Introduction to CTIP

Lee, Dong-Ah

2013-03-08

Dependable Software Laboratory

KU KONKUK UNIVERSITY

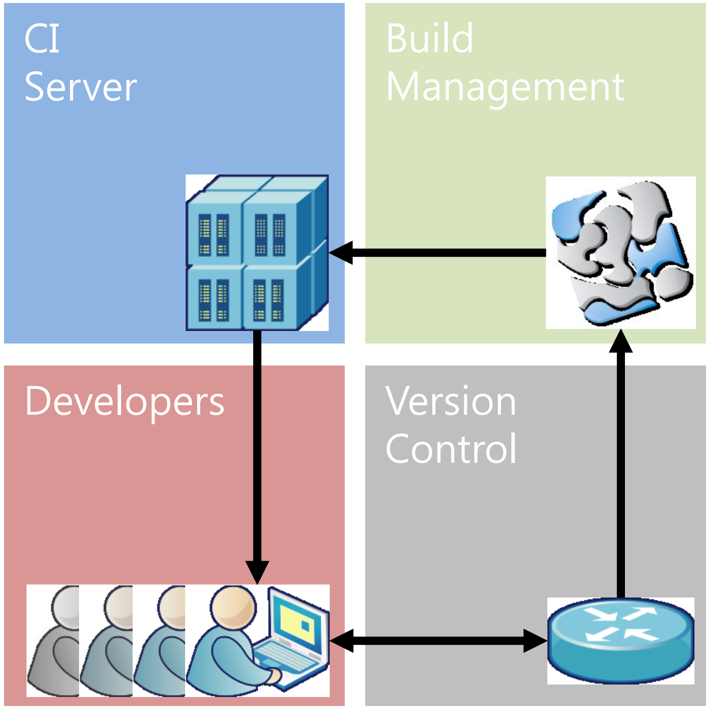
Contents

- CI?
- CTIP?
- COMPONENTS?
- ADVANTAGES!
- DISADVANTAGES?
- 2013 SOFTWARE VERIFICATION

CI?

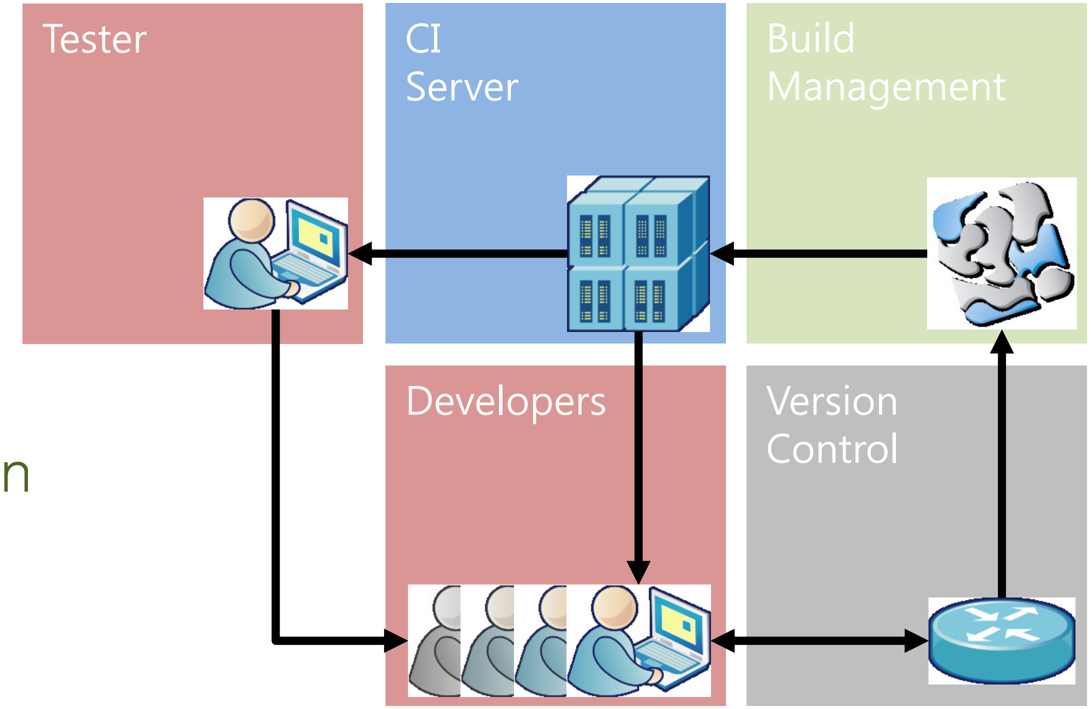
Continuous Integration

- As soon as the work on a task is complete, it **is integrated into the whole system.**
- After such integration, all the **unit tests** in the system must pass.



CTIP?

Continuous Test & Integration Platform



- Continuous Integration

  + Continuous Test

COMPONENTS?

Tool	Name
Project Management	OpenProj, GanttProject, Mylyn, Hsarepoint
Quality Management	Splint, PMD, JDepend, Checkstyle,
Requirement Analysis & Management	Jfeature, JRequisite
Design & Modeling	StarUML, VioletUMLEditor, ArgoUML
Configuration Management & Version Control	Trac, Hudson, Maven, CVS, SVN, Bugzillia, Mantis, TortoiseCVS, Mercurial, Cruise Control, Git, Ant, RedMind
Implementation	Eclipse, Valgrind, Netbeans
Testing	TestLink, CPPUnit, OProfile, HttpUnit, EMMA, Jmeter, Nunit, SoapUI, FitNesse
Static Analysis	Coverity, dart, sparrow, cppcheclipse, Resort, sonar

ADVANTAGES! (Wikipedia)

- When unit tests fail or a bug emerges, developers might revert the codebase to a bug-free state, without wasting time debugging
- Developers detect and fix integration problems continuously — avoiding last-minute chaos at release dates, (when everyone tries to check in their slightly incompatible versions).
- Early warning of broken/incompatible code
- Early warning of conflicting changes
- Immediate unit testing of all changes
- Constant availability of a "current" build for testing, demo, or release purposes
- Immediate feedback to developers on the quality, functionality, or system-wide impact of code they are writing
- Frequent code check-in pushes developers to create modular, less complex code [citation needed]
- Metrics generated from automated testing and CI (such as metrics for code coverage, code complexity, and features complete) focus developers on developing functional, quality code, and help develop momentum in a team
- ……………………

DISADVANTAGES? (Wikipedia)

- Initial setup time required
- Well-developed test-suite required to achieve automated testing advantages

2013 Software Verification

Slides & Reports

- TP#1： Junit , Eclipse , Clover & JDepend
- TP#2： Ant & CruiseControl
- TP#3： Mantis , SVN & CTIP
- TP#4： System Test
- TP#5： System Test – Static Analysis
- TP#6： Final Presentation

All your presentations will be recorded.

2013 Software Verification

3학년 "소프트웨어 모델링 및 분석" 수업과 연동해서 진행합니다.

- 3학년 수업의 개발 결과물에 대해서 시스템 테스팅을 수행한 후 결과를 Mantis를 사용하여 공유합니다.
- 3학년 수업에서 단위 테스팅 및 관련 분석 (5,6,7번)을 수행할 수 있도록 CTIP 환경을 제공합니다.
- 테스팅 결과에 대한 의견교환은 Mantis를 사용합니다.
- 테스팅을 위한 소스코드는 SVN을 통해 공유합니다.