

T-76.3601 — Introduction to Software Engineering

Software Project Management

<http://www.soberit.hut.fi/T-76.3601/>

Casper Lassenius
Casper.Lassenius@tkk.fi



Agenda

- Software projects
- Project planning
- Effort estimation and scheduling
- Risk management
- Monitoring and control



What is a Project?

- A project is a planned activity that involves non-routine tasks and has a clearly defined **beginning** and an **end**.
- Other project characteristics:
 - Specific **objectives** are to be met
 - Specific **resources** are assigned for use on the project
 - A **schedule** should be met



Different Types of Projects

- Projects developing
 - One-of-kind customer specific systems
 - Totally new software products
 - New versions of software products
 - New features or improvements to old systems
 - Products having embedded software
- Projects that are
 - Intra-organizationally distributed
 - Using software subcontractors
 - Using ready-made components
 - Developing or using open-source software
- Their size, length and resources used can differ



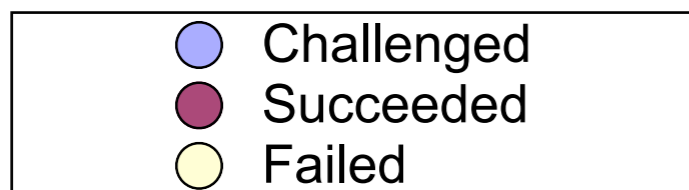
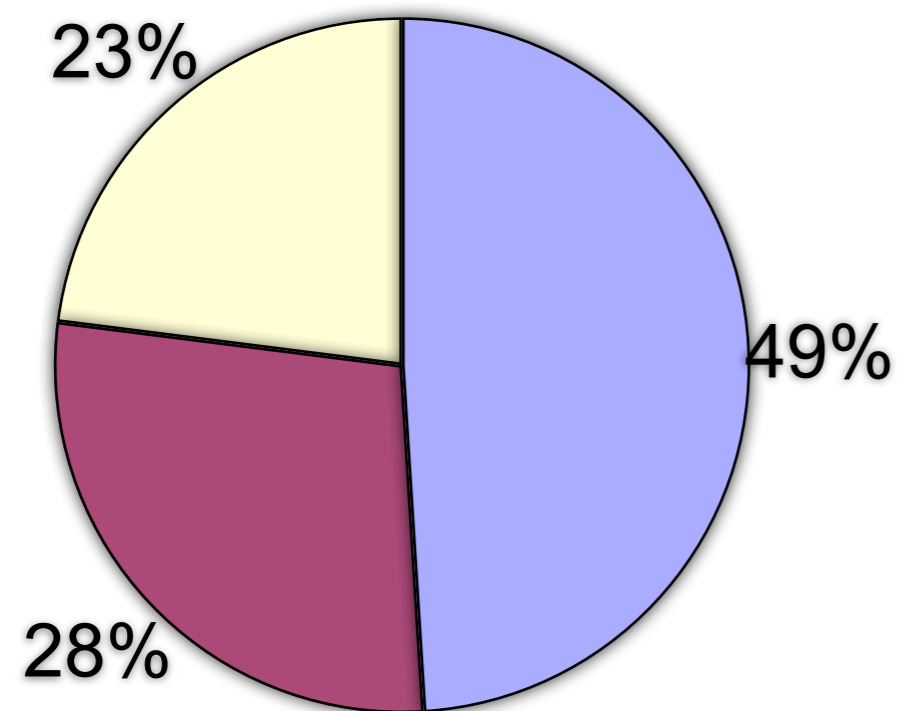
Software Development vs. Other Projects

- Many techniques of general project management are applicable to sw project management
- Software development projects are often very hard to manage
- According to Fred Brooks software is different, because of its
 - **Invisibility**
 - **Complexity**
 - **Conformity** – conform to requirements of human clients
 - **Flexibility** – high degree of change
- Other characteristics of software development
 - Doing a perfect requirements specification in the beginning difficult
 - High productivity differences between individuals
 - Division of tasks—adding workforce in late phase can be harmful
 - Lots of changes—their effect on the system often unknown



Software project success rates 2000

(According the Standish Group, based on US data)



- Successful: on time, on budget, all features
- Challenged: Completed and operational, but over-budget, over time, fewer features
- Failed: Cancelled

Reasons for success and failure

(According the Standish Group, based on US data)

- Reasons for failure
 - “Most projects failed for lack of skilled project management and executive support”
 - “Underestimating project complexity and ignoring changing requirements are basic reasons why projects fail”
 - “The problem – and the solution – lay in people and processes”
- Recipe for success
 - Smaller project size and shorter duration
 - More manageable
 - “Growing”, instead of “developing”, software engages the users earlier and confers ownership.
 - -> Iterative and interactive process



Project Planning



Project Stakeholders

- Identify as early as possible
 - Recognize their motivation and objectives, and try to reconcile them
- Set communication channels
- Stakeholders can be
 - Internal to the project team
 - External to the project team but within the same organization
 - E.g. marketing department
 - External to both the project team and the organization
 - E.g. users, customers, subcontractors



