F1I018: INGEGNERIA DEL SOFTWARE

Developing a Project Plan for Your Application

Lecturer: Prof. Henry Muccini
Università degli Studi dell'Aquila

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Università degli Studi dell'Aquila

COPYRIGHT NOTICE

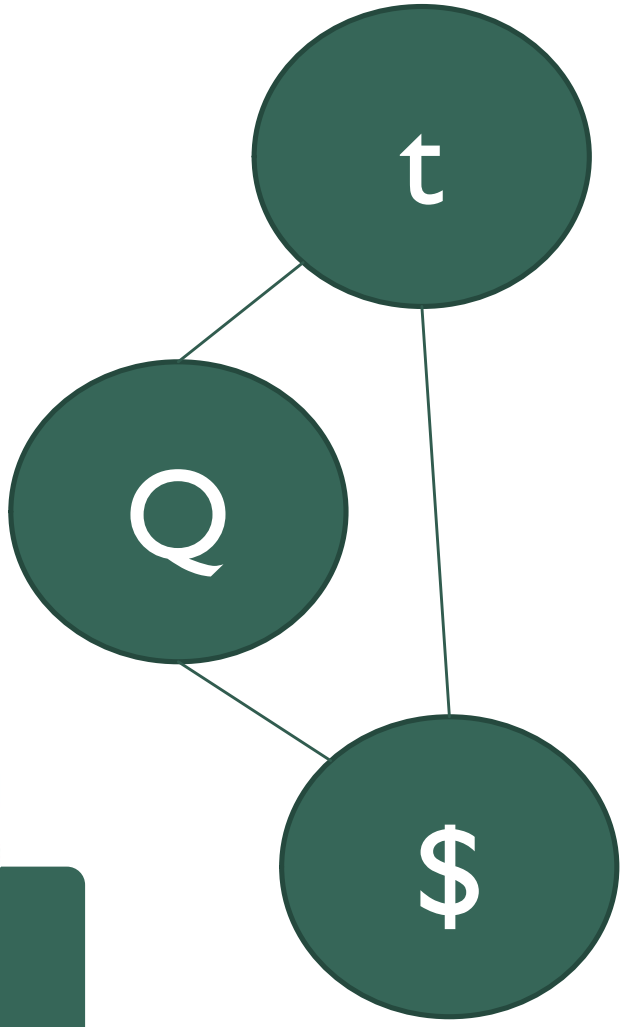
- Il materiale riportato in queste slide puo' essere riutilizzato, parziale o totalmente, a patto che le fonti e gli autori vengano citati

Henry Muccini

DEVELOPING A PROJECT PLAN FOR YOUR APPLICATION

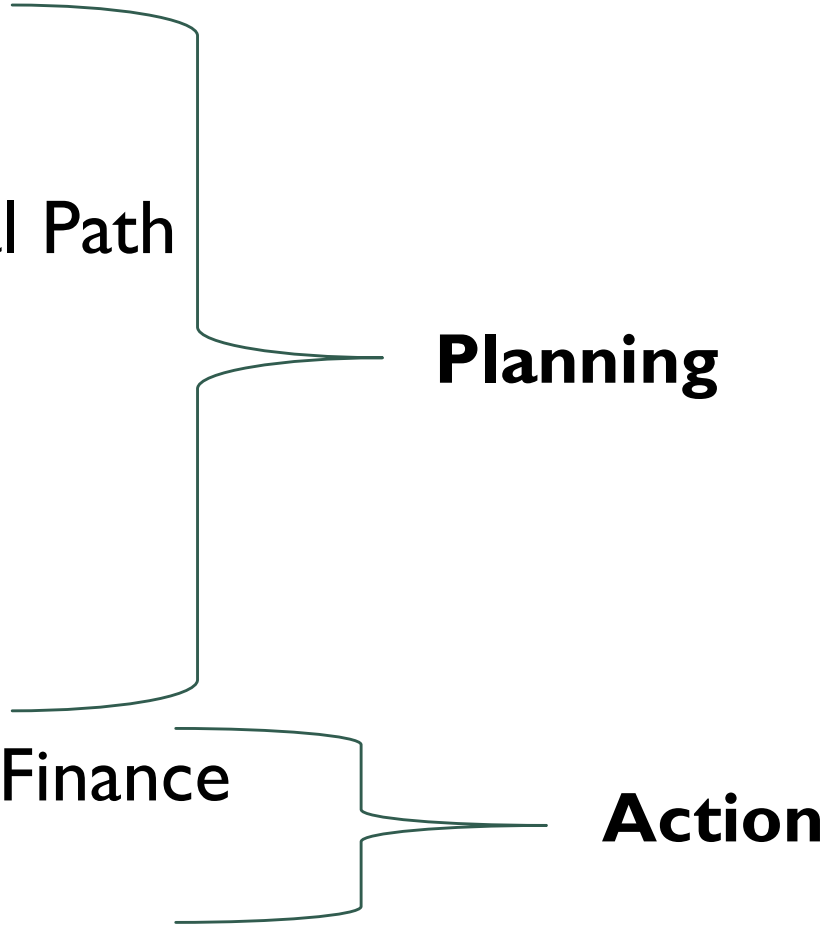
Project Planning activities

- Agreement - 1
- Planning - 7
- Action -3
- Review - 1



Principles of Project Planning

1. **Agreement**
2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks
9-10. Monitor Progress and Finance
11. Reschedule
12. **Review**

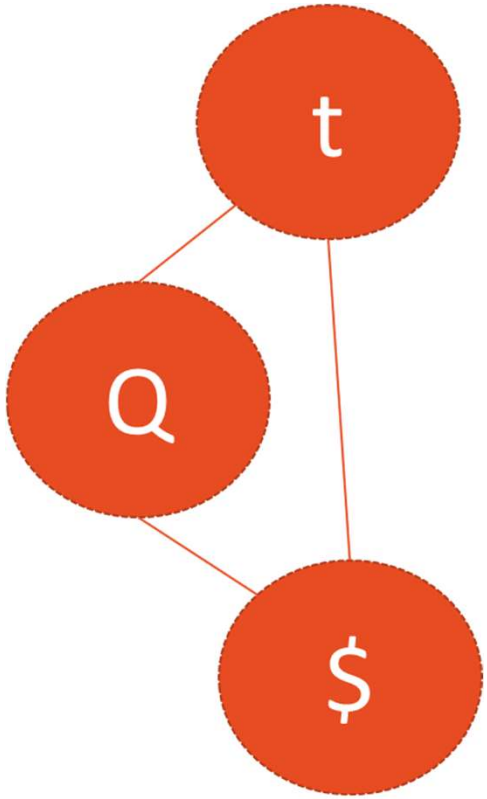


1. Agreement

Agree the succes criteria and constraints with all the customers, in writing

The output is a PID (Project Initiation Document)

