

Software Verification

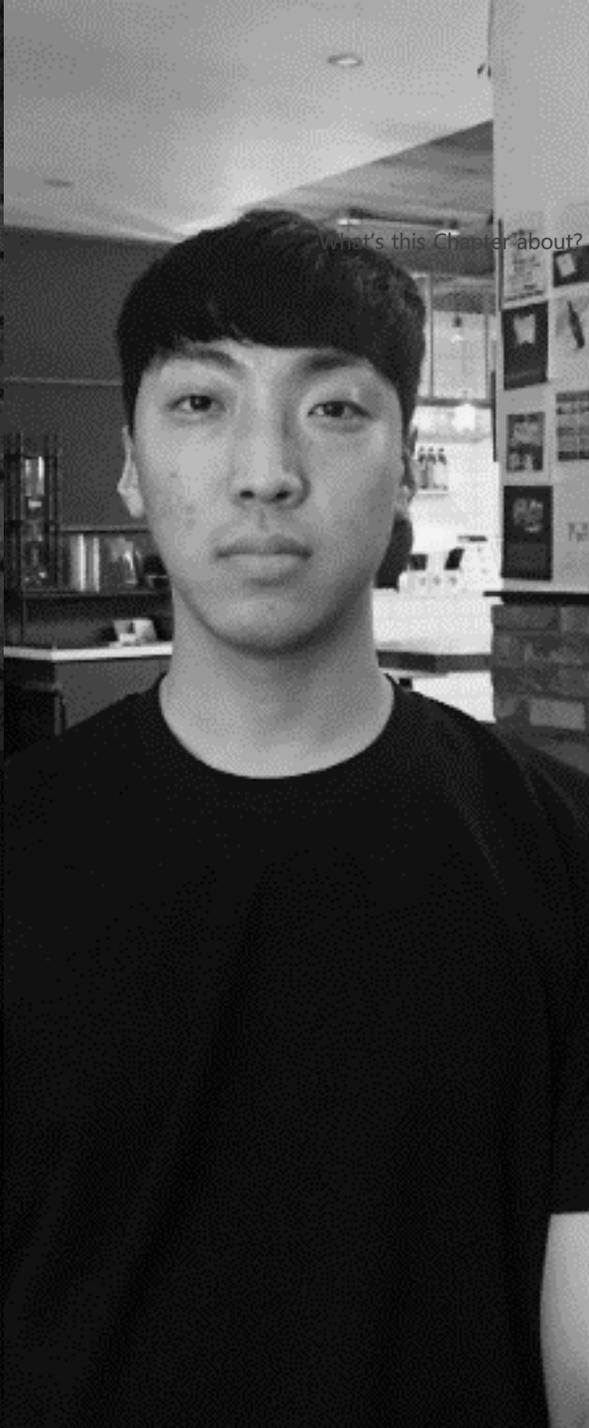
[T1]