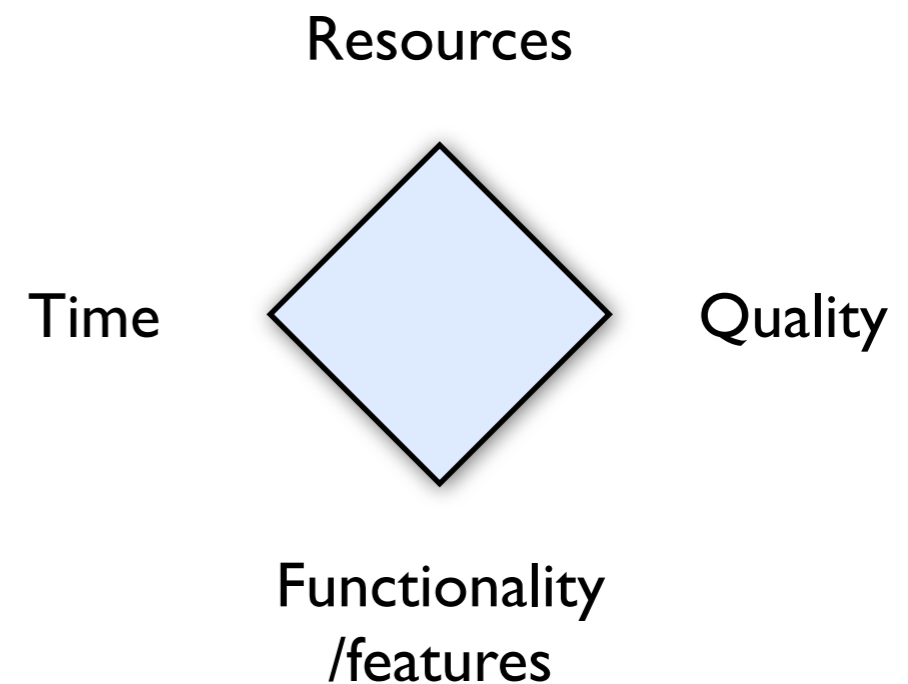
Setting Objectives and Goals

- Project objectives should be clearly defined
- All involved should be informed about the objectives which have to be acceptable for them
- Objectives guide and motivate participants
- Split the project overall objectives into sub-objectives
- Also developer level sub-objectives, that developers can affect
- Objectives should be such that it is easy to determine whether the project has been successful or not
- Which one is better?
 - "To improve customer relations"
 - "To reduce customer complaints by 50 %"



Project constraints

- Projects normally have constraints, such as resources, time, quality and functionality
- These constraints should be addressed when defining objectives
- Quite often one or two are more important than the others, e.g.
 - Time to market
 - Basic functionality
- -> Let everybody know which are the most important ones!



Uses of the Project Plan

- The project plan is often one of the most important project documents
- The primary purpose of a project plan is to
 - document planning assumptions and decisions
 - facilitate communication among shareholders
 - document approved scope, cost and schedule baselines
- In the beginning of the project
 - writing a project plan requires to agree on and consider many important matters
 - the project plan is used to communicate information to different stakeholders
- During the project, project plan is used for
 - checking what was agreed on
 - communicating project info e.g. to new project members



Steps for Doing a Project Plan

- The Project manager is often responsible for writing the project plan
- It is important that all team members participate in planning
- Accepting the project plan
 - e.g. project board
- Delivering the plan to all stakeholders
- The project plan can and should be updated, at least the most important changes
 - version history
 - decide who can do / approve changes, e.g.
- project board / steering group



The Contents of a Project Plan

I. Project overview

- background
- purpose, scope, objectives
- assumptions, constraints
- deliverables
- customer responsibilities
- schedule and budget summary
- evolution of the plan
- references
- definitions

2. Project organization

- external interfaces
- internal structure
- roles and responsibilities

3. Project partitioning

- process model
- project milestones
- project phases /cycles
- release plan

4. Work plan

- work activities
- schedule
- resource allocation

5. Technical plan

- methods, tools, techniques
- infrastructure



The Contents of a Project Plan

6. Support processes

- Staff training
- Quality assurance, reviews, testing
- Configuration / version management
- Documentation

7. Partnering / subcontracting

8. Communication plan

- internal communication practices
- informing

9. Control plan

- project management practices
- reporting
- requirements, schedule, quality, budget control
- change procedure
- metrics collection

