



---

# Project Management

Credits to:  
Hans van Vliet  
Moreno Marzolla,  
Ian Sommerville  
Damian Tamburri

---

# Roadmap



- Who is the project manager?
- Software Project Management
- A scenario example
- Project Planning & Reporting
- Risks and People Management

# Roadmap

---

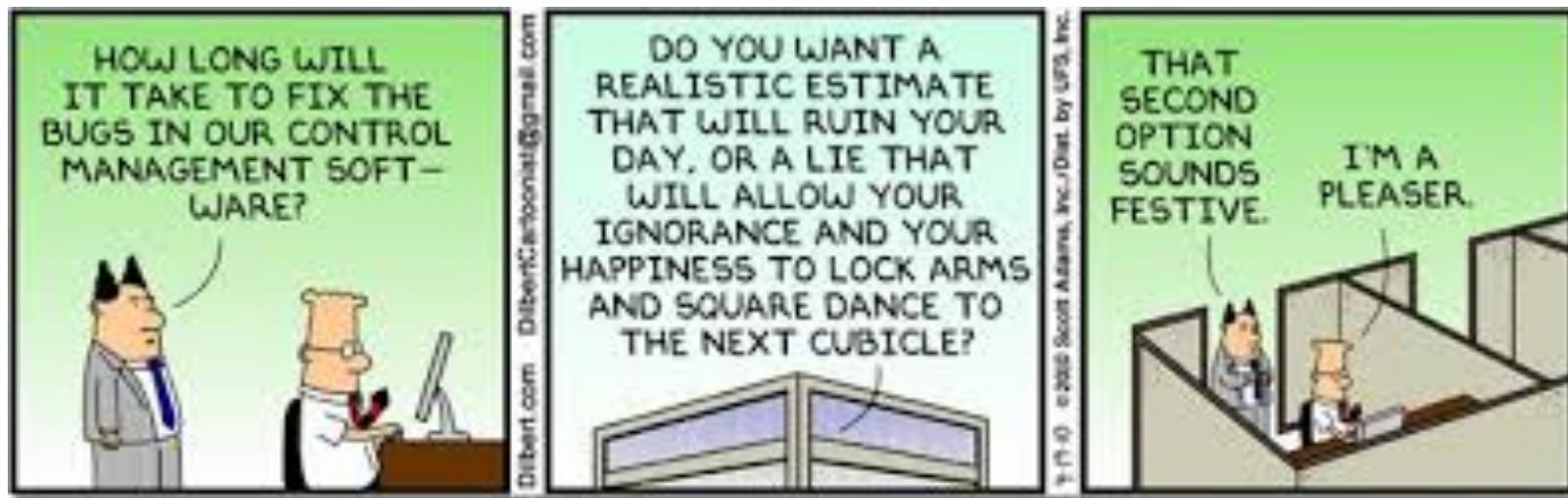


- **Who is the project manager?**
- Software Project Management
- A scenario example
- Project Planning & Reporting
- Risks and People Management

# Who is the project manager?



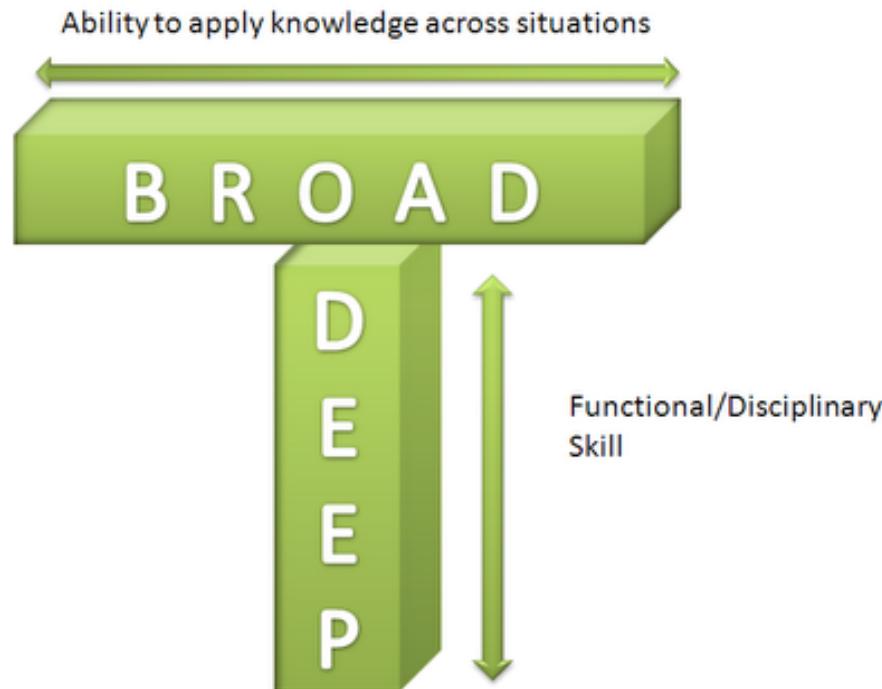
The project in this case has an intriguing name: “cosa-nostra”



# Who is the project manager?



- An experienced and **very bold** professional
- More than **just technical** experience
- T-Shaped Professional figure



# Who is the project manager?

---



- But also...
  - ▶ A **stressed** professional
  - ▶ A **creative** professional
  - ▶ Almost equally good at **writing and communicating** than at **managing people and things**

# Roadmap

---



- Who is the project manager?
- **Software Project Management**
- A scenario example
- Project Planning & Reporting
- Risks and People Management

# Software Project Management: what is a Software Project?

---



- A project is a planned set of activities that
    - ▶ Has a **starting** date
    - ▶ Has an **estimated** ending date
    - ▶ Has a **single** objective
    - ▶ Is realized by **one or more** teams
    - ▶ Exploits a more or less **flexible** set of resources
-

# Software Project Management



- Activities involved in ensuring that software is delivered on time and on schedule and in accordance with context and requirements
- Requires the interaction of ***economical, social, technical and organizational*** aspects
- A well-directed project ***may*** fail, a badly-directed project ***certainly*** fails
- Experience with previous projects is important and has to be considered a company-wide memory, i.e., organizational culture
- It is difficult to teach how to be a good manager... experience certainly helps!