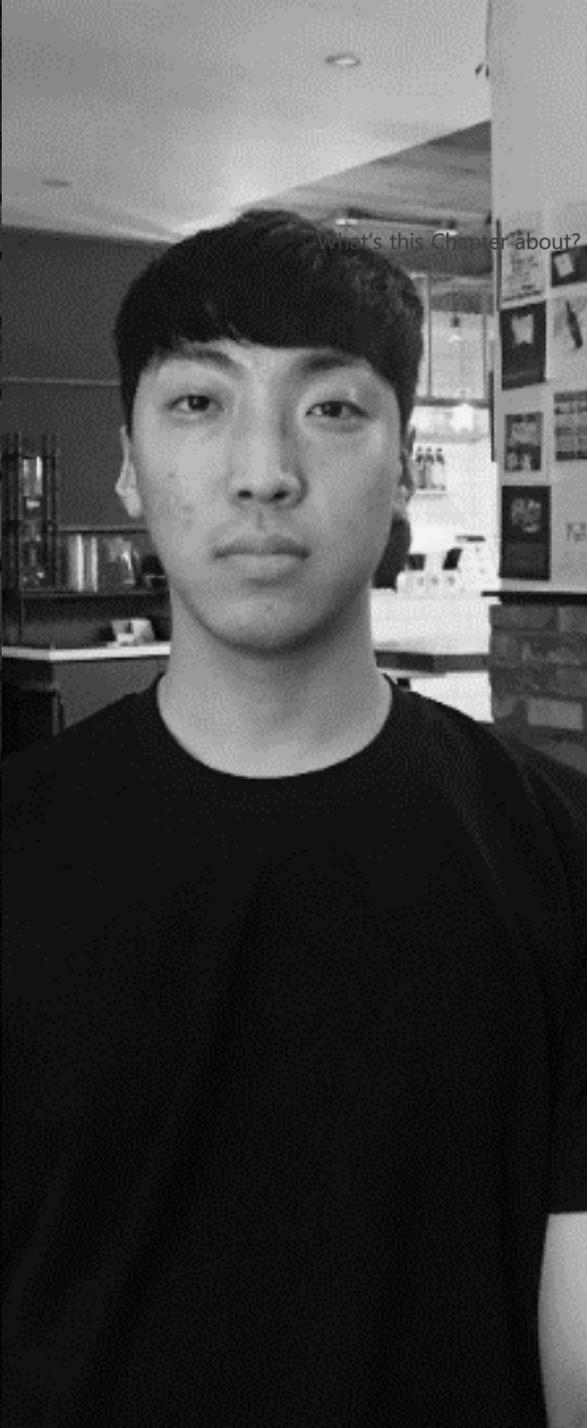
201411276 서수빈

201411291 이금락

201411292 이민환

201411313 장진서





CONTENTS

- CI
- IDE
- Configure management
- Build Environment
- Unit Test
- CI server

CONTENTS

- CI

- IDE

- Configure management

- Build Environment

