



## Version Control Usage in Students' Software Development Projects

**Pekka Mäkiaho, Timo Poranen and Ari Seppi,  
University of Tampere, School of Information  
Sciences.**

**Pekka Mäkiaho ([pekka.makiaho@uta.fi](mailto:pekka.makiaho@uta.fi))**





# Introduction

- Background
- Research problem
- Methods
- Findings
- Conclusions
- Further Work



# Background

- Two courses, year 2012-2013: SW Project management and Project Work
- 13 projects, average size 1109 working hours
- 39 project managers, 59 developers
- 5-10 ECTS
- 6 WWW-, 4 desktop and 3 mobile phone (Android) - applications
- Free to select the set of SW-tools, including VCSs (Version control systems) Subversion repository was provided by SIS



# Research problem

- Not a specific problem but getting an overview on:
  - Student's background experience on VCSs
  - Set of VC-tools (clients) and VCSs selected
  - The differences on VCS-usage like process, number of commits
  - Challenges
- Target to improve practises and teaching on the courses

# Methods (1/2)

- Data was gathered by 2 Moodle-questionnaires
  - 1<sup>st</sup> - January (after 4 months from the begin)
  - 2<sup>nd</sup> – on the end of the March (after the project)
- Extra questions to the project managers
- Weekly reports from the teams included the number of the commits by each project
- Final report written by the project also gave information

## Methods (2/2)

- The answers were anonymized and grouped by the project and by the next themes:
  - The used tools and version control systems
  - Earlier experience and training
  - Number of commits during the project
  - Version control usage and problems
  - Project management and version control

# Findings - used tools

- The used tools and version control systems
  - 2 projects used GIT
  - 11 used Subversion repository provided by University
  - A vast majority of the projects using Subversion on Windows platform used TortoiseSVN as a client
  - The second most popular tools were Eclipse's SVN plugins (Subclipse and Subversive)

# Findings – earlier experience

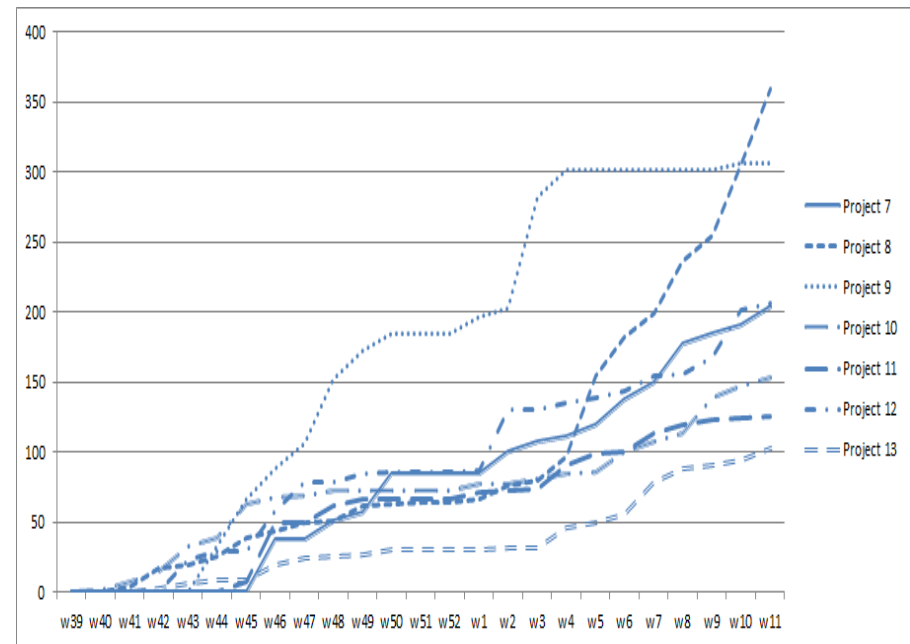
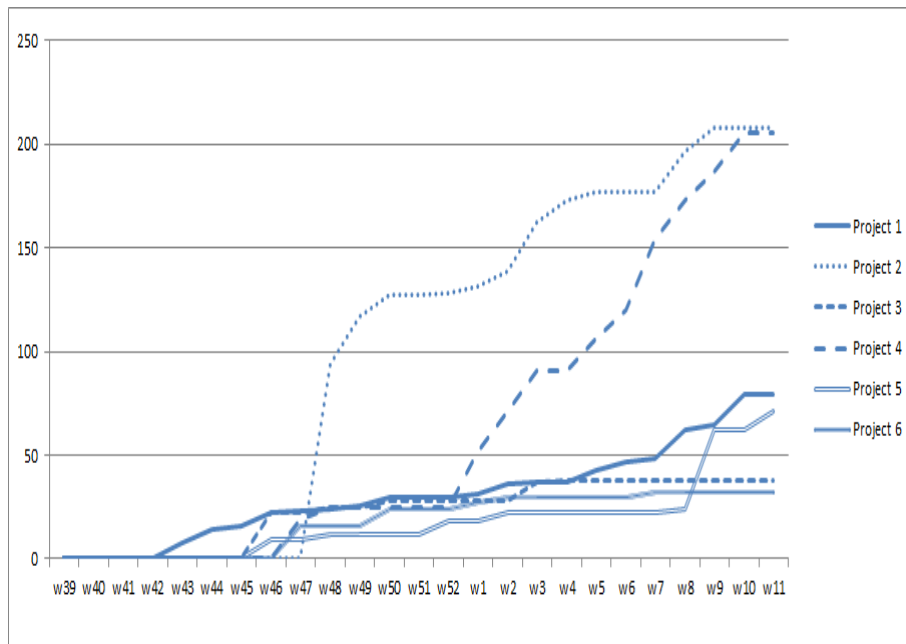
- The earlier experience and training
  - Out of the 13 groups all had at least one member who had no experience on VCSs but at least two members, who did
    - 52% had used VCSs on their own or university projects – 21% had industrial experience
  - Training to non-experienced members were mainly given on need basis. Only two groups had written instructions
  - The level of the training varied a lot



# Findings – number of commits

The average number of commits was 137 (30-160)

The commits by a student varied between 0 and 180 (avg. 23)



# Findings – VC-usage

- On the top of basic functionalities ('add', 'commit', 'update')
  - 2 groups used 'ignore'
  - 6 groups used 'status'
  - 5 groups used branching
- Very little changes on the usage during the projects
  - The only reported changes were 'committing more rapidly on the end of the project'

# Findings – VC-problems

- Three categories divided:
  - Familiarity with the chosen tool (11 groups out of 13)
  - Problems with the environment (7 groups)
  - VCSs principles (only 3 groups)
- Even if 27% of students had no experience
- Some groups did not report these even they certainly had problems on this. (USB-stick usage, etc..)

# Findings – VC and project management

Most groups did not have any releasing schedule

- Common problems related to the lack of experience on VCSs

- Project members did not update their workspace regularly and committed rarely -> **update conflicts and merge problems**

- Even the groups that did not have problems of this scale had difficulties in getting the project members to remember to write commit **comments**

# Conclusions and further work

- We noted that the understanding of the main principles of using version control is not always clear to students and this should be taken in account in teaching.
- The groups that reported lot of problems and used VCS in a non-conventional way also had less commits than the other teams
- The sample was quite small and we did not perform any deeper analysis to detect connection between number of commits, productivity and quality. This is one direction that needs further research

# Thank you!

- Questions?

- [Pekka.Makiaho@uta.fi](mailto:Pekka.Makiaho@uta.fi)