# Software Project Management (2)



- *Project planning*
  - ▶ Project managers are responsible for planning: e.g., estimating and scheduling project development, assigning people to tasks.
- *Reporting*
  - ▶ Project managers are usually responsible for reporting: progress of a project; progress to customers; progress to business IT managers.
- *Risk management*
  - ▶ Project managers assess and monitor risks that may affect a project, taking action if needed.

# Software Project Management (3)



- *People management*
  - ▶ Project managers choose people for their team and establish ways of working for effective team performance
- Addendum Activity: *proposal writing*
  - ▶ The first stage in a software project may involve writing a proposal to win a contract.
  - ▶ The proposal describes the objectives of the project and how it will be carried out.

# Software Project Management (4): Intrinsic Issues in Sw. Engineering



- **The product is intangible.**
  - ▶ Software cannot be seen or touched.
  - ▶ Software project managers cannot see progress by simply looking at the artefact that is being constructed.
- **Many if not all software projects are 'one-off' projects.**
  - ▶ Large software projects are usually different in some ways from previous projects.
  - ▶ Even managers who have lots of previous experience may find it difficult to anticipate problems.
- **Software processes are variable and organization specific.**
  - ▶ We still cannot reliably predict when a particular software process is likely to encounter development problems.

# Software Project Management (5): Delay and Failure

---



- Unrealistic deadlines, e.g., it is imposed by someone external to the technical staff
- Requirements change (too) often
- Effort and resources have been estimated in an overly optimistic way,
- Risks have not been taken into account from the start of the project.
  - Risks can be technical or human difficulties
- Communication problems among the staff members
- Difficulty by the management to recognize recurrent delays and take immediate action
- Subversive stakeholders

# Software Project Management (5): Delay and Failure... **An Example?**

---



- **Unrealistic deadlines, e.g., it is imposed by someone external to the technical staff**
- **Requirements change (too) often**
- Effort and resources have been estimated in an overly optimistic way,
- Risks have not been taken into account from the start of the project.
  - Risks can be technical or human difficulties
- Communication problems among the staff members
- Difficulty by the management to recognize recurrent delays and take immediate action
- Subversive stakeholders

# Software Project Management (5): Delay and Failure... **An Example?**



- Unrealistic deadlines, e.g., it is imposed by someone external to the technical staff
- Requirements change (too) often



# Software Project Management (5): Delay and Failure... **An Example?**



- Too many undiscovered stakeholders were eventually reported
- Unrealistic deadlines were eventually reported by the staff



# Software Project Management (5): Delay and Failure... **An Example?**



**COST: 174,000,000 \$ (give or take)**

<http://www.cio.com/article/2380827/developer/6-software-development-lessons-from-healthcare-gov-s-failed-launch.html>



# Software Project Management (5): Delay and Failure... **Yet Another Example?**

---



- Unrealistic deadlines, e.g., it is imposed by someone external to the technical staff
- Requirements change (too) often
- **Effort and resources have been estimated in an overly optimistic way,**
- **Risks have not been taken into account from the start of the project.**
  - **Risks can be technical or human difficulties**
- Communication problems among the staff members
- Difficulty by the management to recognize recurrent delays and take immediate action
- Subversive stakeholders

# Software Project Management (5): Delay and Failure... **Yet Another Example?**



- “Air India Dreamliner flight diverted after software problems” – Feb. 2014
  - Miscommunicated risks
  - Miscommunicating stakeholders
  - Unknown system interaction patterns



Result: two  
Grounded  
Test Flights...



- Entire project cost... 167 \$, about 25% of which is software-related

# Software Project Management (5): Delay and Failure... **And Another?**

---



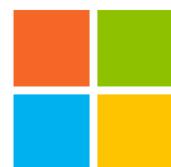
- Unrealistic deadlines, e.g., it is imposed by someone external to the technical staff
- Requirements change (too) often
- Effort and resources have been estimated in an overly optimistic way,
- Risks have not been taken into account from the start of the project.
  - Risks can be technical or human difficulties
- **Communication problems among the staff members**
- **Difficulty by the management to recognize recurrent delays and take immediate action**
- **Subversive stakeholders**

# Software Project Management (5): Delay and Failure... **And Another?**

---



- **Nokia is acquired by Microsoft...**
- **A lot of developers do not want this...**
- **So... They leave!**
  
- **Those who remain become uncooperative with new partners → *Subversive behavior happens!***



**Microsoft**

**NOKIA**

*Consequence? Nokia is virtually out of the cells. Market ☹*

---

# Roadmap

---



- Who is the project manager?
- Software Project Management
- **A scenario example**
- Project Planning & Reporting
- Risks and People Management



## A scenario example (1)

---

- Assume that we have to develop some software within 9 months
- After a while, through an accurate analysis and estimation of risks, you realize that it requires at least 14 months
- What do you do?