10. Risk management

11. Project closeout

- acceptance plan and criteria
- close out plan

12. Budget



Effort Estimation and Scheduling



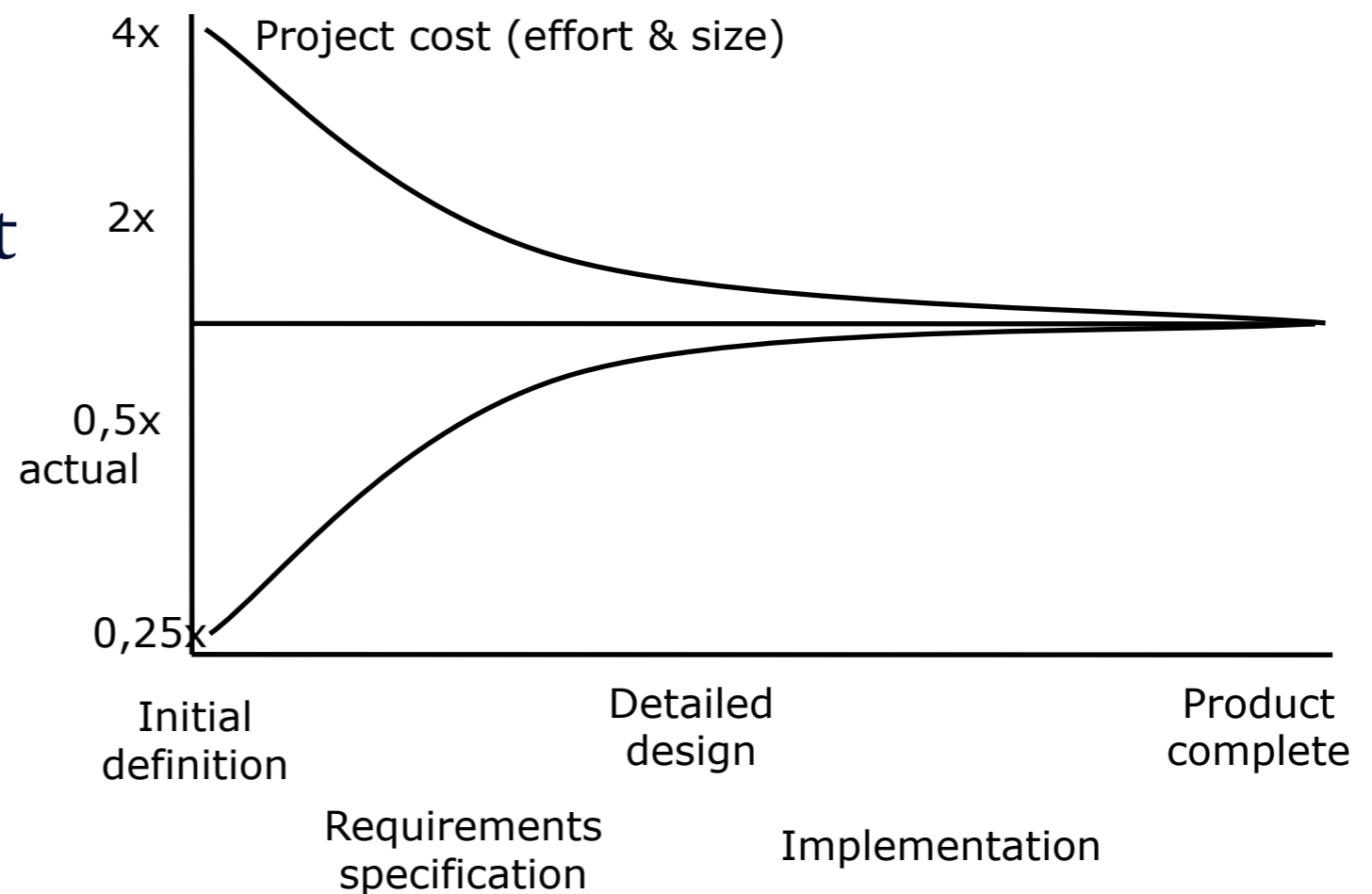
Problems in Effort Estimation

- Basic problem: Predicting the future by looking into the past
- A lack of information on the project to be estimated
 - Most influential decisions are made in the early phases of project, based on inadequate information
- A lack of good historical information
- Estimates are done sloppily
 - "If they cannot be done perfectly, why pay attention to them?"
- Estimates are not followed, respected or trusted
 - An estimate should not be an opinion, as an opinion can be overruled by your superior



Estimates Evolve as the Project Progresses

- As the project progresses you can make better estimates — estimation is a process of gradual refinement
- Problem: new estimations are not done, but the old ones are followed
- Update your estimates!



Estimation Techniques

- Algorithmic models
 - Albrecht & MarkII function points
 - COCOMO 81 and COCOMO II
- Expert judgement
- Estimation by analogy
- Top-down estimation
- Bottom-up estimation



Effort Estimation Best Practices

- Use several estimation techniques and compare them
 - If they converge, you are probably on the right track
 - Find out why the estimates are different
 - Combine several expert opinions
 - Ask several different estimates – optimistic, probable and pessimistic, and compare them
- Avoid off-the-cut estimates
- Allow time for the estimate, and plan it
- Use documented data from previous projects
- Use developer-based estimates



The Terms Used

- Pay attention to terms used:
 - Use HOURS when talking about efforts
 - Use DAYS when talking about schedule
 - Do not mix estimated efforts and calendar time!!!



