Master Software Technology
Software Project Management 2 – [05] Project Phases: Controlling & Corrective Actions

Learning Goals

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

✓ Understand standard controlling methods
✓ Apply the standard controlling methods in practical examples
✓ Understand options for corrective actions from within and from outside the project
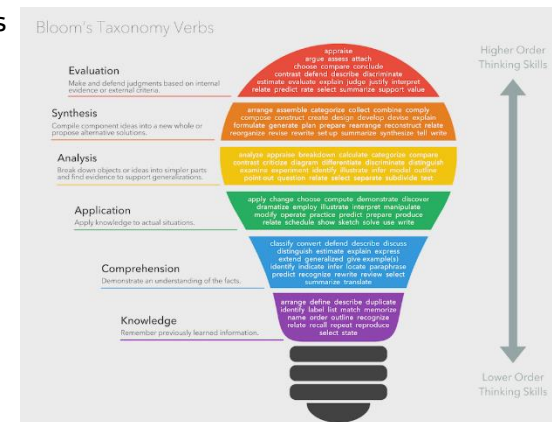
Agenda

Introduction

Methods and Techniques

➢ Milestone Trend Analysis (MTA)

➢ Earned Value Analysis (EVA)

Corrective Actions

– within the project

– from outside the "core" project

Project Controlling: ==Baseline==

Where are we in the overall process?

✓ project has been initially ==planned== (completely)

✓ project has (already) ==started==, but far from being completed

➢ relevant questions now:

– Which are the ==relevant figures (KPIs)== during project execution that we have to collect in order

  – to figure out that something does *not* work out as it is supposed to do and

  – to take corrective actions in time?

– How do we get them?

Project Controlling: <mark>Methods & Techniques</mark>

1. <mark>Controlling Budget</mark>: keep track of the efforts spent
   – very likely: relevant information too late in order to take any corrective actions
   – keeping track *only* of the efforts *not enough*
   – *Gather remaining effort estimations: estimation every time unit!*
2. <mark>Controlling Scope</mark>: determine the degree of completion
   – difficult, if work items are too large!
   – another motivation for sufficiently "small" tasks in the WBS
   – Earned Value Analysis (EVA) and
     Earned Value Management (EVM)
3. <mark>Controlling Time</mark>: Milestone Trend Analysis (MTA)
4. <mark>Agile Controlling</mark>: Burndown Chart
And there are many more…

# **Project Controlling: Milestones**

How to use them:
- set milestone dates in advance
- define start and end of project activities
- detect project delays
- review the project progress and replan milestones if necessary

Used for
- Visible status for all members of the project
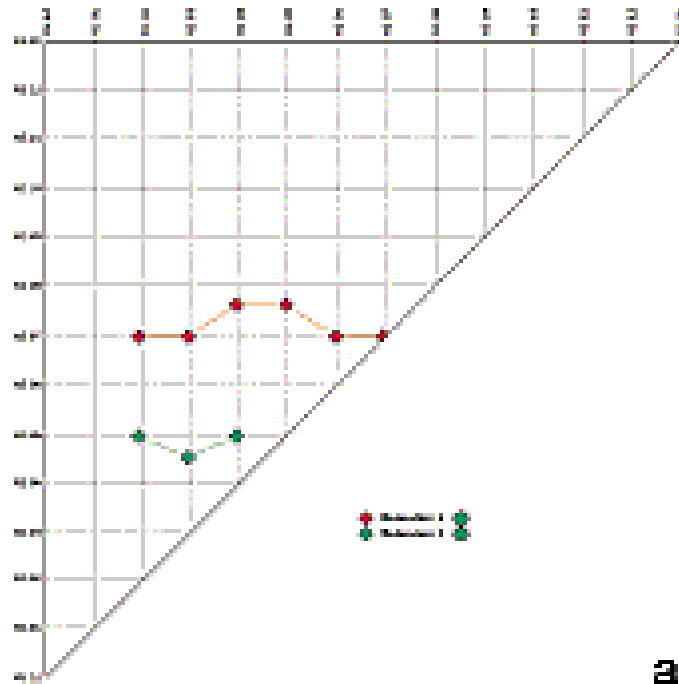- Reporting to the management and customers

Project Controlling Methods:
Milestone Trend Analysis (MTA)

