Summary

A serial process that combines the codes written by developers to create executable products is called the integrated build. It uses CI for continuous and automatic integration. The Continuous Integration means a software development practice that a team consisting of multiple developers integrates their works once or more a day or server time a day.

Description

CI Effects and characteristics

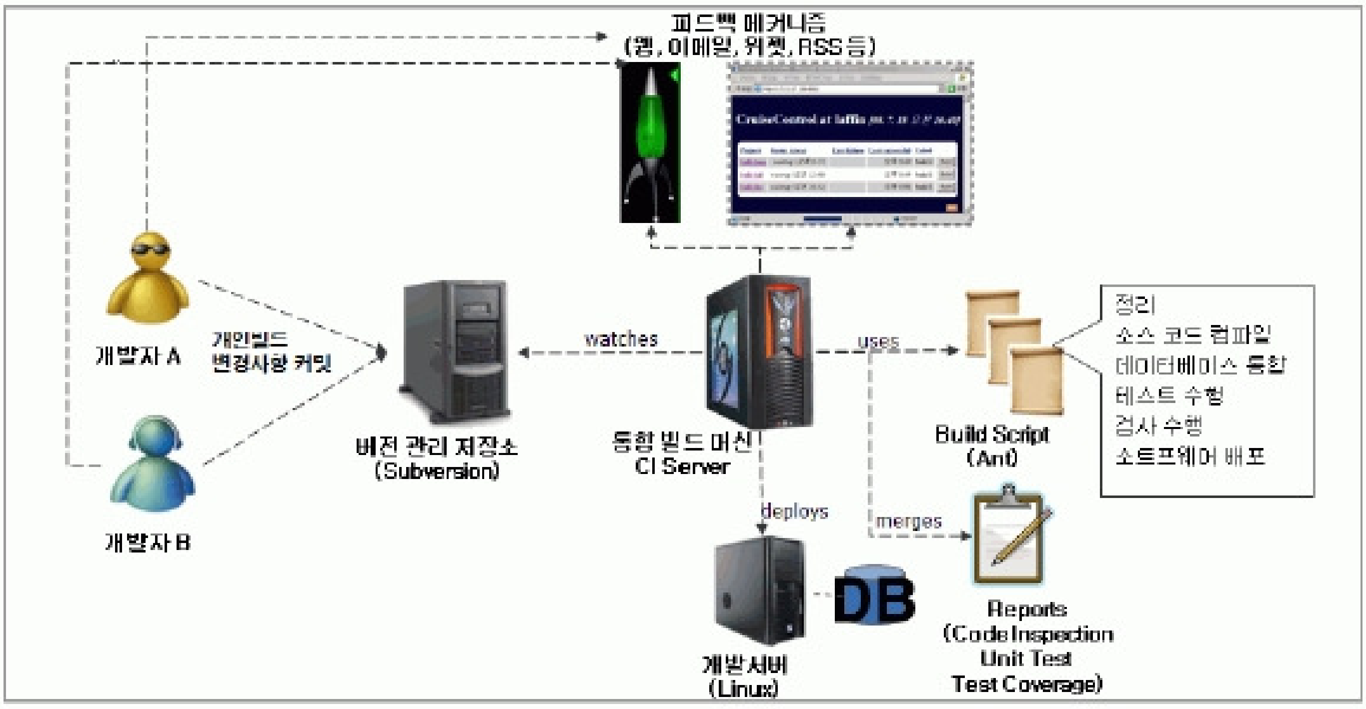
Effects

1. Reduces common and general risks.
2. Errors are early detected. With more integration, the scope of error occurrence becomes smaller, making it easier to discover errors.
3 Software become well working and error free.
4. Software which can be distributed anytime anywhere is provided.
5. Reduces repetitive manual works.
6. Makes the project visibility better.
7. The development team will be more proud of their software.

Characteristics

1. Consistent source code: needs a source management system.
2. Automatic build: Build should be carried out automatically based on the CI tool.
3. Automatic test: Verify the source and function/non-function testing to improve the code quality.
4. Daily check and build: The longer the build cycle becomes, the higher the probability of error becomes in the sources, which will waste the time.

CI (Continuous Integration) is composed of developers, version management storage, CI system, build script, communication device and integrated build machine.



The CI server receives the source codes committed by developers and checked out by the configuration server. It stores them in CI server's local file system and, based on the codes, creates code products (working packages such as WAR or JAR).

Open source CI server: Hudson

Hudson is an open source CI server that provides a mechanism that feeds back the automatic build that uses the source and script and build results to the developer.

Features and functions of Hudson

Features of Hudson

1. Easy installation : Distribute to the servlet container in a form of hudson.war
2. Use a web-based UI to set up easily and feed back instantly
3. Provide plug-in based scalability
4. Simple and well-defined documents and APIs

Functions of Hudson

1. Automatic software build
    - Provides daily or weekly build function
2. Continuous and automatic build verification
    - Uses SCM polling to carry out the latest code-based build
3. Continuous and automatic build test
    - Execute a test suite to verity code quality