## A scenario example (2)

---

- **A possible alternative**

Allocate new budget and include additional people to match allotted timeline



## A scenario example (2)

---

- **A possible alternative**

Allocate new budget and include additional people to match allotted timeline

**Any guesses about this?**



## A scenario example (2)

---

- **A possible alternative**

Allocate new budget and include additional people to match allotted timeline

**WRONG** ☹





## A scenario example (2)

---

- **A possible alternative**

Allocate new budget and include additional people to match allotted timeline

**WRONG ☺ - Brooke's Law:**

“adding manpower to a late software project makes it later.”

**Brooke's Law, a chorollary:**

“9 women can't make a baby in 1 month.”



## A scenario example (3)

---

- **A reasonable alternative**
- Plan for two releases, the first one after 9 months with a limited set of functionalities
  - ... The rest will be eventually completed, e.g., within 14 months

# A scenario example: Lessons Learned

---



- Software Project Management needs **flexibility and adaptation**
- **Adapting means software and people**
- Management in Sw.Eng. is **as much a work of improvisation than precision**
- Sometimes (or actually most of the times) **products may well have to be ``good-enough''**

# Roadmap

---



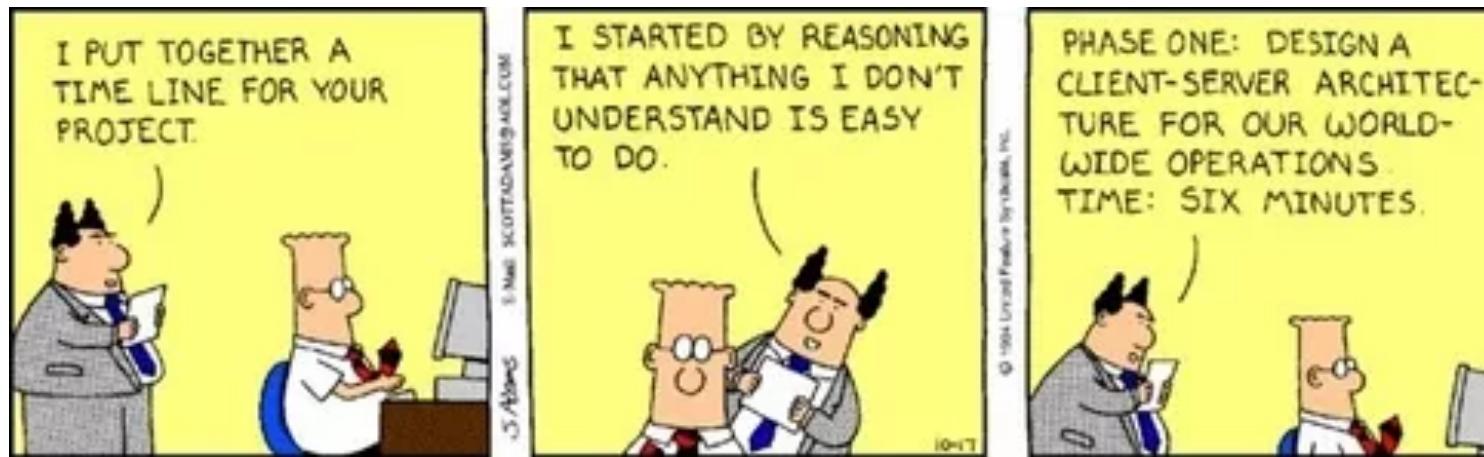
- Who is the project manager?
- Software Project Management
- A scenario example
- **Project Planning & Reporting**
- Risks and People Management