- „time/time diagrams"
- development of schedule dates for several milestones at different reporting dates
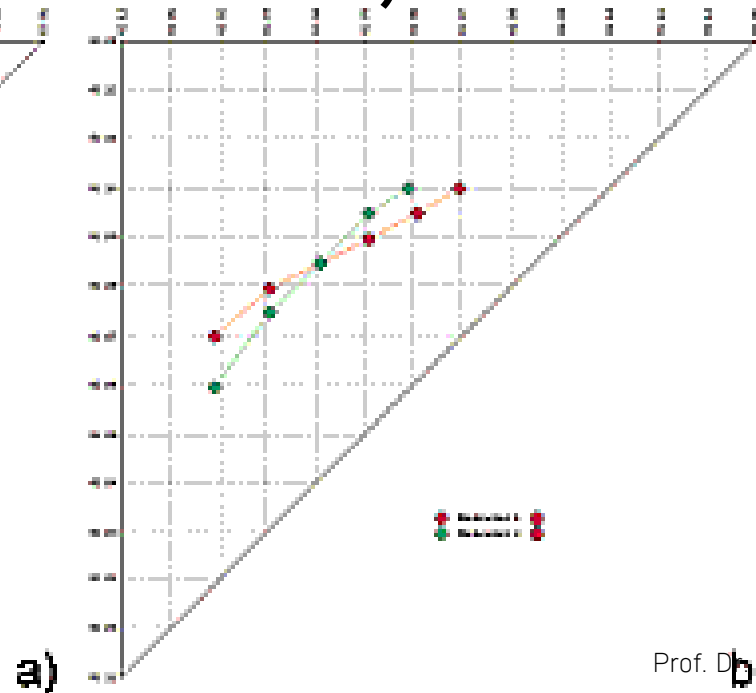- e.g. monthly milestone forecast, also reported to the management
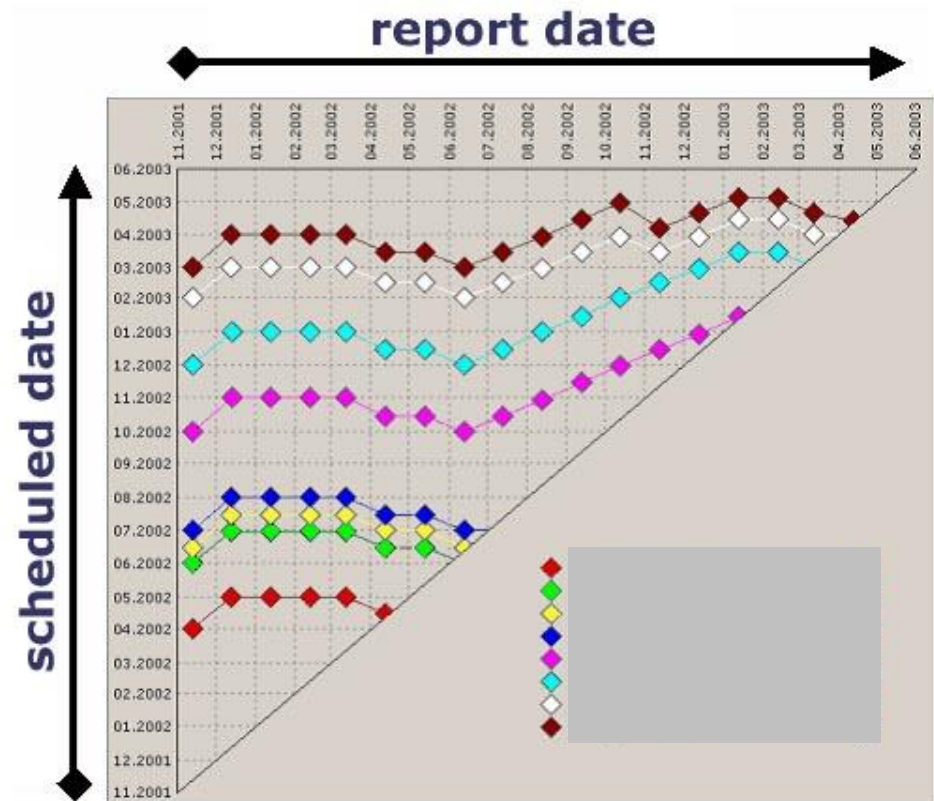
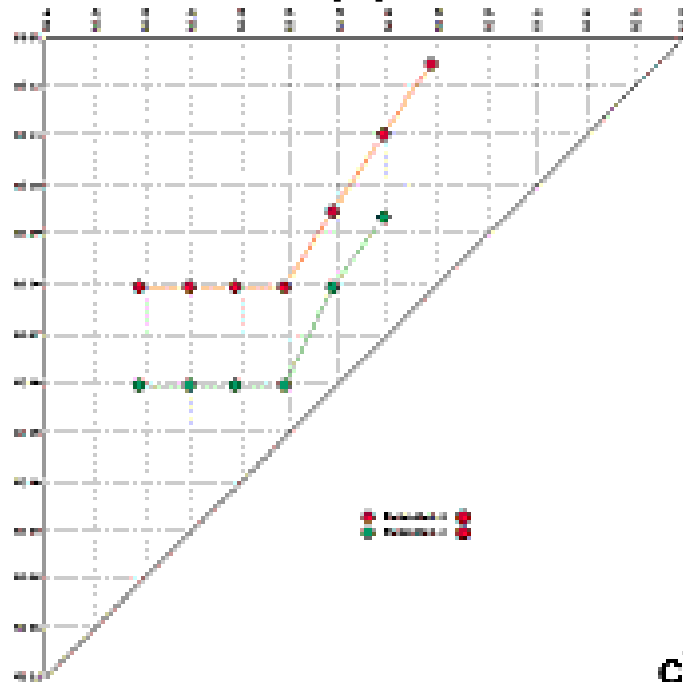# Milestone Trend Analysis (MTA): Examples & Interpretation

a) milestones on time

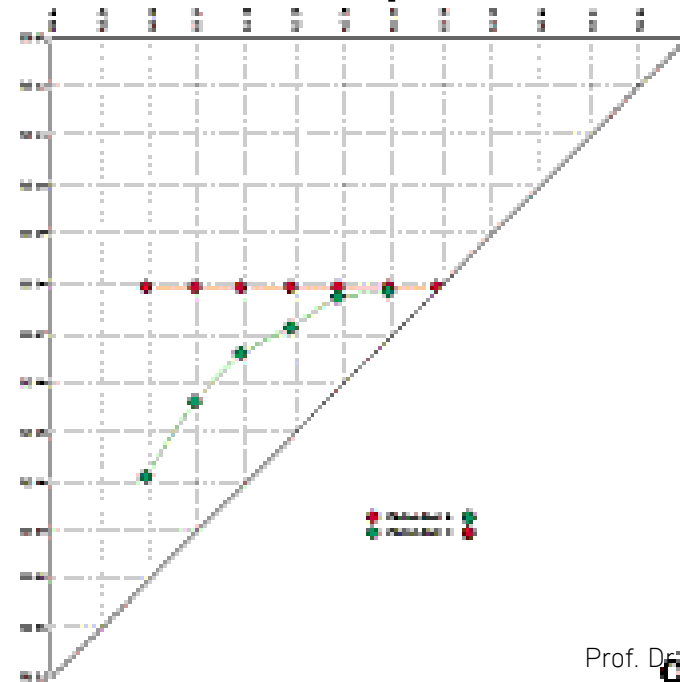b) milestones constantly delayed from the start

Milestone Trend Analysis (MTA): Examples & Interpretation (2)