- Unit Test

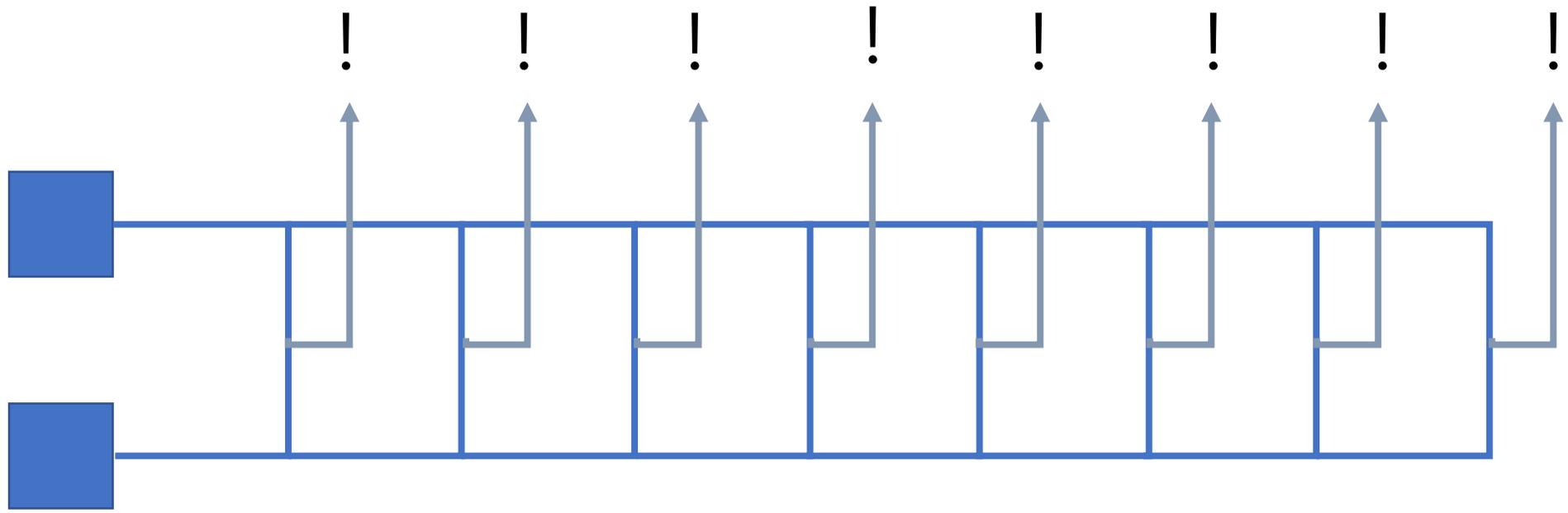
- CI server



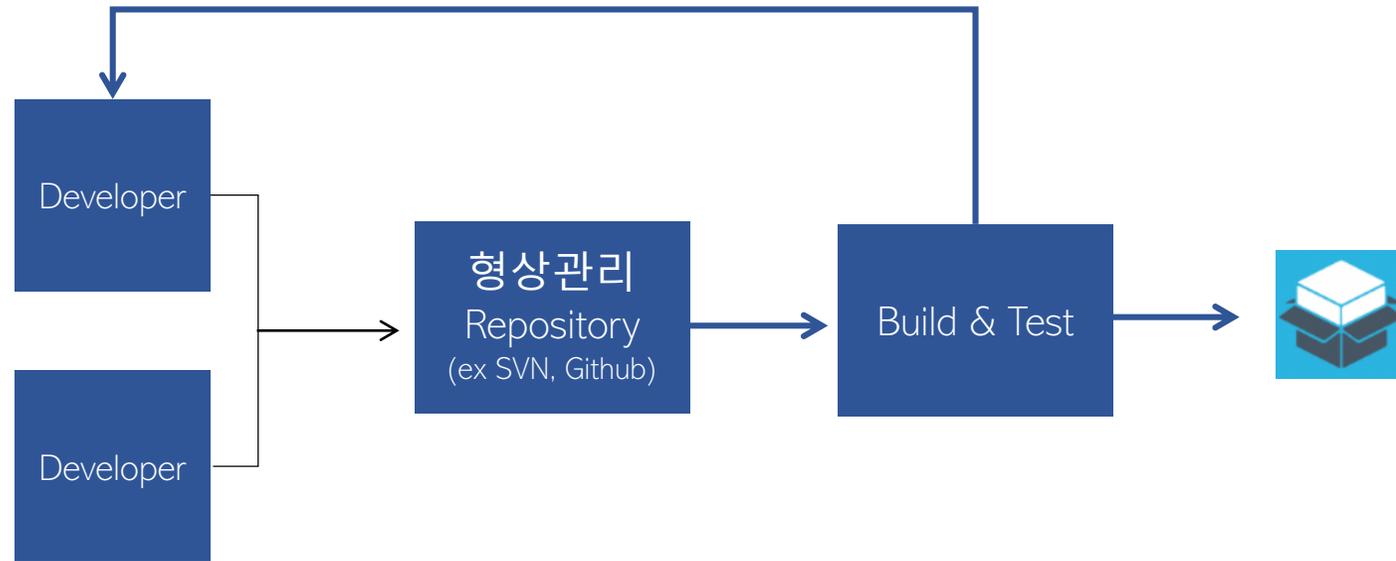
existing

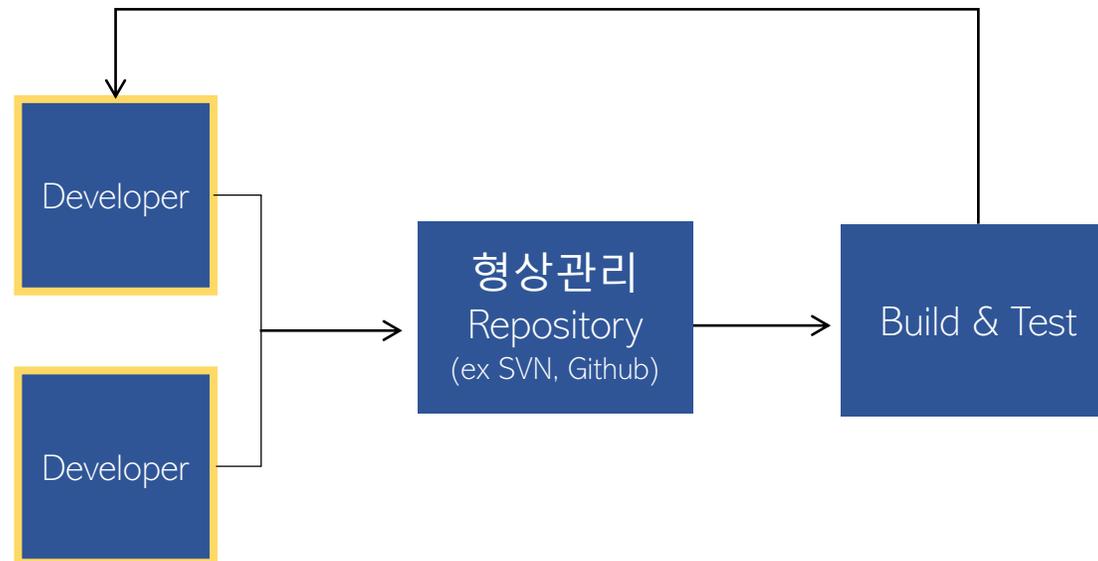


existing



CI





Pros of both IDEs

Eclipse

풍부한 최적화된 플러그인과 패키지 구성