# The planning process

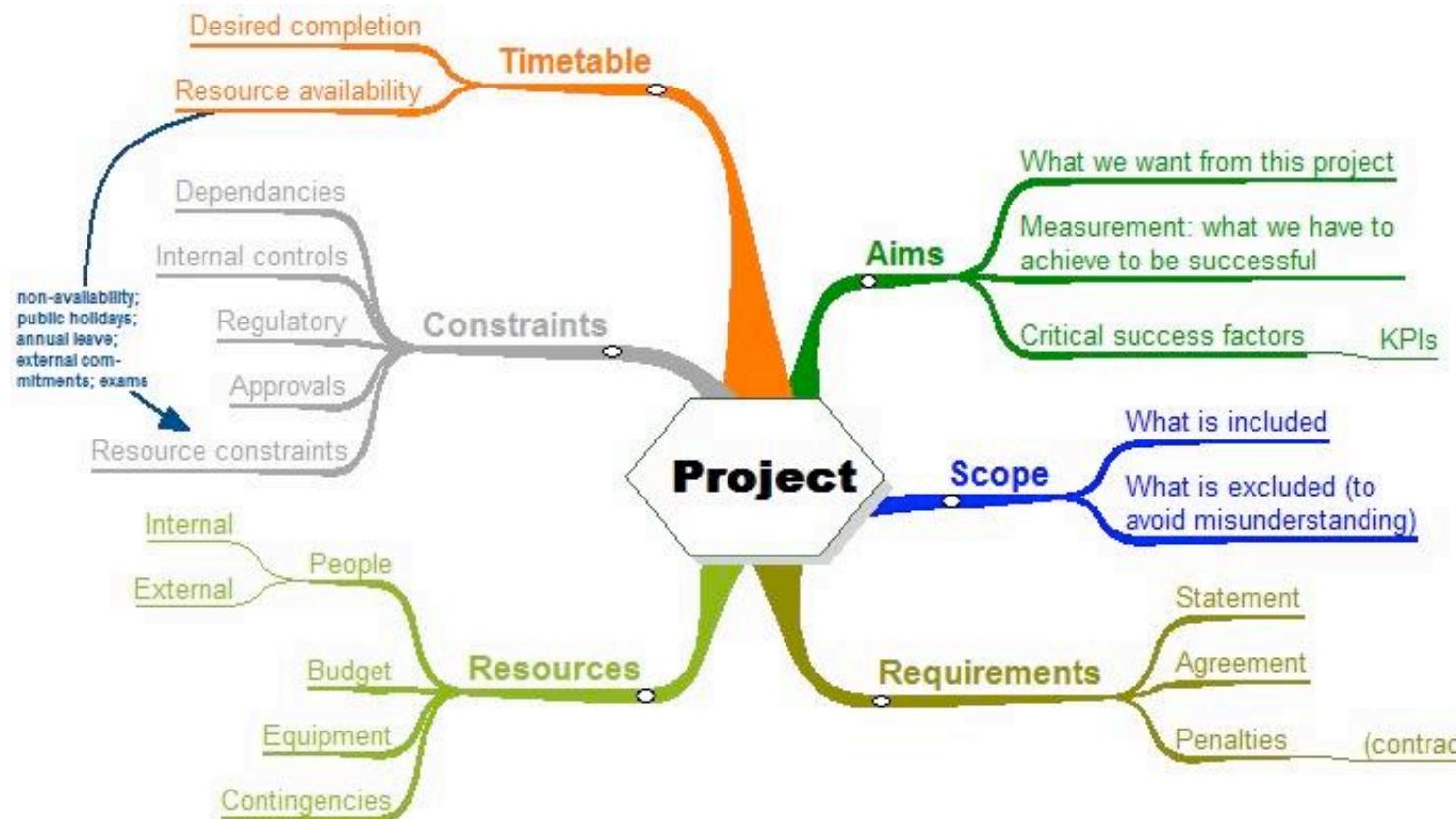
---



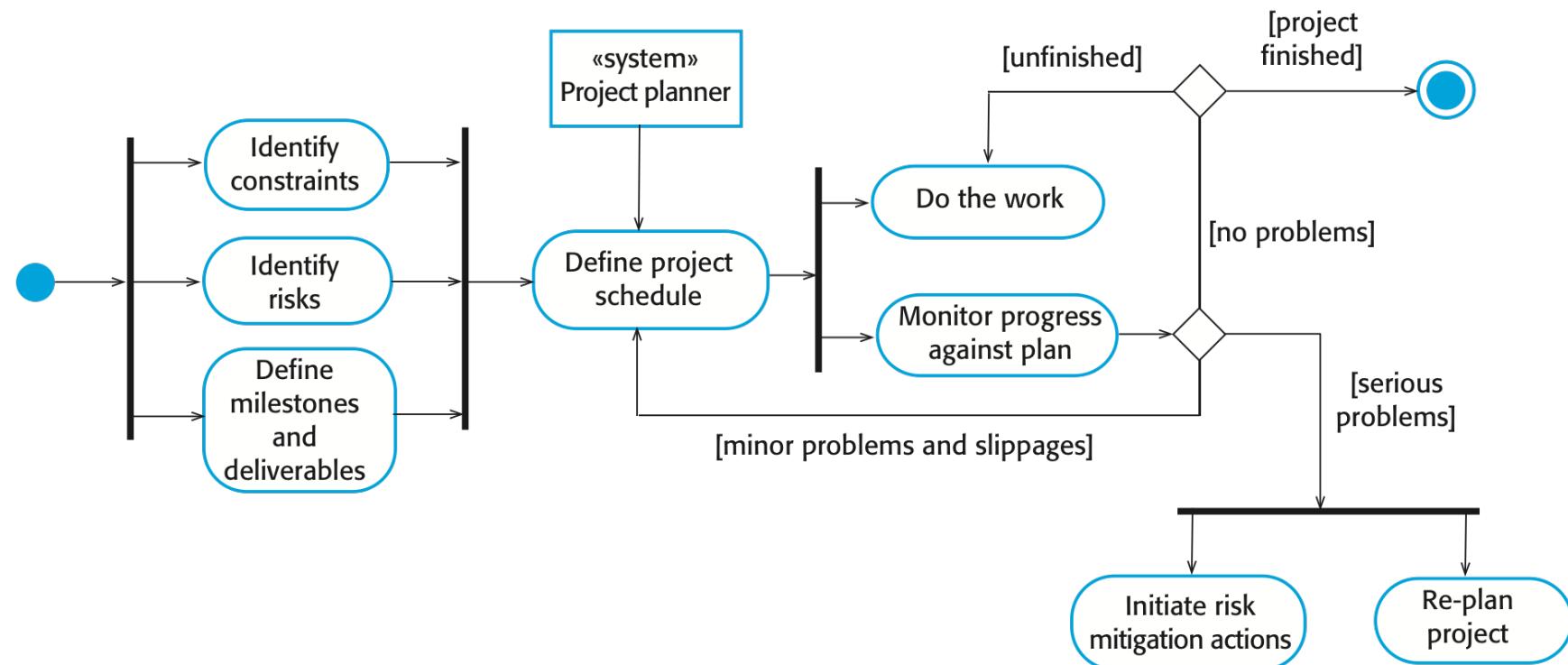
- ✧ Project planning is an iterative process that starts when you create an initial project plan during the project startup phase.
  - ✧ Plan changes are inevitable.
    - As more information about the system and the project team becomes available during the project, you should regularly revise the plan to reflect requirements, schedule and risk changes.
    - Changing business goals also leads to changes in project plans. As business goals change, this could affect all projects, which may then have to be re-planned.
-