4. Automated post-build procedures
    ▪ Packaging and test reporting of the compiled code

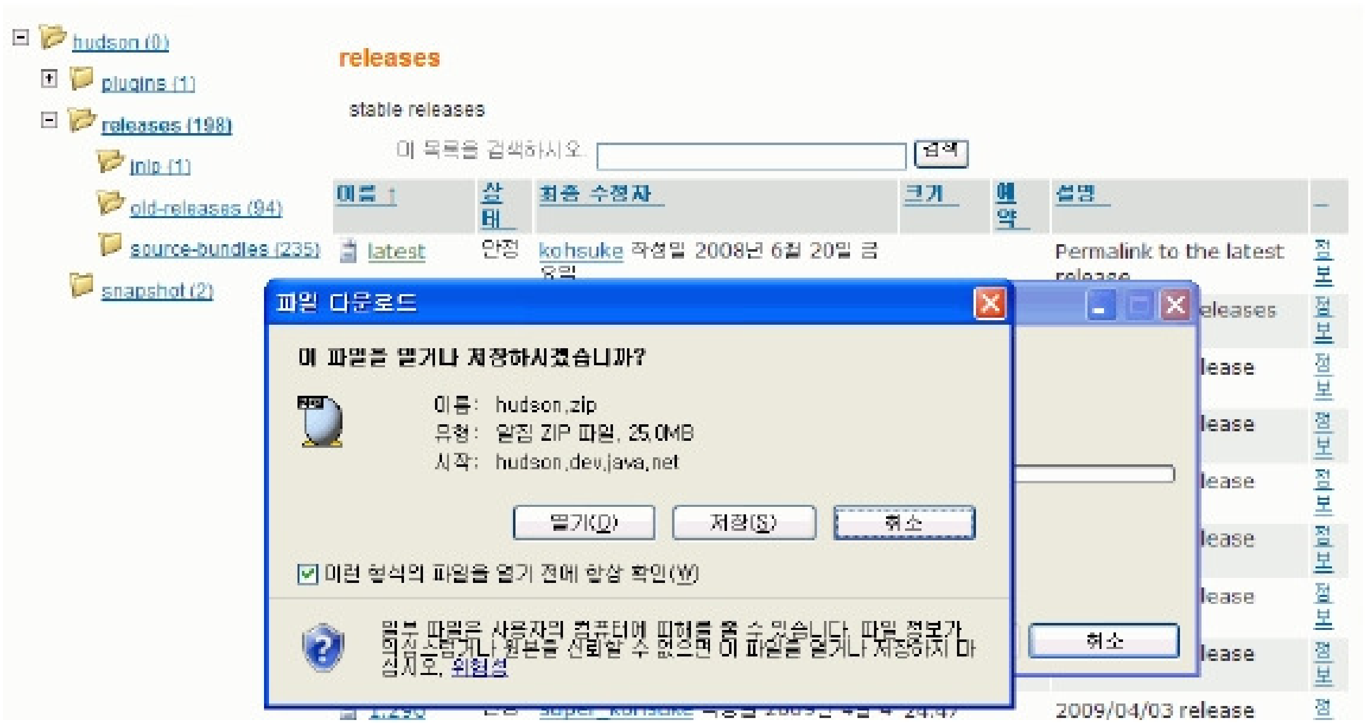
Environmental settings

Install Hudson

1. Download a hudson.war file.

Hudson download

[https://hudson.dev.java.net/servlets/ProjectDocumentList?folderID=2761&expandFolder=2761&folderID=0]



hudson (0)
plugins (1)
releases (198)
inip (1)
old-releases (94)
source-bundles (235)
snapshot (2)

releases

stable releases

이 목록을 검색하시오. 검색

이름 ↑	상태	최종 수정자	크기	예약	설명
latest	안정	kohsuke 작성일 2008년 6월 20일 금요일			Permalink to the latest release
1.296		super_kohsuke			2009/04/03 release

파일 다운로드

이 파일을 열거나 저장하시겠습니까?

이름: hudson.zip
유형: 알집 ZIP 파일, 25.0MB
시작: hudson.dev.java.net

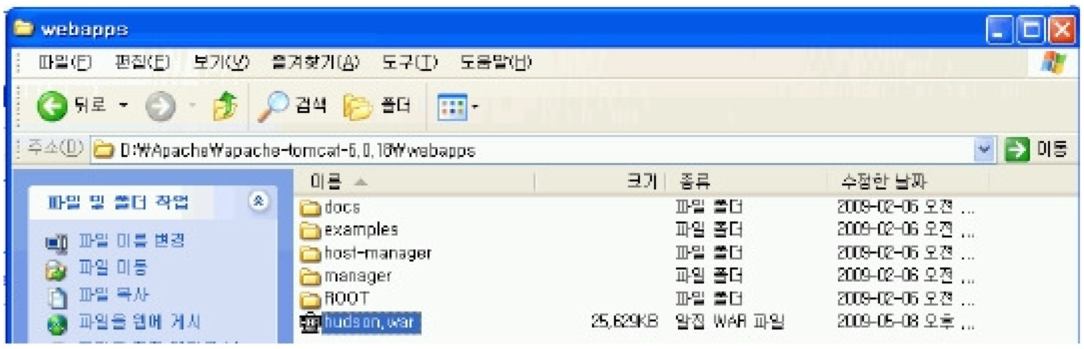
열기(O) 저장(S) 취소

☑ 이런 형식의 파일을 열기 전에 항상 확인(W)

일부 파일은 사용자의 컴퓨터에 피해를 줄 수 있습니다. 파일 정보가 의심스럽거나 원본을 신뢰할 수 없으면 이 파일을 열거나 저장하지 마십시오. 위험성

2. Change the extension of hudson.zip to .war.

3. Place the downloaded hudson.war in the webapp directory of Tom Cat.



webapps

파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)

주소(D) D:₩Apache₩apache-tomcat-6.0.18₩webapps

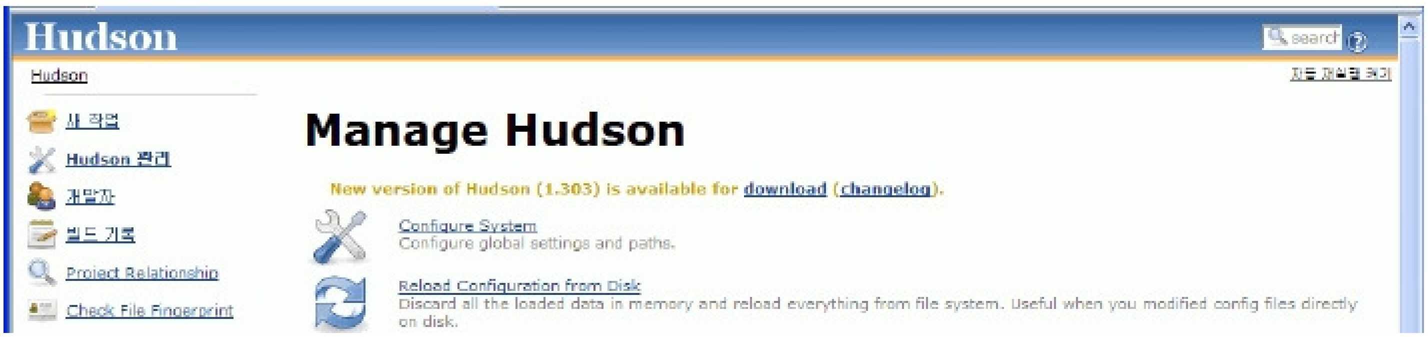
파일 및 폴더 작업
파일 이름 변경
파일 이동
파일 복사
파일을 웹에 게시

이름	크기	종류	수정한 날짜
docs		파일 폴더	2009-02-06 오전 ...
examples		파일 폴더	2009-02-06 오전 ...
host-manager		파일 폴더	2009-02-06 오전 ...
manager		파일 폴더	2009-02-06 오전 ...
ROOT		파일 폴더	2009-02-06 오전 ...
hudson.war	25,629KB	알집 WAR 파일	2009-05-08 오후 ...

4. Run startup.bat to start up Tom Cat server.

Hudson System setup

- Use Configure System menu to designate the installation locations for JDK, ANT and Maven for software build and feedback.