-1000가지 이상의 쉽고 쓰기 편한 디버깅 툴 제공

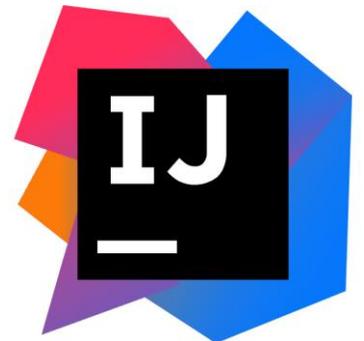


Intellij

쾌적한 코딩 환경 제공

-현재 코드의 context상 자리할 가능성이 큰 symbol 을 빠르게 검색

-코드 확인, checkstyle 및 findbugs 같은 built-in 기능 제공



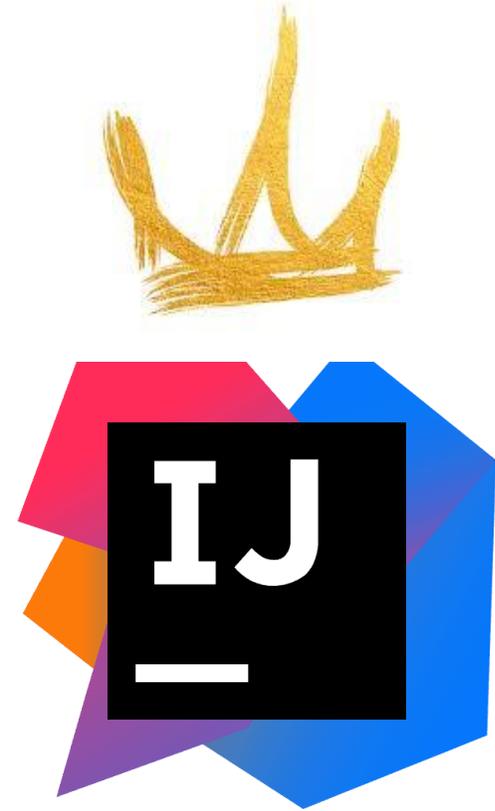
Why is it IntelliJ though?