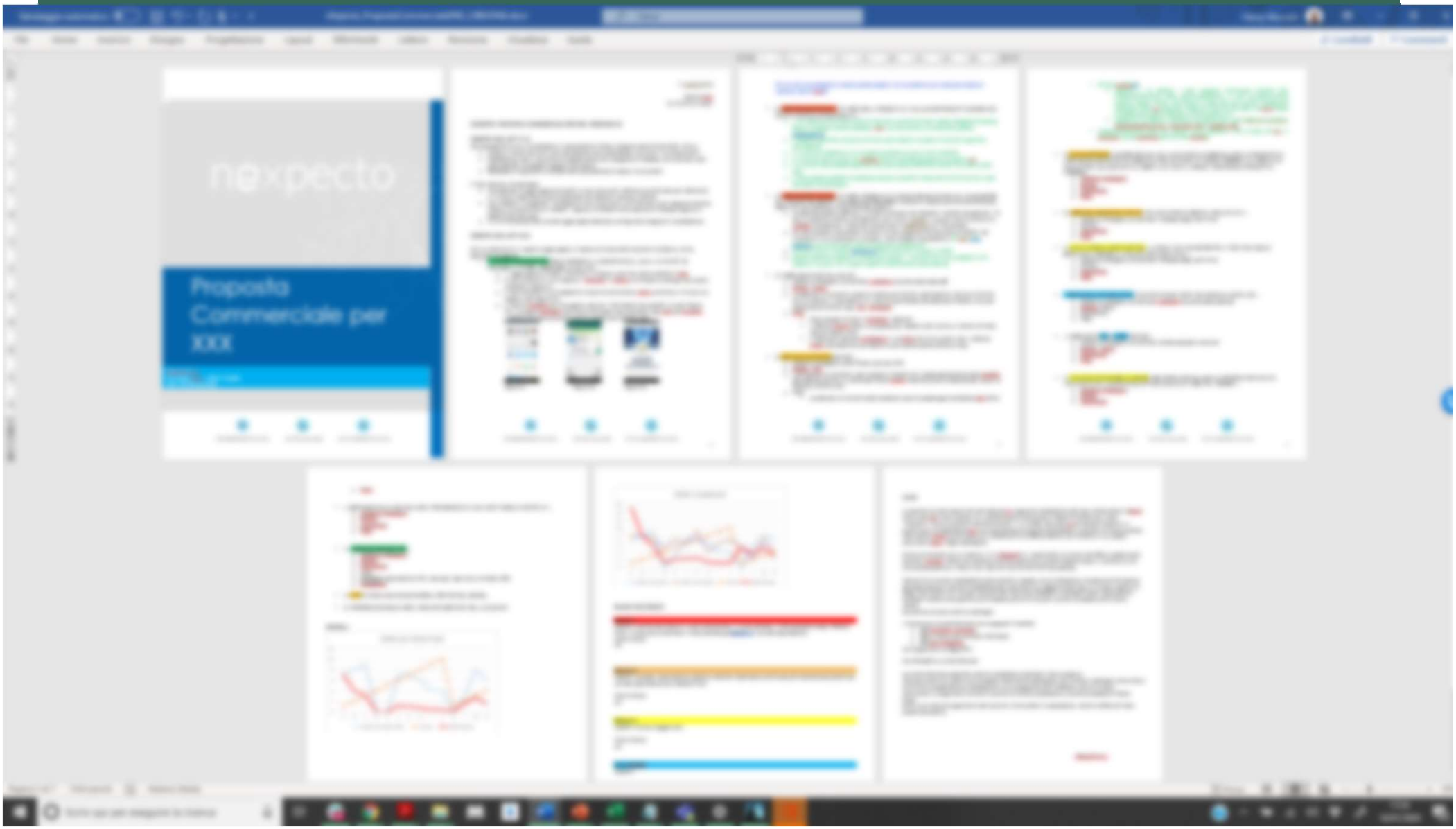
- Do not say MayBe!!!
- Define the key driver
  - The most important
  - Between Q, t, and $



Agreement

STESURA DI UN CONTRATTO

Agreement



PUNTI FONDAMENTALI DI UN CONTRATTO

- Data e Versione
- Autore
- Definizione di fasi incrementali di sviluppo
- Servizi:
  - Descrizione servizio
  - Condizioni necessarie per la realizzazione del servizio
  - Delivery type
- Tempistiche del progetto
  - Tempo 0
  - Milestones
  - #riunioni coperte dal buget
- Costi del progetto
  - Modelli di business
  - Costi di Manutenzione
  - Piano per lavori futuri

Planning

2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks

Planning: list the tasks

[►] CREATE A TASK LIST

- **Brainstorming:** Visualize what you need to do at every stage of the site creation process.
- **Work break down:** Then break that down into tasks that need to be accomplished.
  - Note whether a task is dependent on the completion of an earlier one.
- **Ask an expert**

- **Type of tasks:**
  - **Sw development** (associated to requirements and features);
  - **Hw acquisition tasks** (devices required for realizing the hw/sw infrastructure);
  - **Admin tasks** (approvals, internal procedures);
  - **Communication tasks** (internal, with customer, with clients, …)
  - Etc.

Planning: list the tasks

- Sw tasks:
  - realizzazione del modello statistico;
  - Realizzazione del modello di ottimizzazione;
  - Realizzazione del modello adattivo;
  - Sw per chioschi, big screens, readers, …
- Hw tasks:
  - Selezione e configurazione chioschi, big screens, readers, counters, etc
  - Creazione infrastruttura hw
- Admin tasks:
  - Gestione contratto con Uffizi
  - Acquisizione risorse umane
- Communication:
  - Con Direttore e responsabili Uffizi
  - Con UnivAQ
  - Team

Planning: costs > team

[►] ASSEMBLE A TEAM

- You may not have all the skills or time to do everything that needs to be done. Here are some common roles:

1. **Team Leader:** Every team needs someone who is able to make the final decisions.
2. **Designer:** Depending on the scale of the project, besides the familiar graphic designer, the team may need a user experience (UX) designer, creative designer, or interaction designer.
3. **Client-side developer:** Also known as a user interface (UI) designer, this developer specializes in creating interfaces that function efficiently on the iPad/iPhone platform.
4. **Server-side developer:** If you are building a dynamic site, you will want to have a developer skilled in programming languages like PHP or JavaScript to handle the server side code.

