POLITECNICO
MILANO 1863

Software Engineering 2

Project management

M Camilli, E Di Nitto, M Rossi

POLITECNICO
MILANO 1863

Project management

Roadmap

- Overview on project management
- Specific techniques for software cost and effort estimation

Project

Many definitions, expressing similar content, we take the one reported by the PRINCE2 methodology

"a temporary organization (time, resources) that is created for the purpose of delivering one or more business products (scope) according to an agreed Business Case (scope, cost, time, quality, risks)".

https://en.wikipedia.org/wiki/PRINCE2

Project Management

POLITECNICO MILANO 1863

According to PMBOK, project variables include but are not limited to: scope, quality, schedule (time), budget (cost), resources, risks

Project management is used to manage (plan, monitor and control) these variables in order to make the project successful.

A well-directed project ***may*** fail, a badly-directed project ***certainly*** fails

https://www.pmi.org/pmbok-guide-standards/foundational/pmbok

Project Management

POLITECNICO MILANO 1863

Main questions	Main concerns
What problem are you solving?	Scope
How are you going to solve this problem?	Project strategy
What is your plan?	• The work to be done in detail • How long it may take • The resources you need and how much they cost • How you intend to manage communication, risks, quality, changes …
How will you know when the project is completed?	Definition of success criteria
How well did the project go?	Assessment of success criteria

Project Management

The project management process can be split into different phases, we take as example those identified by the PMBOK

1. Initiating
2. Planning
3. Executing
4. Monitoring and Controlling
5. Closing

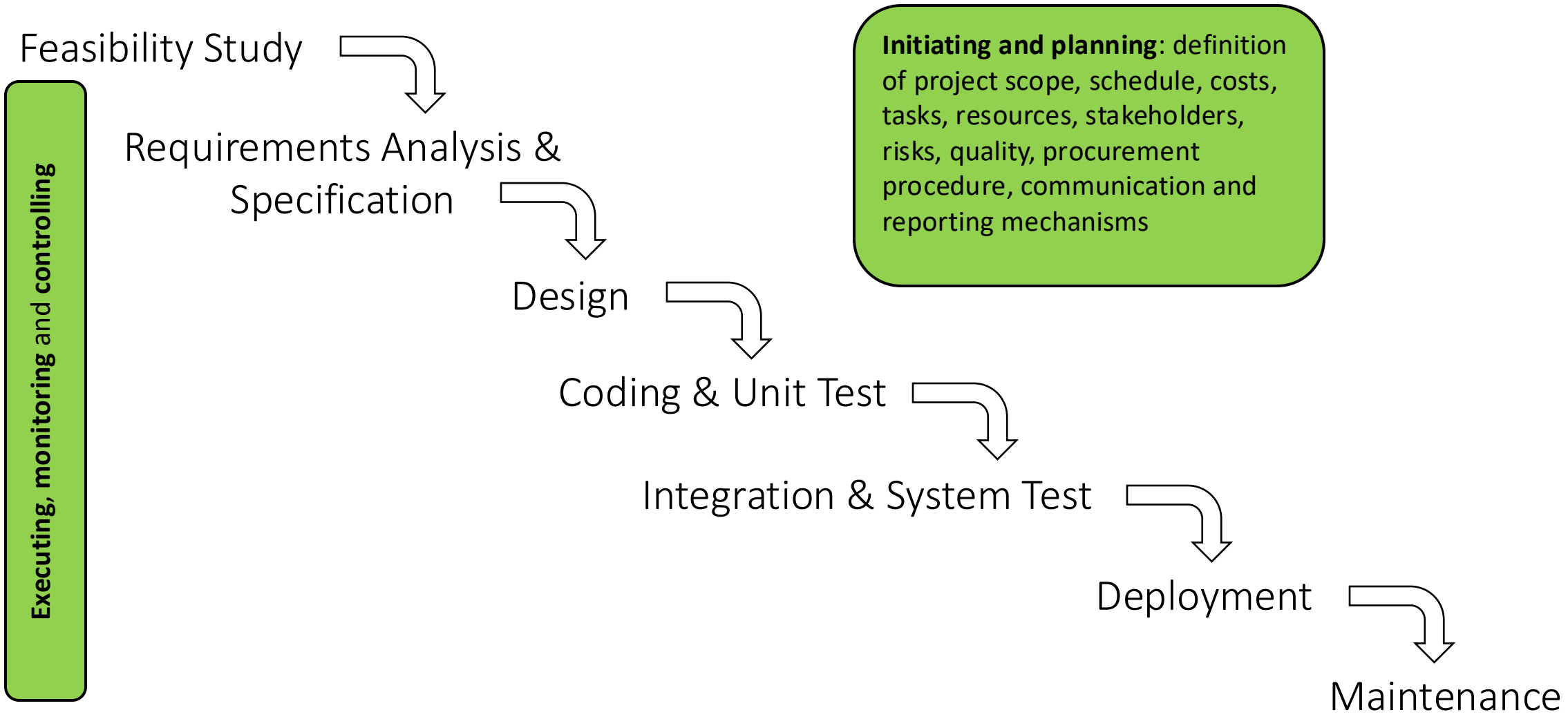
Software Project Management

POLITECNICO MILANO 1863

- The success of most software organizations depends on their ability to deliver the right product in time and within budget.

- Project management practices are adopted as part of the software processes

Software Project Management



Managing SW projects: Delays and failures - An Example? Obamacare

The Affordable Care Act of 2010 (known as Obamacare) provided for the implementation of health benefit websites to facilitate the comparison and purchase of health insurances.

States could either join the federal website (healthcare.gov) or implement their own: most of them decided to adopt the federal one

Healthcare.gov was officially launched on 1 October 2013 and problems emerged immediately…

I'm going to try and download every movie ever made, and you're going to try to sign up for Obamacare, and we'll see which happens first [Jon Stewart]

Healthcare.gov observed issues on the first day

- High website demand (250,000 users [5 times more than expected]) caused the website to go down within 2 hours from launch.
- The website design and implementation was incomplete
  - Drop-down menus were not complete
  - Users' sensitive data were sent over insecure HTTP
  - …
- The login feature was able to handle even less load than the rest of the system
  - This feature was used even by the sys admin, who could not intervene
- Only 6 users could complete and submit their application and select an insurance plan

https://d3.harvard.edu/platform-rctom/submission/the-failed-launch-of-www-healthcare-gov/

https://blog.isthereaproblemhere.com/2013/10/healthcaregov-sent-my-username-and-more.html

Healthcare.gov after the first day…

- After new contractors came in and more management, the website could handle 35000 concurrent users at a time by December 1
- A total of 1.2 million customers signed up for a healthcare plan by 28 December, when the open enrolment period officially ended.

Healthcare.gov failure root causes

POLITECNICO MILANO 1863

- Lack of relevant experience: project managers did not have sufficient knowledge about software products development processes
- Lack of leadership and communication: no clear division of responsibilities between the multiple government offices involved
  - Example: the low workload estimated for the login feature was based on the assumption that login would have come into place only after customers had identified the product they wanted to buy. This decision was changed without communication.
- Schedule pressure: the launch date was mandated regardless the completion of testing and debugging
- The healthcare.gov website costed in the end $1.7 Billions and was initially budgeted for $93.7 Millions

Initiating and planning processes

Initiating: It is all about obtaining the commitment to start the project.

1. Define the project scope and strategy

Planning: It includes

1. Define the schedule
2. Define the stakeholders and risks
3. Estimate cost, resources
4. Define the project management processes needed for execution, monitoring and controlling
    1. When and how to communicate
    2. How to manage procurement
    3. …

POLITECNICO
MILANO 1863

Project scheduling

Planning process – schedule

- What has to be done and when
- How the project parts fit together
- What work people have to do

Terminology

- Tasks: activities that must be completed to achieve the project goal, also called Work Packages (WPs)

- Milestones: points in the schedule in which you can assess progress
  - for example, the handover of the system for testing