Hudson

search

자동 갱신 켜기

Hudson

새 작업
Hudson 관리
개발자
빌드 기록
Project Relationship
Check File Fingerprint

Manage Hudson

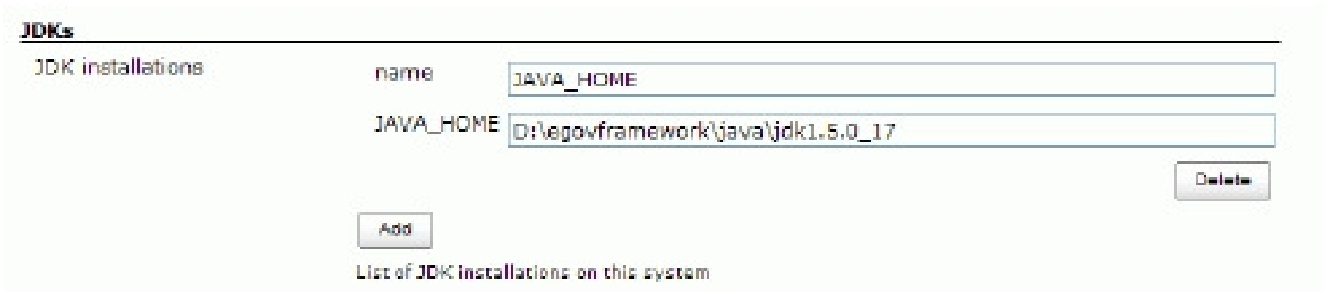
New version of Hudson (1.303) is available for download (changelog).

Configure System
Configure global settings and paths.

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Set JDK, Maven, Ant, build result feedback mechanism to the defaults.

1. JDK Home setup

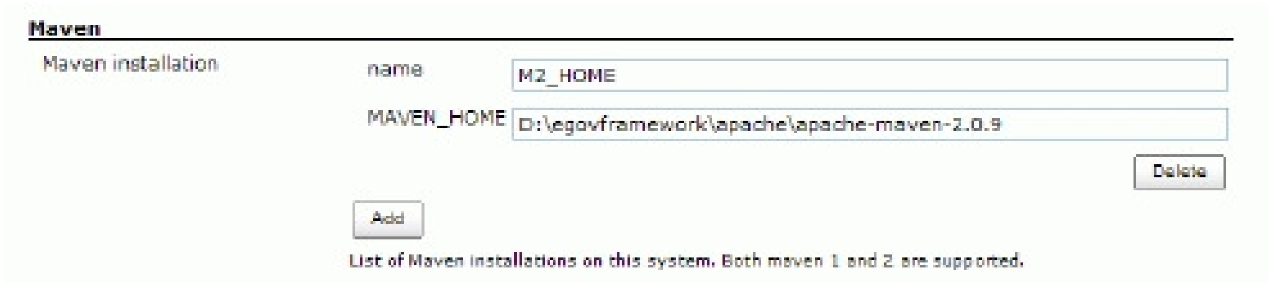

JDKs

JDK installations	name	JAVA_HOME
	JAVA_HOME	D:\egovframework\java\jdk1.5.0_17

List of JDK installations on this system

2. Maven Home setup

Designate the Maven Home for Maven build

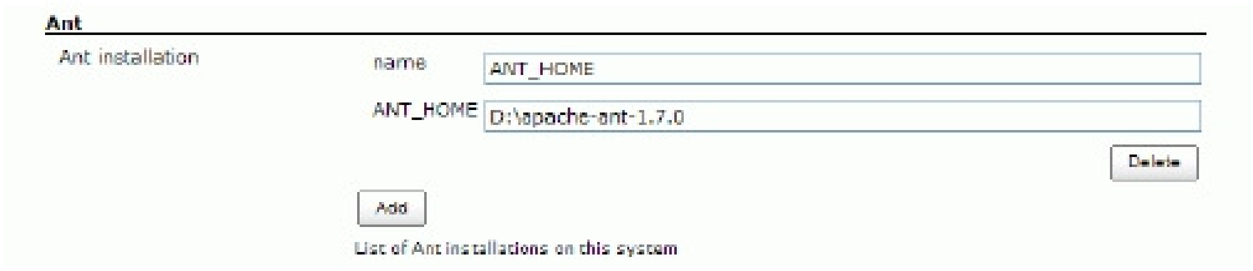

Maven

Maven installation	name	M2_HOME
	MAVEN_HOME	D:\egovframework\apache\apache-maven-2.0.9

List of Maven installations on this system. Both maven 1 and 2 are supported.

3. Ant Home setup

Designate the location for Ant Home

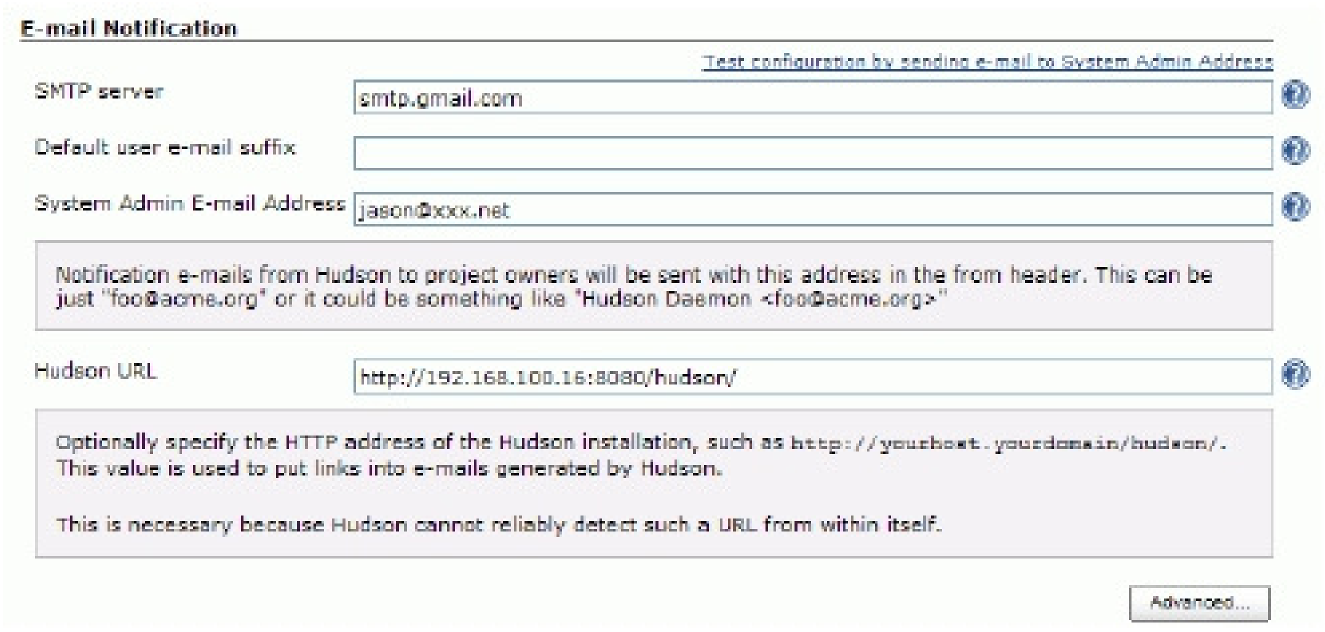

Ant

Ant installation	name	ANT_HOME
	ANT_HOME	D:\apache-ant-1.7.0

List of Ant installations on this system

4. Set up e-mail Notification for feedbacks

SMTP server, Hudson system Admin email address and installed Hudson’s URL



E-mail Notification

Test configuration by sending e-mail to System Admin Address

SMTP server	smtp.gmail.com
Default user e-mail suffix	
System Admin E-mail Address	jason@xxx.net