5. **Database administrator:** Depending on the complexity of the site, you may also need a database specialist to set up and maintain a database

- Hardware Integrator and IoT expert
- Statistics expert
- Optimization expert
- Mobile App development expert
- Web development expert
- Data Scientist
- Team leader
- UI developer
- Backend developer
- DB administrator

Principles of Project Planning

1. Agreement (studio di fattibilità, SOTA, contratto)
2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks
9-10. Monitor Progress and Finance
11. Reschedule
12. Review

COST ESTIMATE

HALF WAY BETWEEN AVERAGE AND WORST CASE

Project plans are based on effort estimates!

Simple Estimation Techniques

Discuss

- Guessing
- Parkinson's Law
- Pricing to win
- Budget method

Better Estimation Techniques

Based on experience or hard data collection

- Such techniques requires to "extensively record historical data"

Informal:

- Expert judgment
- Estimation by analogy
- Variation: Delphi method

Formal:

- Algorithmic cost modeling

Informal

Expert judgment

+ Relatively cheap estimation method.
+ Can be accurate if (AND ONLY IF) experts have direct experience with similar systems
- Does not use hard data

Estimation by analogy

+ Accurate if project data available
- **Impossible if** no comparable project has been undertaken.
- Estimates can be inaccurate if details overlooked.
-Subsequent similar projects can be quicker.

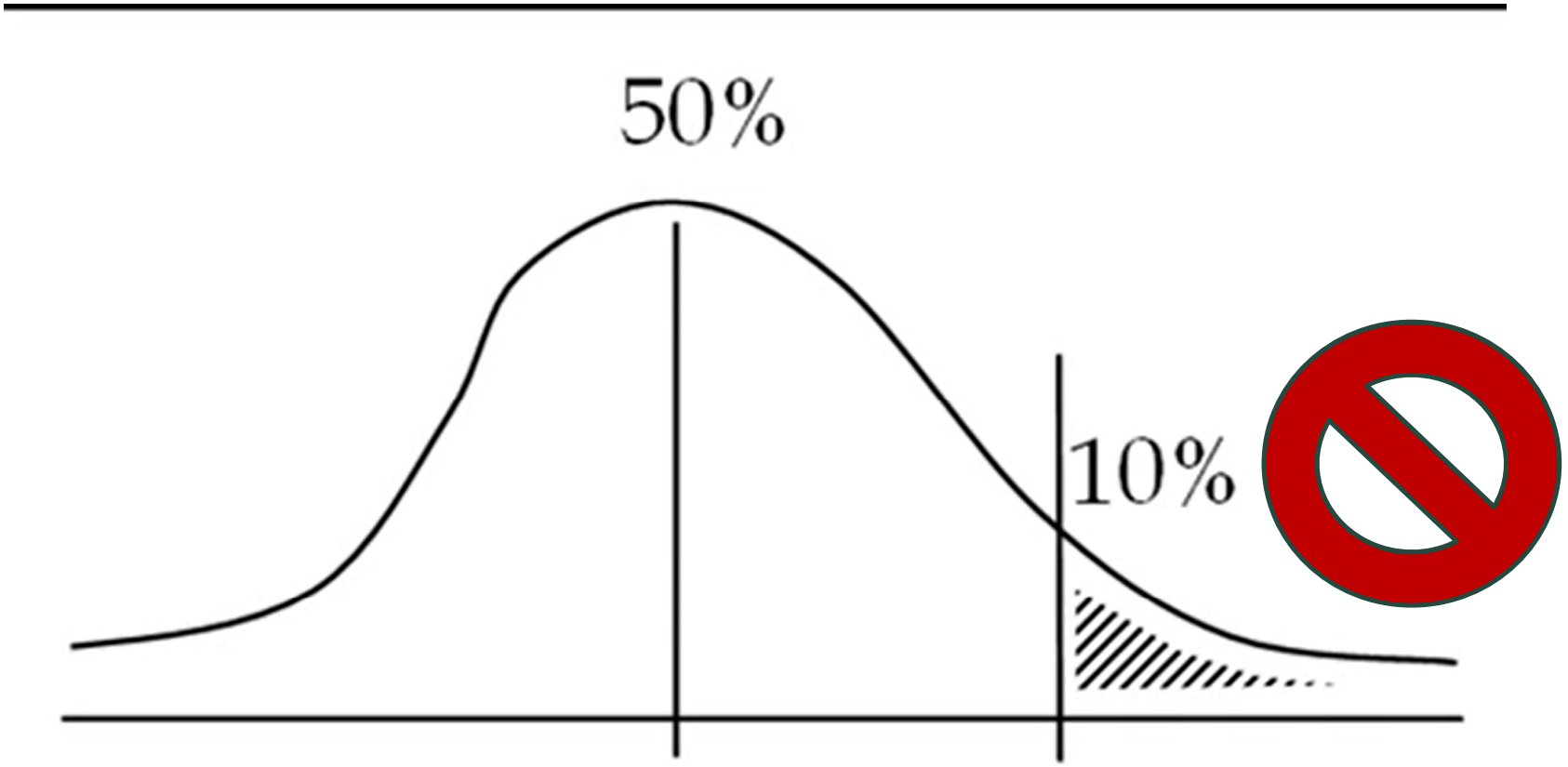
Informal: Delphi Method

Idea: Create a group expert opinion, while counterbalancing personality factors in process

Group of expert estimators + moderator

1. Experts independently create estimates.
2. Moderator collects written estimates from individuals.
3. Estimates are distributed to group.
   - → No names
4. Experts deliver new estimates based on new information from moderator.
5. Continue until consensus is reached.

