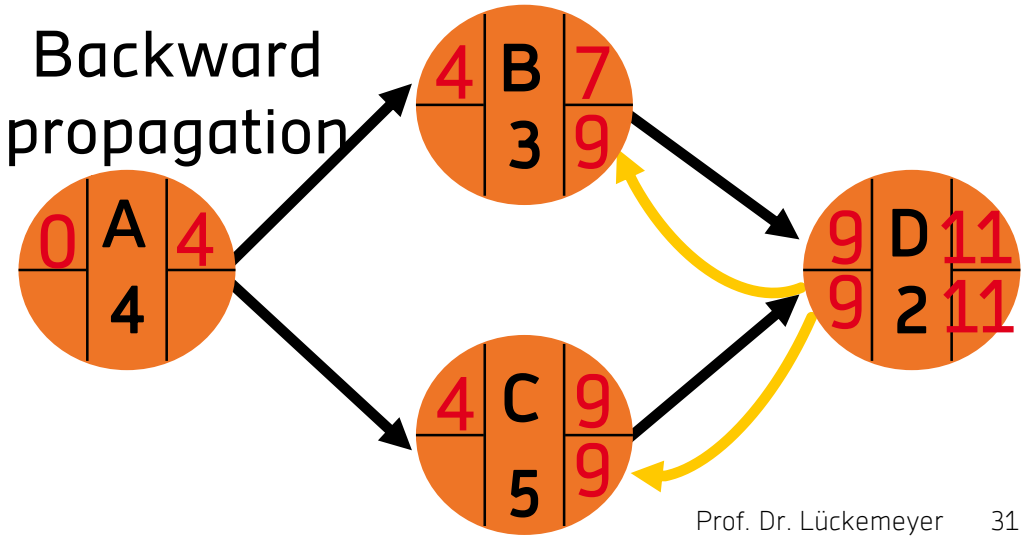
The *earliest* the project can complete is *t =11*

Start reverse calculation

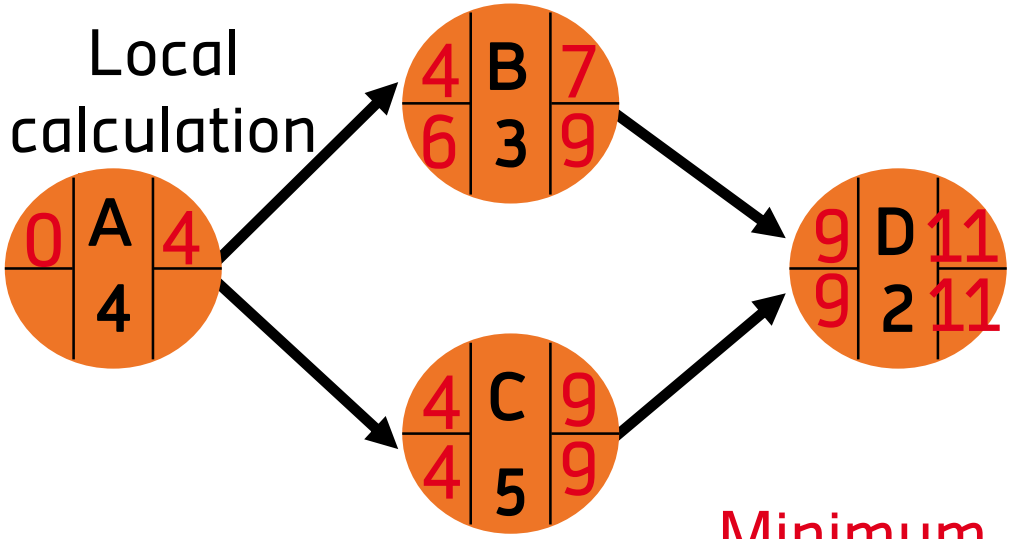


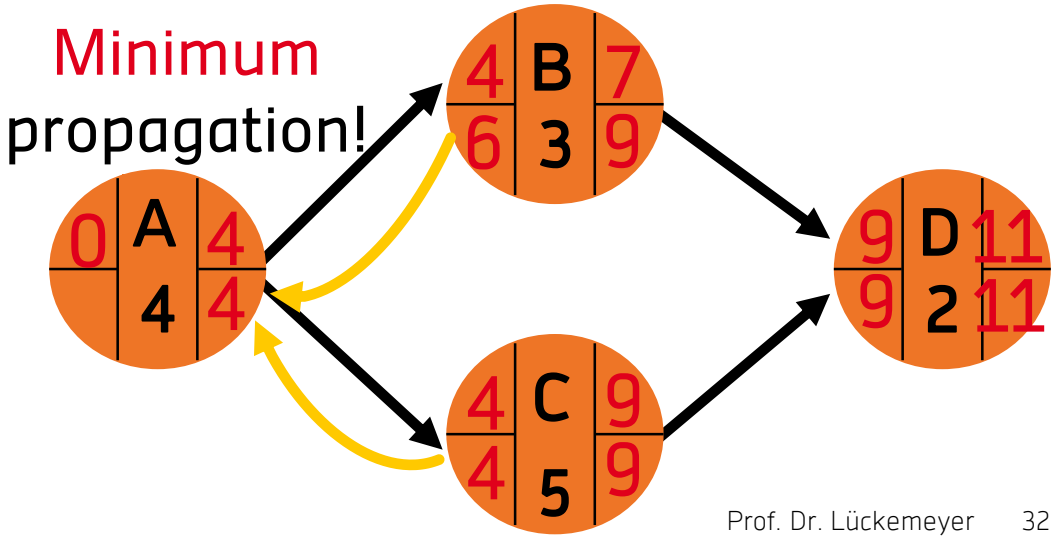
CPM: Calculation Example (5)