Scheduling Software Projects

- The relationship between the number of staff working on a project, the total effort required and the development time is not linear.
 - Increasing staff increases the communication and management costs.
 - Software project work cannot be partitioned infinitely
- A rough estimate: only 60-70% of work time is efficient
- Remember vacations, sick leaves, etc.
- To get a realistic schedule accepted can be the most difficult part of the project (McConnel, 1994)
 - Have a good reasoning behind your schedule estimates
- Do not present over-optimistic schedules
 - They will be accepted & guarantee your project will be late -> if the schedule is fixed, cut the scope



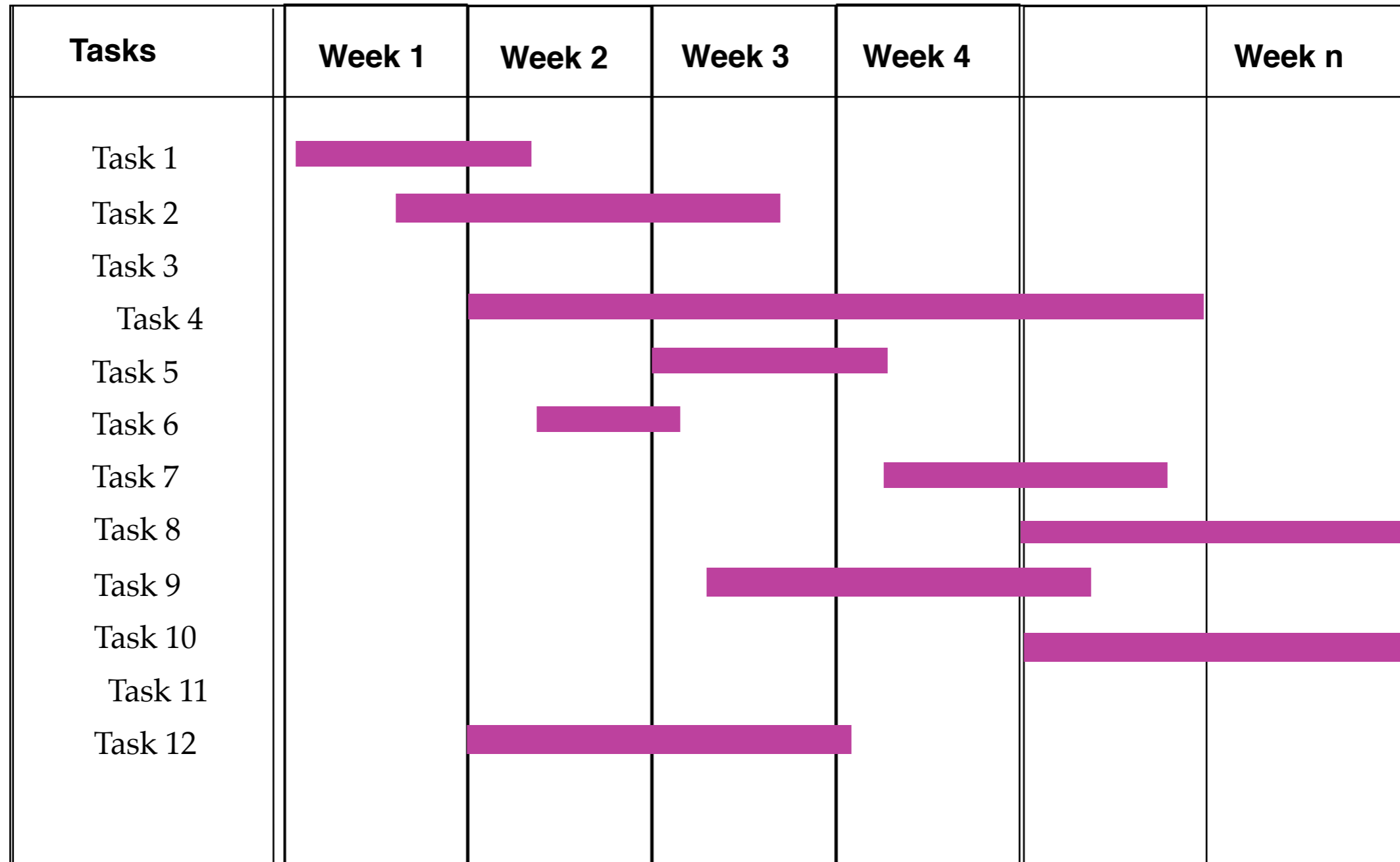
The Reasons for Scheduling

- A good schedule will enable us to:
 - Ensure that the appropriate resources will be available when required
 - Avoid different activities competing for the same resources at the same time
 - Produce a detailed schedule showing which staff carry out each activity
 - Produce a detailed plan against which actual achievement may be measured – and replanned if needed

