

Bamboo에서 Java 빌드 자동화 구성

이 문서는 Bamboo에서 Java 언어에 대한 지속적인 통합(Continuous Integration:CI)를 하는 방법에 대한 가이드를 공유한다.

- 1. Project와 Plan 생성
- 2. 소스 저장소 연결
- 3. 빌드 트리거 선택
- 4. Task 구성
 - 소스코드 체크아웃 Task
 - 빌드 Task
- 5. 실행

1. Project와 Plan 생성

Bamboo Plan은 소스 코드 저장소, 빌드에서 실행되는 태스크 및 빌드를 트리거할 시점을 지정한다. 새로운 Plan을 생성하기 위해 다음을 수행한다.

1. Plan 생성 권한이 있는 사용자로 Bamboo 로그인
2. Create > Create a new plan 메뉴 선택

모든 Plan은 프로젝트에 포함된다. 프로젝트를 아직 생성하지 않았다면, Project > New Project를 선택하고 Project와 Plan을 위한 세부정보를 입력한다.

Bamboo는 Plan 명, Plan 키, Plan 설명을 입력할 수 있다.

2. 소스 저장소 연결

Bamboo는 소스 코드 저장소의 위치를 알아야하며 저장소에 접근 권한이 있어야 빌드를 실행할때 코드를 체크아웃 할 수 있다.

기본적으로 새로운 연결로 확인할 수 있는 저장소는 다음과 같다.

- Bitbucket Cloud
- Bitbucket Server / Stash
- CVS
- Git
- GitHub
- Mercurial
- Perforce
- Subversion

만약 Bitbucket과 Bamboo의 어플리케이션 링크가 되어 있다면, 권한이 있는 Bitbucket 저장소의 목록을 선택할 수 있다.

Link repository to new build plan

Repository host* ☐ Previously linked repository
☒ Link new repository

Bitbucket Server / Stash ▾

Display name* e.g. My repository (branch alpha)

Bitbucket Server / Stash details

Server Bitbucket ▾

Repository Curvc Homepage / curvc ▾ ?

A new public key for the selected repository will be stored as a repository access key.

Branch master ▾

Repository access ☒ Allow all users to reuse the configuration of this repository
☐ Only you are allowed to reuse the configuration of this repository

3. 빌드 트리거 선택

빌드 트리거는 빌드를 유발하는 주체를 말한다. 현재 Bamboo에서는 다음과 같은 5가지 종류의 빌드 트리거를 제공한다.

- Bitbucket Server repository triggered
Bitbucket 서버의 커밋이 발생하면 빌드를 수행 한다.
- Remote trigger
원격 저장소 IP를 입력하며 커밋이 발생하면 빌드를 수행 한다.
- Repository polling
주기적인 Polling을 통해 변경이 감지되면 빌드를 수행한다.
- Scheduled
특정 시간 스케줄에 맞추어서 변경을 확인하고 변경이 감지되면 빌드를 수행한다.
- Single daily build
특정 시간에 한번 빌드를 수행한다.

Select trigger



Bitbucket Server repository triggered
 Bitbucket Server triggers the build when changes are committed



Remote trigger
 Repository triggers the build when changes are committed



Repository polling
 Bamboo polls source repository and builds when new changes are found



Scheduled
 Run according to schedule



Single daily build
 Run once a day

Cancel

4. Task 구성

각 Plan은 적어도 하나의 Task가 필요하다. Task는 계획의 실제 업무를 수행한다.

소스코드 체크아웃 Task

새로 작성된 Plan에는 이전에 지정된 소스 저장소에서 소스코드를 가져오는 기본 소스코드 체크아웃 Task가 존재한다.

2 agents have the capabilities to run this job

Source Code Checkout
Checkout Default Repository

Final tasks Are always executed even if a previous task fails
Drag tasks here to make them final
Add task

Source Code Checkout configuration [How to use the Source Code Checkout task](#)

Task description
Checkout Default Repository
☐ Disable this task
You can check out one or more repositories with this Task. You can choose to check out the Plan's *Default Repository* or specify a *Specific Repository*. You can add additional repositories to this Plan via the [Plan configuration](#).
Repository*
CURVC Homepage
Default always points to Plans default repository.
Checkout Directory

(Optional) Specify an alternative sub-directory to which the code will be checked out.
☐ Force Clean Build
Removes the source directory and checks it out again prior to each build. This may significantly increase build times.
+ Add repository
Save Cancel

빌드 Task

코드를 컴파일하고 유닛 및 통합 테스트를 실행하도록 빌드 업무를 추가한다.

Source Code Checkout
Checkout Default Repository

Maven 3.x
Final tasks Are always executed even if a previous task fails
Drag tasks here to make them final
Add task

Maven 3.x configuration [How to use the Maven task](#)

Task description

☐ Disable this task
Executable
Maven 3 [Add new executable](#)
Goal*
clean test
The goal you want to execute. You can also define system properties such as -Djava.awt.Headless=true.
Build JDK*
JDK 1.8.0_101 [Add new JDK](#)
Which JDK do you need to use for the build? the JAVA_HOME will be added as an environment variable.
Environment variables

Extra environment variables. e.g. MAVEN_OPTS="-Xmx256m -Xms128m". You can add multiple parameters separated by a space.
Working subdirectory

Specify an alternative subdirectory as working directory for the task.

5. 실행

Plan을 enable하고 Create 버튼을 클릭한다.

Plan을 실행하였을때 다음을 확인할 수 있다.

- 저장소 연결
- 소스코드 체크아웃
- 코드 컴파일

그리고 Plan Summary에서 빌드가 성공했는지 여부를 확인할 수 있다.