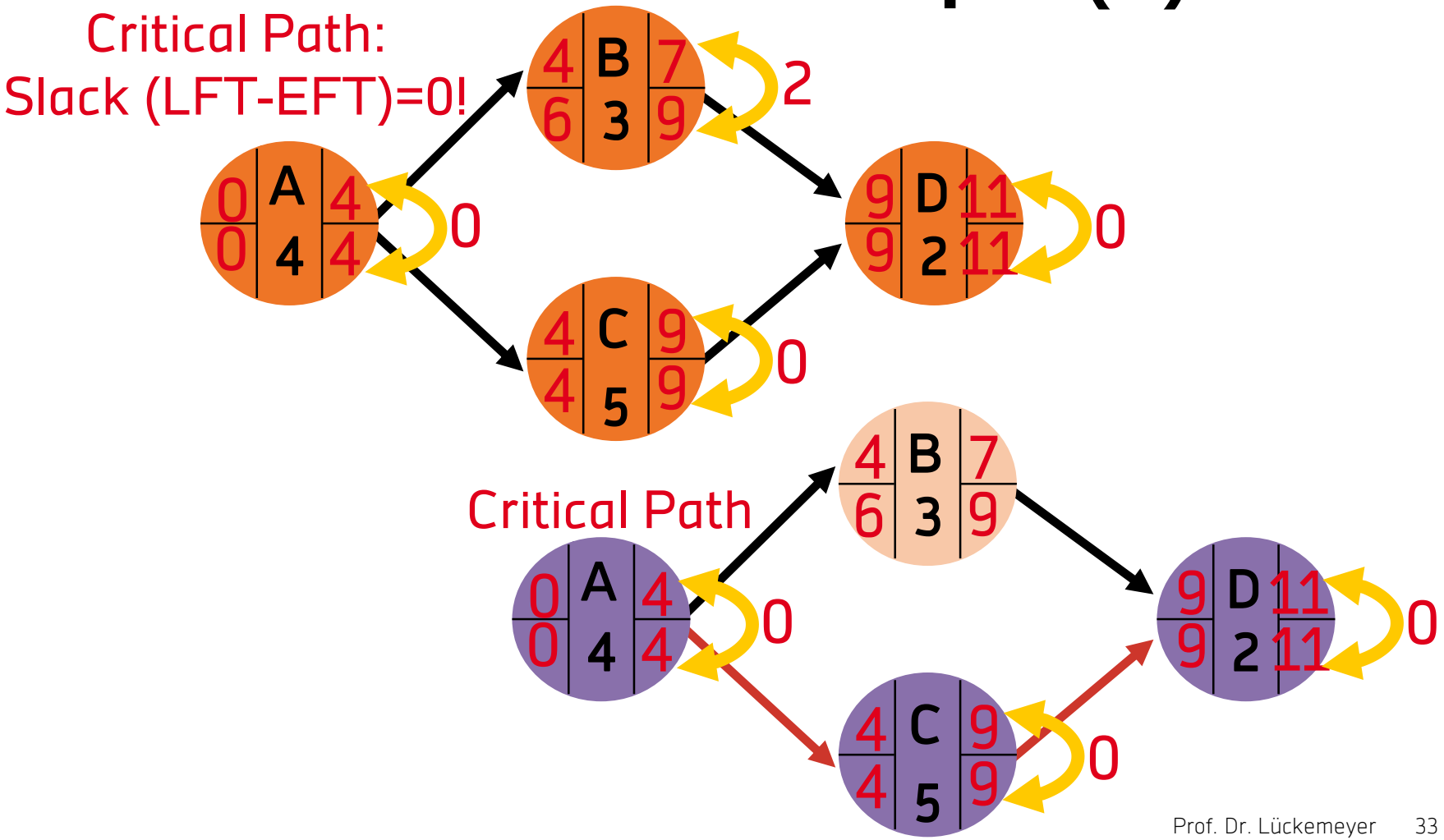


CPM: Calculation Example (6)





CPM: Calculation Example (7)



CPM: Exercise

A project has the following activities:

Activity	Predecessors	Time (d)
A. Analysis mod. alpha	none	10
B. Analysis mod. beta	none	30
C. Design mod. beta	B	15
D. Prototype mod. alpha	A	40
E. Integration mod. gamma	A,C	30
F. Re-implementation mod. delta	B	40
G. Integration	E, F	25
H. On-site test	D,G	10

a) In this example, there are two initial activities A and B which may start instantaneously. How do you cope with this situation?

b) Use CPM to calculate all relevant figures (assuming a single resource is doing all of the tasks) for the individual activities and for the overall project!

c) Figure out the critical path!

Project Planning: Common Pitfalls

- too much in too little time
- plan based on unrealistic deadlines
- management insists on dates that are completely unrealistic
- plans not discussed with all stakeholders
- insufficient data base (lack of information and/or experience) for sound planning
- staffing not appropriately
- estimates too optimistic, not based on any experience
- future tasks cannot be estimated yet

Questions? Questions!

THANK YOU VERY MUCH FOR YOUR ATTENTION!