Université de Paris

Génie Logiciel
Elements of a software project – Part 2

Sylvain Lobry

01/10/2021

Resources: www.sylvainlobry.com/GenieLogiciel

Elements of a software project

Wooclap

https://www.wooclap.com/L3GL4

Before we start…

- Changements de TD: pas autorisé!

Before we start…

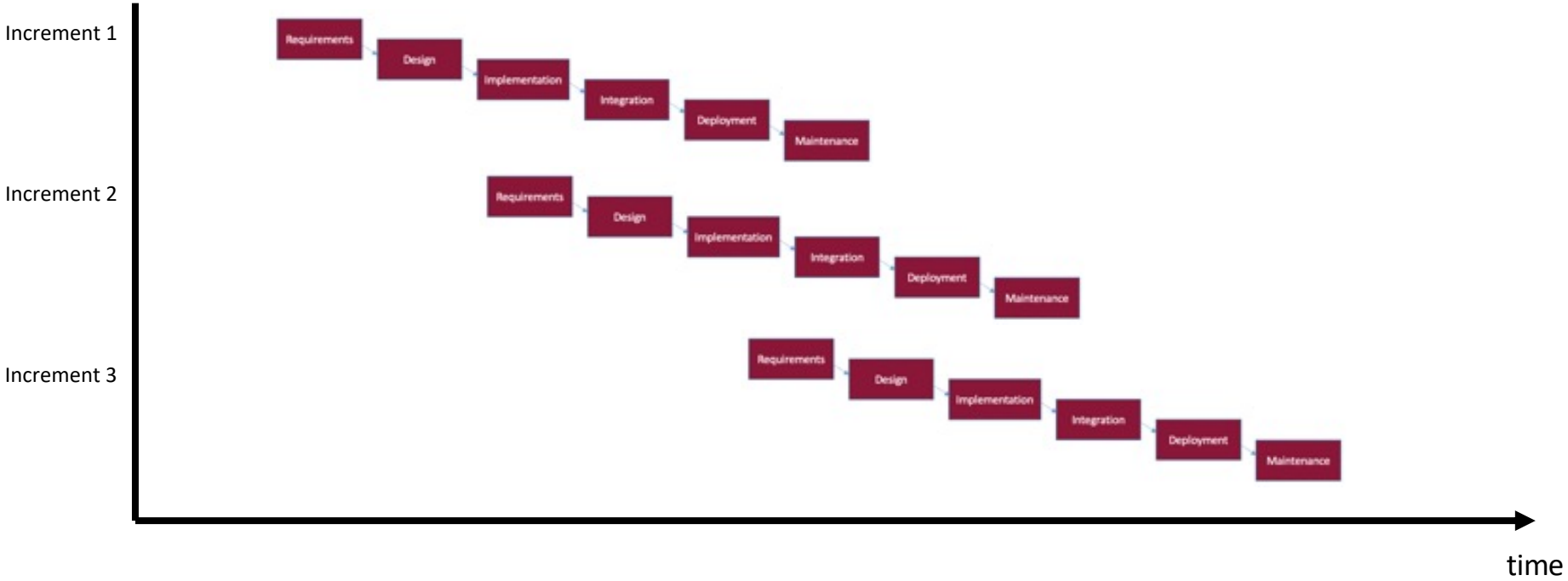
- La dernière fois, on a vu:
  - La définition d'un projet logiciel
  - Les acteurs du projet (MOA/MOE)
  - Les processus
  - Les différentes activités d'un projet logiciel
  - Les cycles de développement (cascade, en V, itératif, spirale)

Before we start…

- Objectives defintion
- Requirement analysis
- Feasability analysis
- Requirements specification

Before we start…

- Incremental SDLC n’est pas équivalent à développement en parallèle.