Hughes, Cotterell, 2002



Timeline Charts



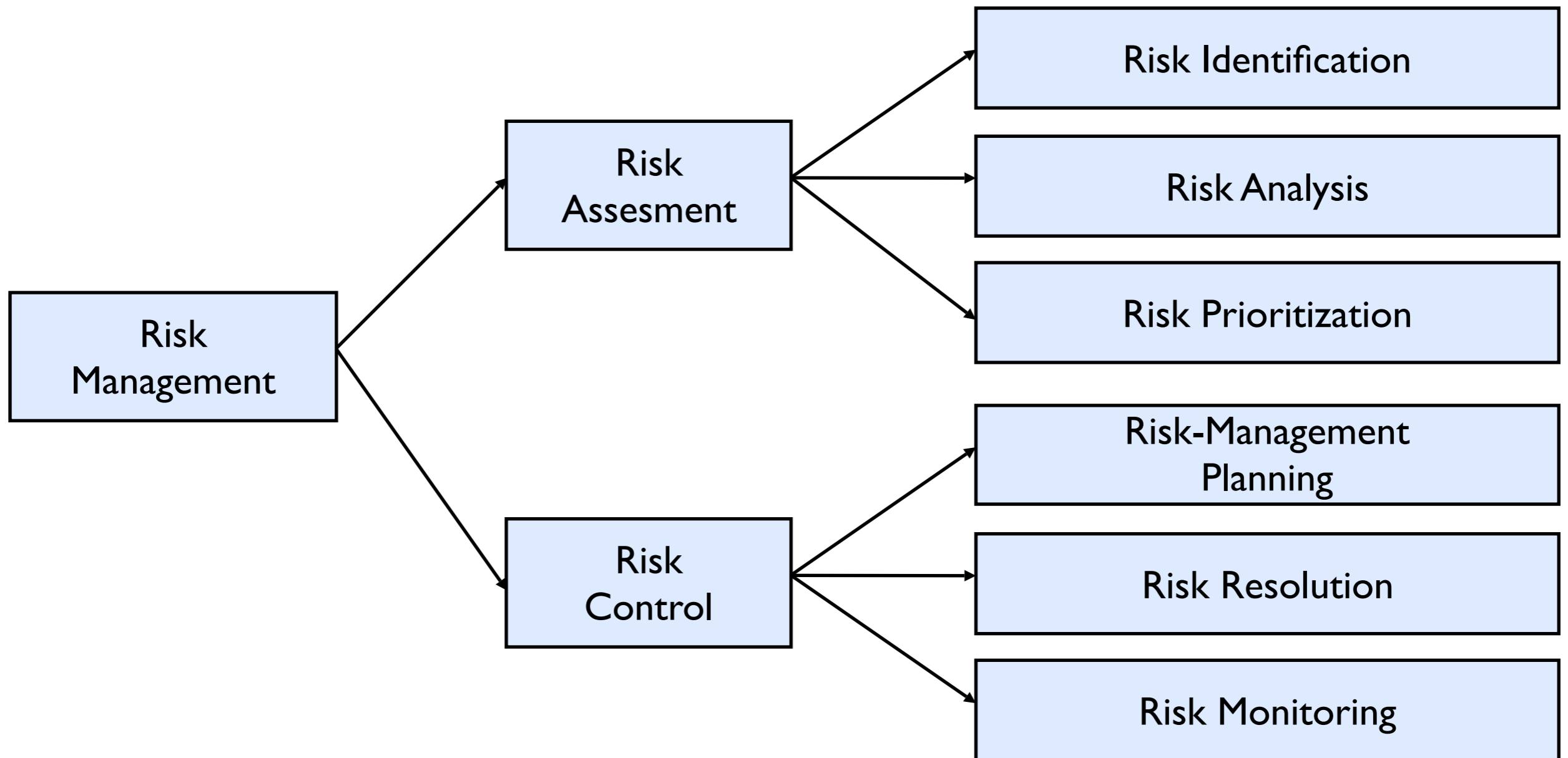
Risk Management

“If you don't actively attack the risks, they will actively attack you.”

