



# Project Planning and Activities

Andrea Polini

Software Project Management  
MSc in Computer Science  
University of Camerino

A.Y. 2017/2018

# Phases in SPM

At the macro level the management of a software project typically requires to go through the following phases:

- Feasibility study
- Planning
- Execution

# Feasibility

Is it worth starting the project?

- Identification of a business case
- Market analysis and strategies
- Cost assessment (accuracy at this level cannot be high)

A small prototype could be developed to clarify project needs

Different situations can change the recipe

- You act as the Contractor
- In house software (build or buy): software that you need to run your business
- Participation to research and innovation proposals
  - Feasibility is “somehow assumed” once you deliver the proposal
  - the proposal generally includes detailed planning

# Feasibility

Is it worth starting the project?

- Identification of a business case
- Market analysis and strategies
- Cost assessment (accuracy at this level cannot be high)

A small prototype could be developed to clarify project needs

Different situations can change the recipe

- You act as the Contractor
- In house software (build or buy): software that you need to run your business
- Participation to research and innovation proposals
  - Feasibility is “somehow assumed” once you deliver the proposal
  - the proposal generally includes detailed planning

# Feasibility study result

A document reporting the effort of feasibility study generally include:

- Introduction and background
- The proposed project
- The market
- Organizational and operational infrastructure
- The benefits
- Outline of the implementation plan
- Costs assessment
- The financial case
- Risk analysis
- Management plan

# Project Execution

Project execution refer to the real performance of the activities foreseen in the plan and it includes the activities of **monitoring** and **control**

# Project planning

## Planning

Thinking **carefully** about something before you do it - even with uncertain projects this is worth doing as long as plans are **seen as provisional**

Planning requires to define at **best the work** to be performed by teams in order to reach the **project objectives** in minimum time and cost

- express objectives as kind of “**post conditions**” - e.g. “*customers should be able to get hot drinks*” is better than “*develop a coffee machine*”
- objectives are detailed in sub-objectives. Well defined objectives are “**SMART**”
  - Specific
  - Measurable
  - Achievable
  - Relevant
  - Time Constrained

# Step Wise method

The **Step Wise** method is a PM method covering the planning of a project that is compatible with Prince2:



Bob Hughes and Mike Cotterell

*Software Project Management, 5th Ed.*

McGraw-Hill, 2009



# Steps in Step Wise

The method foresees the following steps:

- 0 Select Project
- 1 Identify project scope and objectives
- 2 Identify project infrastructure
- 3 Analyse project characteristics
- 4 Identify project products and activities
- 5 Estimate effort of each activity
- 6 Identify activity risks
- 7 Allocate resources
- 8 Review/publicize plan
- 9 Execute plan

# Step 1: Identify project scope and objectives

- 1 Identify objectives and practical measures of the effectiveness in meeting those objectives
- 2 Establish a project authority
- 3 Stakeholder analysis - identify all stakeholders in the project and their interests
- 4 Modify objectives in the light of stakeholder analysis
- 5 Establish methods of communication with all parties
  - external parties need to be involved

## Step 2: Identify project infrastructure

- 1 Identify relationship between the project and strategic planning
- 2 Identify installation standards and procedures
- 3 Identify project team organization

## Step 3: Analyse project characteristics

- 1 Distinguish the project as either objective or product driven
- 2 Analyse other project characteristics
- 3 Identify high level project risks - e.g. acceptance from users
- 4 Take into account user requirements concerning implementation
- 5 **Select development methodology and life-cycle approach**
- 6 Review overall resource estimates

## Step 4: Identify project product and activities

- 1 Identify and describe project products (or deliverables)
  - work will produce intermediate products - documents, software, test suites, etc . . .
  - Product Breakdown Structure (PBS)
  - to each product should be described with a name, purpose, derivation, composition, form, standards, quality criteria to decide if it is acceptable
- 2 Document generic product flows - relations to be described with Product Flow Diagram (PFD)
- 3 Recognize product instances
- 4 Produce ideal activity network
- 5 Modify the ideal to take into account need for stages and checkpoints

## Step 5: Estimate effort for each activity

- 1 Carry out bottom up estimates - elapsed time vs. effort
- 2 Revise plan to create controllable activities
  - long activities make project control difficult
  - activities should not be running when checkpoints have been defined - it is generally a good idea to **alienate activity reporting with monitoring and controlling** activities

## Step 6: Identify activity risks

- 1 Identify and quantify activity based risks
- 2 Plan risk reduction and contingency measures where appropriate
- 3 Adjust overall plans and estimate to take account of risks

