Software Engineering 2

Software Project Management

Dr. Petr SPACEK[1]

[1]petr.spacek@fit.cvut.cz
Department of Software Engineering
Faculty of Informatics
Czech Technical University in Prague

Winter Semester, 2020

Outline

Outline

1. Intro
   - Connection to the previous course
2. Project Management
   - What it is?
   - How to do it?
   - PM in a nutshell
3. Conclusion
   - Summary
   - Literature

Connection to the previous course

Software engineering

The 4 main activities of Software engineering:

- Specification
- Development
- Validation
- Evolution & maintenance

Connection to the previous course

Software engineering

The 4 main activities of Software engineering:

- **Specification**
- **Development**
- **Validation**
- **Evolution & maintenance**

Outline

1. Intro
   - Connection to the previous course
2. Project Management
   - What it is?
   - How to do it?
   - PM in a nutshell
3. Conclusion
   - Summary
   - Literature

What it is?

... according to NASA

Stages:

1. begin with the planing and understanding the target domain
2. define the technical approach
   - selection and adaptation of appropriate life cycle model
   - selection of appropriate activities, methods and products
3. complete the project plan, define the leadership approach
   - organization, estimating, scheduling, ...
4. implementation of the project (the project plan execution)
   - monitoring, control, the project plan maintenance ...
5. close the project

What it is?

... according to Sybase

Stages:

1. initiation
   - the definition of the problem and possible solutions
   - project scheduling
2. execution
   - implementation of the plan
   - monitoring and management of the implementation (progress)
3. closeout
   - completion of the project

What it is?

... according to "best practices"

Project Management requires:

- planning
  - defining goals for product
  - structuring the project
  - review of the plan
  - scheduling the project
  - testing plan
  - costing/expenses the plan
  - re-factoring plan, according to project changes in certain circumstances
- execution of the current plan
- inclusion of changes into the plan and the project
- achieving of objectives related to the product, resources, quality
- coordinating the efforts of groups and individuals
- quality assurance execution

PMs relations

to the SE topis

Project Management has strong relations to:

- Software Requirements management
- Software Architectures
- Software Configuration Management
- Software Quality Assurance
- ...

They can be summarized into:

"*the more you know about what you manage the better you can manage*"