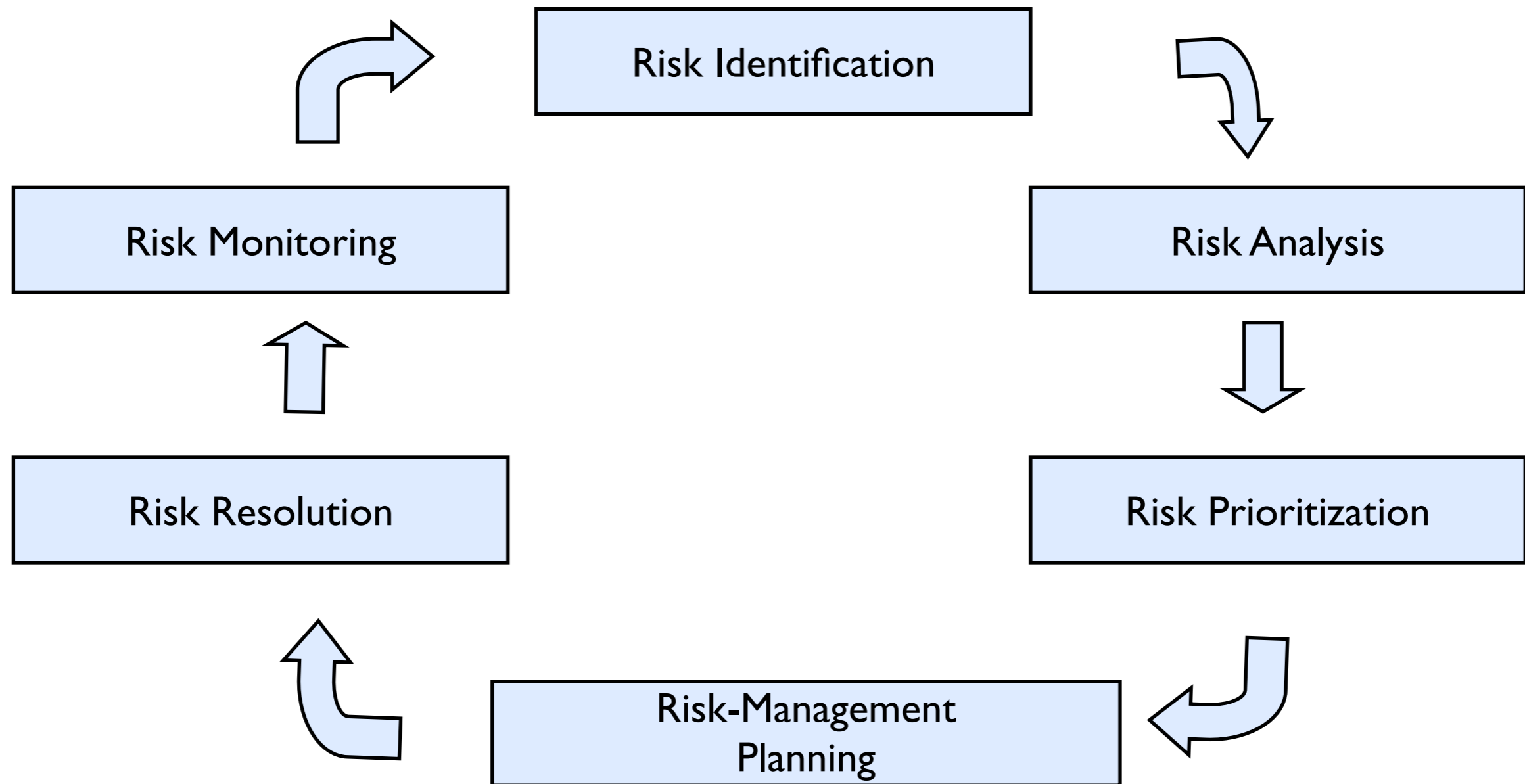
Tom Gilb



Elements of Risk Management



Risk Management Cycle



Risk Exposure

- Different ways to measure, e.g.
 - Time:
 - Probability of loss * Size of loss in weeks = Risk exposure in weeks (e.g. 50 % * 5 weeks = 2,5 weeks)
 - Suitable when you are concerned only with schedule risks
 - Money:
 - Probability of loss * Size of loss in money = Expected value of loss (e.g. 50 % * 100 000 € = 50 000 €)
 - Monetary value is easy to understand, but not always easy to estimate. High loss risks become visible.
 - Scores:
 - Likelyhood (scale 1-10) * Impact (scale 1-10) = Risk exposure (scale 1-100) (e.g. 5*10 = 50)
 - Easy to use



”Top 10 Risks” List

- One of the risk-monitoring tools is the use of ”Top-10 Risks” list
- Identify risk, estimate risk exposure and prioritize risks
- List top 10 risks
- List contains:
 - Each risk’s current rank
 - Its previous rank
 - The number of times on the list
 - Summary of the steps taken to resolve the risk since the previous review
- List should contain also risks moved off the list since the last review
- Top-10 list should be reviewed once a week, e.g. project manager and his boss, or in weekly meetings
- Appointing a risk officer can be useful
 - looks for all reasons for project to fail
 - psychological reasons
 - the role is given to a team member



Example of a "Top-10 Risks" List

This week	Last week	Weeks on list	Risk	Risk resolution progress
1	1	5	Feature creep	Staged delivery approach adopted, need training
2	-	1	Change of CM system	Evaluation under way
3	5	5	Optimistic schedule	New estimation and functionality prioritization under way
4	2	5	Program speed	Negotiations about additional resources under way
5	7	5	Slow customer feedback	Meeting with customer scheduled
...