3. ESTIMATE TIME AND COST



Formal: Algorithmic Cost Modeling

Cost and development time for a project is estimated **from an equation**

Effort estimates are based on size

- Highly inaccurate at start of project

Size is usually given in lines of code, which **not** reflect difficulty

- Some short programs are harder to write than long ones
- Lines of code ≠ effort
  - Not all activities produce code
- Programming Language: Java vs. assembler

Formal: Algorithmic Cost Modeling

$$EFFORT = A \times SIZE^{B} \times M$$

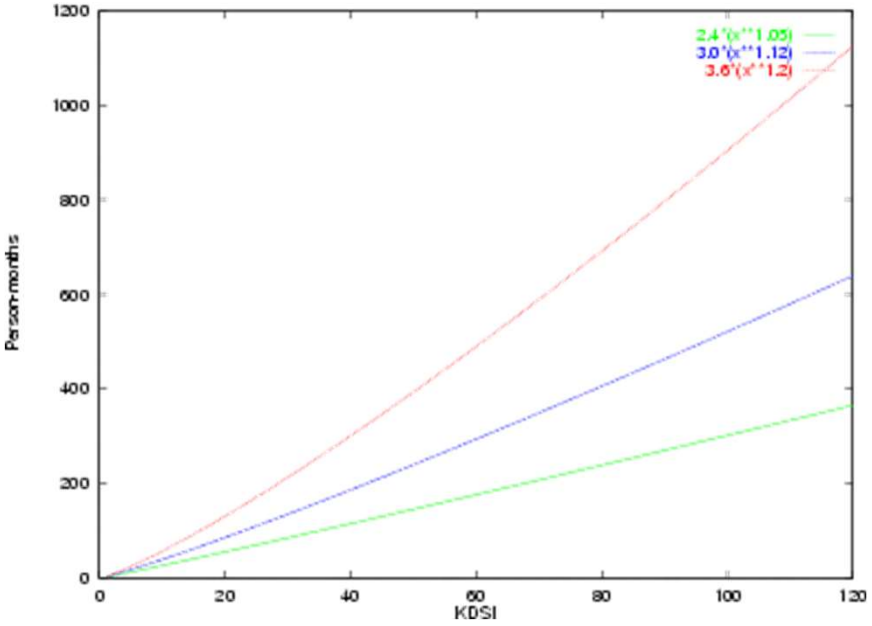
- A = is a constant factor that depends on local organizational practices and the type of software that is developed.
- SIZE = code size or function points
- B = expresses the non-linearity of costs with project size. As the size of the software increases, extra costs are emerged. The value of exponent B usually lies between 1 and 1.5.
- M = is a multiplier made by combining process, product and development attributes, such as the dependability requirements for the software and the experience of the development team.

Formal: Algorithmic Cost Modeling

$$EFFORT = A \times SIZE^{B} \times M$$

Basic COCOMO Formula

- Organic mode: PM = 2.4 (KDSI)$^{1.05}$
- Semi-detached mode: PM = 3 (KDSI)$^{1.12}$
- Embedded mode: PM = 3.6 (KDSI)$^{1.2}$



KDSI: thousands of delivered source instructions

PM: person-months

Formal: Algorithmic Cost Modeling

COCOMO -- Time to Develop

- Organic mode: $TDEV = 2.5(PM)^{0.38}$
- Semi-detached mode: $TDEV = 2.5(PM)^{0.35}$
- Embedded mode: $TDEV = 2.5(PM)^{0.32}$

TDEV: time (months) to develop

Example:

- Organic mode project, 32KLOC
- $PM = 2.4(32)^{1.05} = 91$ person months
- $TDEV = 2.5(91)^{0.38} = 14$ months
- $N = 91/14 = 6.5$ people

WORK BREAKDOWN

LIST THE TASKS AND PERT

Work breakdown and Planning

A Work breakdown reflects the decomposition of a project into subactivities down to a level needed for effective planning and control

Project planning involves scheduling all activities such that the **constraints** are satisfied and resource limits are not exceeded.

Activities

Have a beginning and end

- Often marked by milestones
- A milestone is a scheduled event for which some person is held accountable and which is used to measure and control progress.

Consume resources

- e.g. people and computer time

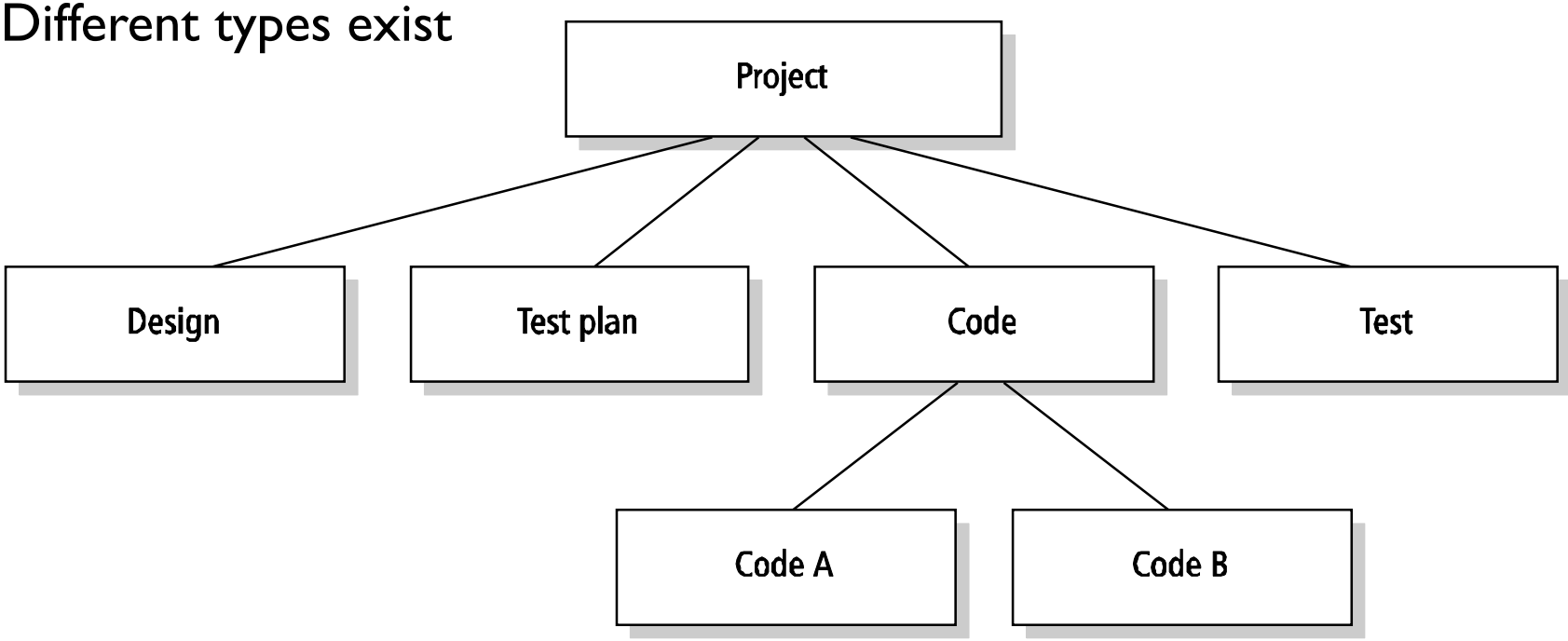
Have dependencies

- e.g. Can't code before we have a specification
- Expressed as constraints, a.k.a. precedence relations
- Usually temporal, but sometimes deliverables