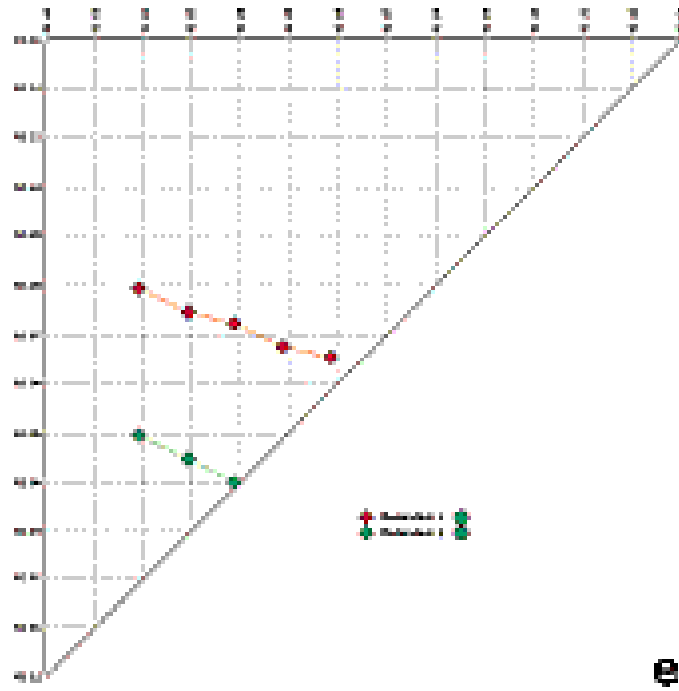c) 80%-syndrome, as deadline approaches

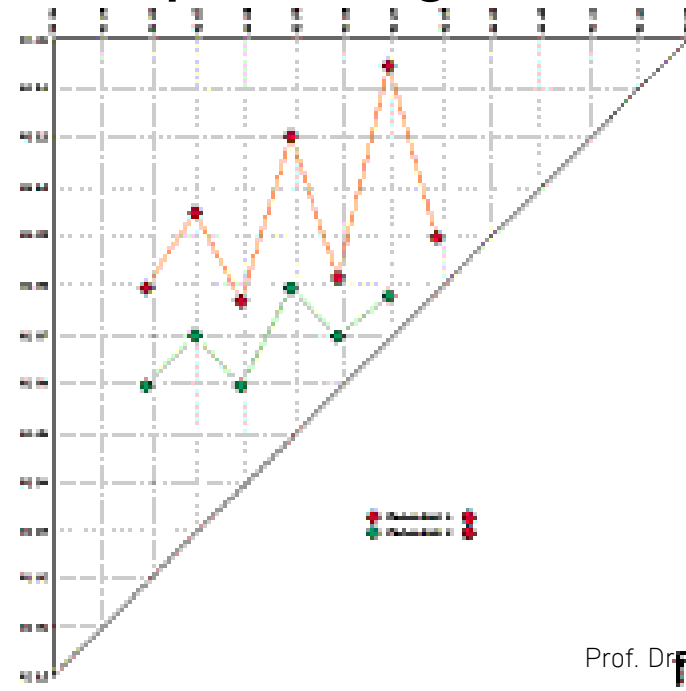d) forced deadline by subsequent milestone

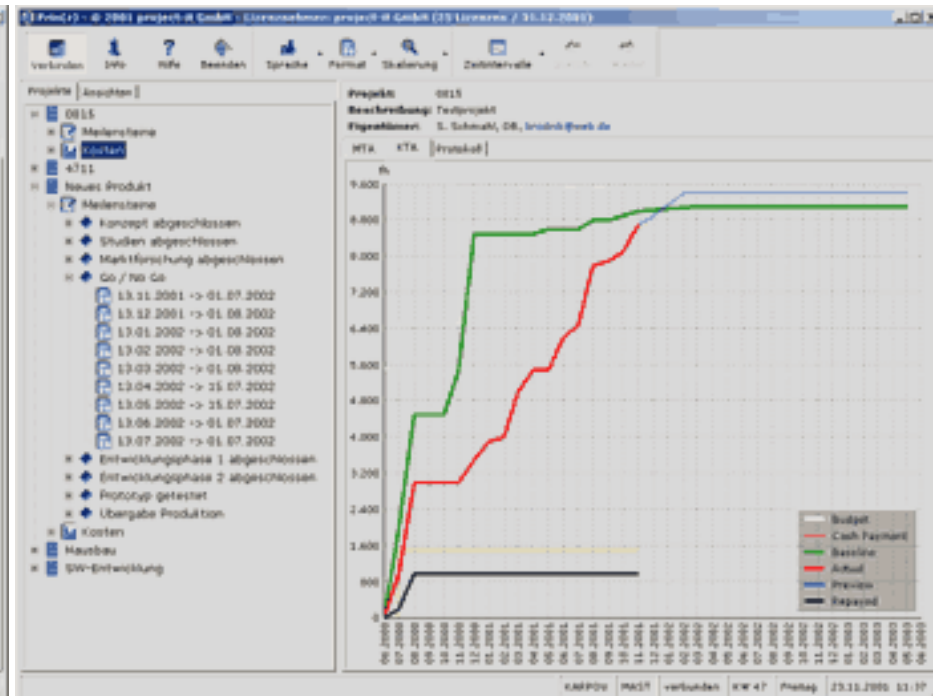# Milestone Trend Analysis (MTA): Examples & Interpretation (2)

e) finished earlier

f) uncertainty in planning

Tools – Prin(z)

Milestone Trend Analysis Cost Trend Analysis

Project Controlling: <mark>Earned Value Analysis (EVA)</mark>

background
- method for project controlling
- compulsory, e.g., for governmental projects in the USA
- well suited as well for large projects (>60 activities & 10% completion)

basic idea
- ✓ <mark>we cannot earn more than we have (originally) planned</mark>
  - although actual costs may be higher!
- ✓ allows to judge project status w.r.t. both *time* and *costs*

Project Controlling: EVA - Terminology

basic figures to be calculated during project execution
– Planned Value: PV
– Actual Cost: AC
– Earned Value: EV

derived figures
– Schedule Variance: SV in €
– Cost Variance: CV in €
– Cost Performance Index: CPI in %
– Schedule Performance Index: SPI in %