Important in Risk Management

- Risks management should not be forgotten right after identifying the risks in the beginning of the project -> MONITORING
- More important than exact calculations of risks is to identify the most important risks early enough and react to the findings
 - Remember that all numbers used are only ESTIMATES and they can give only direction
- A simple method of following the risks is better than nothing (e.g. updated "Top-10 Risks" list that is checked regularly)
- Separate FACTS from RISKS



Monitoring and Control

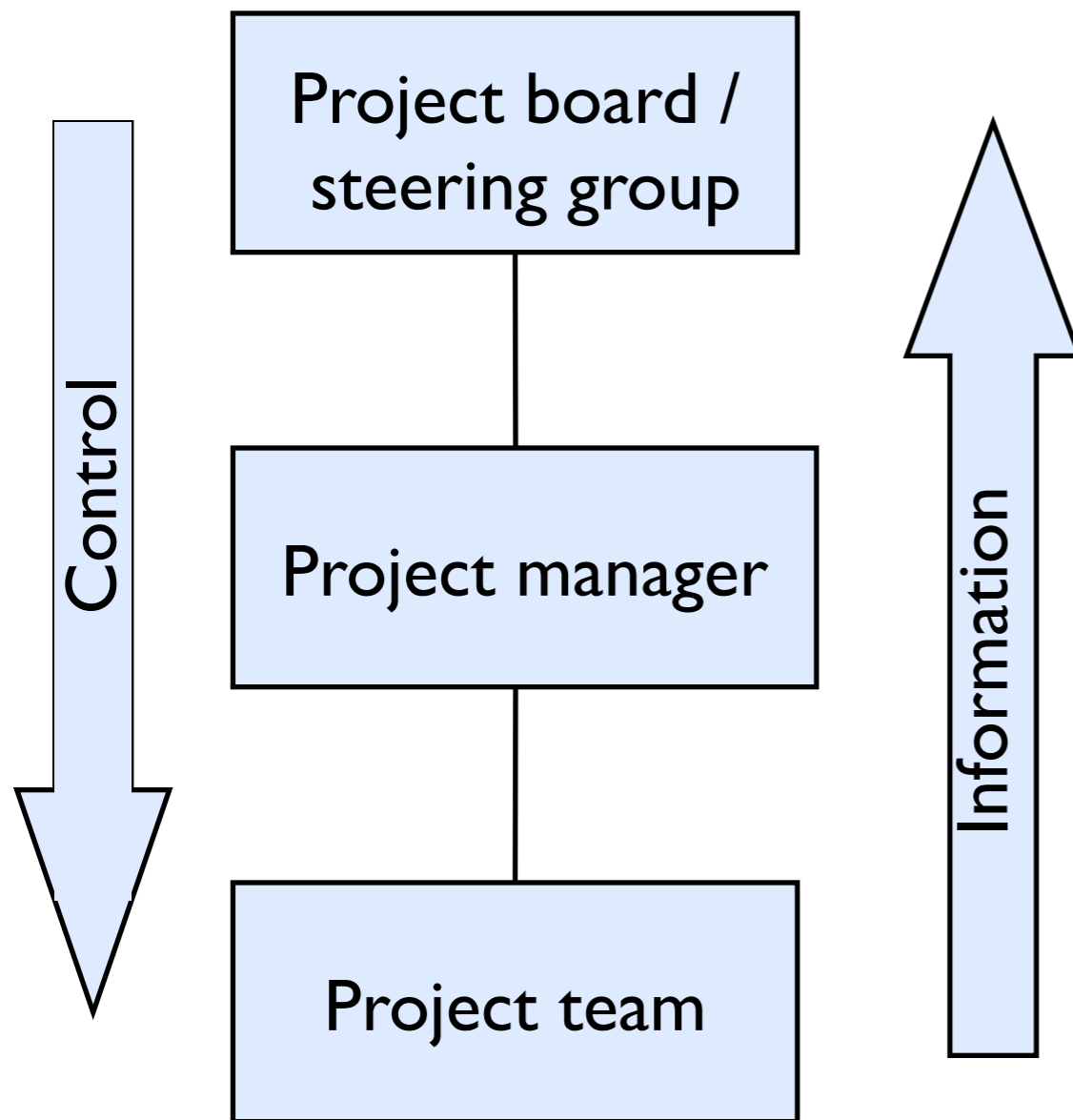


Monitoring and Control

- Monitoring:
 - What is happening?
 - Compare to the plan
- Control:
 - Use monitoring information
 - React to slippage
 - Replan to bring the project back on target or revise the target
- Plan monitoring and control in the beginning of the project and state in the project plan
 - Define practices, e.g. progress reports, meetings
 - Assign roles and responsibilities, e.g. reporting responsibilities, reacting to deviations
 - To follow the progress you need a detailed plan against which to compare the progress!