# Project planning: A Mind-Map



# The project planning process



# PM terms

---



- Tasks: activities which must be completed to achieve the project goal
  - Milestones: are points in the schedule against which you can assess progress, for example, the handover of the system for testing.
  - Deliverables: are work products that are delivered to the customer, e.g. a requirements document for the system.
-

# Project scheduling

---



- ✧ Project scheduling
  - ✧ deciding how the work in a project will be organized as separate tasks,
  - ✧ when and how these tasks will be executed.
- ✧ Estimates for
  - ✧ the calendar time needed to complete each task,
  - ✧ the effort required
  - ✧ who will work on the tasks that have been identified.
  - ✧ estimate the resources needed to complete each task (e.g., disk or server space, time required on specialized hardware, simulators, travel budget)

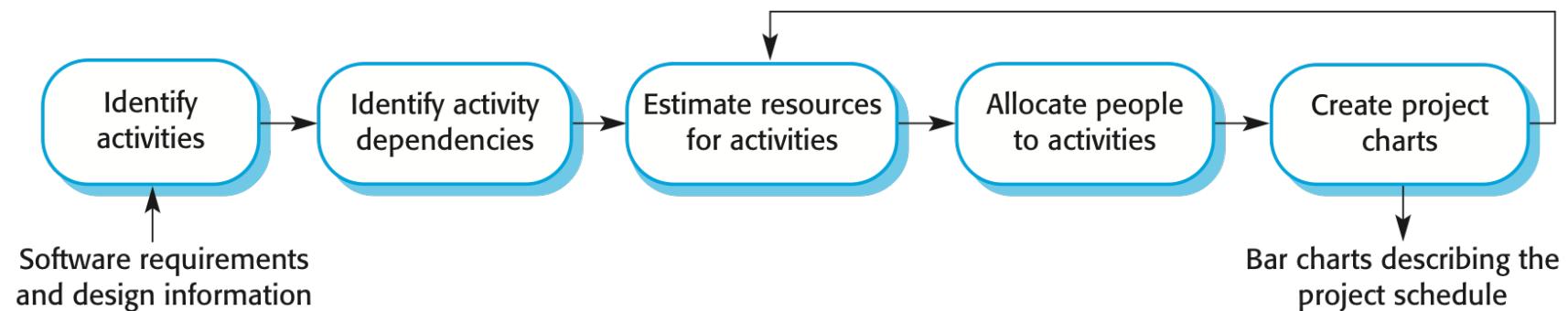
# Project scheduling activities

---



- ✧ Split project into tasks and estimate time and resources required to complete each task.
  - ✧ Organize tasks concurrently to make optimal use of workforce.
  - ✧ Minimize task dependencies to avoid delays caused by one task waiting for another to complete.
  - ✧ Dependent on project managers intuition and experience.
-

# The project scheduling process



# Scheduling problems



- ✧ Estimating the difficulty of problems and hence the cost of developing a solution is hard.
- ✧ Productivity is not proportional to the number of people working on a task.
- ✧ Adding people to a late project makes it later because of communication overheads.
- ✧ The unexpected always happens. Always allow contingency in planning.

# Schedule representation

---



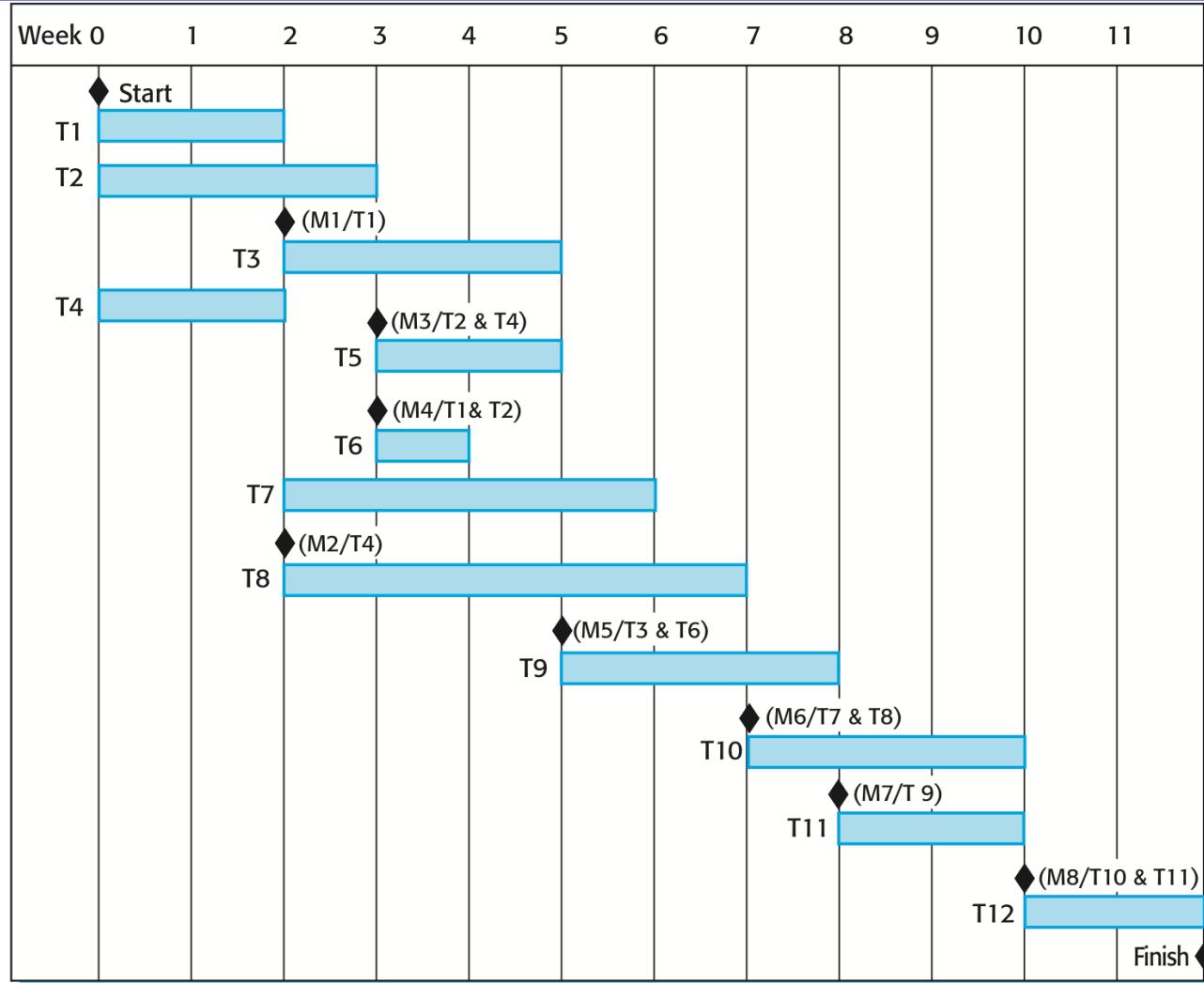
- ❖ Graphical notations are normally used to illustrate the project schedule.
  - ❖ These show the project breakdown into tasks. Tasks should not be too small. They should take about a week or two.
  - ❖ Bar charts are the most commonly used representation for project schedules. They show the schedule as activities or resources against time.
-