Outline

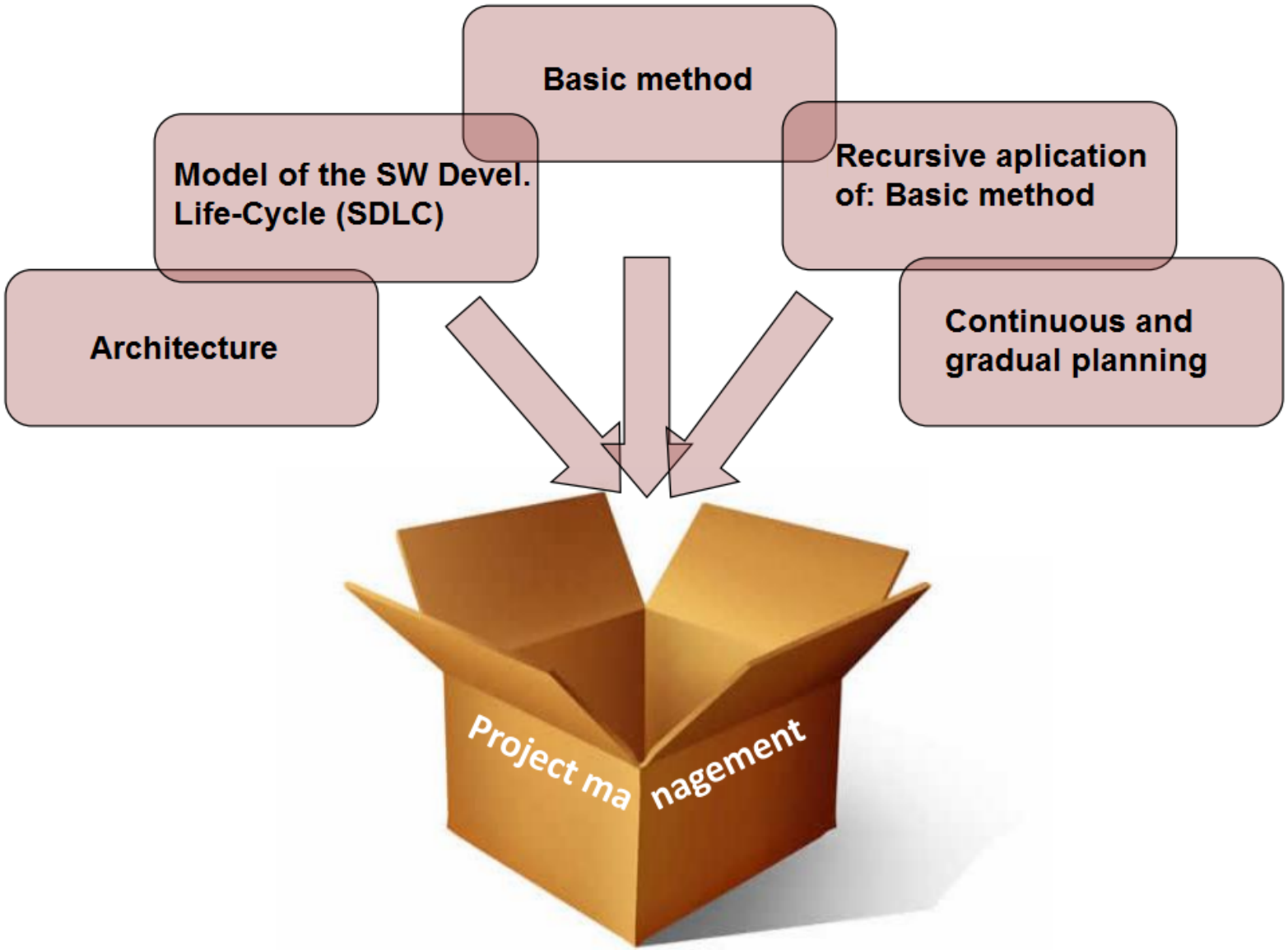
Basic method

- setup the project plan
- find out a understand "*what to do*"
- identify components, i.e. divide & conquer
  - create estimations
  - assign requirements to resources
  - analyse dependencies
  - create schedule
- assign resources to components
- measure, monitor, record
- learn from mistakes, i.e. fix and don't repeat
- record a store all project-data for the future

Organizing team

- how to assign work
- how to know who is working on what
- how to plan for humans
- estimation vs. real Man-Days (MDs)
- roles - pm, team-leader

Basic method in the context



Outline

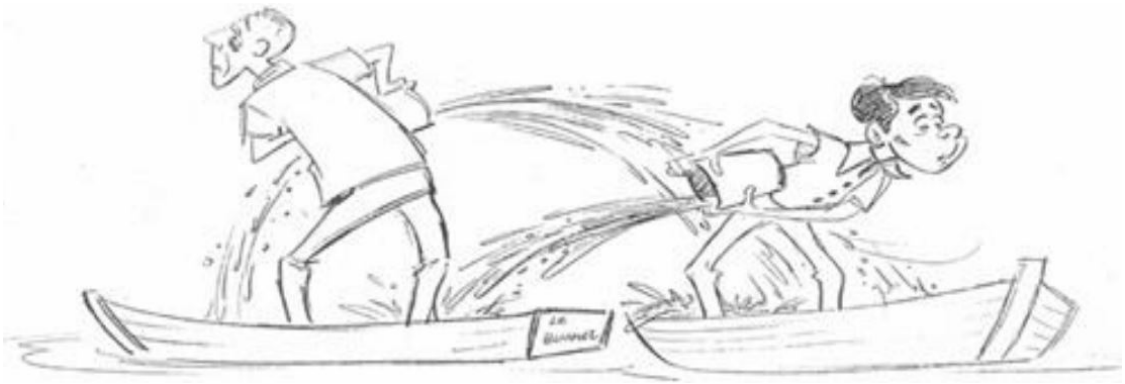
PM in a nutshell

- you must create and maintain a plan with a view to sufficiently distant future

- you need to be clear
  - the important dates
  - your and third-part commitments
- your people must know exactly
  - what to do in the next few days (about one week)
  - so that they can organize their own work

PM in a nutshell

- you must take care of the efficiency of your own people and to have everything well organized, measure and think about the economics of the project, to learn quickly from own and theirs mistakes
- you must understand the system, understand its complexity and to be able to decide if the client pushes you to a (scope, terms)-commitment on a meeting
- you need to communicate with the client, go to meetings, write notes, pass protocols, . . .
- you must have a summary record of all the problems and risks, and view their solutions (or elimination)
- you must be ever vigilant and **guard the scope**!



PM state tests

BREATHALYZER TEST

The Project Breathalyzer questions provide program managers with a "quick look" at software project health. The Breathalyzer determines whether key program elements exist, without which the program is not likely to succeed. If a program manager cannot answer the following questions about current project status, or must answer in the negative, the project should be scheduled for immediate review.

1. **Do you have a current, credible activity network supported by a Work Breakdown Structure (WBS)?**
2. **Do you have a current, credible schedule and budget?**
3. **Do you know what software you are responsible for delivering?**
4. **Can you list the current top ten project risks?**
5. **Do you know your schedule compression percentage?**
6. **What is the estimated size of your software deliverable? How was it derived?**
7. **Do you know the percentage of external interfaces that are not under your control?**
8. **Does your staff have sufficient expertise in the key project domains?**
9. **Have you identified adequate staff to allocate to the scheduled tasks at the scheduled time?**

The Joel Test

1. Do you use source control?
2. Can you make a build in one step?
3. Do you make daily builds?
4. Do you have a bug database?
5. Do you fix bugs before writing new code?
6. Do you have an up-to-date schedule?
7. Do you have a spec?
8. Do programmers have quiet working conditions?
9. Do you use the best tools money can buy?
10. Do you have testers?
11. Do new candidates write code during their interview?
12. Do you do hallway usability testing?

Outline

Honor, respect and adapt your project plan.

Outline

For Further Reading I

Abran, Alain and Bourque, Pierre and Dupuis, Robert and Moore, James W.
Guide to the Software Engineering Body of Knowledge - SWEBOK.
Piscataway, NJ, USA: IEEE Press. 2001.