Representing Activities

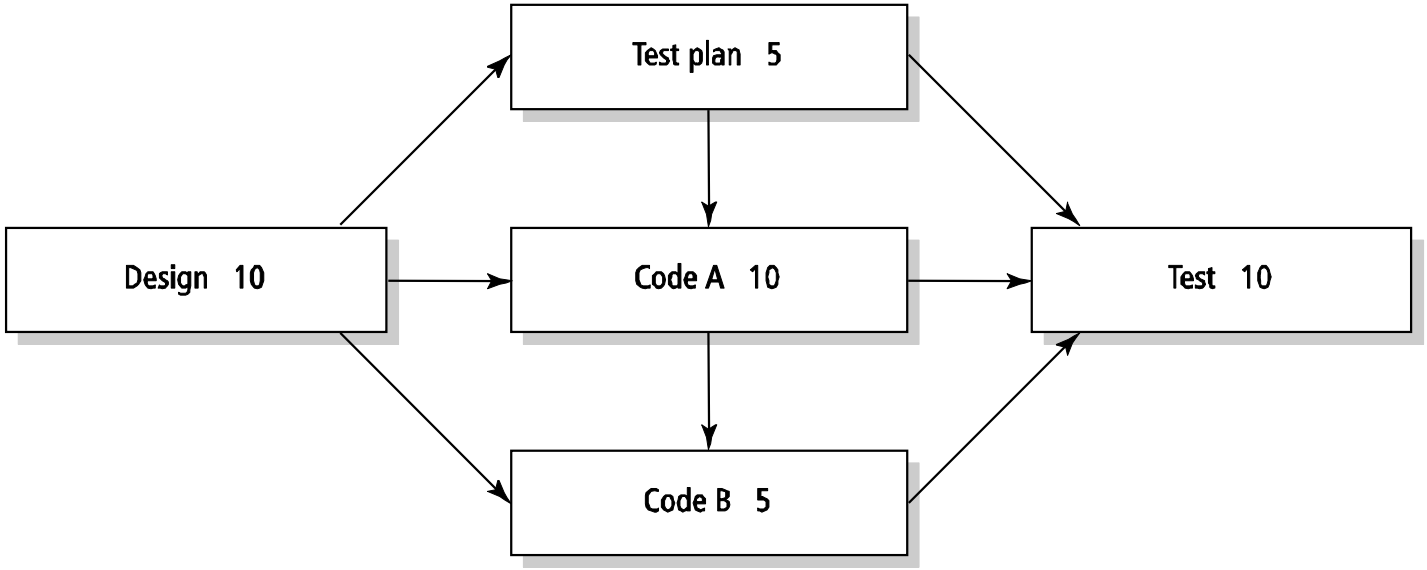
Work Breakdown Structure (WBS)

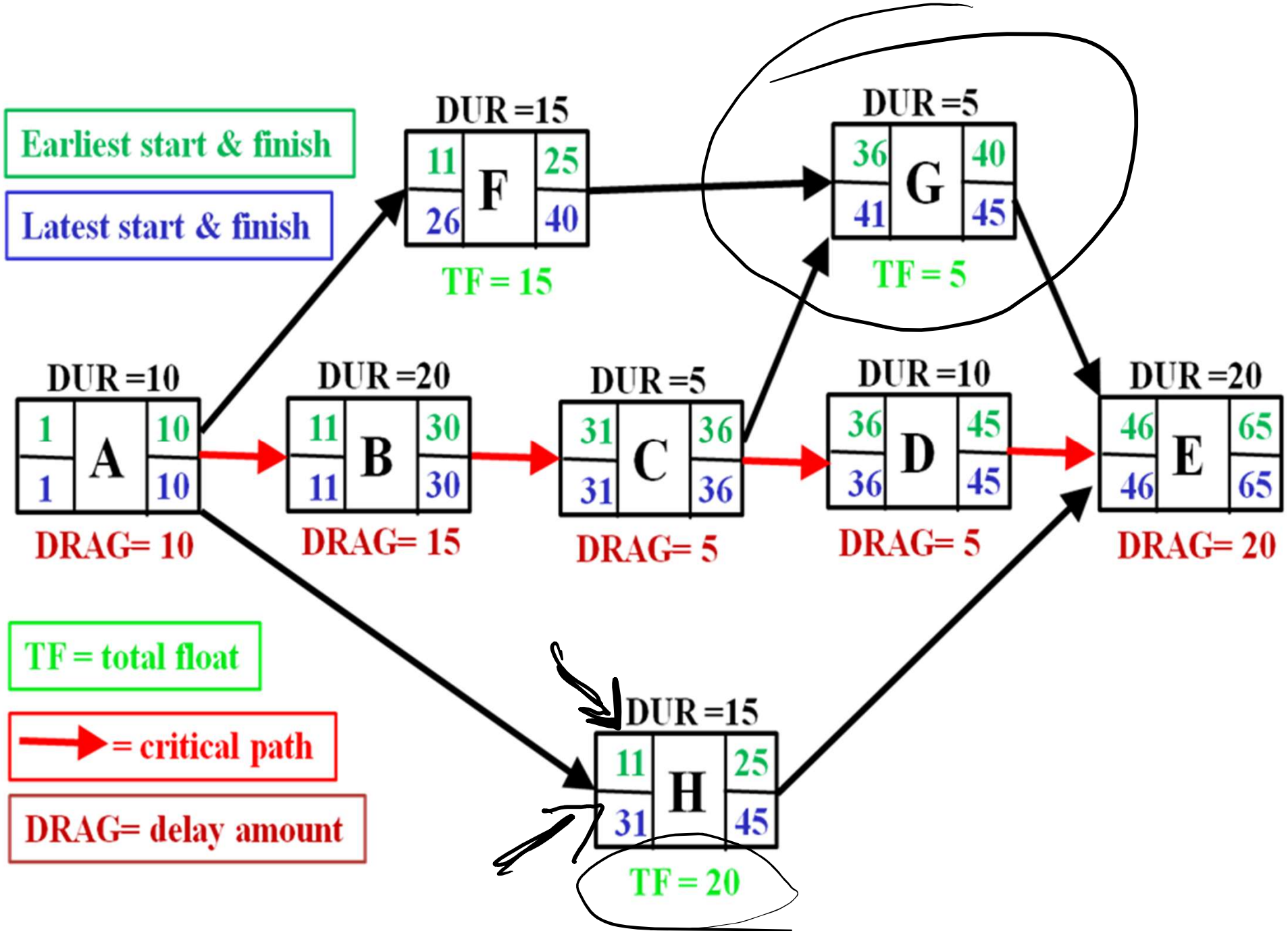
- Graphical representation of a project and constituent activities
- Can be abstract or highly detailed
- Different types exist