Notification e-mails from Hudson to project owners will be sent with this address in the from header. This can be just "foo@acme.org" or it could be something like "Hudson Daemon <foo@acme.org>"

Hudson URL	http://192.168.100.16:8080/hudson/

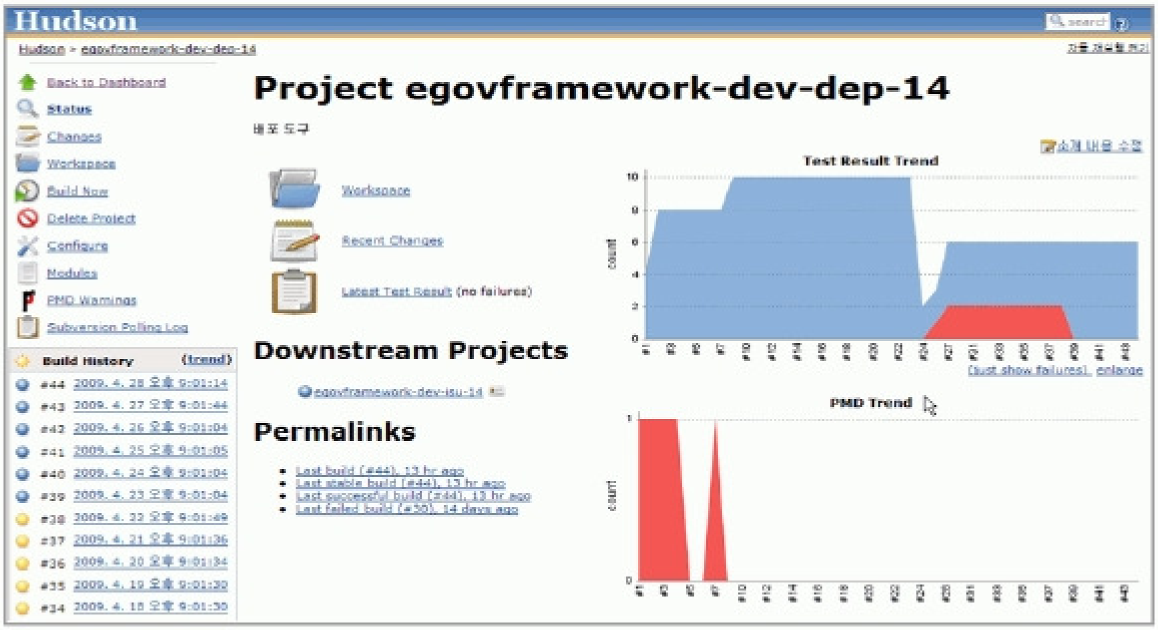
Optionally specify the HTTP address of the Hudson installation, such as `http://yourhost.yourdomain/hudson/`. This value is used to put links into e-mails generated by Hudson.

This is necessary because Hudson cannot reliably detect such a URL from within itself.

Manual

Hudson dashboard

Hudson provides the dashboard function that shows build results and test results.



Project egovframework-dev-dep-14

배포 도구

#44	2009. 4. 28 오후 9:01:14
#43	2009. 4. 27 오후 9:01:44
#42	2009. 4. 26 오후 9:01:04
#41	2009. 4. 25 오후 9:01:05
#40	2009. 4. 24 오후 9:01:04
#39	2009. 4. 23 오후 9:01:04
#38	2009. 4. 22 오후 9:01:49
#37	2009. 4. 21 오후 9:01:36
#36	2009. 4. 20 오후 9:01:34
#35	2009. 4. 19 오후 9:01:30
#34	2009. 4. 18 오후 9:01:30

Downstream Projects

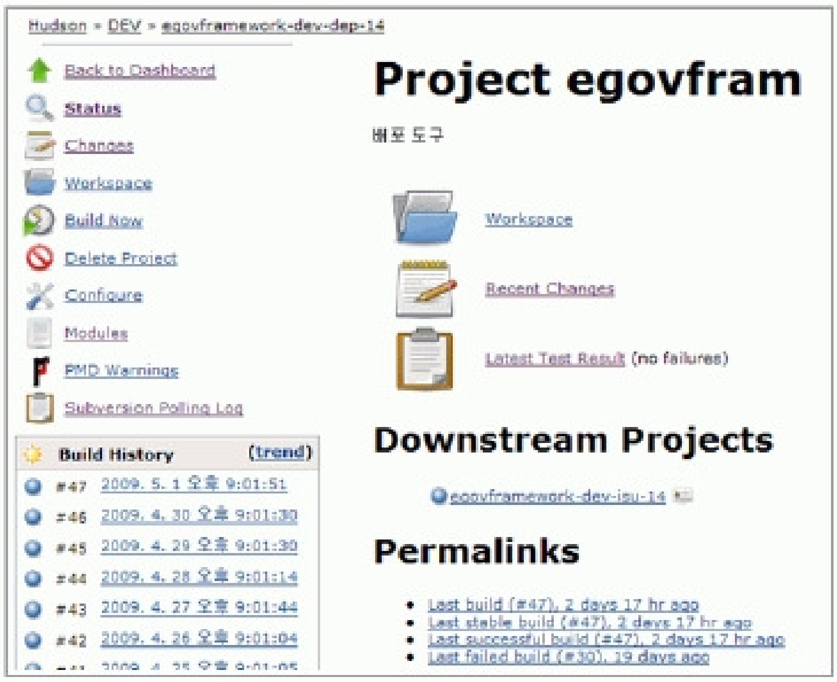
egovframework-dev-isu-14

Permalinks

- Last build (#44), 13 hr ago
- Last stable build (#44), 13 hr ago
- Last successful build (#44), 13 hr ago
- Last failed build (#30), 14 days ago

Hudson project menu

Hudson provides a menu to track the changes to source codes, manual build menu, and a menu that shows the status of project build.