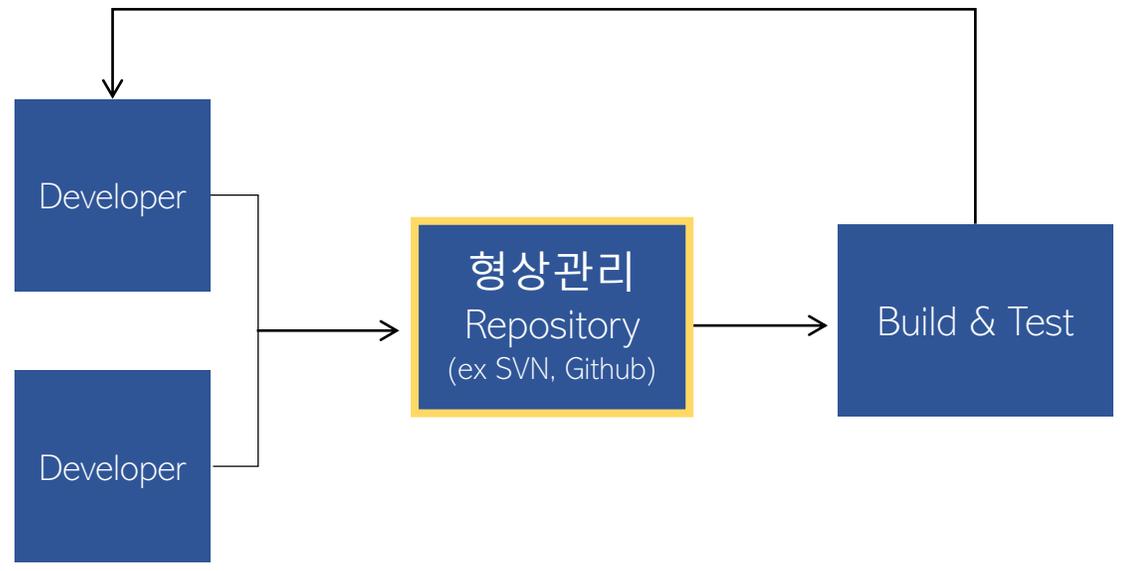
안정적인 plugin 구성

디버깅 정보 바로 확인

편리한 버전 관리

내장 빌드 및 테스트 툴



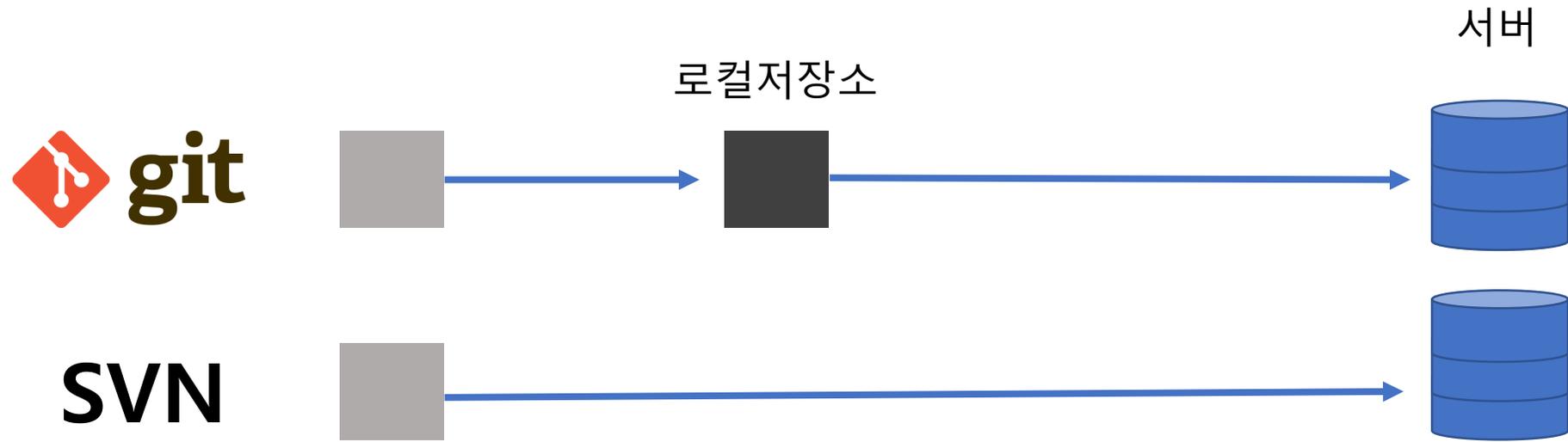




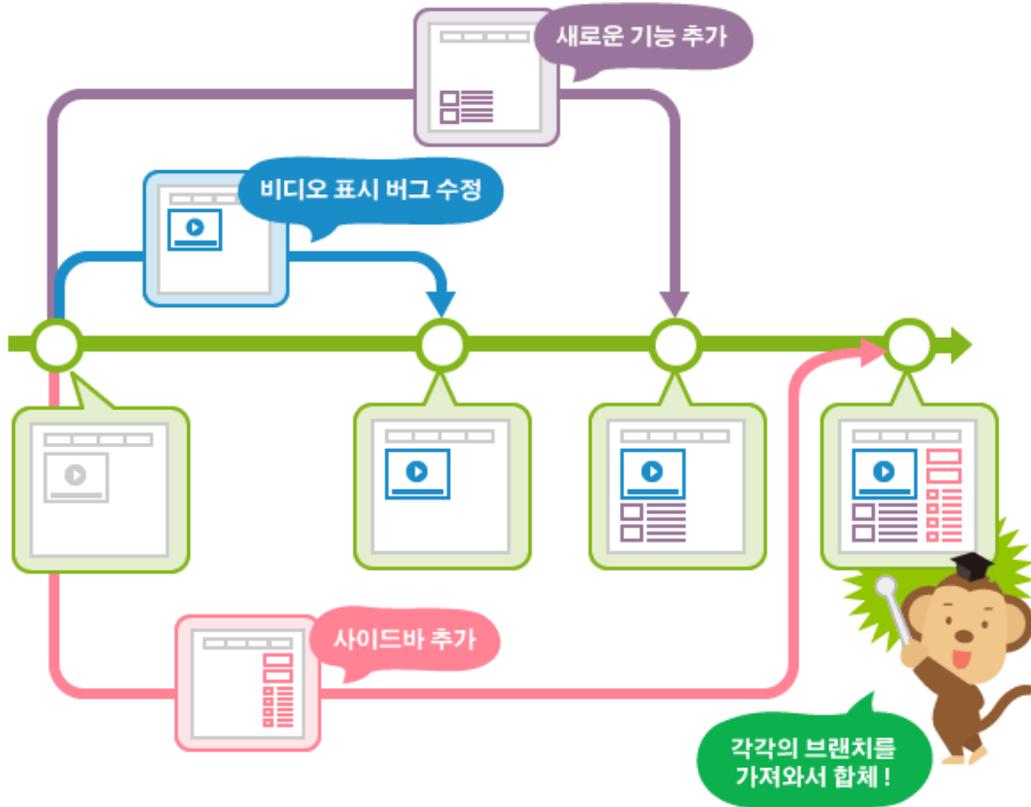
SVN

어느 파일에 저장할까요?		
파일 1		00:14:53 2019/02/15 20:27
파일 2		00:19:31 2019/02/15 20:49
파일 3		00:45:06 2019/02/19 20:08
파일 4		00:44:57 2019/02/15 21:30