PERT CHARTS

- PERT = Program Evaluation and Review Technique
  - First used in 1950s on Polaris missile program
- Most useful for finding dependencies and minimum schedule time




Earliest start & finish
Latest start & finish
TF = total float
= critical path
DRAG= delay amount
DUR =10
1
1
A
10
10
DRAG= 10
DUR =15
11
26
F
25
40
TF = 15
DUR =20
11
11
B
30
30
DRAG= 15
DUR =5
31
31
C
36
36
DRAG= 5
DUR =5
36
41
G
40
45
TF = 5
DUR =10
36
36
D
45
45
DRAG= 5
DUR =20
46
46
E
65
65
DRAG= 20
DUR =15
11
31
H
25
45
TF = 20

Critical Path

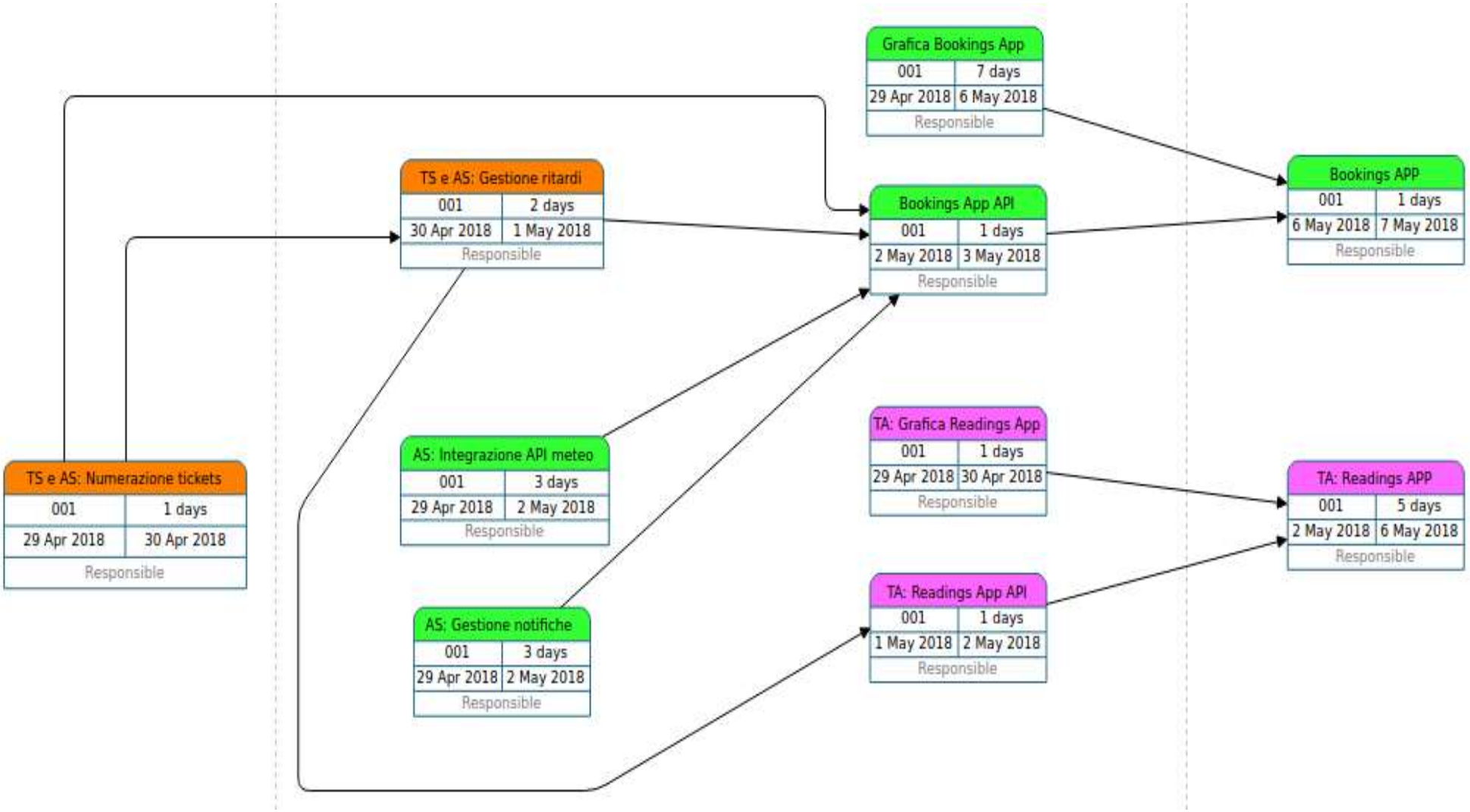
A **critical path** is a sequence of activities without slack time.

- example

If activities on a critical path are delayed, the total project is delayed as well.