Planning a project

Planification

- Scope: balance between
  - Quality
  - Time
  - Cost

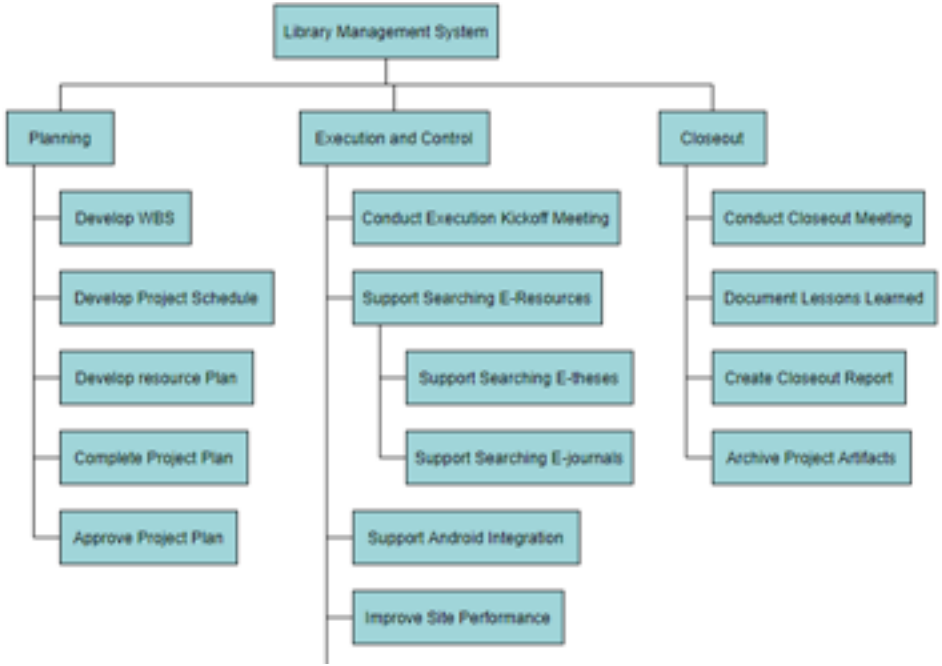
Planning a project

Planification

- Planification is an essential component of a successful project
- We know the main tasks from our SDLC model. We need to decompose them in smaller, achievable tasks
- Work Breakdown Structure (WBS) or Organigramme des Tâches (OT)
- Tree structure with at least 3 levels:
  - Level 1: name of the project
  - Level 2: main activities seen before
  - Level 3 and more: sub tasks

Planning a project

Example of a WBS



Source

Planning a project

WBS rules

5 rules to follow:

1) It has to be a tree structure
2) Each task should be clearly defined, including potential deliverables
3) Each task should have a clear finsihing action
4) Each deliverable should be associated to a task
5) Achievement of every sub-task implies the achievement of the parent task

Planning a project

PERT

- Program Evaluation and Review Technique (PERT) is a method of analyzing the different tasks in the project.
- In particular, allows to analyze:
  - Dependencies between tasks
  - Duration of the tasks
  - Duration of the project (through critical path)
- Often represented as a diagram

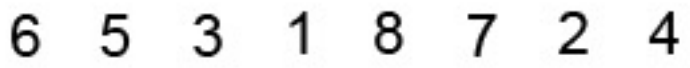
Planning a project

PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Planning a project

Bubble sort?



An example of bubble sort. Starting from the beginning of the list, compare every adjacent pair, swap their position if they are not in the right order (the latter one is smaller than the former one). After each iteration, one less element (the last one) is needed to be compared until there are no more elements left to be compared.

