Université de Paris

Génie Logiciel
Elements of a software project – Part 2

Sylvain Lobry

30/09/2022

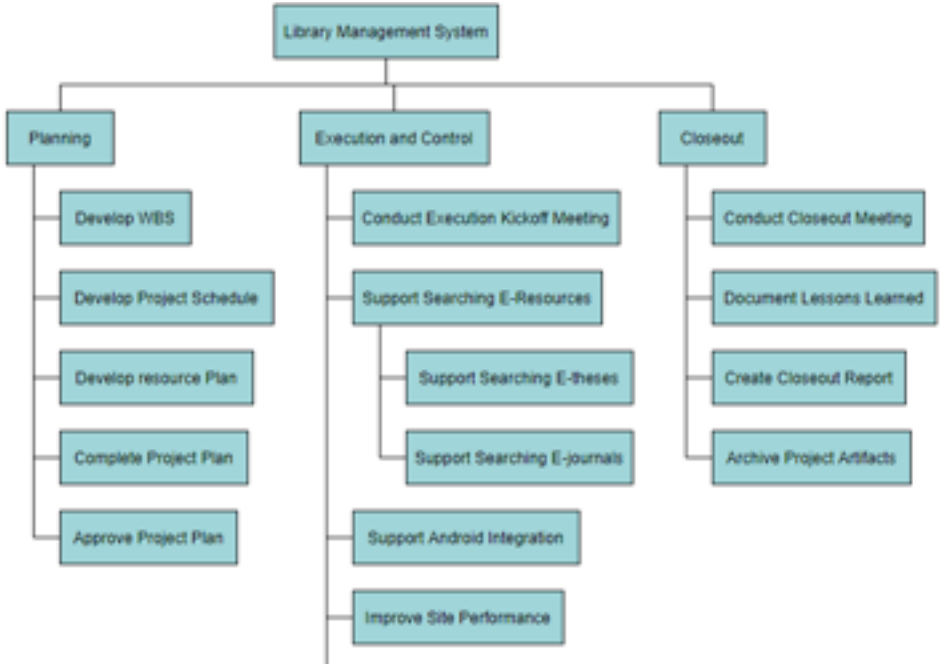
Planning a project

Planification

- Planification is an essential component of a successful project
- Remember: scope of a project depends on Quality, Time, Cost
- 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