Repository

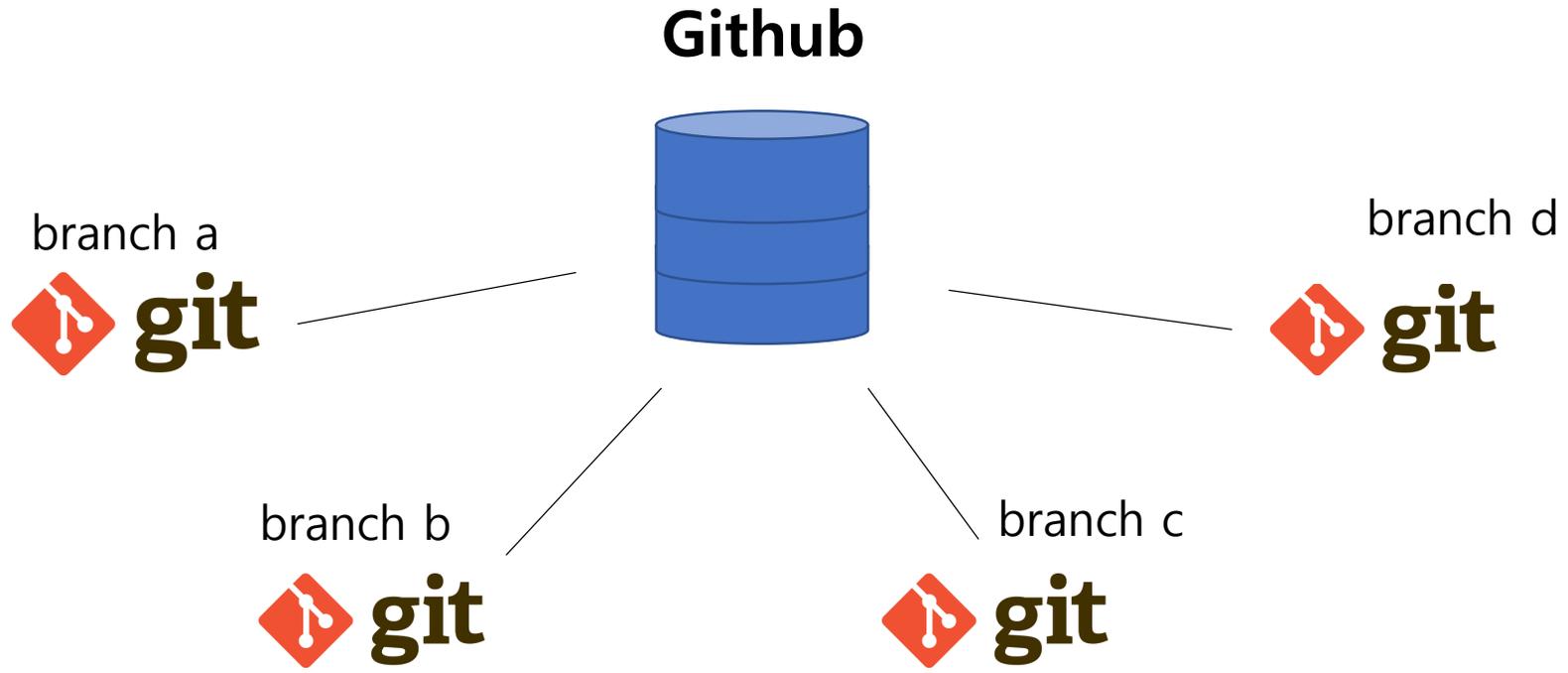


Repository



1. 로컬저장소와 원격저장소의 분리
-> 커밋에 부담이 없음, 빠른 처리
2. Branch 생성이 쉽다

Repository



Repository

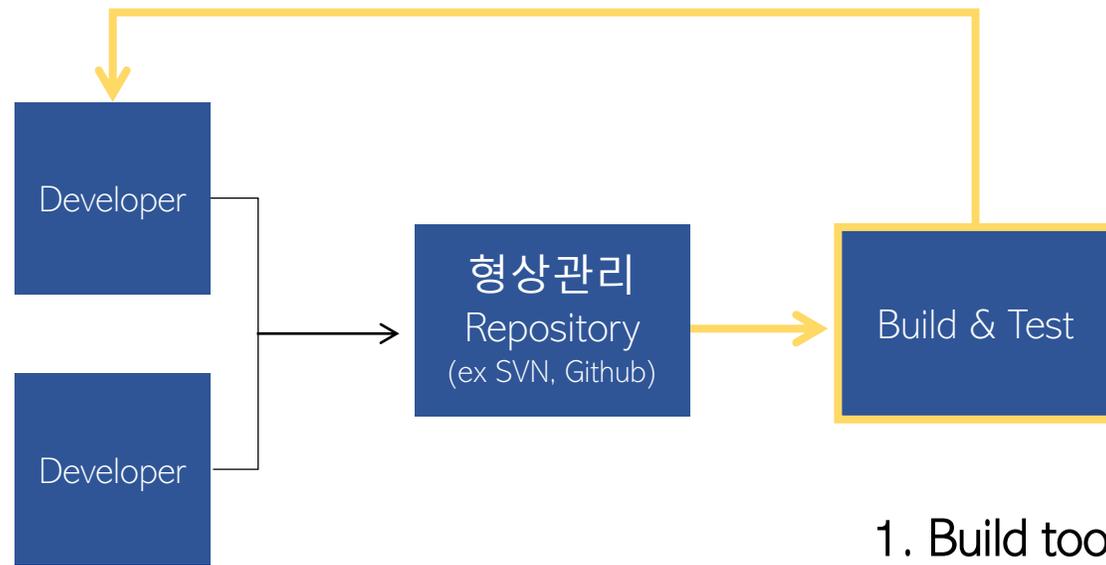


- 무료
- 비공개(최대 3명)



cloud source repositories

- 유료(최대 5명 무료)
- 비공개(자유)



1. Build tool
2. Unit test tool
3. CI server

Build Environment

Build?

Build Environment

Build?

compile

testing

inspection

deploy

Build?



자동화

Build Environment



maven



Build Environment



절차적

유연성

재사용 X

복잡

Build Environment

maven

표준화

상속

다양한 플러그인

XML

가독성

Build Environment



Groovy

가독성

API

유연성

Configuration
Injection

Build Environment

```

1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4   <groupId>info.solidsoft.rnd</groupId>
5   <artifactId>speck-10-groovy-24-gradle-maven</artifactId>
6   <version>0.0.1-SNAPSHOT</version>
7   <properties>
8     <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
9     <surefire.version>2.19.1</surefire.version>
10  </properties>
11  <build>
12    <plugins>
13      <plugin>
14        <groupId>org.codehaus.gmavenplus</groupId>
15        <artifactId>gmavenplus-plugin</artifactId>
16        <version>1.4</version>
17        <executions>
18          <execution>
19            <goals>
20              <goal>compile</goal>