Hudson » DEV » egovframework-dev-dep-14

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure
- Modules
- PMD Warnings
- Subversion Polling Log

Build History	(trend)
#47	2009. 5. 1 오후 9:01:51
#46	2009. 4. 30 오후 9:01:30
#45	2009. 4. 29 오후 9:01:30
#44	2009. 4. 28 오후 9:01:14
#43	2009. 4. 27 오후 9:01:44
#42	2009. 4. 26 오후 9:01:04

Project egovfram

배포 도구

Workspace

Recent Changes

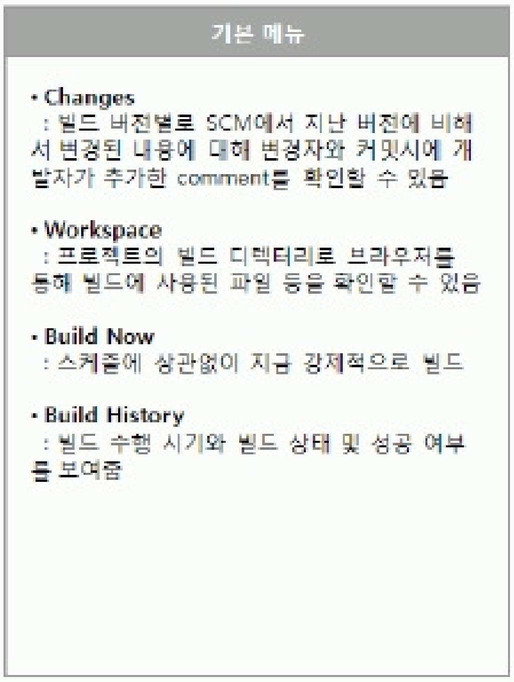
Latest Test Result (no failures)

Downstream Projects

egovframework-dev-isu-14

Permalinks

- Last build (#47), 2 days 17 hr ago
- Last stable build (#47), 2 days 17 hr ago
- Last successful build (#47), 2 days 17 hr ago
- Last failed build (#30), 19 days ago


기본 메뉴

• Changes
: 빌드 버전별로 SCM에서 지난 버전에 비해서 변경된 내용에 대해 변경자와 커밋시에 개발자가 추가한 comment를 확인할 수 있음

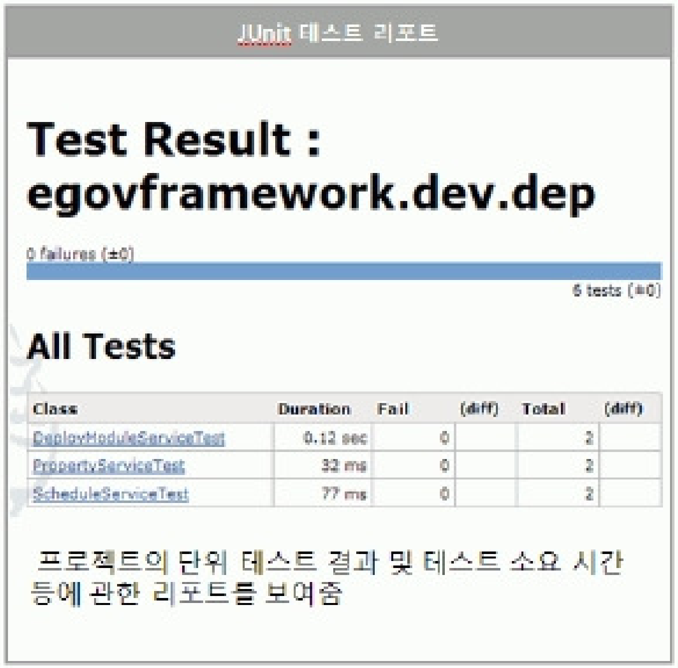
• Workspace
: 프로젝트의 빌드 디렉터리로 브라우저를 통해 빌드에 사용된 파일 등을 확인할 수 있음

• Build Now
: 스케줄에 상관없이 지금 강제적으로 빌드

• Build History
: 빌드 수행 시기와 빌드 상태 및 성공 여부를 보여줌

Hudson – interworking with JUnit test

Hudson, by default, provide a function that output JUnit test report, on the browser, which is created by using the build script and shows the test's accumulative graph on the home screen of the project.


JUnit 테스트 리포트

Test Result : egovframework.dev.dep

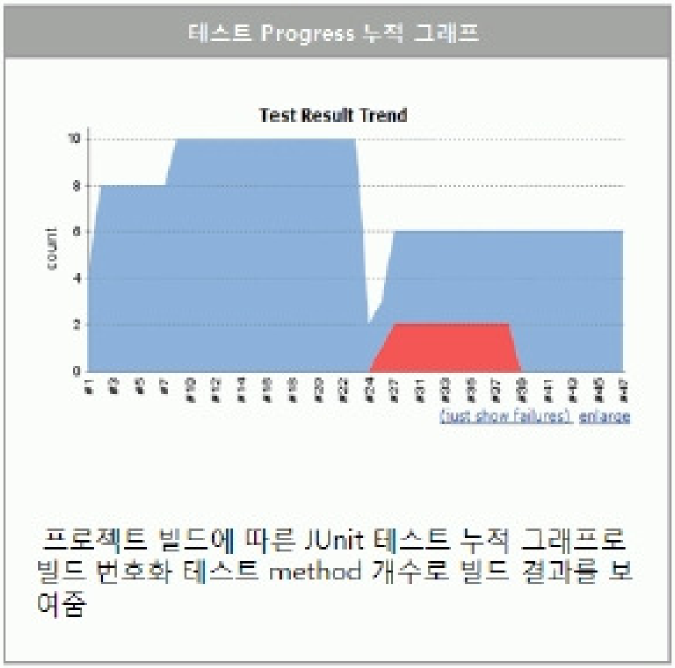
0 failures (±0)

6 tests (±0)

All Tests

Class	Duration	Fail	(diff)	Total	(diff)
DeployModuleServiceTest	0.12 sec	0		2	
PropertyServiceTest	32 ms	0		2	
ScheduleServiceTest	77 ms	0		2	

프로젝트의 단위 테스트 결과 및 테스트 소요 시간 등에 관한 리포트를 보여줌

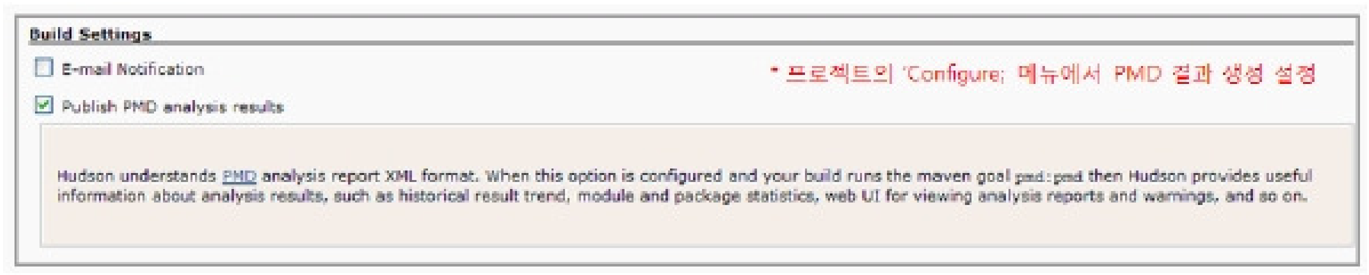

테스트 Progress 누적 그래프

프로젝트 빌드에 따른 JUnit 테스트 누적 그래프로 빌드 번호화 테스트 method 개수로 빌드 결과를 보여줌

Hudson – interworking with code inspection tool

Hudson provides PMD plug-in, a code inspection tool. It provides a function that interworks with PMD plug-in to show the PRM results of executing the build script on the Hudson screen

.