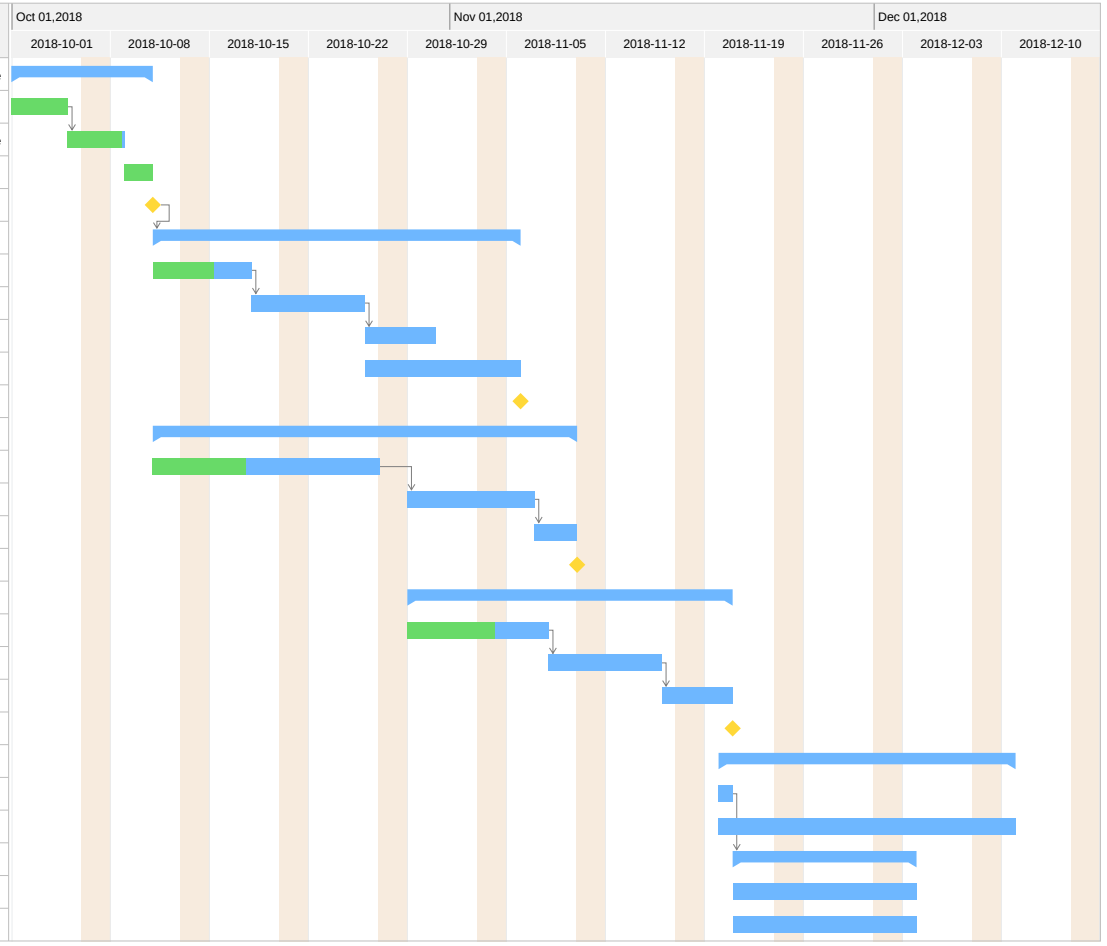
- Deliverables: work products that are delivered to the customer
  - e.g., components of the final software solution

Planning process – schedule

- Graphical notations are normally used to illustrate the project schedule
- Gantt charts are the most commonly used representations for project schedules. They show the schedule as activities or resources against time

Planning process – schedule
GANTT chart

ID	Task	Start	Finish	Duration	Progress	Priority	Resources
1	Analysis	2018-10-01	2018-10-10	8 d	98.9%	1	Rita; William; Tyler; Wenger; Steve
2	Requirement Meetings	2018-10-01	2018-10-04	4 d	100%	1	Rita; William
3	Communication with Stakeholders	2018-10-05	2018-10-08	2 d	95.5%	3	Rita; William; Tyler; Wenger; Steve
4	Document Current System	2018-10-09	2018-10-10	2 d	100%	3	William
5	Analysis Finished	2018-10-10	2018-10-10	1 d	0%	1	
6	Design	2018-10-11	2018-11-05	18 d	14.9%	2	Steve; Yvette; Zoe
7	Design Database	2018-10-11	2018-10-17	5 d	62.4%	1	Steve
8	Software Design	2018-10-18	2018-10-25	6 d	0%	2	Yvette
9	Interface Design	2018-10-26	2018-10-30	3 d	0%	4	Zoe
10	Create Design Specifications	2018-10-26	2018-11-05	7 d	0%	2	Steve
11	Design Finished	2018-11-05	2018-11-05	1 d	0%	2	
12	Development	2018-10-11	2018-11-09	22 d	22.5%	3	Tyler; Wenger
13	Develop System Module	2018-10-11	2018-10-26	12 d	41.2%	1	Tyler; Wenger
14	Integrate System Module	2018-10-29	2018-11-06	7 d	0%	1	Tyler
15	Perform Initial Testing	2018-11-07	2018-11-09	3 d	0%	2	Wenger
16	Development Finished	2018-11-09	2018-11-09	1 d	0%	3	
17	Testing	2018-10-29	2018-11-20	17 d	29.4%	4	Vicky; Mike
18	Perform System Testing	2018-10-29	2018-11-07	8 d	62.5%	3	Vicky
19	Document Issues Found	2018-11-08	2018-11-15	6 d	0%	4	Mike
20	Correct Issues Found	2018-11-16	2018-11-20	3 d	0%	1	Mike
21	Testing Finished	2018-11-20	2018-11-20	1 d	0%	3	
22	Implementation	2018-11-20	2018-12-10	15 d	0%	1	Tyler; Mike
23	On-Site Installation	2018-11-20	2018-11-20	1 d	0%	5	Tyler
24	Support Plan for the System	2018-11-20	2018-12-10	15 d	0%	4	Mike
25	Completion	2018-11-21	2018-12-03	9 d	0%	2	Rita; William
26	System Maintenance	2018-11-21	2018-12-03	9 d	0%	3	Rita
27	Evaluation	2018-11-21	2018-12-03	9 d	0%	1	William



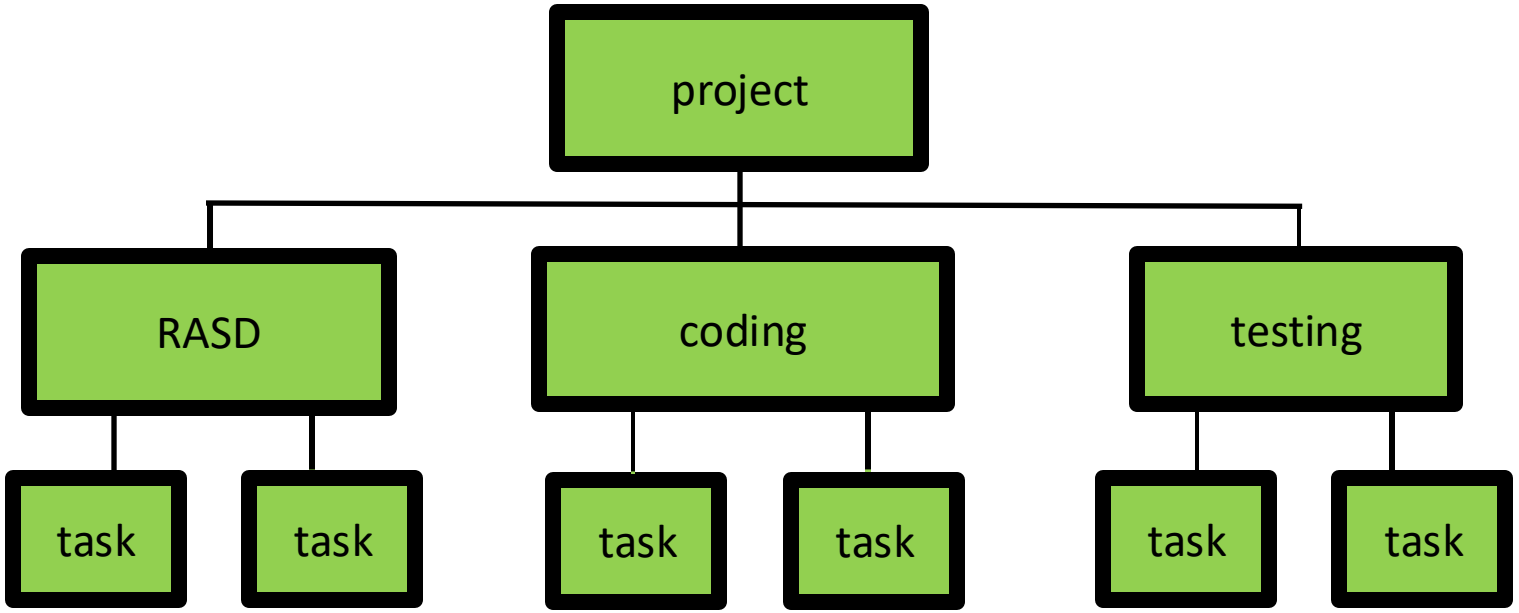
https://www.edrawsoft.com/gantt-chart-examples.html

Creating project schedules

- Gantt diagrams often result from one or more of the following methods
  - Work Breakdown Structure
  - Precedence Diagram Method
  - Critical Path Method

Planning process – schedule

- **Work Breakdown Structure (WBS)**
  - Details the work that must be done, breaks down the work into tasks that can be easy to estimate, assign and track

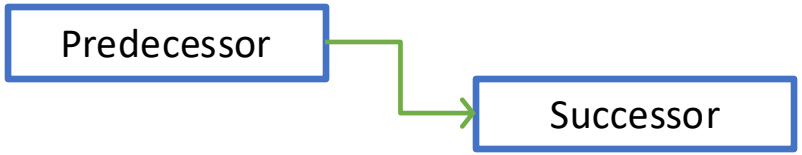


Planning process – schedule

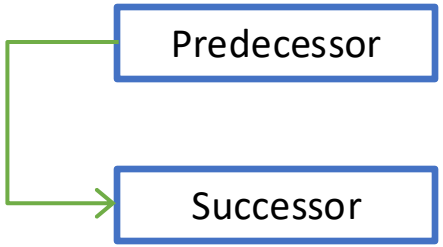
- **Sequencing tasks: the Precedence Diagram Method (PDM)**
  - It is a technique used for constructing a schedule starting from the precedence relationships between activities
- Define which task (predecessor) triggers the other (successor)
- Define the principal **types of dependencies**:
  - Finish to start
  - Start to start
  - Finish to finish
  - Start to finish
- Introduce a delay with the LAG time (positive or negative)