highly aggregated derived figure
– Critical Ratio: CR
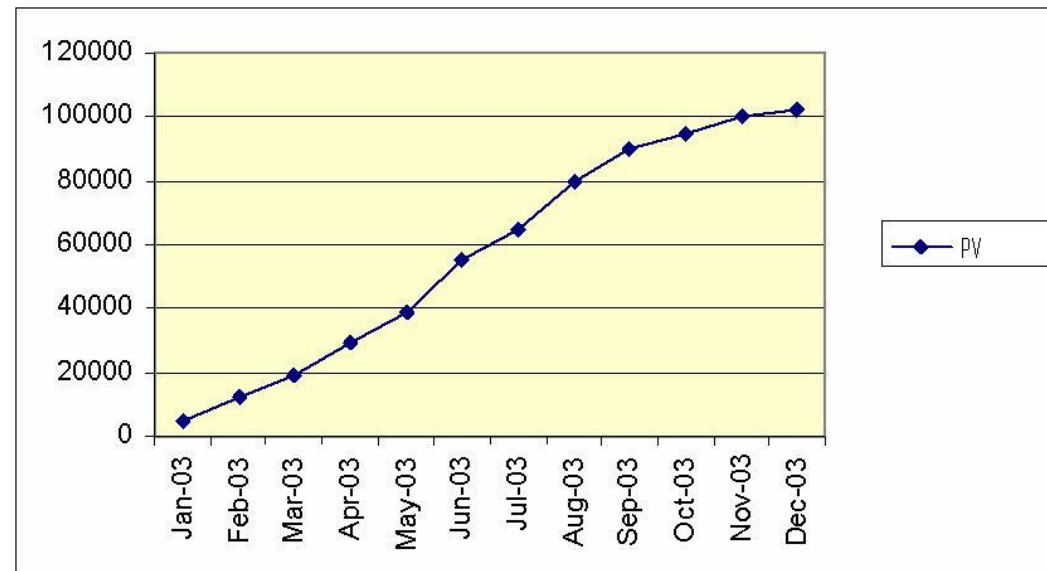
forecast
– Estimated Cost: EAC

EVA - Basic Figure: Planned Value (PV)

Definition: planned cost of the total amount of work scheduled to be performed

calculated for each task that is represented in the schedule
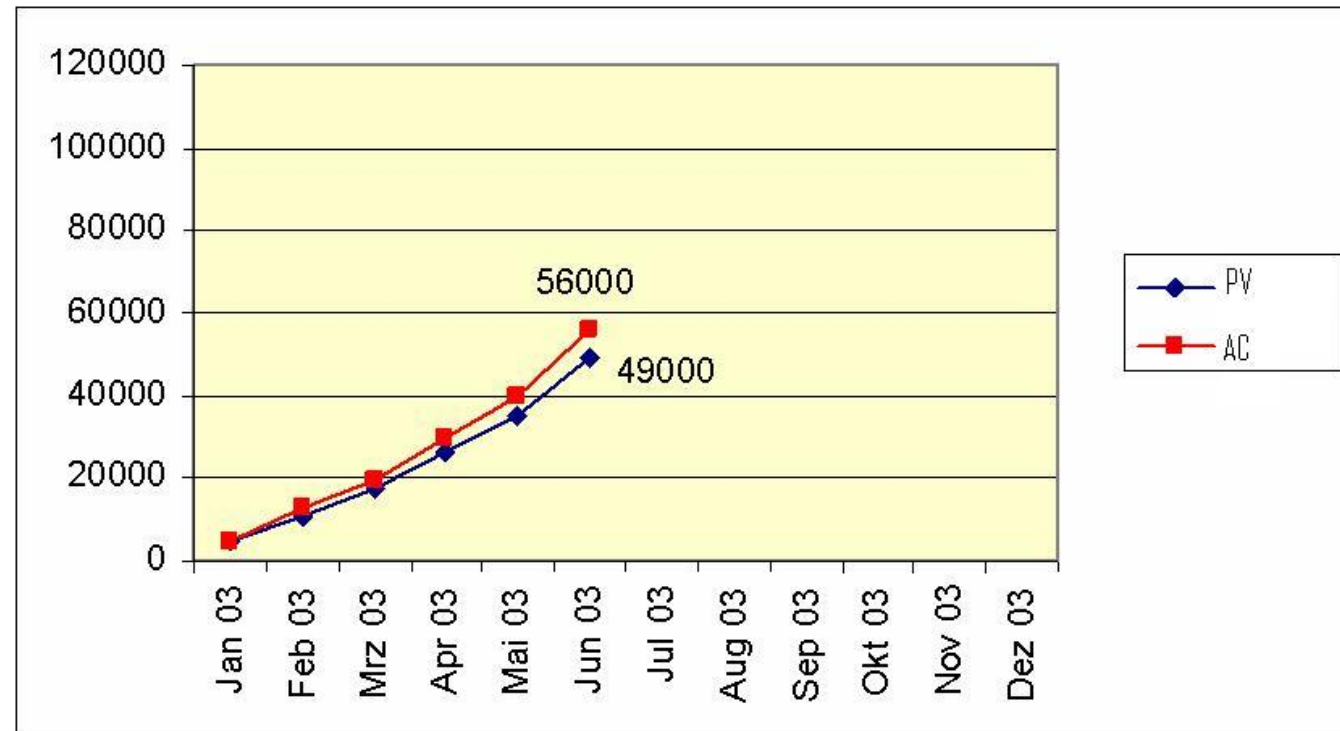
PV_i: effort planned for task i

budget at completion =

$\sum (PV_k)$ for all tasks k

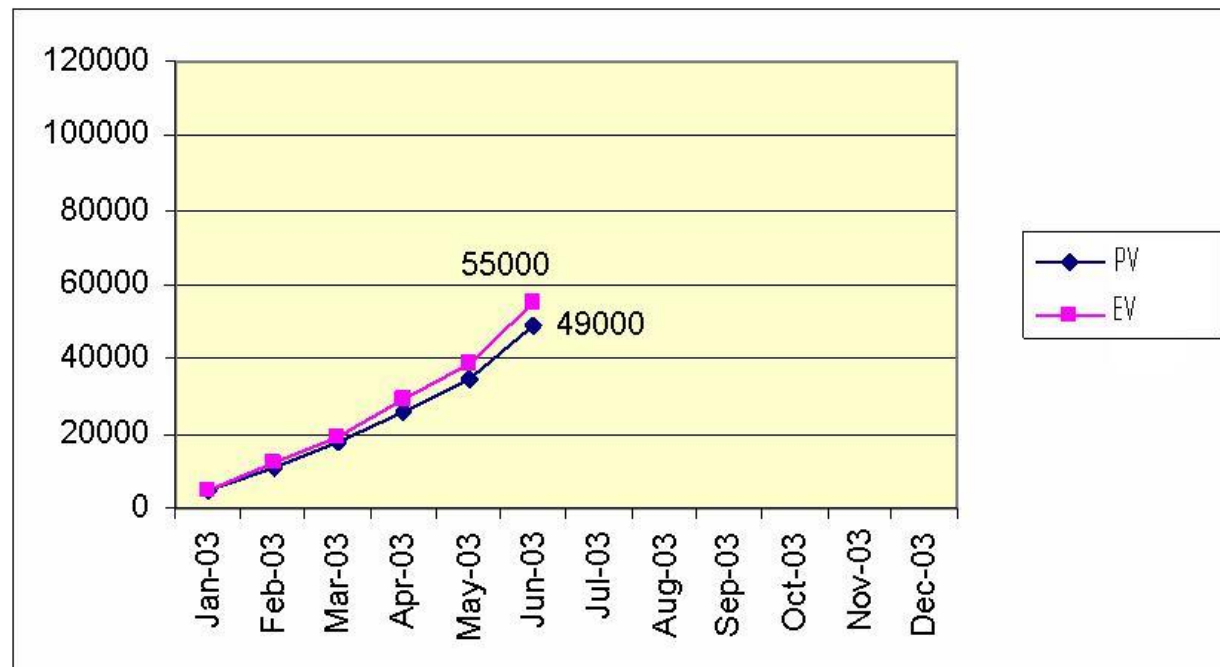# EVA - Basic Figure: Actual Cost (**AC**)