## Step 7: Allocate resources

- 1 Identify and allocate resources
- 2 Revise plans and estimate to take into account resource constraints - a needed expert could be fully booked in certain period



## Step 8: Review publicize plan

- 1 Review quality aspects of the project plan - which are the conditions that **establish when an activity can be closed**
- 2 Document plans and obtain agreement

# Activity Planning

# Detailed planning

A detailed project planning requires to precisely define start and end of activities. This will permit to:

- ensure that the appropriate resources are available when required
- avoiding 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
- produce a timed cash flow forecast
- replan the project in case of drift from the plan

Planning generally aims at shortening project duration. One strategy, when possible, is to put activities in parallel

# Project schedules

Before the start of the project (project execution) the **plan should define when each activity will start and end, and when and how much each resource will be required**. Then the scheduling proceed in 4 consecutive steps:

- what activities need to be carried on, and build an **ideal activity plan** - infinite resources
- Consider for each activity the **associated risks**, and then revise the plan
- Now consider the **real resources that will be available** to run the project, and put **constraints related to their availability**. The result constitute the scheduling of the project

# Defining activities

General recommendations:

- Project can start when at least one activity is ready to start and finish when all activities are declared completed
- Activities start could depend on the **availability of specific artefacts and/or dependencies from other activities**

Three approaches to activity identification:

- **Activity based**
- **Product based**
- **Hybrid**

# Activity identification approaches

## Activity based approach

- Create a **list of activities** that the project is thought to require
  - studying, reporting, market research, . . .
- with large project could be helpful to consider a **breakdown structure for general project life-cycle** – Work Breakdown Structure (WBS)

## Product based approach

The activities are related to the product transformation they produce. List product and sub-product and **analyse how they can be produced** from each other - uses PBS

## Hybrid based approach

Results as a clear mix

# Hybrid approach example

## IBM Proposal

IBM proposes that five levels should be used in WBS:

- 1 Project
- 2 Deliverables
- 3 Components
- 4 Work-packages
- 5 Tasks

# Network planning models

Activities and relations are modeled as a network – the “**Critical Path Method**” can then applied.

Network characteristics:

- one start and one end
- nodes have a duration
- links have no duration
- An arrows originating in A and ending in B affirm that B cannot start until A has not finished
- no loop
- no dangling activities

|                                      |          |                 |
|--------------------------------------|----------|-----------------|
| Earliest Start                       | Duration | Earliest Finish |
| Activity label, activity description |          |                 |
| Latest Start                         | Float    | Latest Finish   |



# Project duration and definition of activities start/end

|   | <b>Activity</b>         | <b>Duration (weeks)</b> | <b>Precedents</b> |
|---|-------------------------|-------------------------|-------------------|
| A | Hardware selection      | 6                       |                   |
| B | System Configuration    | 4                       |                   |
| C | Install Hardware        | 4                       | A                 |
| D | Data Migration          | 4                       | B                 |
| E | Draft Office Procedures | 3                       | B                 |
| F | Recruit Staff           | 10                      |                   |
| G | User Training           | 3                       | E,F               |
| H | Install and Test System | 2                       | C,D               |

In order you can apply:

- Forward pass
- Backward pass
- Critical path identification
- Activity float identification - free float and interfering float
- Shortening project duration and conflicts handling

# Arrow based approaches

It is possible to use alternative representations where **activities are used to mark arcs**. Nodes are now representing start/end of activities. Similar analysis can be carried on

- carefully consider unnecessary constraints introduced by the representation and use dummy activities

# Exercise CPM

Consider a project in which the foreseen activities and dependencies are the ones specified in the Table, where the duration is expressed in weeks, and the effort is expressed in terms of needed Senior Developers (SD) and Junior Developers (JD) respectively:

| Task | Duration | Effort      | Depends on |
|------|----------|-------------|------------|
| T1.1 | 10       | 20 SD/10 JD |            |
| T1.2 | 12       | 24 SD/12 JD |            |
| T1.3 | 6        | 6 SD/0 JD   | T1.1       |
| T1.4 | 2        | 2 SD/2 JD   |            |
| T2.1 | 2        | 4 SD/2 JD   |            |
| T2.2 | 6        | 12 SD/6 JD  | T1.3,T2.3  |
| T2.3 | 8        | 0 SD/8 JD   | T2.1       |
| T3.1 | 4        | 12 SD/8 JD  | T1.2, T1.4 |
| T3.2 | 16       | 0 SD/32 JD  | T3.1       |
| T4.1 | 6        | 12 SD/6 JD  | T3.2       |
| T4.2 | 4        | 8 SD/0 JD   | T2.2       |

- Report the CPM network and derive the total duration of the project, as well as the various floats for each activity