Planning process – schedule

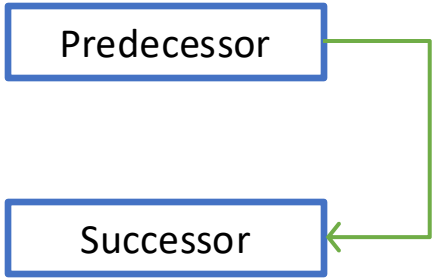
Sequencing tasks



Finish to start



Start to start

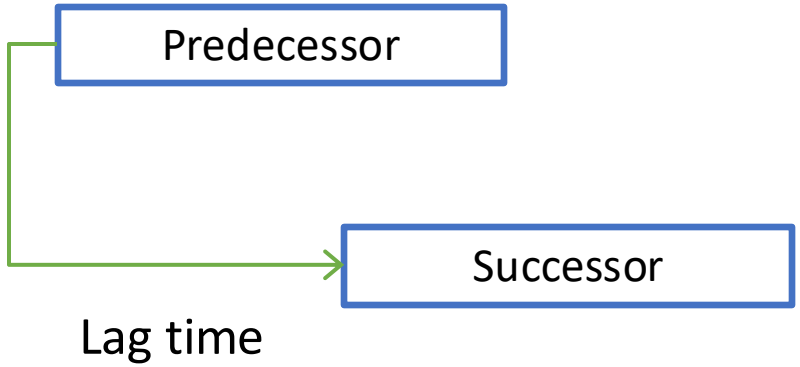


Finish to finish

Start to finish

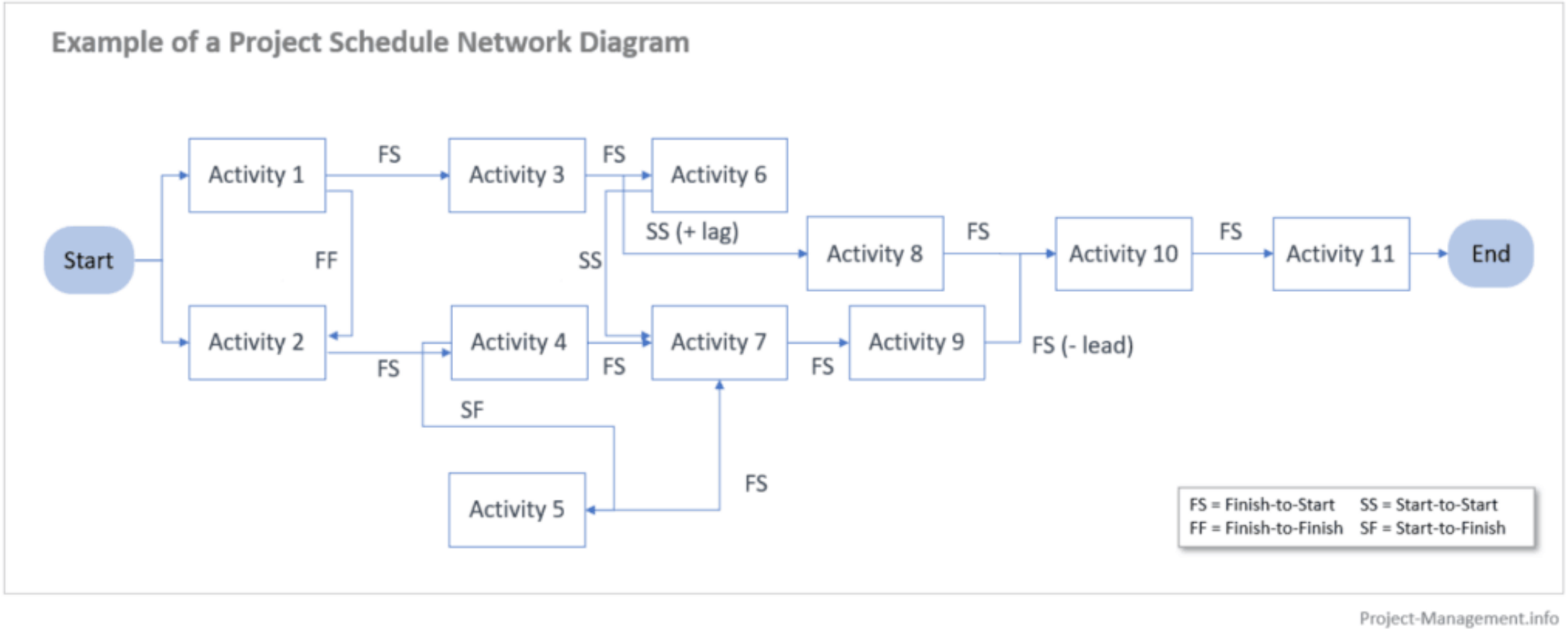
Planning process – schedule

Lag time



Start to start

Planning process – schedule



From https://project-management.info/project-schedule-network-diagram/

Planning process – schedule

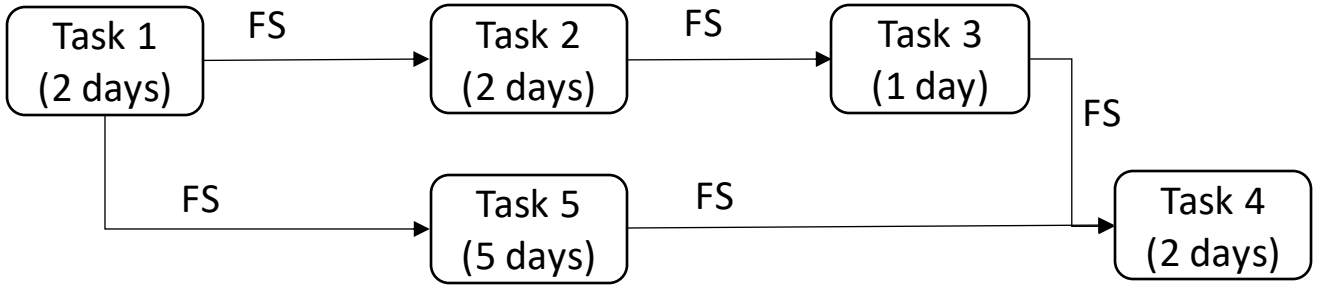
- Tasks Constraints
  - **Flexible**
    - Start as soon as possible
  - **Partially flexible**
    - Start not earlier than
    - Finish not later than (deadline)
  - **Inflexible**
    - A task that should occur on a specific date

Planning process – schedule

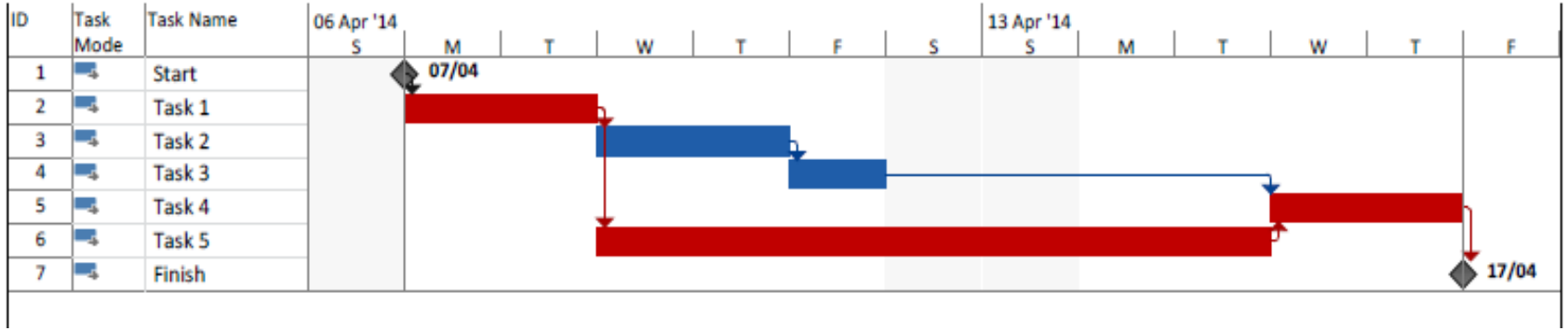
- **Critical Path Method (CPM)**
  - It is used to estimate the minimum project duration.
  - Given a project schedule network diagram and an estimation of activities duration, calculate the early start, early finish, late start, and late finish dates that do not delay the project finish date or any schedule constraint.
  - The difference to the finish date is called task float. If it is zero, then the task is critical.
  - Critical path results in a connected sequence of tasks that runs from the start till the end of the project.
  - Any change to the tasks on the critical path changes the project finish date.