cost incurred to accomplish the work that has been done to date

EVA - Basic Figure: Earned Value (==EV==)

the value of work that has been achieved to date

$EV = \sum (PV_k * \text{Degree of Completion}_k)$

EVA - **Derived Figures**

Cost Variance (CV)	Cost Performance Index (CPI)
compares actual costs to the budgeted costs	compares budget with actual cost
CV = EV – AC	CPI = EV / AC
interpretation	interpretation
– CV = 0	– CPI = 1
– CV < 0	– CPI < 1
– CV > 0	– CPI > 1

EVA - Derived Figures (2)

Schedule Variance (SV)

compares planned work to completed work

SV = EV – PV

interpretation

– SV = 0

– SV < 0

– SV > 0

Schedule Performance Index (SPI)

compares the budget with the plan
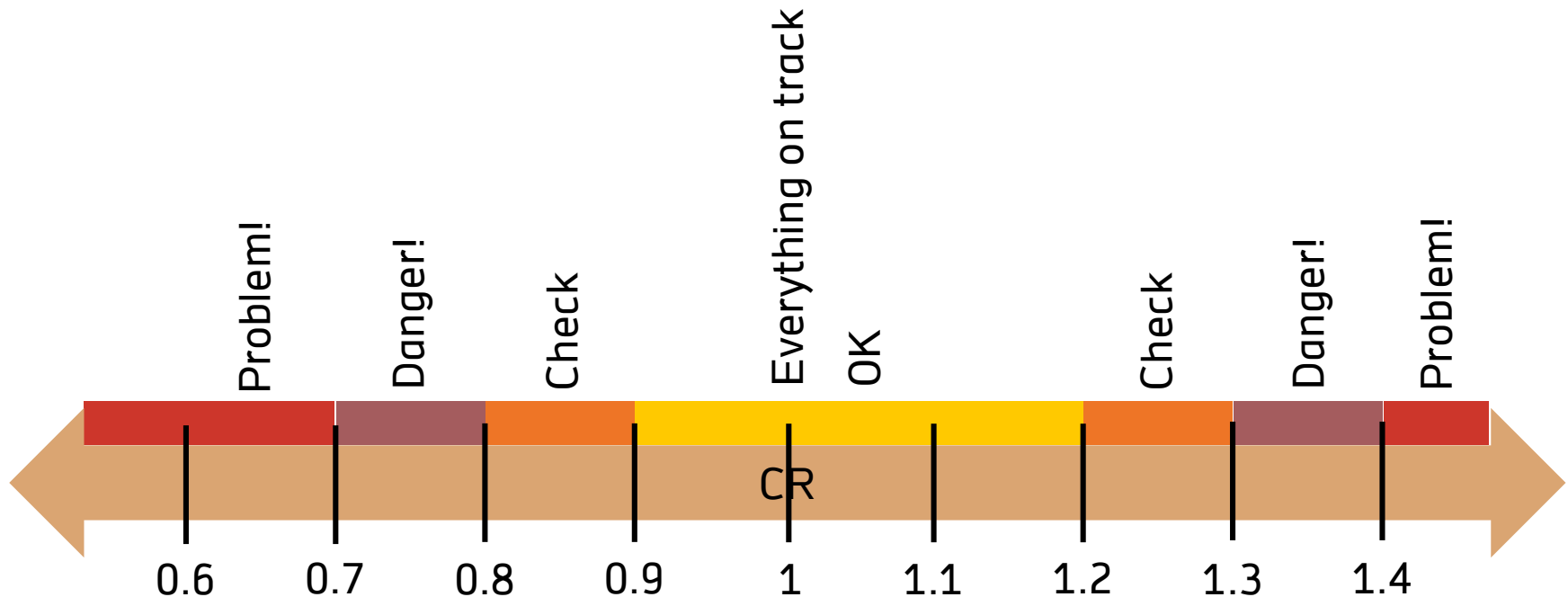
SPI = EV / PV

interpretation:

– SPI = 1

– SPI < 1

– SPI > 1

EVA – Derived Figures (3): <mark>Critical Ratio</mark>

<mark>critical ration represents whole projects performance</mark>
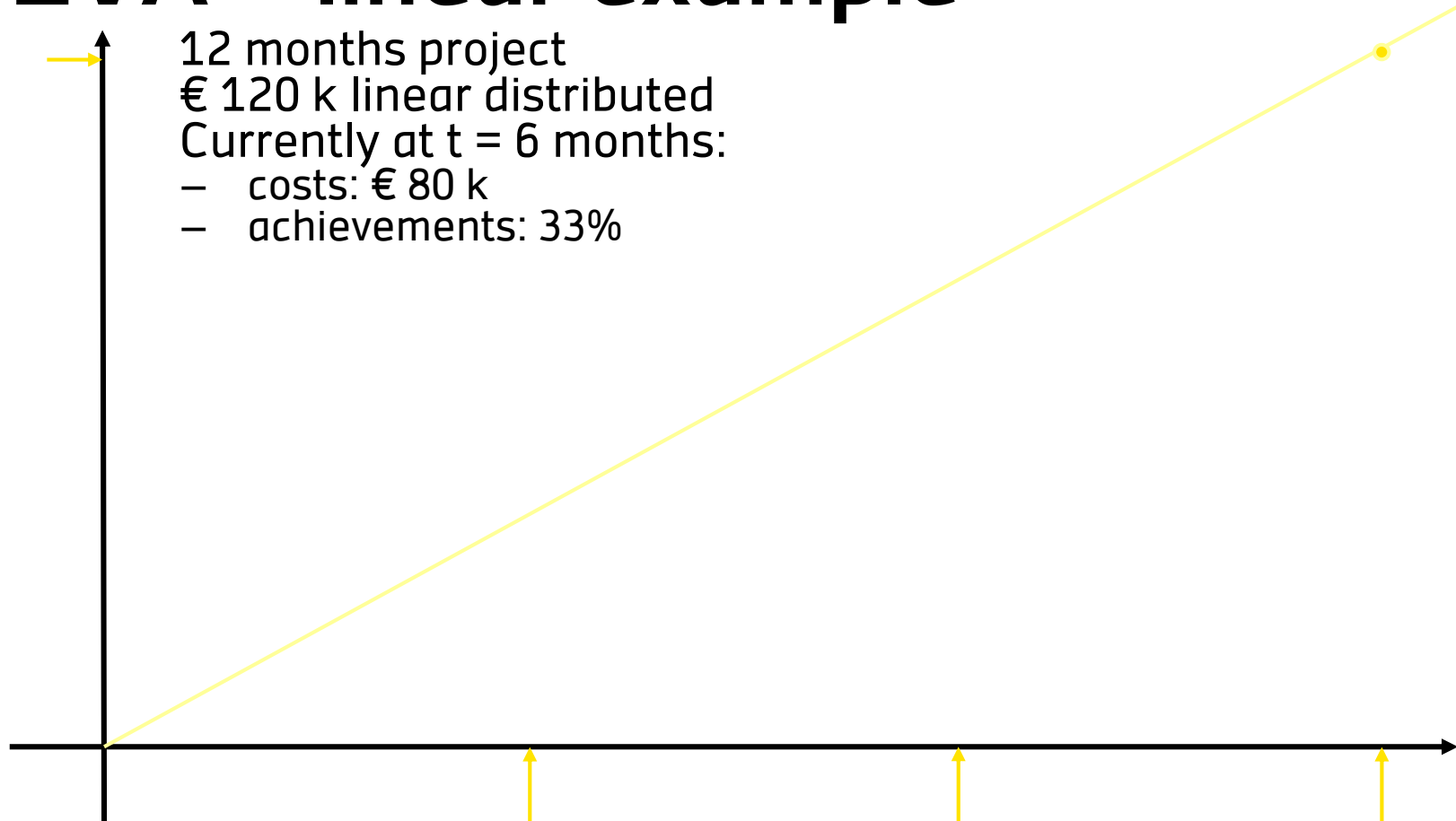CR = SPI * CPI

EVA - Forecast: Estimated Cost (EAC)

used to predict total cost of the project on completion in advance
EAC=(Starting PV-Remaining Budget)/CR =AC/(CPI*SPI)

Degree of Completion: alternative approaches
1. linear in time
–   no improvement (as compared to "standard" guesses)
2. fixed formula
    a) 0/100%
–   only those tasks *finished completely* contribute to the earned value!
–   „pessimistic"
    b) 50/50%
–   as soon as a task has started, 50% of its earned value are taken into account; the other 50% upon completion
    c) further alternative: 25/75%
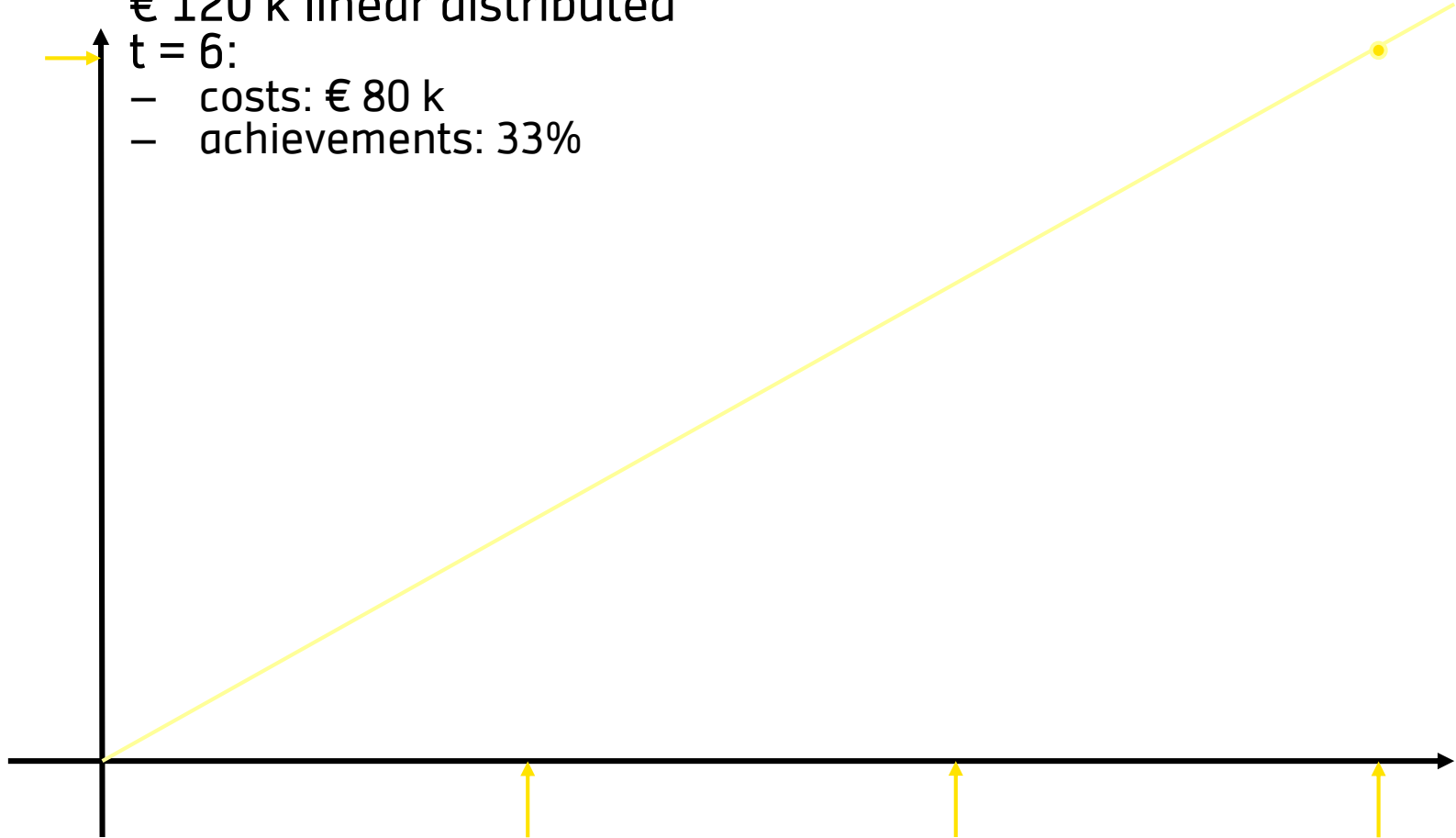
EVA – linear example