Image and caption, wikipedia

Elements of a software project

Wooclap

https://www.wooclap.com/L3GL4

Planning a project

Estimating time

- Comes with experience
- In practice, we tend to underestimate the time necessary
- in PERT, we can compute the expected time as a weighted average of
  - o, the optimistic time (everything goes perfectly) weight = 1
  - p, the pessimistic time (everything goes wrong) weight = 1
  - m, the most likely time, weight = 4

- expected time = $\frac{o+p+4m}{6}$
- Derivated from Beta distribution

Planning a project

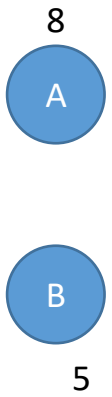
PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Planning a project

PERT

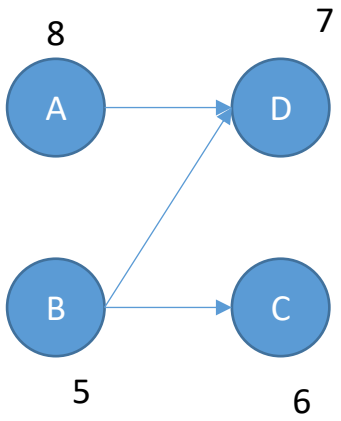
Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G



Planning a project

PERT

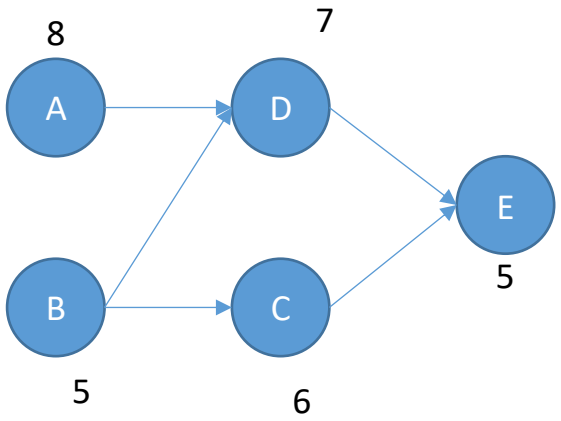
Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G



Planning a project

PERT

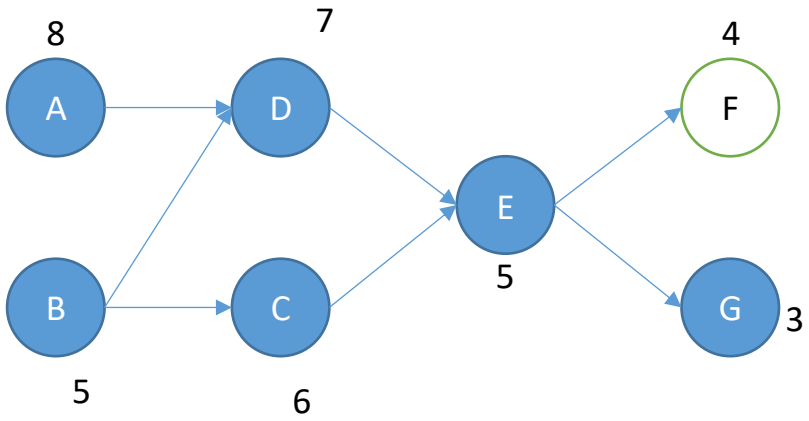
Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G



Planning a project

PERT

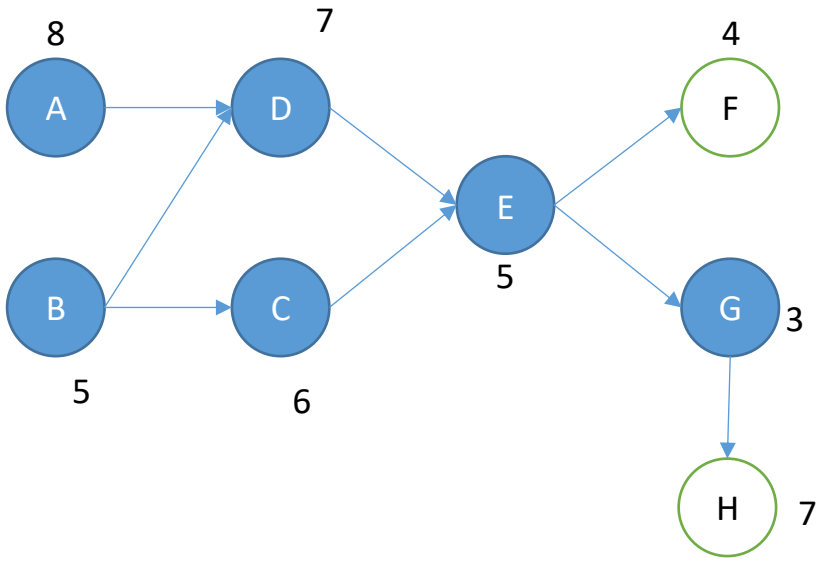
Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G