# Tasks, durations, and dependencies

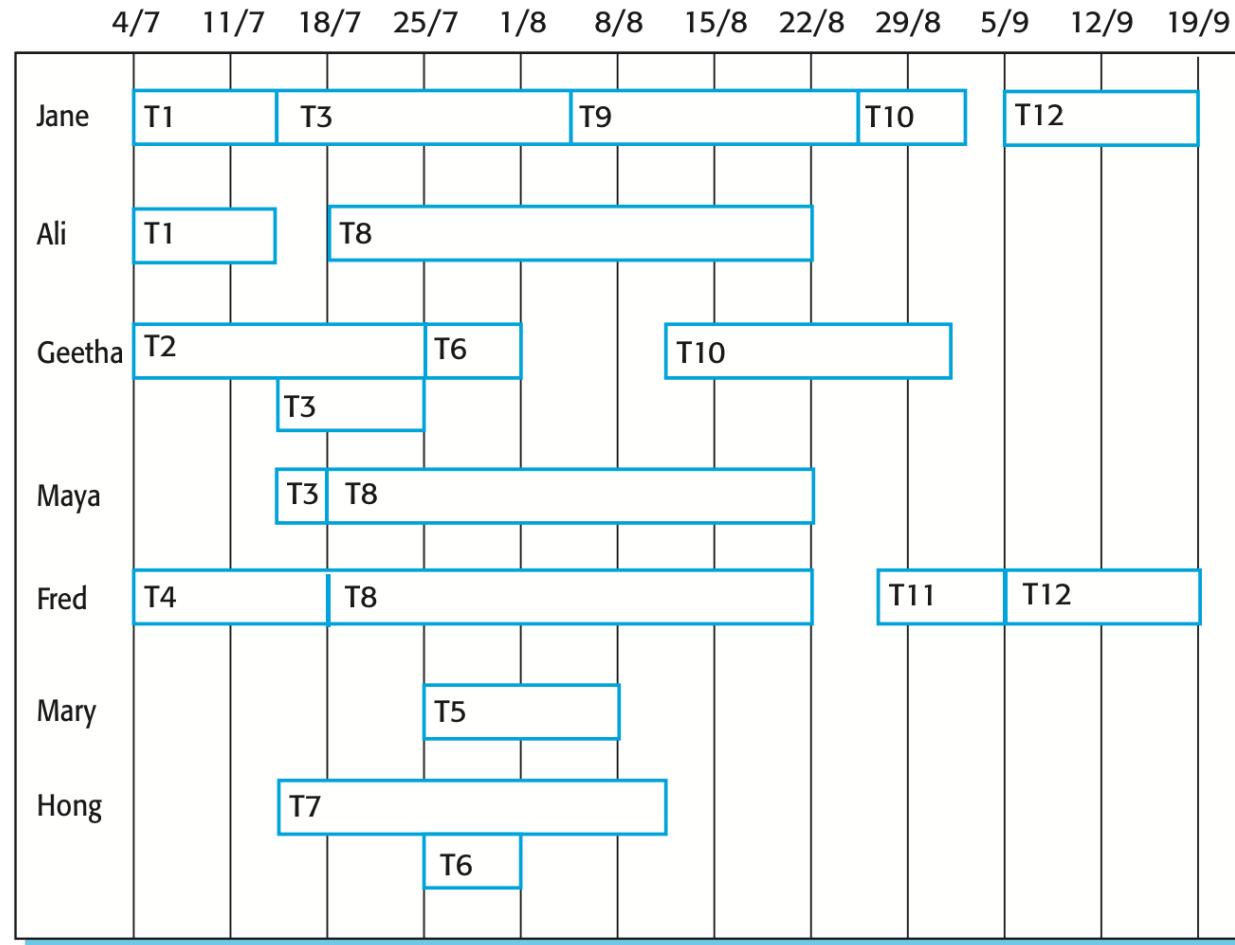


Task	Effort (person-days)	Duration (days)	Dependencies
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)