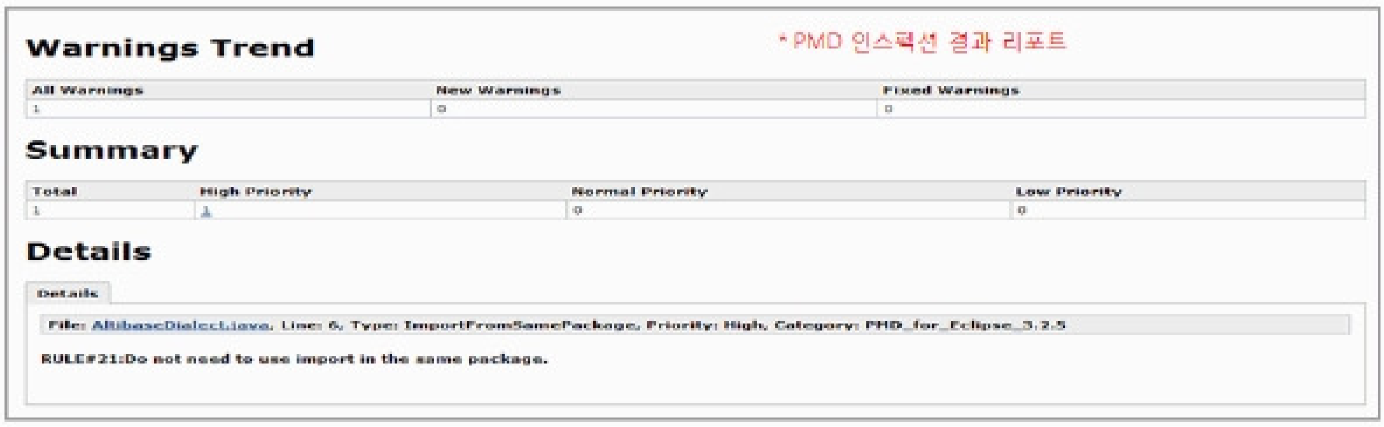
Build Settings

☐ E-mail Notification

☑ Publish PMD analysis results

Hudson understands PMD analysis report XML format. When this option is configured and your build runs the maven goal pmd:pmd then Hudson provides useful information about analysis results, such as historical result trend, module and package statistics, web UI for viewing analysis reports and warnings, and so on.

Select Configure in the project and set up PMD.



Warnings Trend

All Warnings	New Warnings	Fixed Warnings
1	0	0

Summary

Total	High Priority	Normal Priority	Low Priority
1	1	0	0

Details

File: AltibaseDialect.java, Line: 6, Type: ImportFromSamePackage, Priority: High, Category: PMD_for_Eclipse_3.2.5

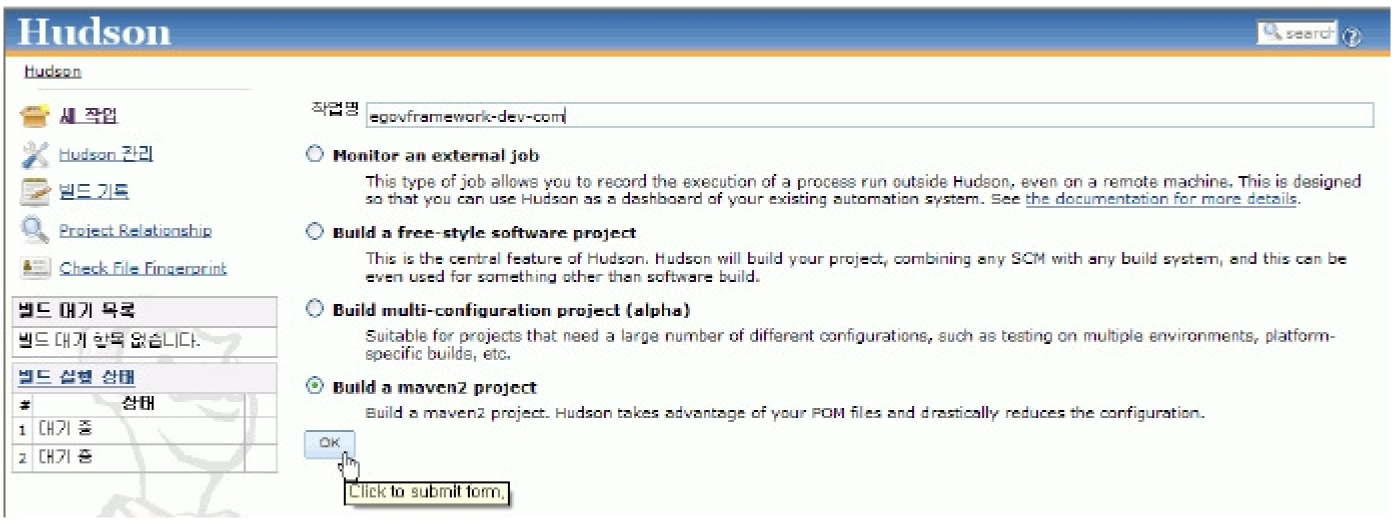
RULE#21:Do not need to use import in the same package.

Check RMD results on the Hudson screen

Register software project task in Hudson

1. Prerequisites for setup
   1. There should be an accessible source code repository.
   2. The repository should contain the source code of the task (project) to be registered.
   3. The project should contain the script for the automatic source build.

2. Task registration scenario
   1. Enter the task name to be registered.
   2. Select Build a maven2 project. (The e-government standard framework is based on Maven project.)
   3. Click OK button to create a Hudson project.



작업명: egovframework-dev-com

( ) Monitor an external job
This type of job allows you to record the execution of a process run outside Hudson, even on a remote machine. This is designed so that you can use Hudson as a dashboard of your existing automation system. See the documentation for more details.