Planning process – schedule

Example: project must be finished by April 17th.
Duration of activities and network diagrams is as follows



Critical path



POLITECNICO
MILANO 1863

Risk management plan

Risk management plan

- **Objective** = Define a framework to identify, analyze, treat, and monitor the risk constantly
- **Risk** (potential problem, or threat)
  - Example: organizational financial problems force reductions in the project budget, for the project team

Risk management plan – identification

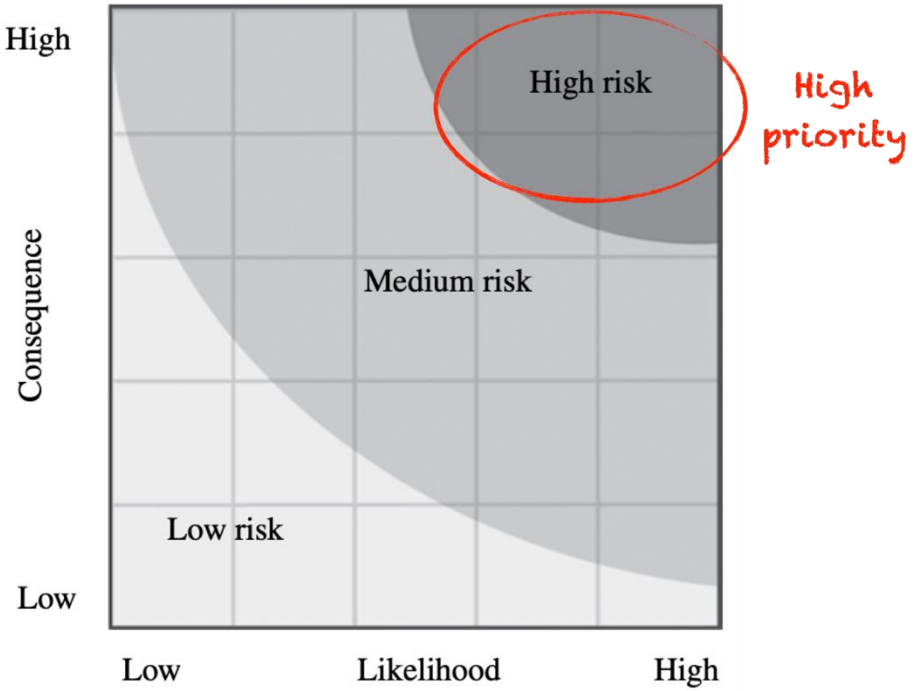
- Identification can be based on
  - Prior experience
  - Brainstorming
  - Lessons learned from similar projects
  - Checklists (well-known risks)
- Rule of thumb
  - Pose each risk in the form “if <situation>, then <consequence>, for <stakeholder>”
  - **Examples of incorrect risk formulation**
    - statement with consequence, but no stakeholder → no impact on the project
    - consequence, but no situation → cause-effect not properly understood (further analysis required)

Risk management plan – identification

- Template: if <situation>, then <consequence>, for <stakeholder>
- Examples
  - If inadequate data storage tools are employed, then significant response time delays can occur, for end users
  - If a global financial crisis occurs, then currency fluctuations may cause a financial loss, for major funding partners
- Exercise: produce risk statement examples for these situations
  - functional requirements are not completely met
  - functional requirements are not completely understood

Risk management plan – measure

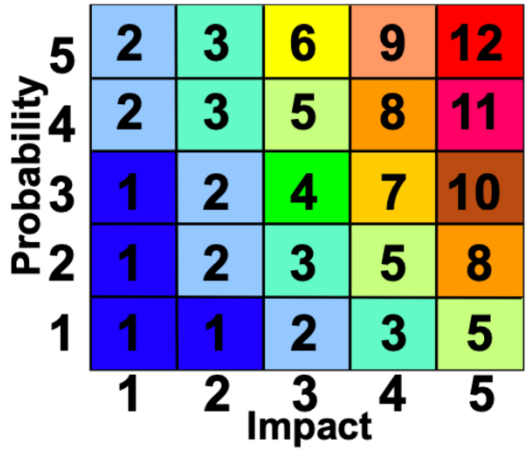
- Evaluation of
  - Likelihood = expressed as probability
  - Impact = Depends on the nature of the event
    - e.g., lost investment, insufficient performance, …



Risk management plan – NASA risk sector plot

- 5 probability levels + 5 impact levels

Attribute: Probability		
Level	Value	Criteria
5	Near certainty	Everything points to this becoming a problem, always has
4	Very likely	High chance of this becoming a problem
3	Likely (50/50)	There is an even chance this may turn into a problem
2	Unlikely	Risk like this may turn into a problem once in awhile
1	Improbable	Not much chance this will become problem



Attribute: Impact				
Level	Value	Technical Criteria	Cost Criteria	Schedule Criteria
5	Catastrophic	Can't control the vehicle OR Can't perform the mission	> $10 Million	Slip to level I milestones
4	Critical	Loss of mission, but asset recoverable in time	$ 10 M ≤ X < $ 5 Million	Slip to level II milestones
3	Moderate	Mission degraded below nominal specified	$ 5 M ≤ X < $ 1 Million	Slip to level III milestones
2	Marginal	Mission performance margins reduced	$ 1 M ≤ X < $ 100 K	Loss of more than one month schedule margin
1	Negligible	Minimum to no impact	Minimum to no impact	Minimum to no impact

Exercise: reformulate and re-evaluate the risks in a correct way

Risk	Probability	Impact
Organizational financial problems force reductions in the project budget, for project team.	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 must be repaired before these components are reused.	Moderate	Serious
Changes to requirements that require major design rework are proposed	Moderate	Serious

The risk register example

Risk	Probability	Impact
If organizational financial problems arise then they will force reductions in the project budget, for project team.	Unlikely	Catastrophic
If the labour market lacks candidates with the skills required for the project, then this will cause a reduction in the team workforce, for project team.	Very likely	Critical
If key staff is sick at critical times in the project, then this determines that the project is delayed, for project team.	Likely	Moderate
If reusable software components show defects, then this causes delays in the project, for project team.	Likely	Moderate
If changes in requirements that require major design rework are proposed, then this determine the need for an extra effort, for project team.	Likely	Moderate

Adding mitigation strategies

POLITECNICO MILANO 1863

Risk	Probability	Impact	Mitigation Strategy
If organizational financial problems arise then they will force reductions in the project budget, for project team.	Unlikely	Catastrophic	Reduce the project scope and revise requirements accordingly
If the labour market lacks candidates with the skills required for the project, then this will cause a reduction in the team workforce, for project team.	Very likely	Critical	Plan for training available staff
If key staff is sick at critical times in the project, then this determines that the project is delayed, for project team.	Likely	Moderate	Assign more than one staff member to critical components so that it is less likely that all of them are sick at the same time
If reusable software components show defects, then this causes delays in the project, for project team.	Likely	Moderate	Plan for a careful testing and bug fixing for these reusable components before their usage
If changes in requirements that require major design rework are proposed, then this determine the need for an extra effort, for project team.	Likely	Moderate	Identify the involved components, replan a new development iteration to address these changes

Executing, monitoring and controlling

Project execution

- Launch the project : kick off meeting
- Acquire and manage project team (internal and external resources)
- Acquire the required equipment and materials and external services
- Perform the work identified in the schedule
- Perform monitoring and controlling activities

Monitoring and controlling

POLITECNICO MILANO 1863

- **Monitoring** consists of collecting data about where the project stands, since projects never stick to the initial plans due to changes, problems, etc.

- **Controlling** is where you implement corrections to get your project back on track

Monitoring process – data gathering

- Gather up-to-date data (actual dates, time worked, money spent)
  - In detail:
    - Track when tasks start
    - Track actual work hours or actual duration
    - Track how much work or duration remains
    - Explore additional costs
- Update the schedule (the original schedule is our baseline)

Monitoring schedule and cost

- Compare your actual schedule with the baseline schedule.
  - Some warning signs:
    - Tasks running late
    - Tasks not started that should be
    - Haven’t completed as much work as planned
- Compare actual costs to assigned budget

Monitoring process - risks

- Monitor risks and, in case they become reality, follow the mitigation strategy defined in the risk management plan

- Revise the mitigation strategy if needed

Monitoring process – Earned Value Analysis

- Methodology to assess project performance and progress
- The variables scope, time and cost related to a project's activity cannot be compared to each other
- Earned Value is the **financial equivalent** derived from the three project variables
  - it reduces all variables to financial values so they can be monitored and compared.