Elements of a software project

Wooclap

https://www.wooclap.com/L3GL3

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

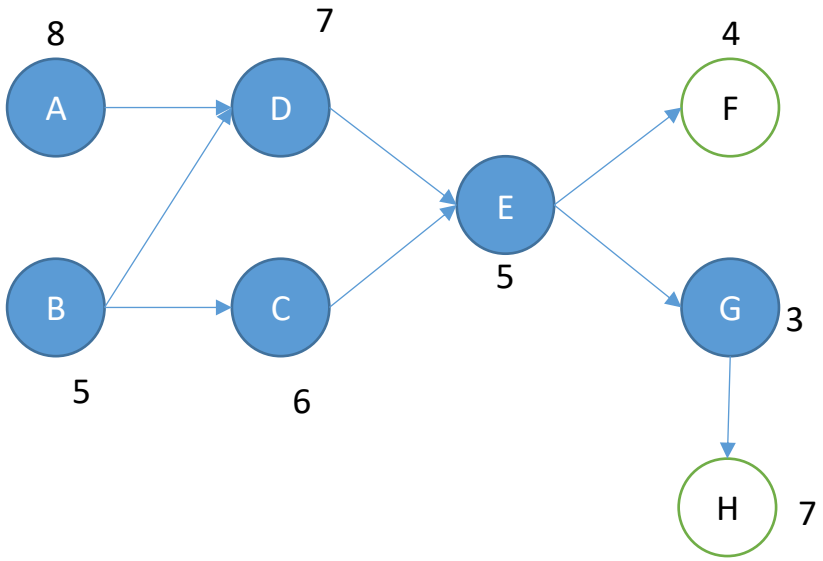
PERT

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

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

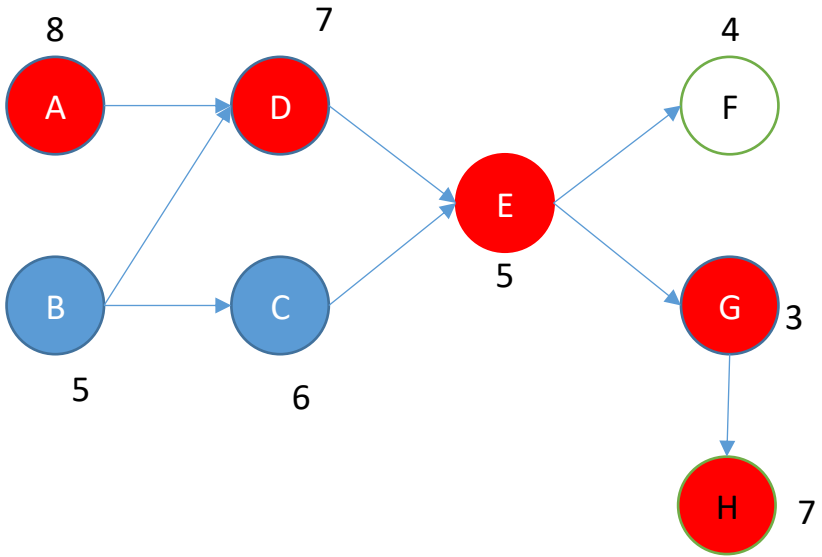
PERT

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



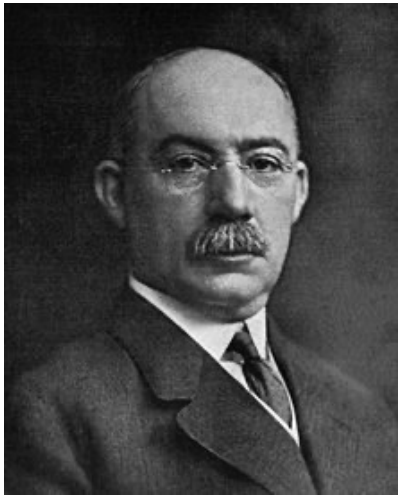
Task name	Start date	End date	Critical path
A	0	8	X
B	0	5	
C	5	11	
D	8	15	X
E	15	20	X
F	20	24	
G	20	23	X
H	23	30	X

Planning a project

Gantt chart

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

- Introduced by Henry Gantt around 1910

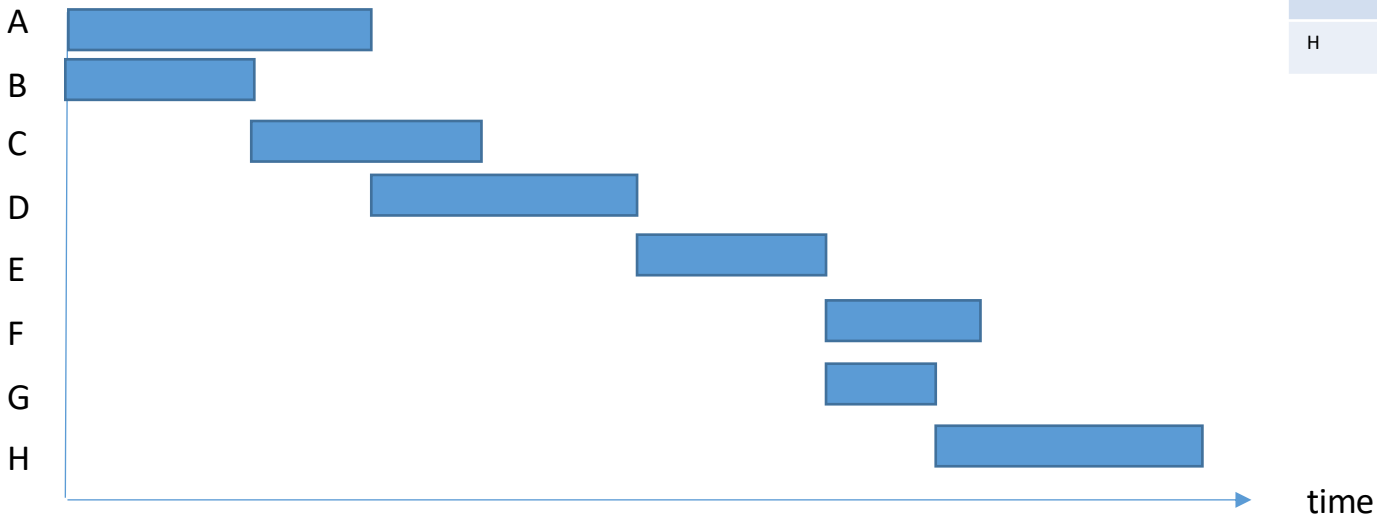


Planning a project

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

Risk management

Risk

- Needs to be taken into account in a software project
- Risk: what is the probability that a negative event will happen AND what will be the impact of this negative event on the project
- Different from uncertainty

Risk management

Types of risk

- Classification 1 : Human/Management/Technical
- Classification 2: Process/Quality/Viability
- Classification 3: Impact only a given project or all projects

Risk management

Potential causes of risk

- Human risks

- Plan or budget misestimated
- Requirements not well estimated

Management risks

- External causes
- Underwhelming performances

Technical risks

- Other risks?
- Potential solutions?