Found by performing a breadth-first search while tracking the duration

ESEMPIO REALE DI PERT



Gantt Chart

Invented by Henry Gantt in 1910

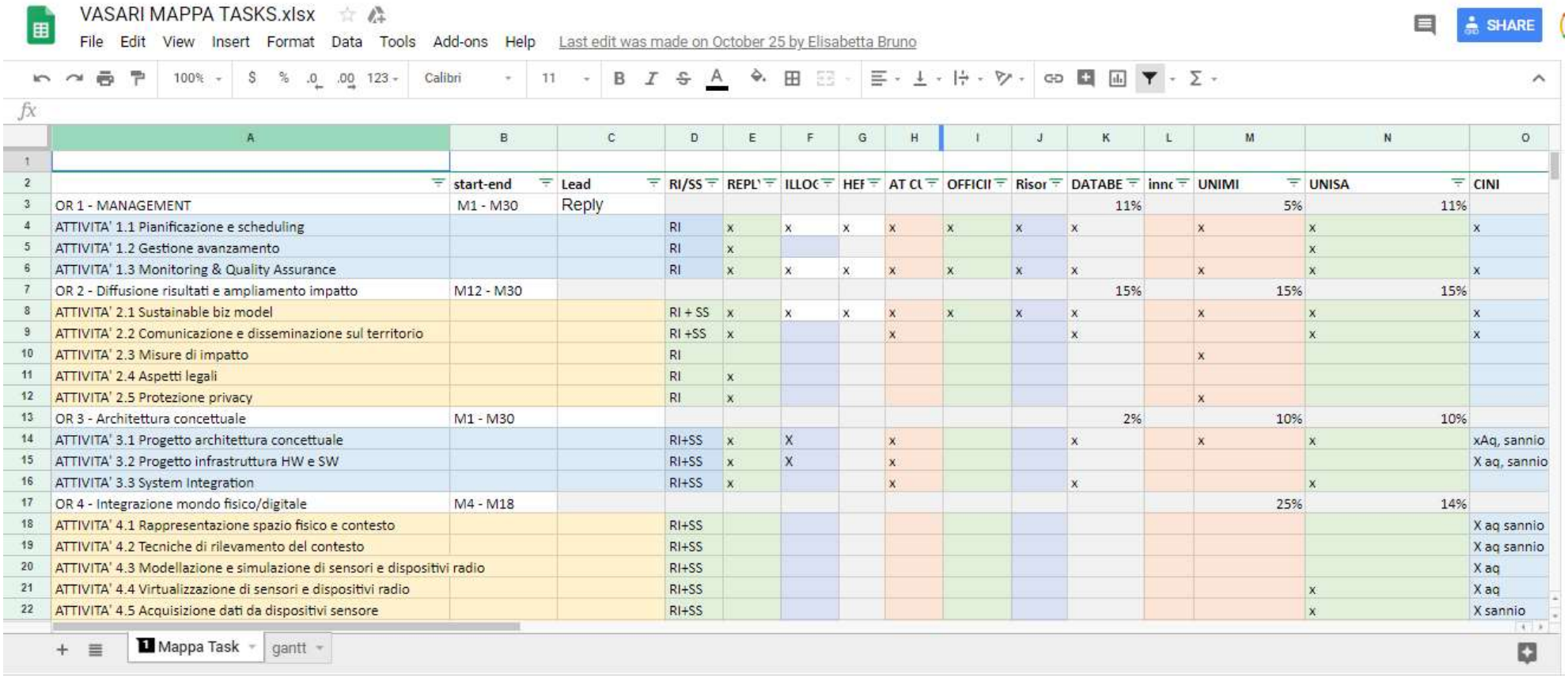
- Engineer and management scientist
- Used on Hoover dam project

Like a bar chart version of PERT chart with **extra features**

- Shows activities on a calendar
- Depicts additional temporal dependencies
  - Start activity after the start of…
  - Start activity before the end of…
- Allocate resources to tasks
  - Budgeting
  - Can ask what-if questions about allocations

GANTT

- https://docs.google.com/spreadsheets/d/1KhxzvuC-IB4mHqAB1DNt1tlvo5RFuLyYu1RYo5IEVAo/edit#gid=560140332



VASARI MAPPA TASKS.xlsx

File Edit View Insert Format Data Tools Add-ons Help Last edit was made on October 25 by Elisabetta Bruno

SHARE

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1															
2		start-end	Lead	RI/SS	REPL	ILLOC	HEF	AT CL	OFFICII	Risor	DATABE	innc	UNIMI	UNISA	CINI
3	OR 1 - MANAGEMENT	M1 - M30	Reply								11%		5%	11%	
4	ATTIVITA' 1.1 Pianificazione e scheduling			RI	x	x	x	x	x	x	x		x	x	x
5	ATTIVITA' 1.2 Gestione avanzamento			RI	x									x	
6	ATTIVITA' 1.3 Monitoring & Quality Assurance			RI	x	x	x	x	x	x	x		x	x	x
7	OR 2 - Diffusione risultati e ampliamento impatto	M12 - M30									15%		15%	15%	
8	ATTIVITA' 2.1 Sustainable biz model			RI + SS	x	x	x	x	x	x	x		x	x	x
9	ATTIVITA' 2.2 Comunicazione e disseminazione sul territorio			RI +SS	x			x			x			x	x
10	ATTIVITA' 2.3 Misure di impatto			RI									x		
11	ATTIVITA' 2.4 Aspetti legali			RI	x										
12	ATTIVITA' 2.5 Protezione privacy			RI	x								x		
13	OR 3 - Architettura concettuale	M1 - M30									2%		10%	10%	
14	ATTIVITA' 3.1 Progetto architettura concettuale			RI+SS	x	X		x			x		x	x	xAq, sannio
15	ATTIVITA' 3.2 Progetto infrastruttura HW e SW			RI+SS	x	X		x							X aq, sannio
16	ATTIVITA' 3.3 System Integration			RI+SS	x			x			x			x	
17	OR 4 - Integrazione mondo fisico/digitale	M4 - M18											25%	14%	
18	ATTIVITA' 4.1 Rappresentazione spazio fisico e contesto			RI+SS											X aq sannio
19	ATTIVITA' 4.2 Tecniche di rilevamento del contesto			RI+SS											X aq sannio
20	ATTIVITA' 4.3 Modellazione e simulazione di sensori e dispositivi radio			RI+SS											X aq
21	ATTIVITA' 4.4 Virtualizzazione di sensori e dispositivi radio			RI+SS										x	X aq
22	ATTIVITA' 4.5 Acquisizione dati da dispositivi sensore			RI+SS										x	X sannio

Mappa Task | gantt

Planning: risk management

2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks

RISK MANAGEMENT

Risk management

Risk management is concerned with **identifying** risks and drawing up **plans to minimise** their effect on a project.