Planning a project

PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G



Planning a project

PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Task name	Start date	End date
A		
B		
C		
D		
E		
F		
G		
H		

Planning a project

PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Task name	Start date	End date
A	0	8
B	0	5
C	5	11
D	8	15
E	15	20
F	20	24
G	20	23
H	23	30

Planning a project

Critical Path

- Critical path: the set of tasks that allow to obtain the shortest time to finish the project
- Consequence: if one of the tasks from the critical path takes longer to be performed, the project will take longer to finish

- Algorithm for critical path:
  1) Select the taks with the latest finish date
  2) Put the selected task in the critical path
  3) Select the predecessor(s) of the selected task with the latest finishing date
  4) Repeat 2-3 until reaching starting node(s)

Planning a project

PERT

1) Select the taks with the latest finish date
2) Put the selected task in the critical path
3) Select the predecessor(s) of the selected task with the latest finishing date
4) Repeat 2-3 until reaching starting node(s)

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Task name	Start date	End date	Critical path
A	0	8	
B	0	5	
C	5	11	
D	8	15	
E	15	20	
F	20	24	
G	20	23	
H	23	30	

Planning a project

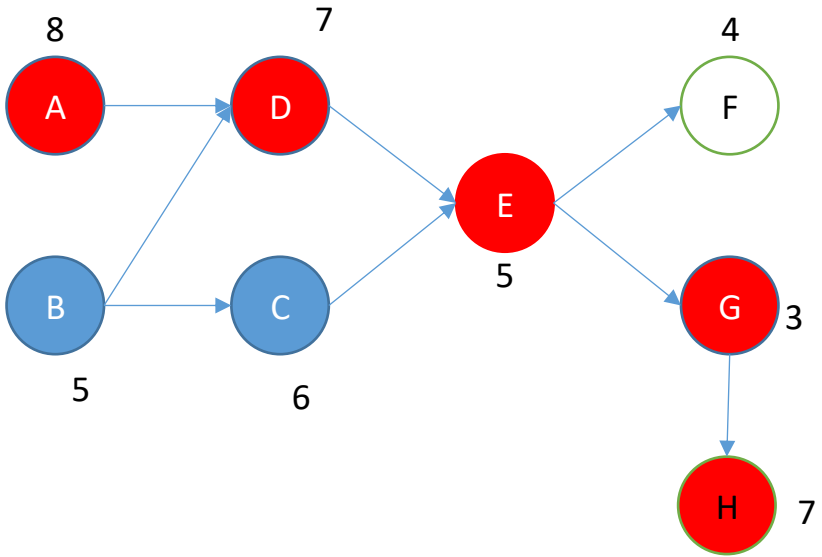
PERT

Task name	Time allocated	Predecessor(s)
A	8	
B	5	
C	6	B
D	7	A, B
E	5	C, D
F	4	E
G	3	E
H	7	G

Task name	Start date	End date	Critical path
A	0	8	X
B	0	5	
C	5	11	
D	8	15	X
E	15	20	X
F	20	24	
G	20	23	X
H	23	30	X