Monitoring process – Earned Value Analysis

- It is based on :
  - Budget at completion (BAC): total budget for the project
  - Planned value (PV): budgeted cost of work planned
  - Earned value (EV): budgeted cost of work performed
  - Actual cost (AC): actual cost for the completed work
- Costs and work completed as of today

Earned Value Analysis: OS servers' upgrade

Scope : upgrade 4 servers

Time : 4 weeks

Budget : 400 EUR

Schedule

time	scope	budget (EUR)
week 1	server 1	100
week 2	server 2	100
week 3	server 3	100
week 4	server 4	100

Earned Value Analysis: OS servers' upgrade

- Monitoring at the end of week 2:
  - Actual spent : 150
  - 1 server upgraded

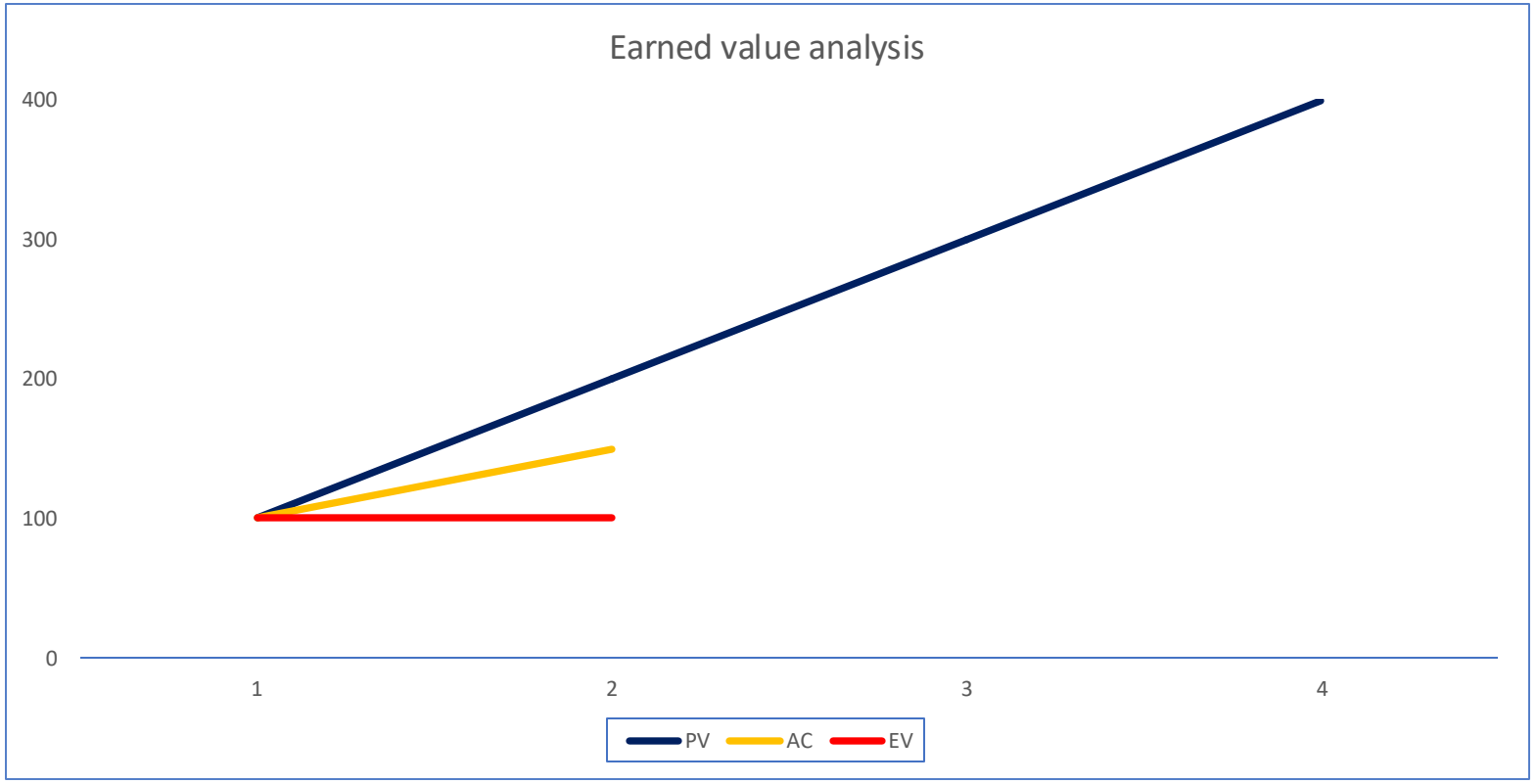
- Traditional PM:
  - Actual spent 150; Planned spent 200 ⟶ under budget
  - 1 Server upgraded instead of 2 ⟶ behind schedule

Earned Value Analysis: OS servers' upgrade

POLITECNICO MILANO 1863

- Monitoring at the end of week 2
  - Actual spent : 150
  - 1 server upgraded
- Earned Value (budgeted cost of work performed):
  - 1 server upgrade, budgeted cost 100 euro → EV = 100
- Planned Value (budgeted cost of work planned)
  - 2 servers, 100 euro each → PV = 200
- Actual spent: 150 → AC = 150

Earned Value Analysis: OS servers' upgrade



- Over budget (EV < AC)
- Behind schedule (EV < PV)

Monitoring process – Earned Value Analysis

- Schedule point of view
  - Schedule variance : SV = EV – PV
  - Schedule performance index : SPI = EV / PV

- Cost point of view
  - Cost variance : CV = EV – AC
  - Cost performance index : CPI = EV / AC

Monitoring process – Earned Value Analysis

- CPI and SPI allow us to perform an estimation of the budget at completion of the project, given one of the following assumptions:
  1. continue to spend at the same rate
     - same CPI
  2. continue to spend at the baseline rate
  3. continue at the same cost and schedule performance index
     - same CPI and SPI

Monitoring process – Earned Value Analysis

- **Cost Estimate At Completion** (EAC)
  - Assumption 1: continue to spend at the same rate
    - EAC = BAC/CPI
  - Assumption 2: continue to spend at the original rate
    - EAC = AC + (BAC – EV)
  - Assumption 3: both CPI and SPI influence the remaining work
    - EAC = [AC + (BAC – EV)] /(CPI*SPI)

Earned Value Analysis: OS servers' upgrade

- CPI = EV/AC = 100/150 = 0.67
- SPI = EV/PV = 100/200 = 0.5
- If we continue to spend at the same rate (same CPI)
  - EAC = BAC/CPI = 400/0.67 = 597
- If we continue to spend at the baseline rate
  - EAC = AC + (BAC – EV) = 150 + (400 – 100) = 450
- If we continue at the same cost and schedule performance indexes (same CPI and SPI)
  - EAC = [AC + (BAC – EV)] /(CPI*SPI) = 450/(0.67*0.5) = 1343,28

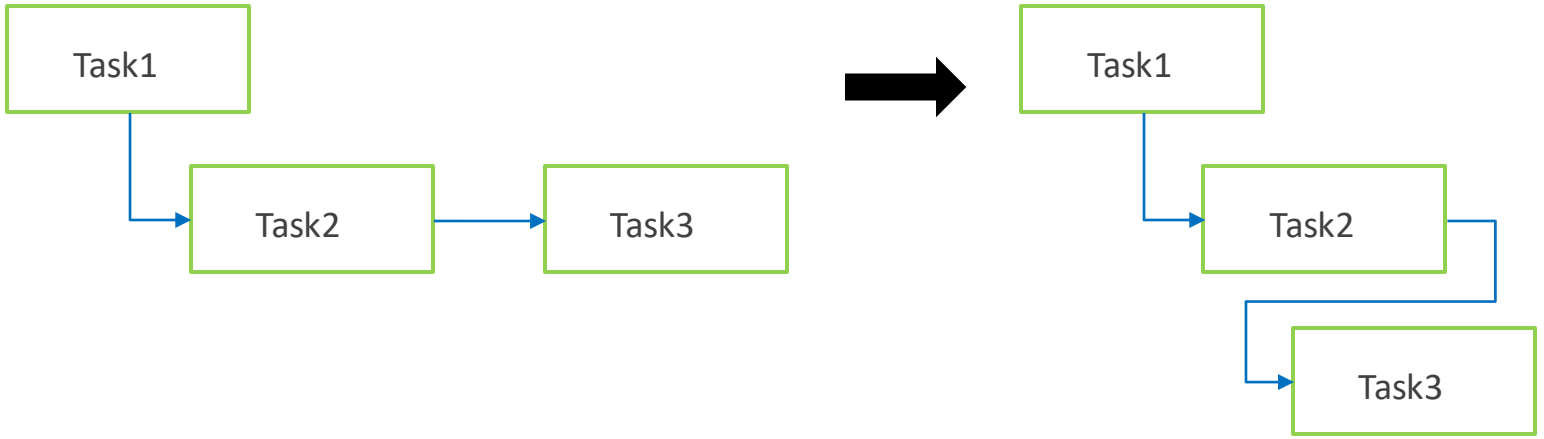
Controlling process

- **Controlling**: Means balancing scope, time, cost, resources and quality
- If schedule is important, you can use techniques like fast-tracking and crashing
- If money is a priority, you can reduce costs by reducing resources or overhead costs
- If schedule, money and resources are not negotiable, then reduce scope eliminating the tasks associated with it

- The choice introduces a risk factor and depends on stakeholders' priority

Controlling process – fast tracking

- You push tasks to start earlier than they would by introducing a negative lag time, if needed
  - This can be done only if the considered tasks can actually be parallelized
- There is no cost increase but a higher risk