Risk management

Cost of a risk

- Expected value = probability of a risk x cost
- Possibility to represent the risks as a decision tree
- Example: should I do a prototype ?

If you do the prototype, it will cost you $100,000; and, of course, if you don't pursue it, there will be no cost. If you do the prototype, there is 30 percent chance that the prototype might fail, and for that the cost impact will be $50,000. However, if the prototype succeeds, the project will make $500,000. If you do not do any prototype, you're already taking a risk, the chance of which is 80 percent with a failure impact of $250,000. But, again, without a prototype, should you succeed, the project will make the same money as mentioned before. What should you do?

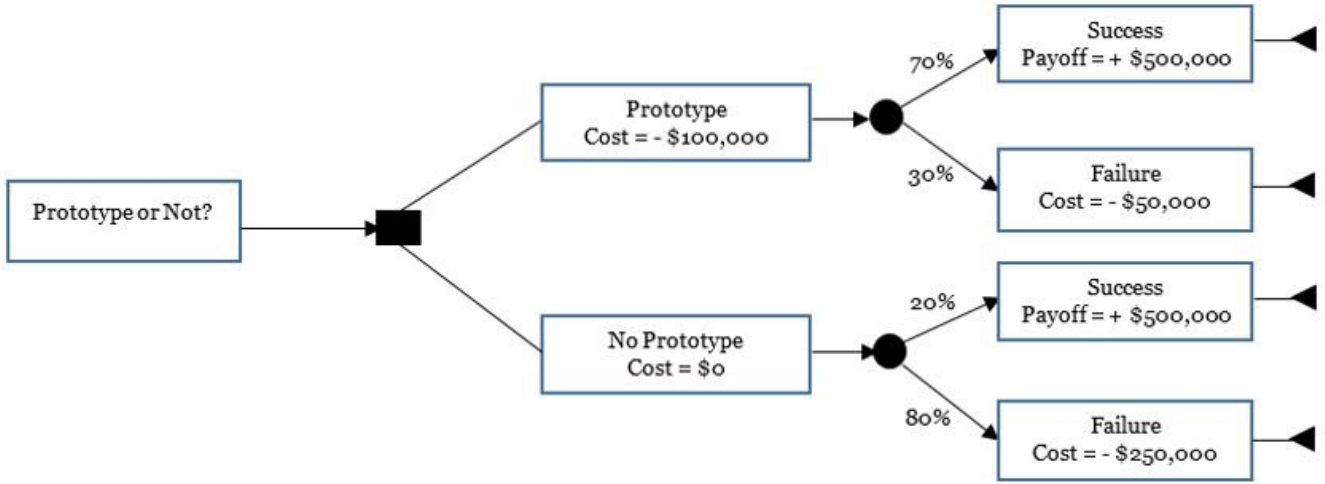
MPUG

Risk management

Cost of a risk

If you do the prototype, it will cost you $100,000; and, of course, if you don't pursue it, there will be no cost. If you do the prototype, there is 30 percent chance that the prototype might fail, and for that the cost impact will be $50,000. However, if the prototype succeeds, the project will make $500,000. If you do not do any prototype, you're already taking a risk, the chance of which is 80 percent with a failure impact of $250,000. But, again, without a prototype, should you succeed, the project will make the same money as mentioned before. What should you do?