# Activity bar chart



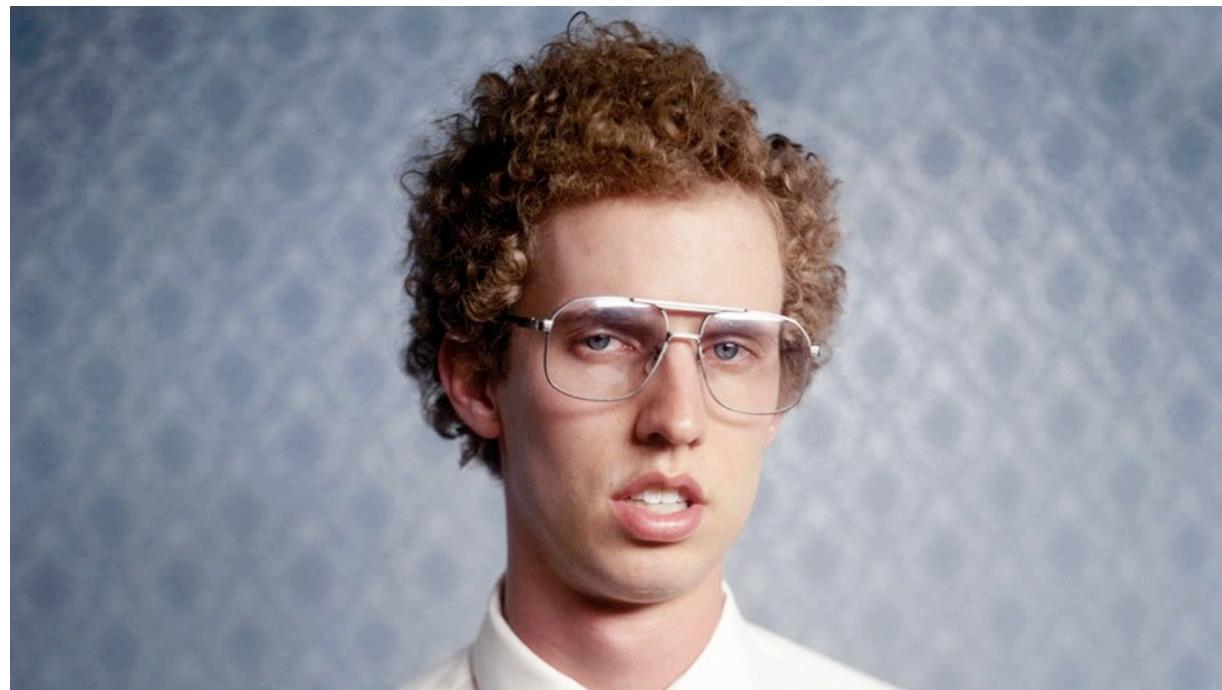
# Staff allocation chart





---

# Any Questions?

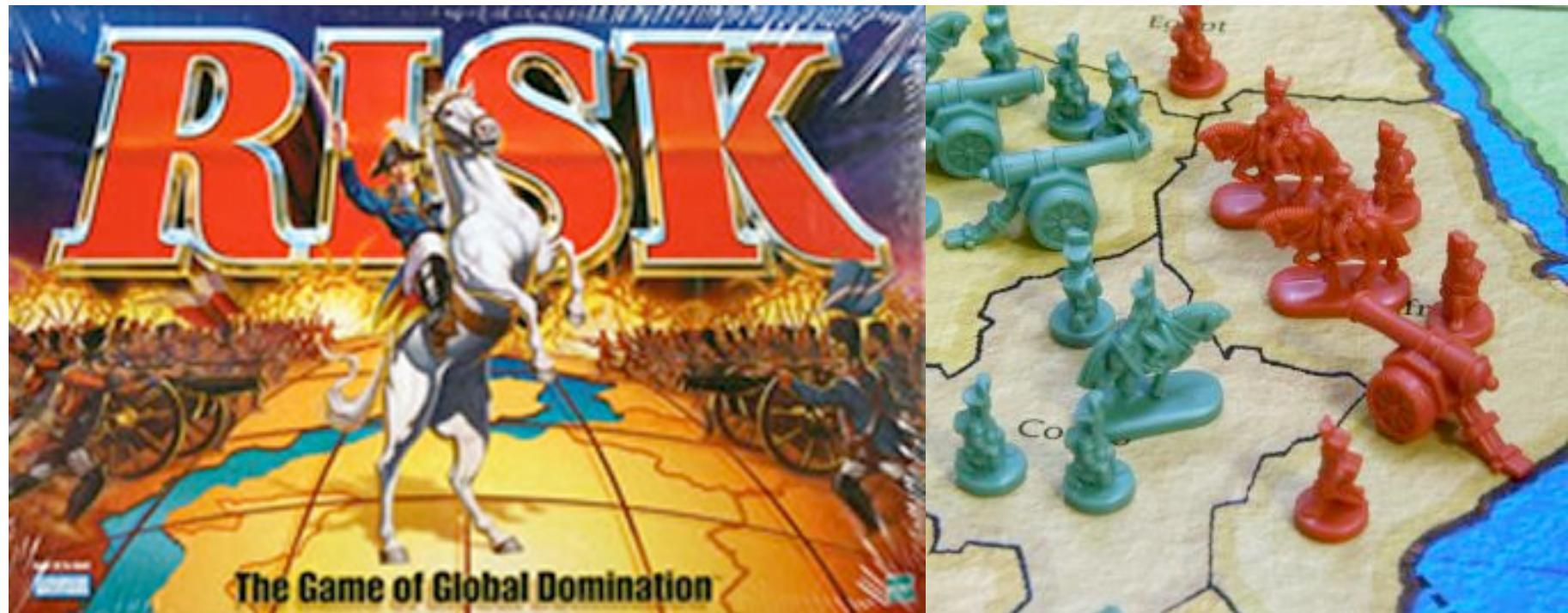


# Roadmap



- Who is the project manager?
- Software Project Management
- A scenario example
- Project Planning & Reporting
- **Risks and People Management**

# What are Software Project Risks?



*All the variables you evaluate when making a move across the software lifecycle...*

# Risk Management

---



- More precisely risk **avoidance**
- Connected to hazards
  - ▶ A Hazard is any **real or potential** condition that can **cause injury**, illness, or death to personnel; damage to or loss of a system, equipment or resource; or damage to the environment.



# Definition of Risk

---

- A risk is a potential problem – it might happen and it might not
- Conceptual definition of risk
  - ▶ Risk concerns future happenings
  - ▶ Risk involves change in mind, opinion, actions, places, etc.
  - ▶ Risk involves choice and the uncertainty that choice entails
- Two characteristics of risk
  - ▶ Uncertainty – the risk may or may not happen, that is, there are no 100% risks (those, instead, are called constraints)
  - ▶ Loss – the risk becomes a reality and unwanted consequences or losses occur



# Top ten risk factors

---



- Personnel shortfall
- Unrealistic schedule/budget
- Wrong functionality
- Wrong user interface
- Goldplating
- Requirements volatility
- Bad external components
- Bad external tasks
- Real-time shortfalls
- Capability shortfalls



# Risk Categorization – Approach #1

---

- Project risks
  - ▶ They threaten the project plan
  - ▶ If they become real, it is likely that the project schedule will slip and that costs will increase
- Technical risks
  - ▶ They threaten the quality and timeliness of the software to be produced
  - ▶ If they become real, implementation may become difficult or impossible
- Business risks
  - ▶ They threaten the viability of the software to be built
  - ▶ If they become real, they jeopardize the project or the product

(More on next slide)



## Risk Categorization – Approach #1 (continued)

---

- Sub-categories of Business risks
  - ▶ **Market risk** – building an excellent product or system that no one really wants
  - ▶ **Strategic risk** – building a product that no longer fits into the overall business strategy for the company
  - ▶ **Sales risk** – building a product that the sales force doesn't understand how to sell
  - ▶ **Management risk** – losing the support of senior management due to a change in focus or a change in people
  - ▶ **Budget risk** – losing budgetary or personnel commitment



# Reactive vs. Proactive Risk Strategies

---

- Reactive risk strategies
  - ▶ "Don't worry, I'll think of something"
  - ▶ The majority of software teams and managers rely on this approach
  - ▶ Nothing is done about risks until something goes wrong