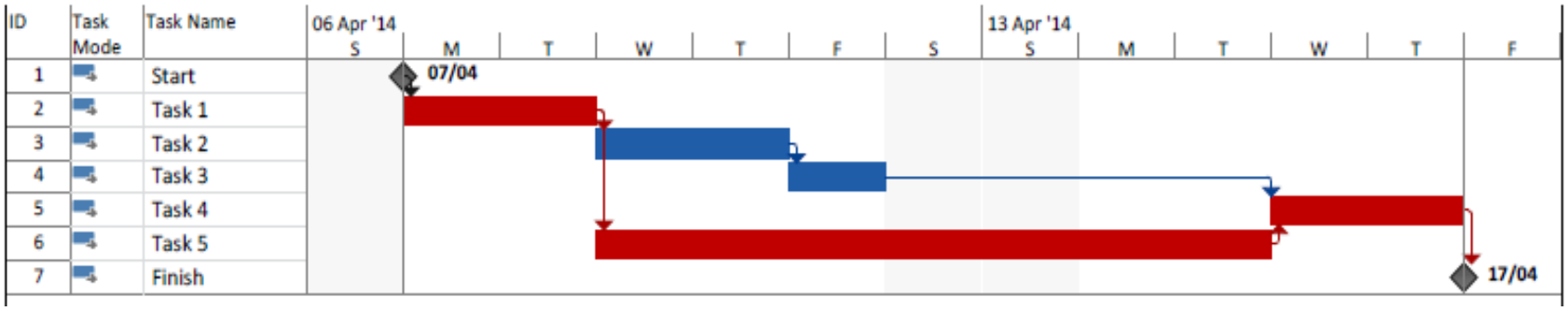


Controlling process – crashing

- Shorten the tasks on the critical path by adding new resources
- It implies a cost increase
- It is feasible only if the addition of new resources is beneficial (remember what Brooks wrote in the Mythical Man Month…)

Controlling process – crashing: an example

- We reduce the project duration from 9 days to 5.5 days at a cost of 1450 (in addition to the planned project budget)



Task name	Duration in days	Crash length	New length	Crash cost per day	Crash cost
task 1	2	0.5	1.5	400	200
task 4	2	1	1	250	250
task 5	5	2	3	500	1000
totals	9		5.5		1450

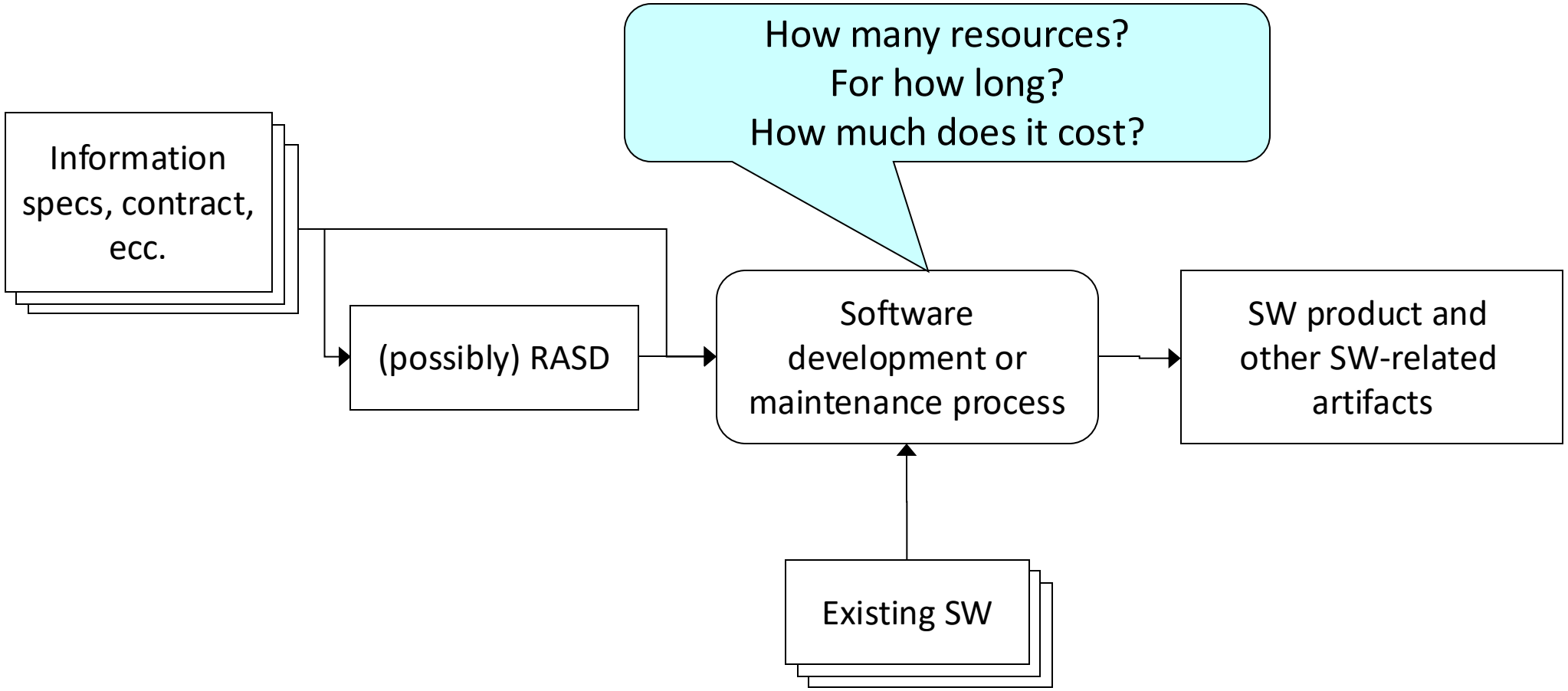
Controlling process – closing

- Ensure project acceptance
- Track project performance
- Lessons learned
- Close contracts
- Release resources

POLITECNICO
MILANO 1863

Effort and Cost Estimation

The estimation problem



Estimation techniques

POLITECNICO MILANO 1863

- Organizations need to make software effort and cost estimates. Two types of techniques:
  - Experience-based techniques: The estimate of future effort requirements is based on the manager's experience of past projects and the application domain
    - Essentially, the manager makes an informed judgment of what the effort requirements are likely to be
  - Algorithmic cost modelling: In this approach, a formulaic approach is used to compute the project effort based on estimates of product attributes, such as size, and process characteristics, such as experience of staff involved

Experience-based approaches

POLITECNICO MILANO 1863

- Experience-based techniques rely on judgments based on experience of past projects
- Steps:
  - Identify the deliverables to be produced in the new project (both documents and software)
  - Document these in a spreadsheet
  - Estimate the effort needed for each of them
  - Compute the total effort required
- It usually helps to get the whole team involved in the effort estimation and to ask each team member to explain their estimate

Algorithmic cost modelling

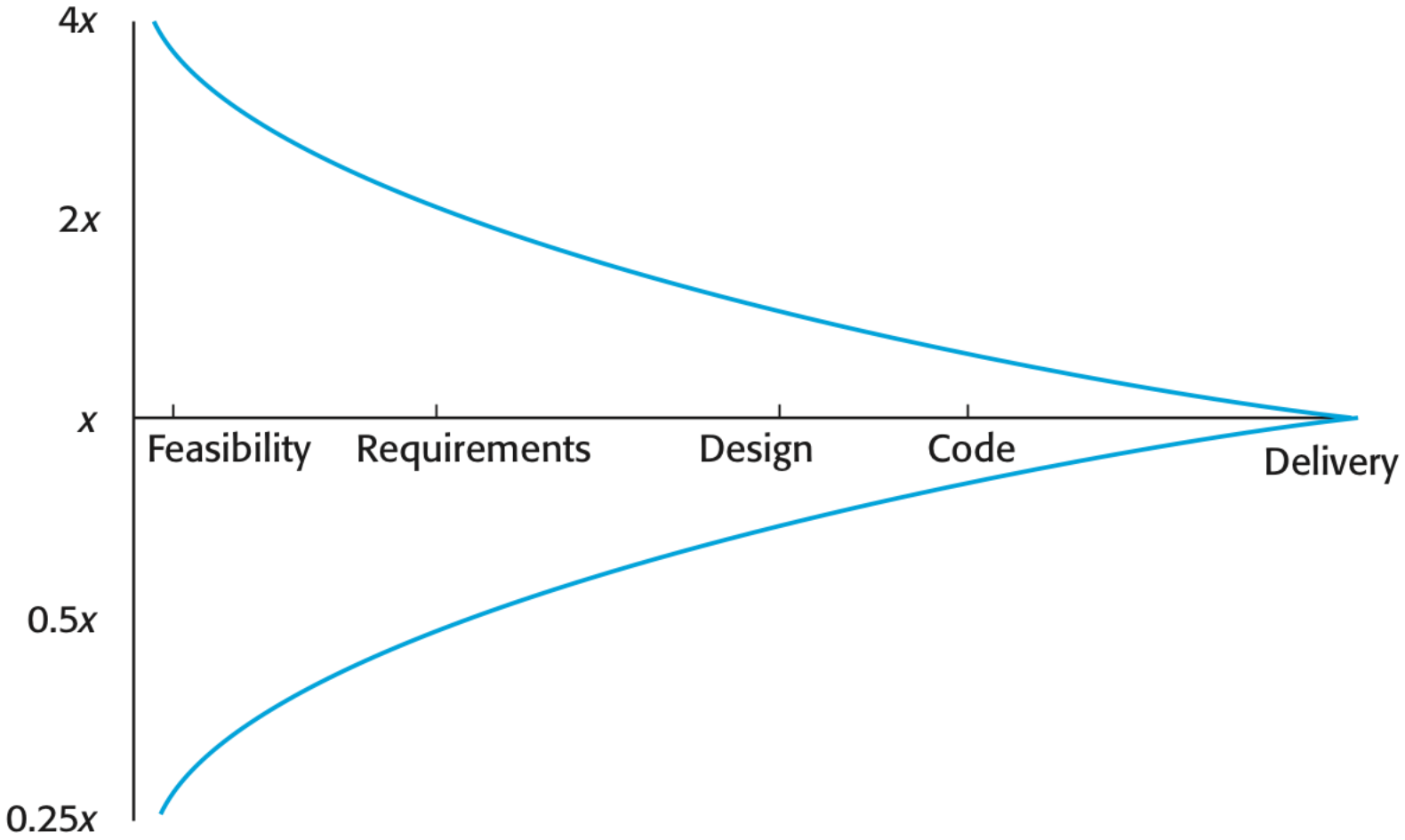
POLITECNICO MILANO 1863

- The estimation is based on a mathematical function of product, project and process attributes whose values are estimated by project managers:
  - Effort = A x Size x B x M
    - A is an organisation-dependent constant,
    - Size is a measure of the "quantity" of product
    - B reflects the disproportionate effort for large projects and
    - M is a multiplier reflecting product, process and people attributes
- Most models are similar, but they use different values for A, B and M