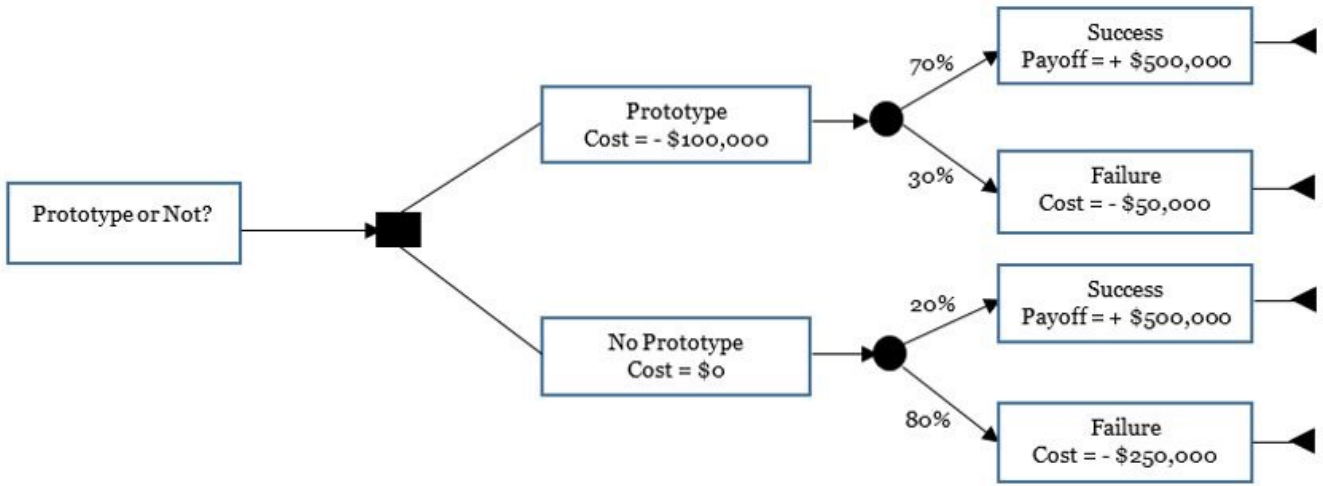
MPUG



Risk management

Cost of a risk

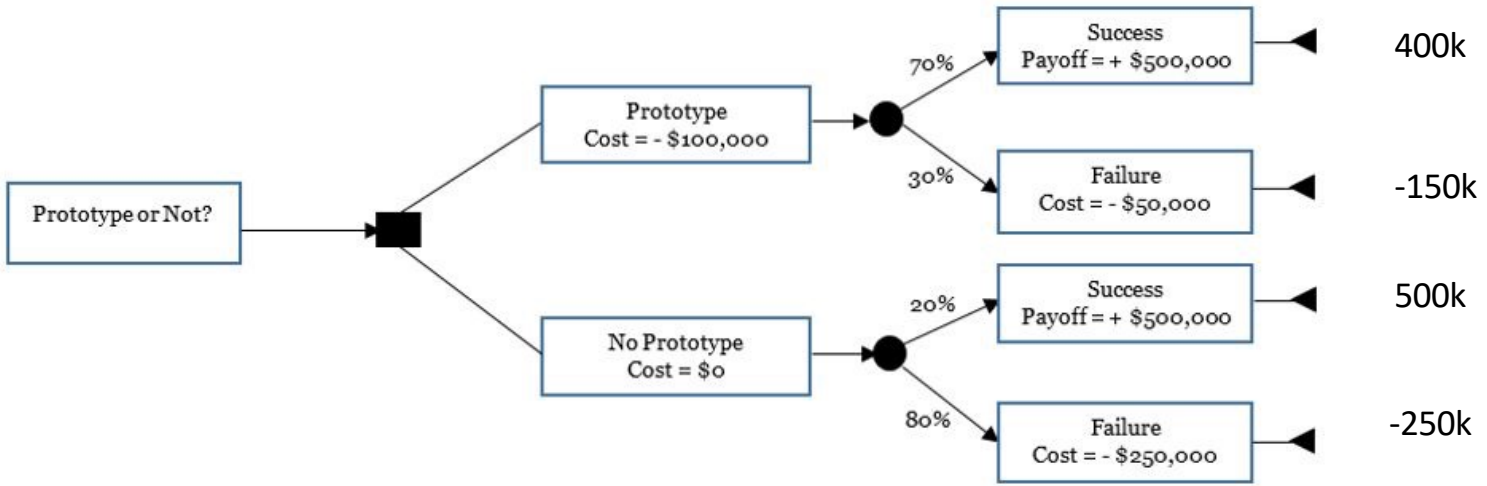
1) Compute Net cost of each path



Risk management

Cost of a risk

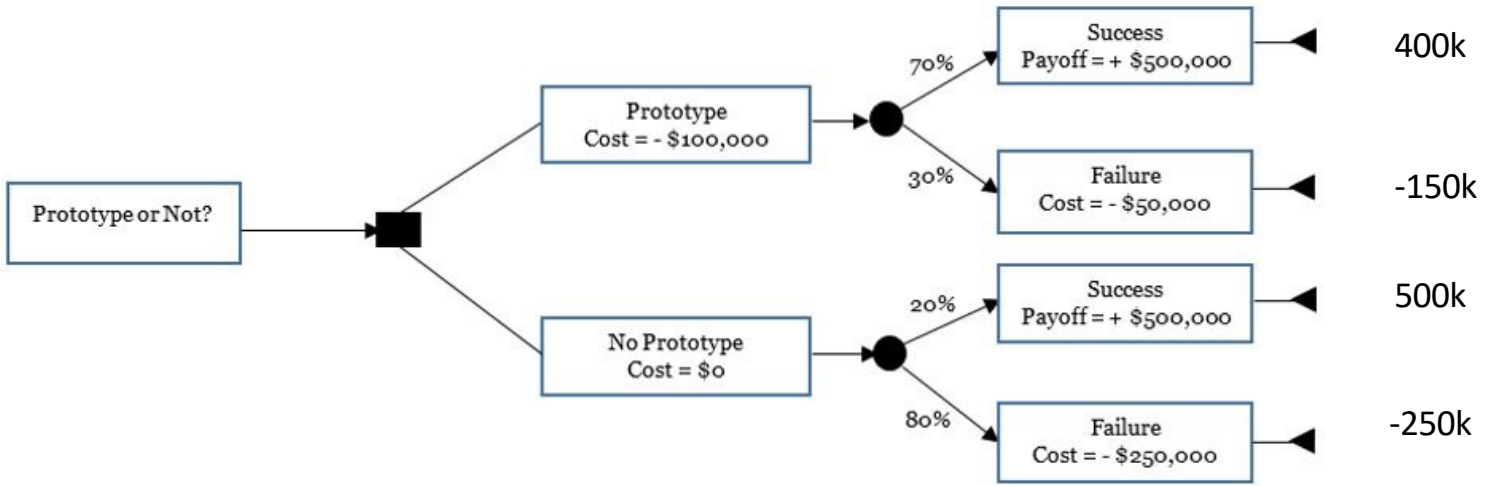
1) Compute Net cost of each path



Risk management

Cost of a risk

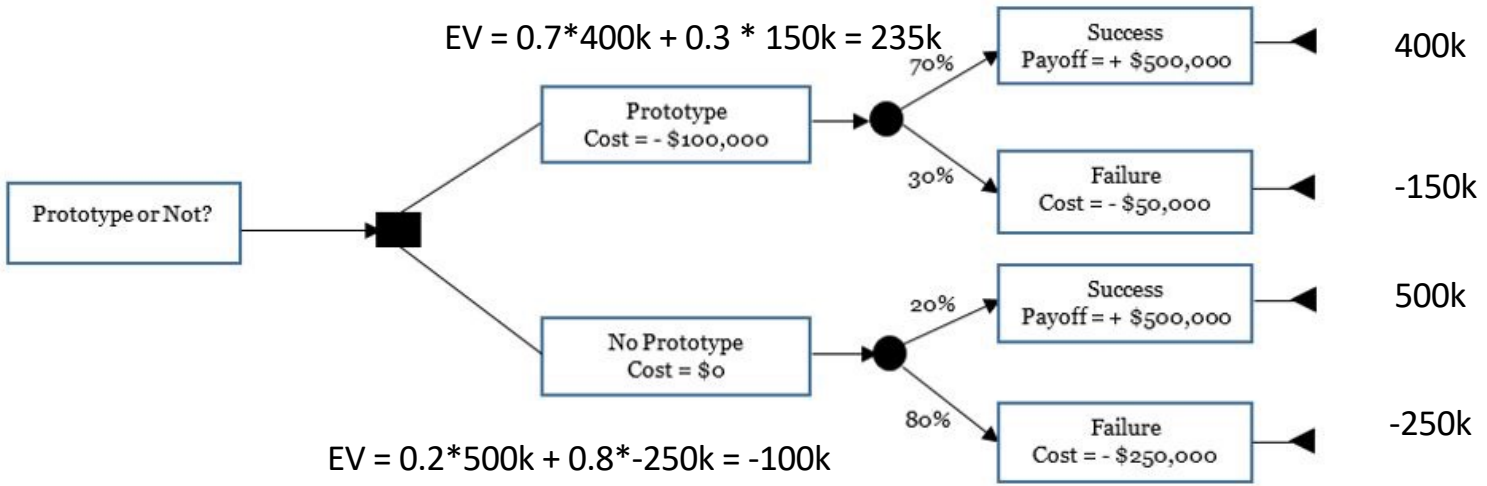
1) Compute Net cost of each path
2) Compute expected value for each path



Risk management

Cost of a risk

1) Compute Net cost of each path
2) Compute expected value for each path



Elements of a software project

- Numerous activities for a software project
- Planning should be one of the first things to do
- Should correspond to the scope of the project