21              <goal>testCompile</goal>
22            </goals>
23          </execution>
24        </executions>
25      </plugin>
26      <plugin>
27        <artifactId>maven-surefire-plugin</artifactId>
28        <version>${surefire.version}</version>
29        <configuration>
30          <includes>
31            <include>/**/*.java</include> <!-- Yes, .java extension -->
32            <include>/**/*.Test.java</include> <!-- Just in case having "normal" .Jnit tests -->
33          </includes>
34        </configuration>
35      </plugin>
36    </plugins>
37  </build>
38  <dependencies>
39    <dependency>
40      <groupId>org.codehaus.groovy</groupId>
41      <artifactId>groovy-all</artifactId>
42      <version>2.4.1</version>
43    </dependency>
44    <dependency>
45      <groupId>org.spockframework</groupId>
46      <artifactId>spock-core</artifactId>
47      <version>1.0-groovy-2.4</version>
48      <scope>test</scope>
49    </dependency>
50  </dependencies>
51 </project>

```

pom.xml

```

1 apply plugin: 'groovy'
2
3 group = "info.solidsoft.rnd"
4 version = "0.0.1-SNAPSHOT"
5
6 repositories {
7   mavenCentral()
8 }
9
10 dependencies {
11   compile "org.codehaus.groovy:groovy-all:2.4.1"
12   testCompile "org.spockframework:spock-core:1.0-groovy-2.4"
13 }
14
15