Levels of Control



- Project board
- Consists of e.g. higher level managers and customers
- Progress reports and/or meetings, e.g. monthly
- Inform often enough
- Inform about possible problems early enough: dividing responsibility
- Project manager reports
- Project manager & project team
- Meetings and/or progress reports, e.g. weekly or even daily

Reporting Progress

- Achievements in reporting period: finished tasks
- Future outlook: Planned tasks, how things are likely to progress during next period
- Problems encountered
 - Focus on real problems - exceptions to planned activity
- Costs — actual costs compared to budgeted (earned value)
- Staffing — joiners, leavers, sickness etc.
- Risk monitoring — Top-10 Risks
- Avoid ‘information overload’
- When information goes to higher management levels summarize more
- Use visualizations
 - graphical representation
 - highlight problems



A Problem

- 90% completion syndrome
 - job reported as 'on time' until last scheduled week
 - job reported as '90% complete' for each remaining week until task is completed

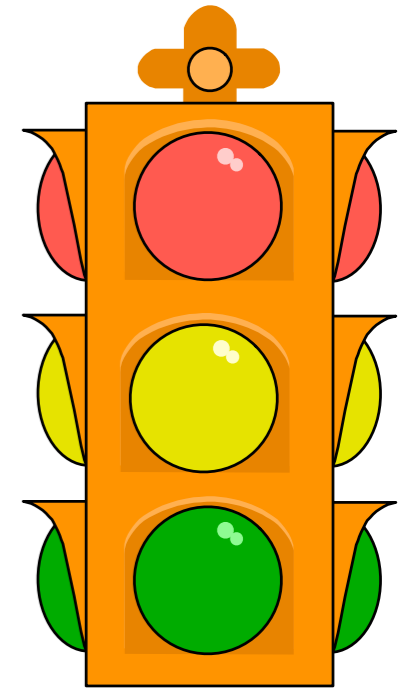


Solution?

- Control on deliverables: report only finished tasks (e.g. tested functionality)
- Estimation & WBS: tasks small enough (a few hours – a few days)
- Define what is meant by "completed", e.g.
 - developer has tested it
 - integration testing is another task
 - possible corrections are separate tasks
- An alternative, when tasks larger:
 - Ask how many hours are already used to accomplish a task
 - Ask for an estimation of hours still needed to complete a task
 - Compare to the original estimate



Visualizing Progress



- Enables to see the project progress quickly and notice the possible slippage
- Stakeholders need the transparency
 - Team members -> motivation
 - Management -> possibility to react
 - Customer -> e.g. payments
- Many possible charts etc.
 - Choose the one best suitable for your project
 - Update frequently
 - React to problems

E.g. Traffic-lights

- Red not on plan:
recoverable only with difficulty
- Yellow not on plan:
recoverable
- Green: on schedule

Week	1	2	3	4	5	6	Comments
Task 1	Green	Green	Green	Green			
Task 2	Green	Yellow	Yellow	Red			
Task 3	Green	Yellow	Red	Yellow			
Task 4	Green	Green	Green	Yellow			
Task 5	Green	Green	Yellow	Red			
Task 6	Green	Green	Green	Green			
...							

Important in Monitoring and Control

- Plan monitoring and control practices in the beginning of the project
- Monitor the progress very frequently, e.g. daily or weekly
- Give immediate feedback to
 - managers
 - team members
- React to deviations fast



Software Development Teams

”It is the People – not the procedures and techniques,
that are critical to accomplishing the project objectives.”



What is a Team?

- A team consists of
 - at least two people, who
 - are working towards a common goal/objective/mission, where
 - each person has been assigned specific roles or functions to perform, and where
 - completion of the mission requires some form of dependency among group members (Dyer)
- Team size
 - Less than 20 people
 - Optimal size is 4-8 persons for software teams
 - In a larger project add the number of teams
- It is optimal that a person works only in one project team at the time



How to Build Effective Teams

- Team **cohesion** (=yhtenäisyys, yhteenkuuluvuus)
 - Collocation
 - Sense of team identity
 - Give frequent, easy opportunities for the team to succeed together and celebrate the achievement (e.g., team dinner after achieving a milestone)
- Challenging **goals**
 - “Establish a vision”
 - Goals must be specific and measurable, represent a significant challenge, be achievable and accepted by team members
 - All team members should participate in defining the team goals
 - Goals should be followed and adjusted if needed



How to Build Effective Teams

- Establishing **plans**
 - Agreeing together a strategy for achieving the goals
 - Team members must
 - feel that the tasks are achievable
 - understand their role and responsibilities
 - agree on how to accomplish them
- **Feedback**
 - Goals must be tracked and progress visibly displayed
 - Frequent and precise feedback motivates
- Maintaining **communication** among team members
 - Most common team problem is poor communication
 - Both formal and informal communication is needed
 - Formal: e.g. regular meetings once a week
 - Informal: Daily contact among team members



Working as a Team Member

- Participate actively in project planning- it is a common task
- Help your fellow team members when they have problems or questions – ask if they need help even they might not ask for that
 - -> They are happy to help you when needed
- Ask help right away when you have problems or don't understand something
- Remember: team goals are your goals -> the project can be successful only when everybody works towards common goals
- Give feedback to your fellow team members and to your project manager – also positive!
- Think about how you could make your project a fun place to work in!



Questions?