    - The team then flies into action in an attempt to correct the problem rapidly (fire fighting)
  - ▶ Crisis management is the choice of management techniques
- Proactive risk strategies
  - ▶ Steps for risk management are followed (see next slide)
  - ▶ Primary objective is to avoid risk and to have a contingency plan in place to handle unavoidable risks in a controlled and effective manner

# Steps for Risk Management



- 
- 1) Identify possible risks; recognize what can go wrong
  - 2) Analyze each risk to estimate the probability [L,M,H] that it will occur and the impact (i.e., damage) that it will do if it does occur
  - 3) Rank the risks by probability and impact
    - Impact may be negligible, marginal, critical, and catastrophic
  - 4) Develop a contingency plan to manage those risks having high probability and high impact



# Risk types and examples

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget.	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project.	High	Catastrophic
Key staff are ill at critical times in the project.	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused.	Moderate	Serious
Changes to requirements that require major design rework are proposed	Moderate	Serious
The organization is restructured so that different management are responsible for the project	High	Serious
The database used in the system cannot process as many transactions per second as expected	Moderate	Serious

# Strategies to help manage risk: examples



Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other's jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.

# Strategies to help manage risk



Risk	Strategy
Organizational restructuring	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business.
Database performance	Investigate the possibility of buying a higher-performance database.
Underestimated development time	Investigate buying-in components; investigate use of a program generator.

# Managing People is Key!



- Software engineering means people, not things
- Civil Engineering is (arguably) easier to manage (less uncertainty)
- People need to be part of the whole, they are skilled professionals not cogs in a machine
- People can become risks as well, e.g., employee turnover, organizational rebellion
- Consistently create a group or community

# Activities involved in managing people

---



- Selecting staff
  - Motivating people
  - Managing groups
  - The people capability maturity model
-

# People management factors



- Consistency
  - ▶ Team members should all be treated in a comparable way without favourites or discrimination.
- Respect
  - ▶ Different team members have different skills and these differences should be respected.
- Inclusion
  - ▶ Involve all team members and make sure that people's views are considered.
- Honesty
  - ▶ You should always be honest about what is going well and what is going badly in a project.

