Robotics Programming Laboratory

Bertrand Meyer
Jiwon Shin
Andrey Rusakov

Software Engineering Tools
A Story of a Woodcutter

..."I must be losing my strength", the woodcutter thought.

- "When was the last time you sharpened your axe?" the boss asked.

- "Sharpen? I had no time to sharpen my axe. I have been very busy trying to cut trees..."
Engineering Tools

Why do we use tools?

Because things we usually create are complicated and with tools we can create them easier.
Software Engineering definition

SE - the multi-person construction of multi-version software

(David Parnas, 1975)
Benefits of Using Tools

Tools

- Minimize time of routine operations
- Minimize human factor
- Provide more information about the system
- Provide more information about the process
Which SE tools do you use?
Text editors vs. IDEs

IDEs provide:

- Syntax highlighting/checking
- Auto completion
- Feature “navigation” (e.g. Go to the definition)
- Refactoring tools (see following slides)

General purpose text editors can also offer some of these features!

One of the main advantages of using general purpose text editor: you don’t have to install any additional software to start writing your code.
Refactoring

Code refactoring is a "disciplined technique for restructuring an existing body of code, altering its internal structure without changing its external behavior"
Refactoring techniques

Techniques that allow for more abstraction
- Encapsulate Field
- Generalize Type
- Replace type-checking code with State/Strategy
- Replace conditional with polymorphism

Techniques for breaking code apart into more logical pieces
- Componentization
- Extract Class
- Extract Method

Techniques for improving names and location of code
- Move Method or Move Field
- Rename Method or Rename Field
- Pull Up
- Push Down
Refactoring Tools: Examples

Integrated refactoring tools:

- Eclipse
- NetBeans
- MS VisualStudio
- EiffelStudio

ReSharper
Debuggers

Integrated debuggers:

- EiffelStudio
- MS VisualStudio
- Eclipse

**GDB (GNU Debugger)** - a command-line debugger for several languages, including C and C++

**DDD (Data Display Debugger)** - is a graphical front-end for command-line debuggers such as GDB

**Valgrind** (memory debugger)
Profilers & Performance analyzers

Integrated profilers

• Eclipse
• MS VisualStudio
• NetBeans
• EiffelStudio

Intel VTune
Testing

Unit testing
• JUnit
• NUnit
• CppUnit
• Autotest

GUI testing
• Selenium (web applications)

Testing multi-threaded applications
• ConTest
Test-Driven Development

TDD cycle:

• Add a test

• Run all tests and see if the new one fails

• Write some code

• Run tests

• Refactor code

• Repeat
EiffelStudio

DEMO
Version control

SVN

Git

Mercurial
SVN

Repository

SVN server

Update
Commit
Update
Commit

Dasha’s machine
Local working copy

Ivan’s machine
Local working copy

Local working copy

SVN
SVN basics

**Commands:**
- checkout
- add
- remove
- update
- revert
- diff
- commit

**Common terms:**
- Diff
- Revision
- Branch
- Merge
Git

Remote repository

Git server

Dasha’s machine

Local repository

Local working copy

Commit

Reset

Ivan’s machine

Local repository

Local working copy

Commit

Reset
Git basics

Commands:
- clone
- add
- rm
- status
- pull
- commit
- push

Git workflows:
- Centralized
- Feature Branch
- Gitflow
- Forking
- ...

Bug trackers, Issue trackers

JIRA
Bugzilla
Redmine
Trac
Jazz
Build tools

Ant

Maven

Cmake
Continious Integration

Jenkins

Teamcity
Object-oriented analysis and design

BON

UML