12 months project
€ 120 k linear distributed
Currently at t = 6 months:
– costs: € 80 k
– achievements: 33%

EVA – linear example

12 months project
€ 120 k linear distributed
t = 6:
- costs: € 80 k
- achievements: 33%

Agile Controlling: Burndown Chart

Source: https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/release-burndown/altern
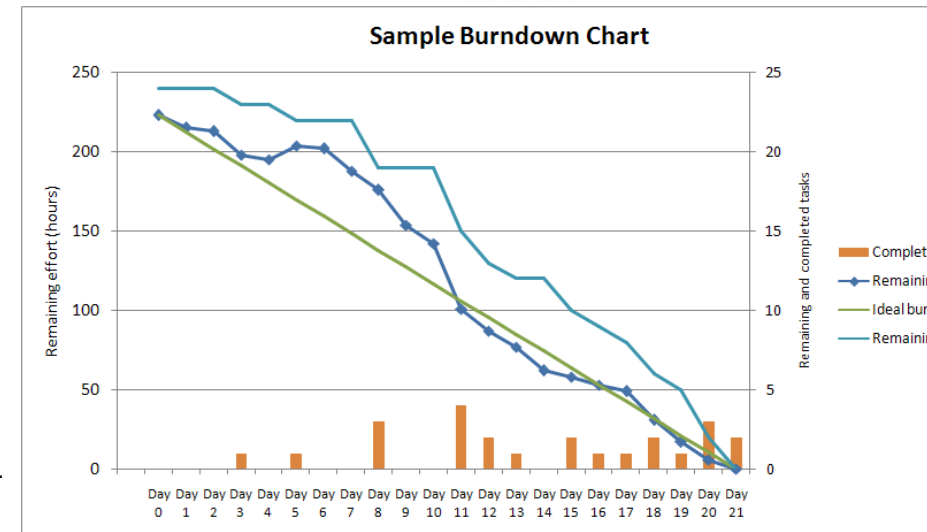
Graphs remaining effort over time
For a single or multiple iterations
Additional effort added to below 0
Crossing of extrapolated trajectories
provides prognosis for end date

Source: https://de.wikipedia.org/wiki/Burn-Down-Chart#/media/File:SampleBurndownChart.png
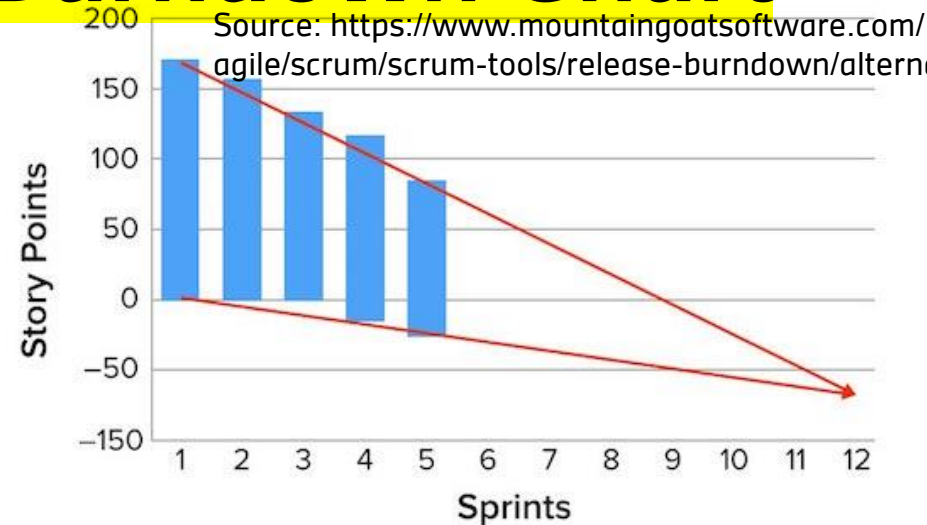
Corrective Actions: options for the project manager

time
– schedule/deadlines
quality
costs
– resources/money
functionality
– "scope"

➢interdependencies among these 4

more   *less*

time   quality

resources/ money   functi-onality
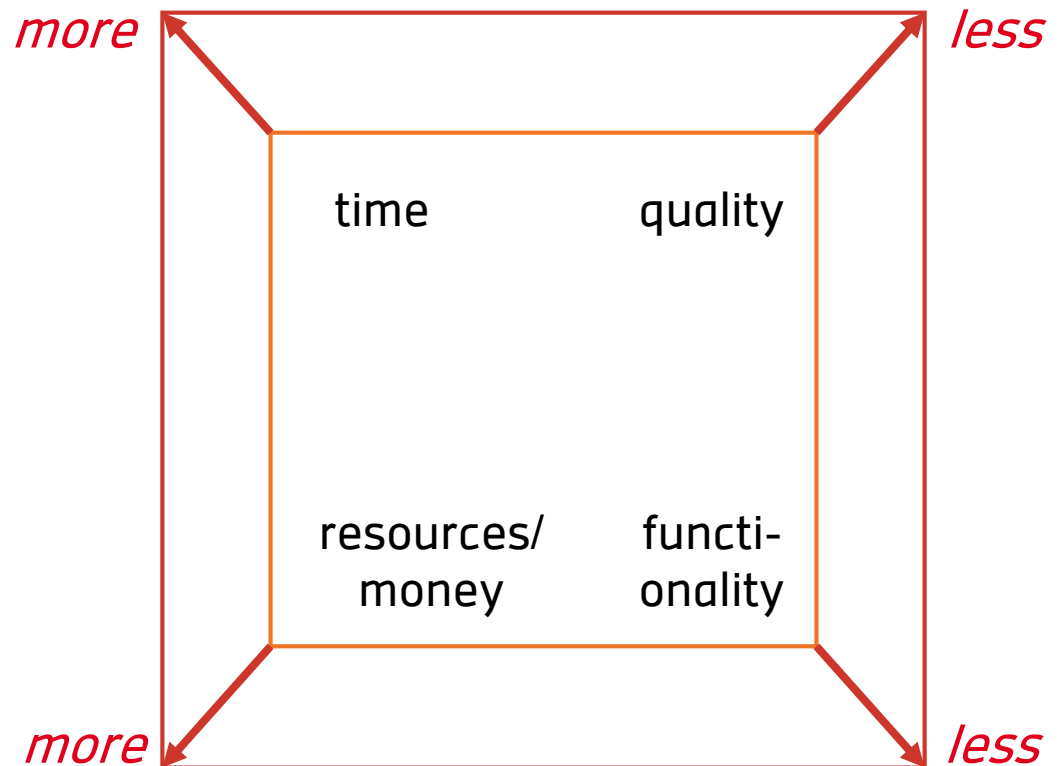
more   *less*