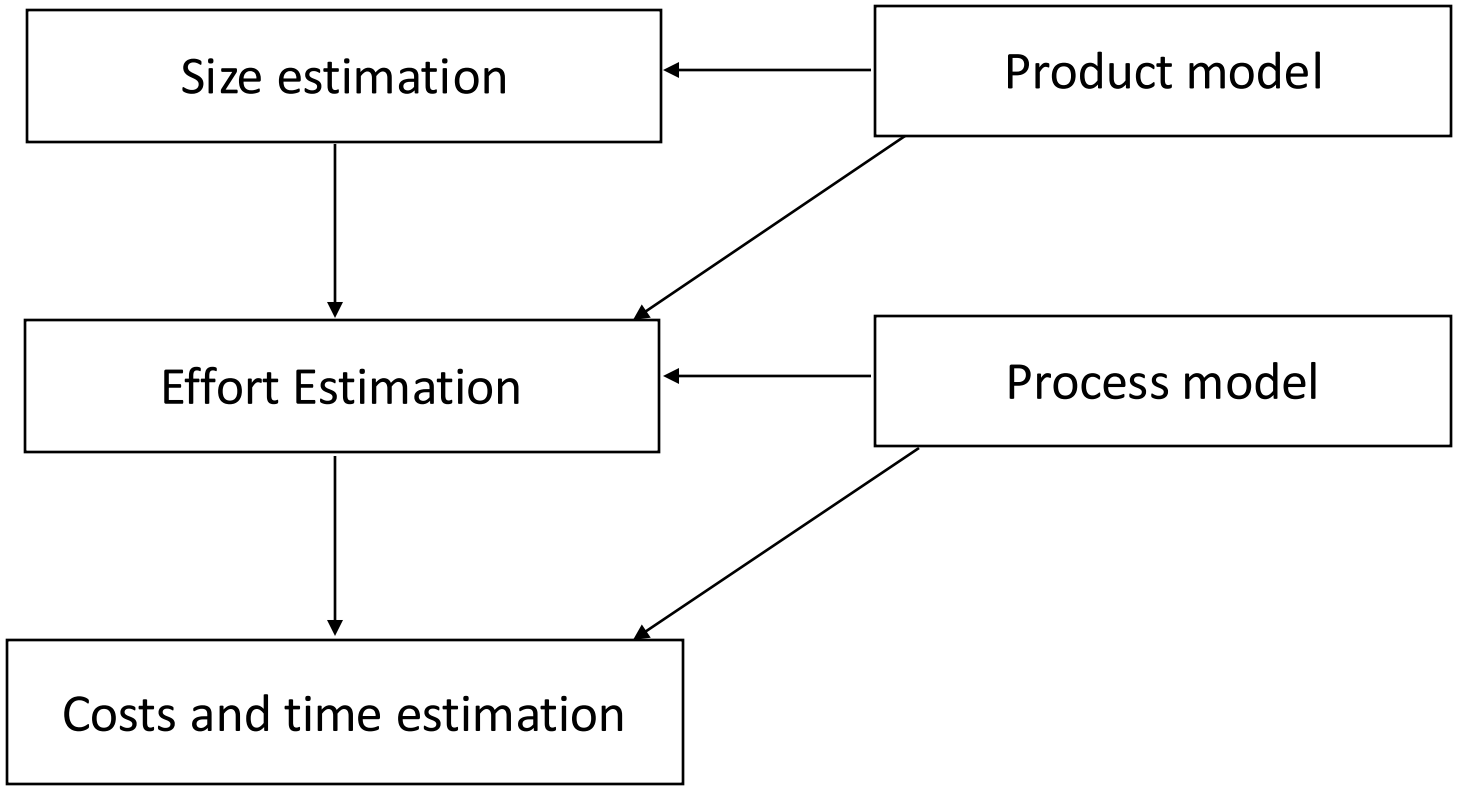
Estimation accuracy

- The size of a software system can be known accurately only when it is finished
- Several factors influence the final size
  - Use of COTS (Components Off The Shelf)
  - Programming language
  - Distribution of the team
- As the development process progresses then the size estimate becomes more accurate
- The estimates of the factors contributing to B and M are subjective and vary according to the judgment of the estimator

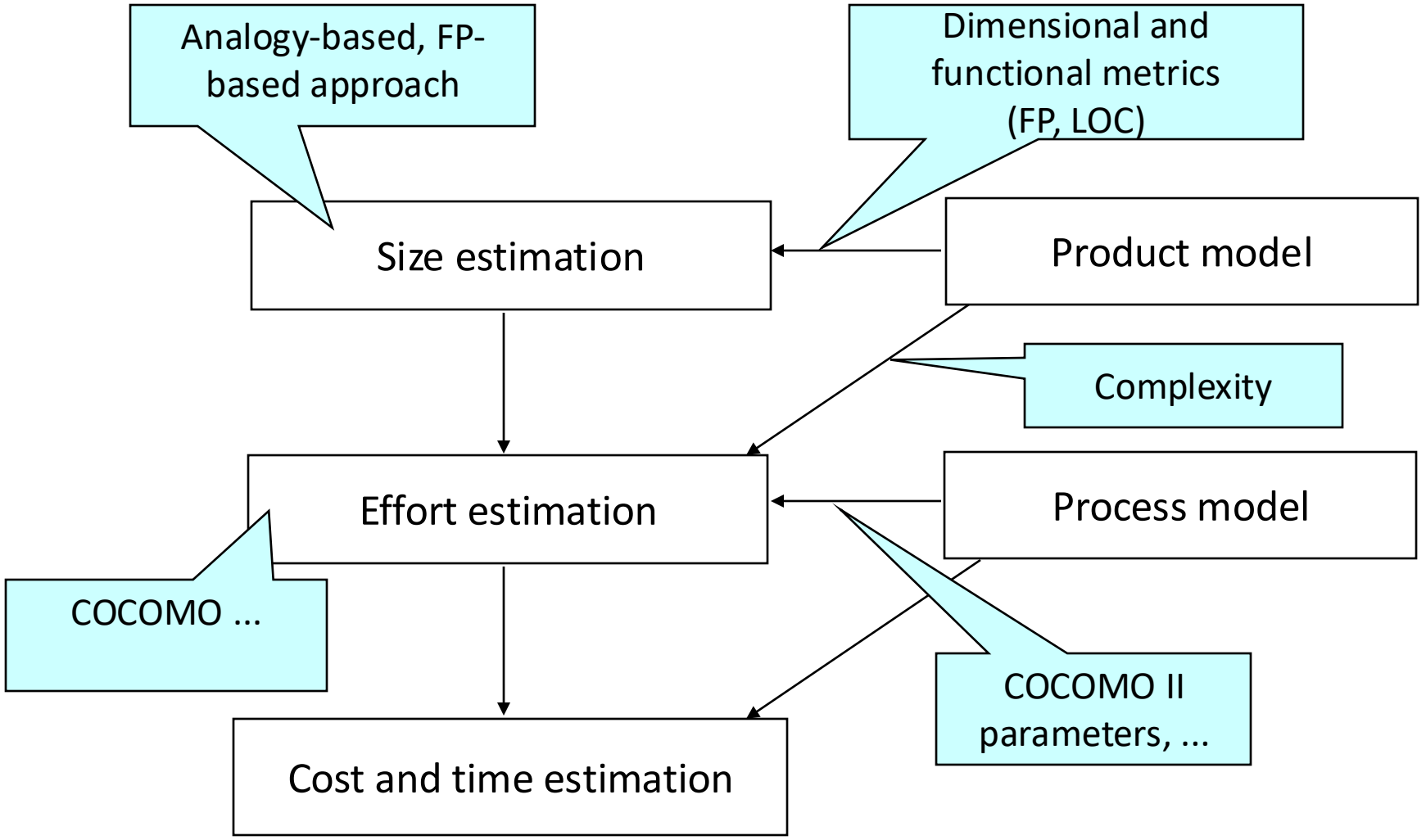
Estimate uncertainty



A possible estimation procedure



Techniques to support the estimation procedure



How do we estimate software size?