( ) Build a free-style software project
This is the central feature of Hudson. Hudson will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

( ) Build multi-configuration project (alpha)
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

[✓ Build a maven2 project]
Build a maven2 project. Hudson takes advantage of your POM files and drastically reduces the configuration.

빌드 대기 목록

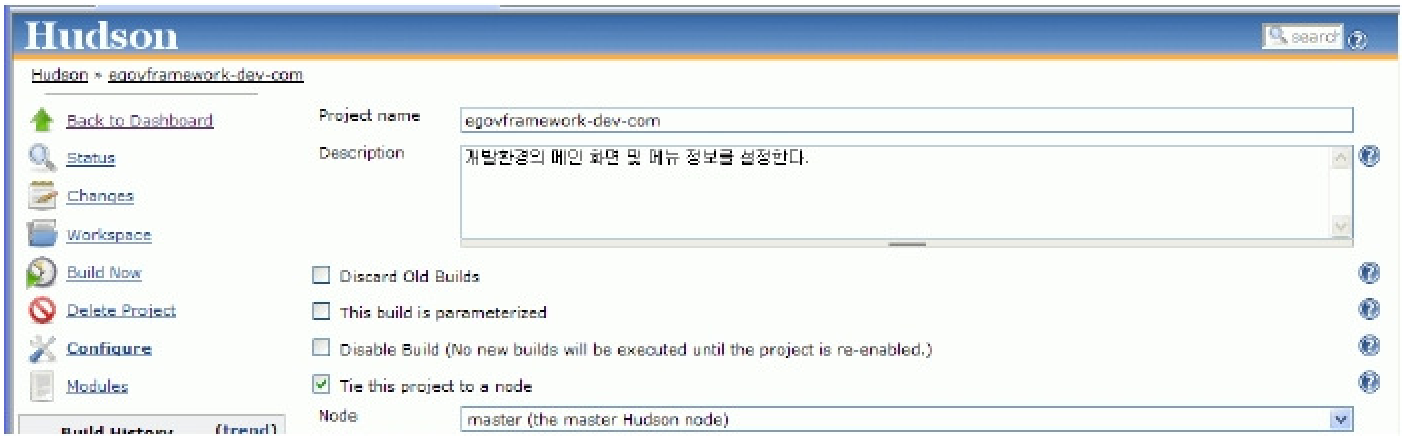
빌드 대기 항목 없습니다.

빌드 실행 상태

#	상태
1	대기 중
2	대기 중

3. Set up registered projects

3.1 Check the registered project and make a description on the project.



Project name: egovframework-dev-com

Description: 개발환경의 메인 화면 및 메뉴 정보를 설정한다.

☐ Discard Old Builds

☐ This build is parameterized

☐ Disable Build (No new builds will be executed until the project is re-enabled.)

☑ Tie this project to a node

Node: master (the master Hudson node)

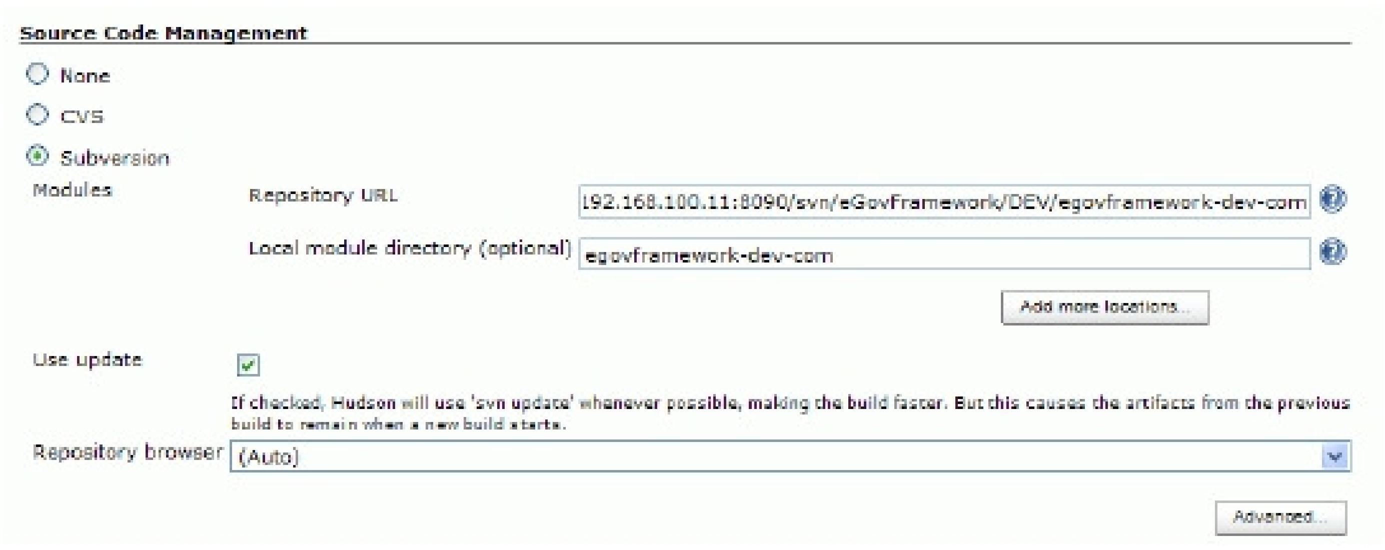
3.2 Select the SCM (repository) where the project code is stored.

- Select the SCM used by the project. - CVS or SVN
- The e-government standard framework provides SVM as the development environment tool.

3.3 Select the type of SCM and set the URL of the repository where the project to be registered in Hudson server is stored.

- The URL should include the project name level registered in the repository.
- Example) http://192.168.100.11:8090/svn/eGovFramework/DEV/egovframework-dev-com

[http://192.168.100.11:8090/svn/eGovFramework/DEV/egovframework-dev-com]



Source Code Management

○ None

○ CVS

◉ Subversion

Modules

Repository URL: 192.168.100.11:8090/svn/eGovFramework/DEV/egovframework-dev-com

Local module directory (optional): egovframework-dev-com

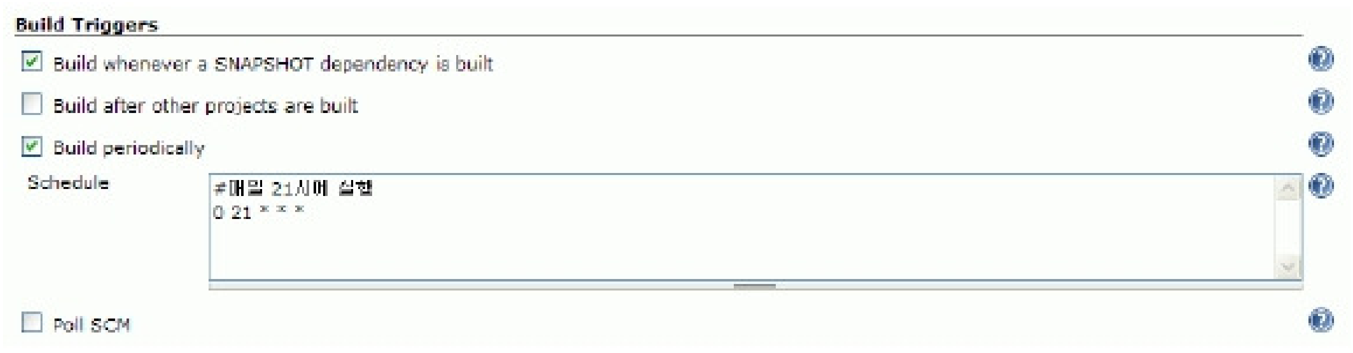
Use update ☑

If checked, Hudson will use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

Repository browser: (Auto)