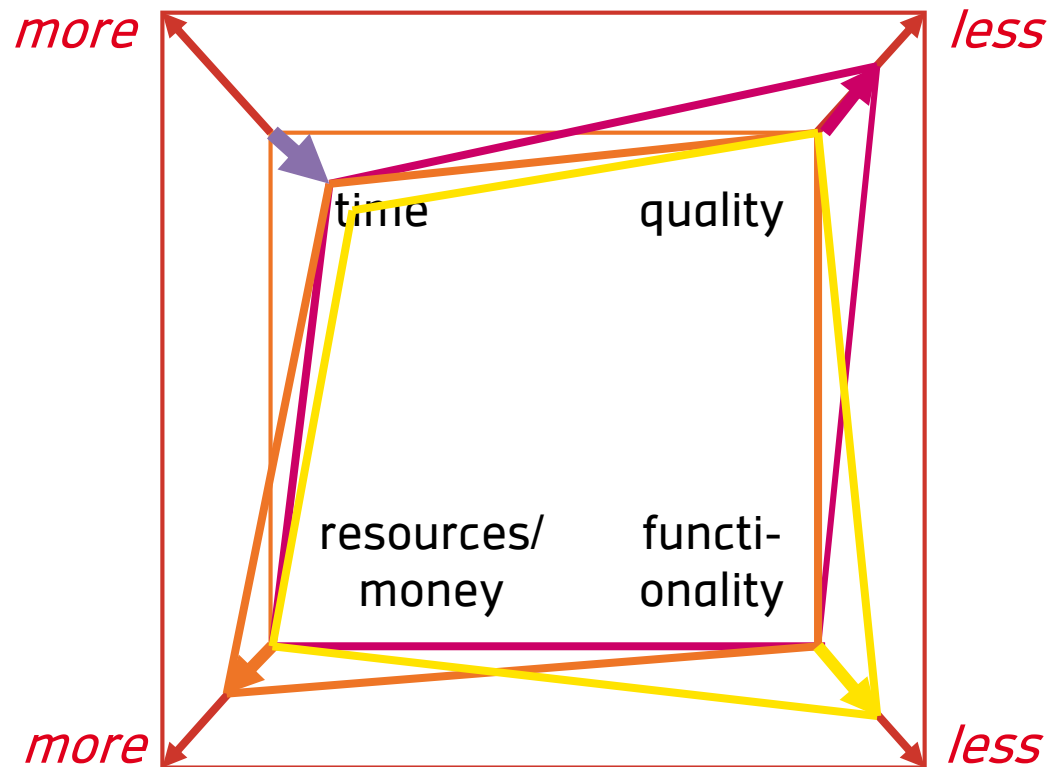# Corrective Actions example: <mark>less time</mark>

potential solutions:
- <mark>more resources / higher costs</mark>
- <mark>less quality</mark>
- <mark>less functionality</mark>

Corrective Actions: example <mark>less time - elaboration</mark>

Increase budget, i.e. more people
- – „Adding more people to a late project makes it even later"
- – effort for getting familiar with the project
- ➢ increases effort for communication and coordination

Reduce quality
- – fewer tests, reviews, …
- ➢ consequences are pretty obvious

Reduce scope, i.e. functionality
- – e.g., for the new release of a standard software
- ➢ any – external or internal – customer is unlikely to appreciate this

Creative planning: perform activities on the critical path in parallel
- – e.g., user documentation – implementation
- ➢ will *increase the overall effort* (requires final consolidation)

Cost problems and potential solutions

reasons

– effort has been underestimated

– technical problems that were not foreseen

approaches: there is no single solution

– fixed price customer project:

  ➢ extremely difficult ☹ ☹ ☹ …

– customer project with fixed limit for the overall effort

  ➢ reduce functionality, shifts to later phases of the project, …

– product development

  ➢ focus on core functionality

Scope (Functionality) problems and potential solutions

common reasons:
- ➢ unrealistic requirements/expectations
- ➢ technical problems

potential approaches
- o consult external experts
- o evaluate alternative implementation approaches
- o outsourcing to subcontractors
- o buy components instead of implementing them
- o …

all of these approaches have certain consequences, none comes for free
- – e.g, they increase costs, take more time, … ☹

Hochschule **für** Technik **Stuttgart**
Quality problems and potential solutions

most often in later phases of the project
only solution:

– test and get rid of them!

prevention

– use well established software engineering methods
  – and be acquainted with them!
– reviews in all phases of the project, not just at the end!

Prof. Dr. Lückemeyer   29

Corrective actions from *outside* the project

starting point: one of the following situations:
- project manager has taken all actions that he/she is allowed to take
- project manager overloaded
- project manager him-/herself is the problem
- …

steering committee, upper management takes appropriate action
- Assign further/different people, potentially with a better/more suitable qualification
- new project manager – difficult, especially in the beginning
- additional support for the project manager:
  - assistant, project office with supporting staff, …
  - division into technical and administrative project management

Remember: escalation is your last resort as a PM! Shows your limited authority, capabilities, BUT escalating too late is the worst case!

Questions? Questions!

THANK YOU VERY MUCH FOR YOUR ATTENTION!