



**HY-464:**

**Συστήματα Ανάκτησης Πληροφορίας**

Information Retrieval Systems

Πανεπιστήμιο Κρήτης, Άνοιξη 2006

**Φροντιστήριο 5**

**Θέμα : Subversion**  
**Ημερομηνία : 6/4/2006**



# Outline

- Introduction
- Basics
  - Subversion cycle
  - Example
- Project
- Subversion interfaces



# Introduction

- **Source Control**
  - Elegant cooperation method beyond plain file-sharing techniques such as email
- **Benefits**
  - Version tracking
  - Easily accessible repository
  - Enhances collaboration
- **Drawbacks**
  - Enhances collaboration?
    - People tend to get used to a working style so a transition period is a necessity.



# Basics

- Subversion lifecycle
  1. **Check out** a project
  2. **Create/edit** files and subdirectories
  3. **Update** local copy from the repository
  4. If you are ready to commit, go to **step 5**, else go to **step 2**
  5. **Commit** changes, go to **step 2**



## Example (1/3)

- **Log in** to a debian linux machine (such as mandarini.csd.uoc.gr)
  - other machines : milo mandarini firiki peponi syko stafyli
- **Check out** a repository for the first time
  - *svn co repository\_url*
    - co stands for “check out”
- **Edit a file**
  - open an editor and edit an existing file
  - *pico some\_file*
- **Update**
  - before checking in make sure nobody else committed changes
  - *svn update*



## Example (2/3)

- **Commit**

- *svn commit filename1 filename2*
- if filename is omitted the whole directory will be committed
- an editor will popup asking you to write a comment regarding the changes you are about to commit
- svn reply:
  - Filename
  - Transmitting file data...
  - Committed revision 3
- Each committed change is assigned a revision number
- *svn log -v -r3* tells you which files were touched in each revision
- If you change your mind about committing just exit the editor



## Example (3/3)

- **Adding a new file**

- Subversion knows when you are changing an existing file but can't tell when you want to add a new file
- *svn add file*

- **Looking around**

- *svn status*
- lets you see what you have edited

- **Help**

- *svn help*

- **In conclusion**

- *svn co, svn update, (edit or add), svn status, svn commit*



# Project

- For each component we will create a directory on <http://www.csd.uoc.gr/~hy463/google>
  - example : crawler => <http://www.csd.uoc.gr/~hy463/google/crawler>
- Each team will have access to all folders and can see what the other teams are doing
- If a team needs to use another teams component they can easily add the other teams package to their project and continue the development in parallel
- If one team member needs to use another members code he only needs to take it from the repository





# Questions





# References

- <http://subversion.tigris.org/>
- Downloads
  - [http://subversion.tigris.org/project\\_packages.html](http://subversion.tigris.org/project_packages.html)
  - TortoiseSyn (adds svn functionality in context menu)
    - <http://tortoisesvn.tigris.org/>