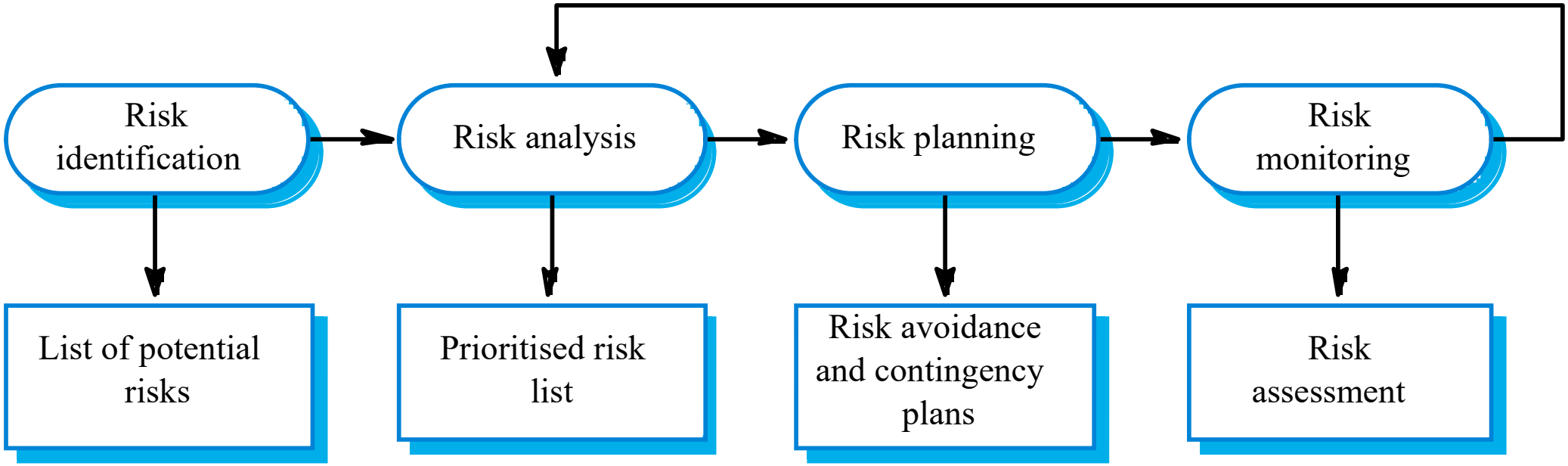
"**A risk is a possible future negative event that may affect the success of an effort**" [VV8.3]

- Project risks affect schedule or resources;
- Product risks affect the quality or performance of the software being developed;
- Business risks affect the organisation developing or procuring the software.

Analogy with "Software Faults and Software Failures"

THE RISK MANAGEMENT PROCESS

[FROM SOMMERVILLE BOOK]



- **PREVENT**
- **MONITOR**
- **FIX**

THE RISK MANAGEMENT PROCESS

1. Risk identification
   - Identify project, product and business risks;

Discuss

2. Risk analysis
   - Assess the likelihood and consequences of these risks;
   - Called "Risk Exposure" in VV

3. Risk planning
   - Draw up plans to avoid or minimise the effects of the risk;
   - From VV:
     - risk avoidance, transfer, acceptance
     - Like in deadlock handling

4. Risk monitoring
   - Monitor the risks throughout the project;

Principles of Project Planning

1. **Agreement**
2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks

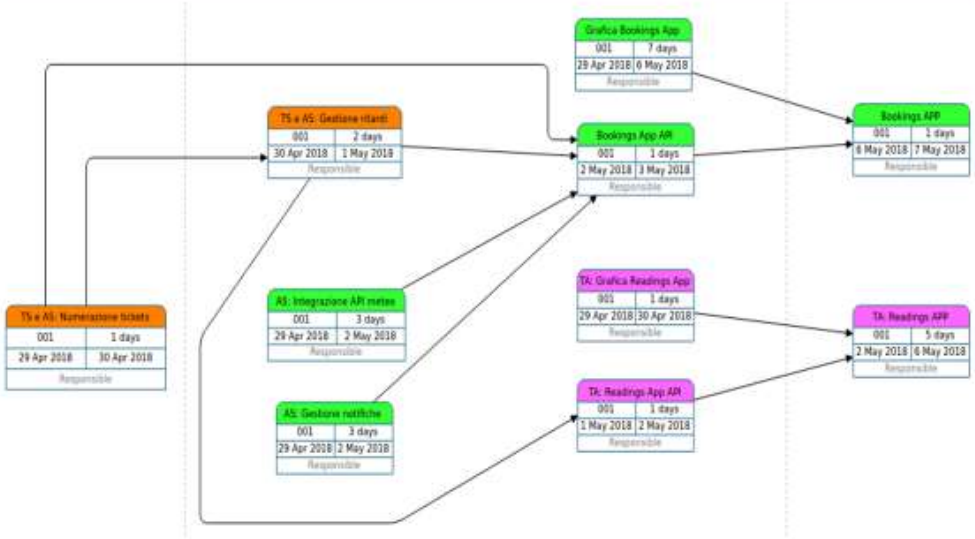
Planning

9-10. Monitor Progress and Finance
11. Reschedule

Action

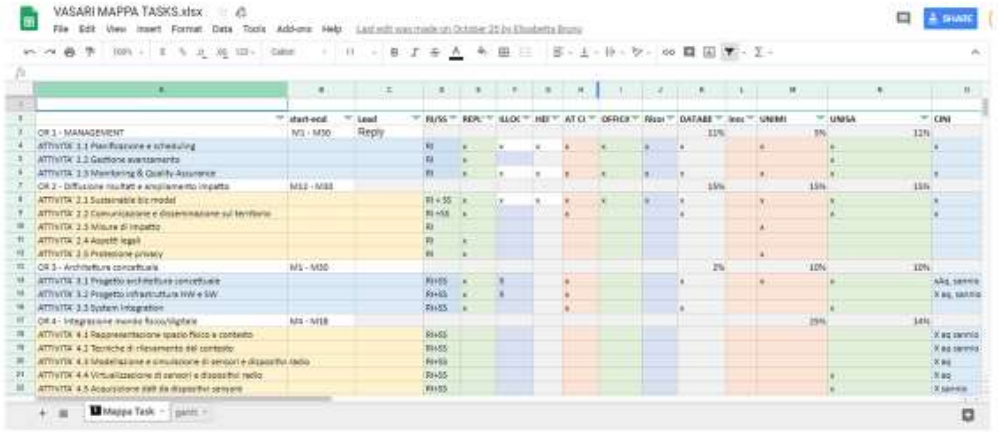
12. **Review**

ESEMPIO REALE DI PERT



GANTT

- https://docs.google.com/spreadsheets/d/1KhxzvuC-lB4mHqAB1DNt1tlvo5RFuLyYu1RYo5lEVAo/edit#gid=560140332



[►] MONITOR AND RESCHEDULE

Esempio 2: Uffizi (chioschi)

17

Review

Good

Bad

Better

Principles of Project Planning

1. Agreement
2. List the tasks
3. Estimate time and Cost
4. Dependencies and Critical Path
5. Crashing
6. GANTT Chart
7. Resource Planning
8. Risks
9-10. Monitor Progress and Finance
11. Reschedule
12. Review