3.4 Set up the automatic build cycle for periodical integration.

- Hudson provides cron expression based scheduling
- Example) Execute build at 21:00 everyday - 0 21 * * *


Build Triggers

[✓] Build whenever a SNAPSHOT dependency is built

[ ] Build after other projects are built

[✓] Build periodically

Schedule

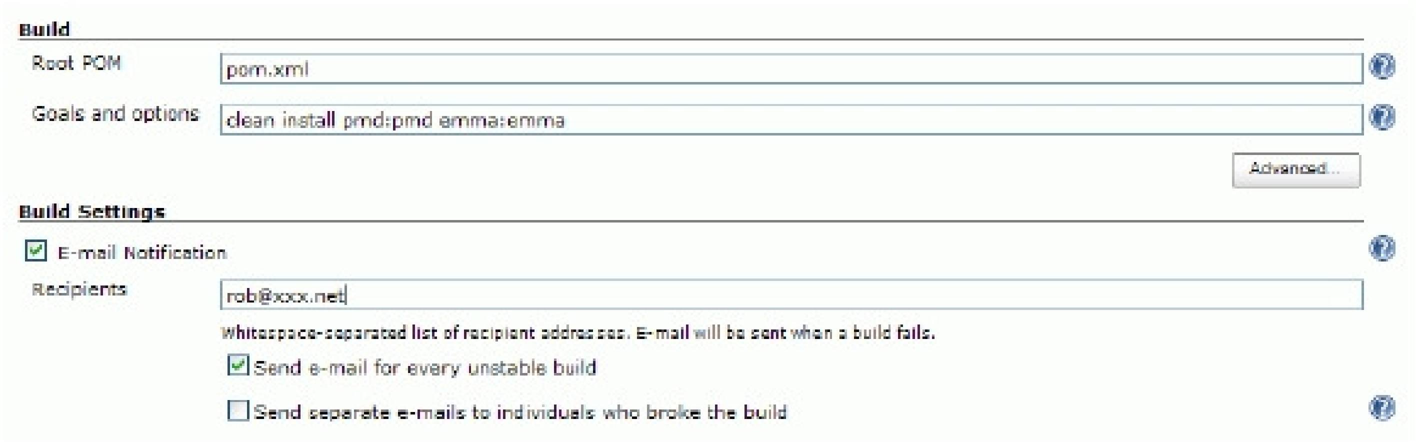
```
#매일 21시에 실행
0 21 * * *
```

[ ] Poll SCM

3.5 Set up the build.

- Root POM – Input pom.xml that contains Maven project information.
- Input Maven build life cycle phase and options.
- Example) clean install pmd:pmd emma:emma

3.6 Set up the build feedback mechanism.


Build

Root POM	pom.xml
Goals and options	clean install pmd:pmd emma:emma

Build Settings

[✓] E-mail Notification

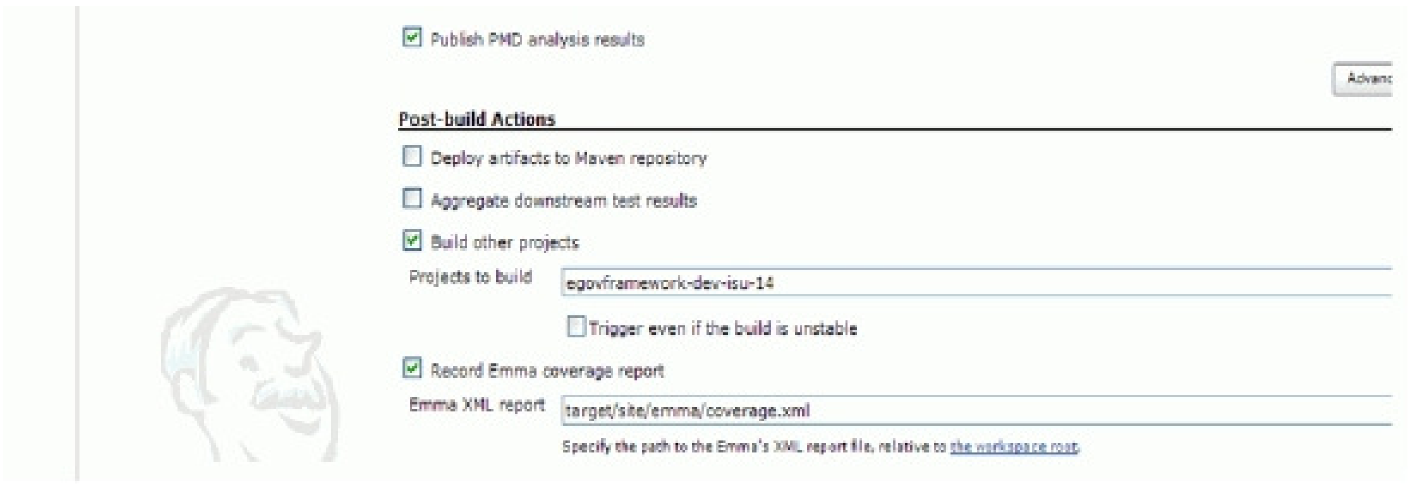
Recipients: rob@xxx.net

Whitespace-separated list of recipient addresses. E-mail will be sent when a build fails.

[✓] Send e-mail for every unstable build

[ ] Send separate e-mails to individuals who broke the build

3.7 Following the build, set up pmd (inspection tool) and emma coverage report creation.


[✓] Publish PMD analysis results

Post-build Actions

[ ] Deploy artifacts to Maven repository

[ ] Aggregate downstream test results

[✓] Build other projects

Projects to build: egovframework-dev-isu-14

[ ] Trigger even if the build is unstable

[✓] Record Emma coverage report

Emma XML report: target/site/emma/coverage.xml

Specify the path to the Emma's XML report file, relative to the workspace root.

References

Martin Fowler - Continuous Integration

[http://martinfowler.com/articles/continuousIntegration.html#AutomateDeployment]