Planning a project

PERT



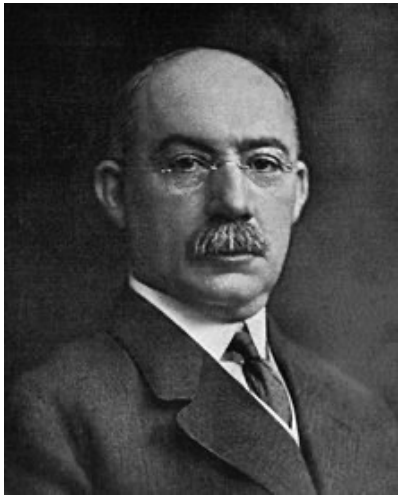
Task name	Start date	End date	Critical path
A	0	8	X
B	0	5	
C	5	11	
D	8	15	X
E	15	20	X
F	20	24	
G	20	23	X
H	23	30	X

Planning a project

Gantt chart

- PERT: statistical tool
- Gantt: Better vizualisation, better communication

- Introduced by Henry Gantt around 1910

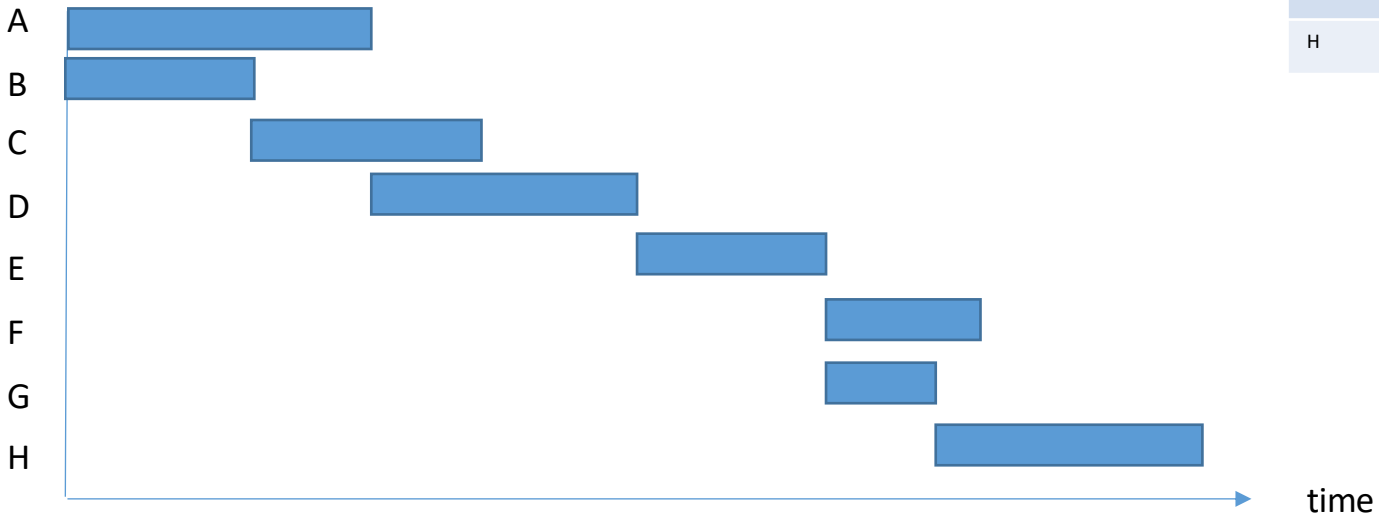


Planning a project

Gantt chart

- Basic principles:
  - y-axis: list of tasks
  - x-axis: time
  - Each task is represented by a rectangle

Task name	Start date	End date	Critical path
A	0	8	X
B	0	5	
C	5	11	
D	8	15	X
E	15	20	X
F	20	24	
G	20	23	X
H	23	30	X



Estimating the costs

Cost estimation

- Gantt chart: good base for bottom-up estimation
- Method 1, by lines of codes (LOC): Cost = $\alpha \times KLOC^{\beta} + \gamma$ with
  - α: marginal cost per 1000 LOC (KLOC)
  - γ: fixed cost of a project
  - β: scale factor
- To be noted: parameters proposed by Boehm in the COCOMO (COnstructive COst MOdel) method (1981)
- Estimation of the number of LOC: beta distribution
- Method 2, by person hours