- Option1:
  - From the RASD or other informal preliminary documents, identify the main characteristics of the software to be and classify them in terms of Function Points
  - From the number of Function Points determine the software LOC estimation
- Option 2:
  - Estimate the LOC directly, based on previous experience/projects
- Any estimation has to be completed by a complexity analysis to identify particularly complex projects

Function points

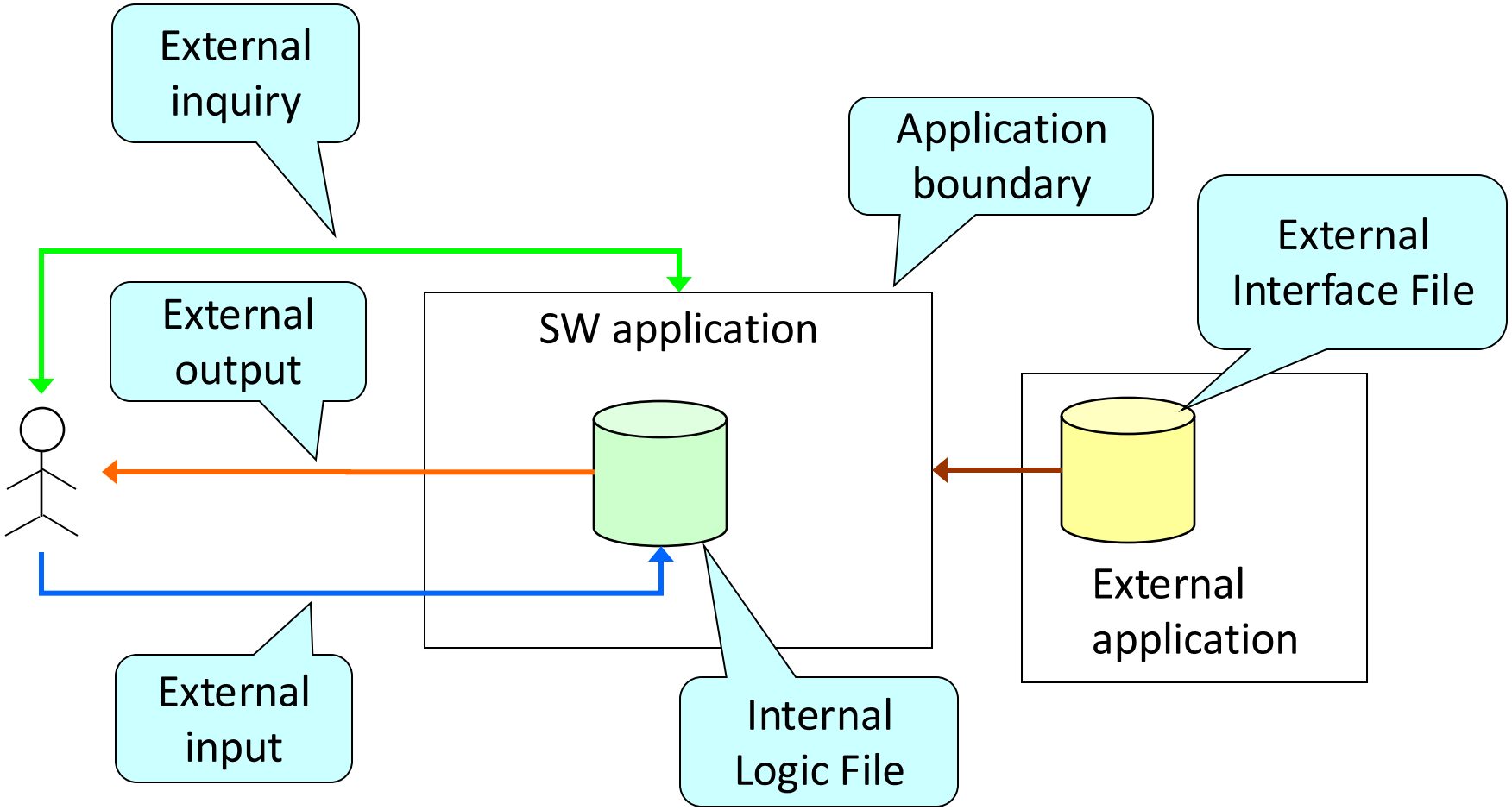
- The Function Points approach was defined in 1975 by Allan Albrecht (IBM)
- Basic assumption:
  - the size of software can be characterized based on the "functions" that it has to offer

What are Function Points (FP)?

POLITECNICO MILANO 1863

- Based on a combination of program characteristics
  - Data structures
  - Inputs and outputs
  - Inquiries
  - External interfaces
- A weight is associated with each of these FP counts; the total is computed by multiplying each "raw" count by their weight and summing all partial values:
  - FP = $\sum$ (number of elements of a given type * type weight)

Function Types (1)



Function Types (2)

- Albrecht's method identifies and counts the number of function types
- They, all together, constitute the "external representation" of an application, that is, its functionality

Function Types (3)

- Internal Logic File (ILF)
  - homogeneous set of data used and managed by the application
- External Interface File (EIF)
  - homogeneous set of data used by the application but generated and maintained by other applications
- External Input
  - Elementary operation to elaborate data coming from the external environment
- External Output
  - Elementary operation that generates data for the external environment
    - It usually includes the elaboration of data from logic files
- External Inquiry
  - Elementary operation that involves input and output
    - Without significant elaboration of data from logic files

How was the approach defined?

- Albrecht considered 24 applications
  - 18 COBOL, 4 PL/1, 2 DMS
  - …ranging from 3000 to 318000 LOC,
  - …from 500 to 105200 person hours
- Based on these he defined the number of FP as the sum of Function Types weighted in order to correlate them to the development effort

Weighting function points

Complexity is evaluated based on the characteristics of the application

Function Types	Weight		
	Simple	Medium	Complex
N. Inputs	3	4	6
N. Outputs	4	5	7
N. Inquiry	3	4	6
N. ILF	7	10	15
N. EIF	5	7	10

Example

- Calculate FPs for an application managing a mobile telephony service
- The application:
  - Manages data about both users and the different types of rates. The system will store the user data (phone number, personal data, the type of rate, any applying promotion). Each rate type is characterized by: call cost, cost of text messages, cost of surfing the internet. Promotions change some of the cost items only for a well-defined period of time.
  - Allows users to access their information, change rates, and activate promotions.
  - Manages user billing based on information retrieved from the network management system. This last system sends to the service management system data concerning the voice calls (call duration, and the user location in case of roaming), SMSs (the information that messages have been sent, and the corresponding user location in case of roaming), and internet surfing (connection duration, amount of downloaded data, user location) of each user.

Example

profiles, rates, promotions, invoices info requests

Login/logout

Select rate

Select / eliminate promotions

Insert/delete new rate /promotion

Insert/delete user

Customer info

Mobile telephony application

users

rates

promotions

invoices

Invoices

calls

sms

internet

Network management system

ILFs

- The application stores the information about
  - users
  - rates
  - promotions
  - As the application also manages billing information, it will have to produce invoices
- Each of these entities has a simple structure as it is composed of a small number of fields
  - Thus, we can decide to adopt for all four the simple weight
- Thus, 4 x 7 = 28 FPs concerning ILFs

EIFs

- The application manages the interaction with the network management system for acquiring information about
  - calls
  - SMSs
  - internet usage
- Three entities with a simple structure
  - Thus, we decide to adopt a simple weight for the three of them
- We get 3 x 5 = 15 FPs for EIFs

External Inputs (1)

- The application interacts with the customer:
  - Login/logout: these are simple operations, so we can adopt the simple weight for them
    - 2 x 3 = 6 FPs
  - Select a rate: this operation involves two entities, the rate and the user. It can still be considered simple, so, again we adopt the simple weight
    - 1 x 3 = 3 FPs
  - Select/eliminate a promotion: These two operations involve three entities, the user, the rate and the promotion. As such, we can consider them of medium complexity.
    - 2 x 4 = 8 FPs

External Inputs (2)

- The application also interacts with the company operators to allow them to:
  - Insert/delete information about a new rate or promotion
  - Insert/delete information about a new user
- All 4 operations can be considered of medium complexity
  - 4 x 4 = 16 FPs
- In summary, we have 6+3+8+16 = 33 FPs for External Inputs

External Inquiries

- The application allows customers to request information about
  - their profiles
  - the list of rates, promotions, and invoices that have been defined for them
- The application also allows the operator to visualize the information of all customers
- In summary, we have 5 different external inquiries that we can consider of medium complexity
  - Thus, 5 x 4 = 20 FPs for External Inquiries

External Outputs

- The application allows the creation of invoices
- This is a quite complex operation as it needs to collect information from ILFs and EIFs
- Thus, we can apply the weight for the complex cases:
  - 1 x 7 = 7 FPs for External Outputs

Total number of FPs

- ILFs: 28
- EIFs: 15
- External Inputs: 33
- External Inquiries: 20
- External Outputs: 7
- Total: 103

Final remarks

POLITECNICO MILANO 1863

- The function point count is modified by the complexity of the project
- FPs can be used to estimate LOC depending on the average number of LOC per FP for a given language
  - LOC=AVC*number of function points
    - AVC is a language-dependent factor varying from 200-300 for assembly language to 2-40 for a 4GL
    - See here for a possible mapping http://www.qsm.com/resources/function-point-languages-table
- FPs are very subjective, they depend on the estimator

The International Function Point Users Group

- Mission: take care of the evolution of FPs
- Produces a manual for supporting the adoption of the FP approach
- Defines various versions of FP to apply them outside the initial context
- Offers examples useful as a reference
- http://www.ifpug.org/