```

build.gradle

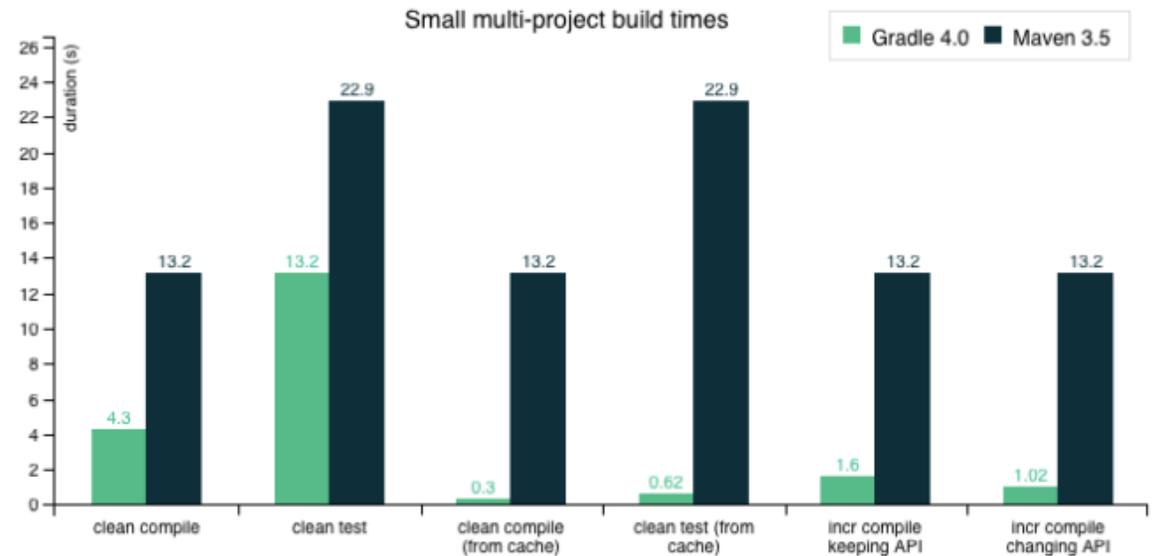
```

1 rootProject.name = 'speck-10-groovy-24-gradle-maven'
2

```

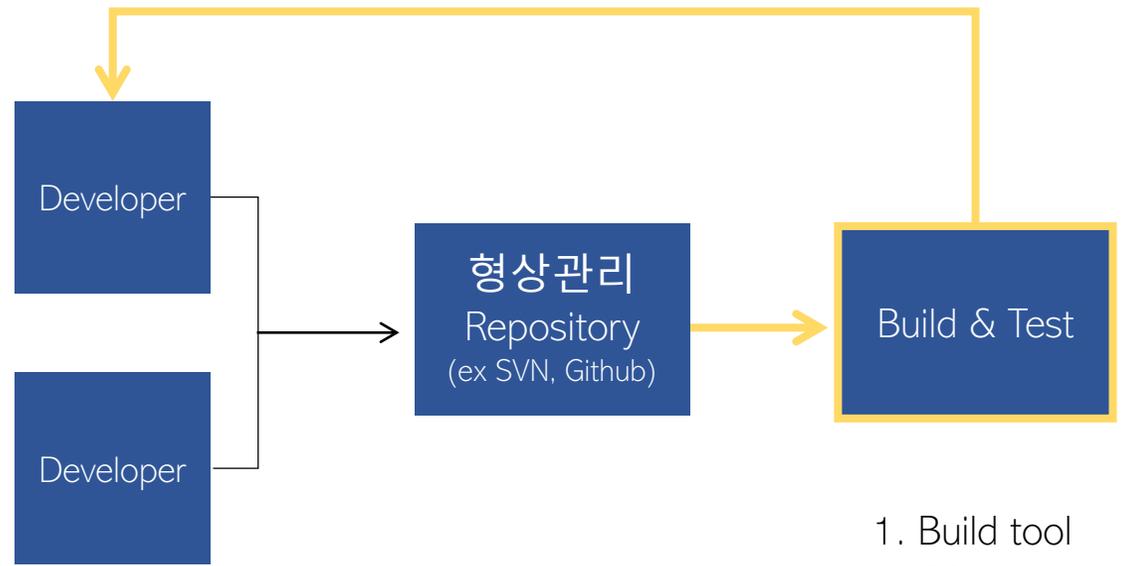
settings.xml



Results below were generated on a MacBook Pro 2018, 2.9 GHz Intel Core i9, 32 GB of RAM, SSD, OSX Mojave (10.14.2), using mean of 10 runs.

test project has 10 module. (50 source files, 50 tests)



- 1. Build tool
- 2. **Unit test tool**
- 3. CI server

UNIT TEST



Unit Test의 장점

이상적인 API 디자인 제공

기능의 문서화

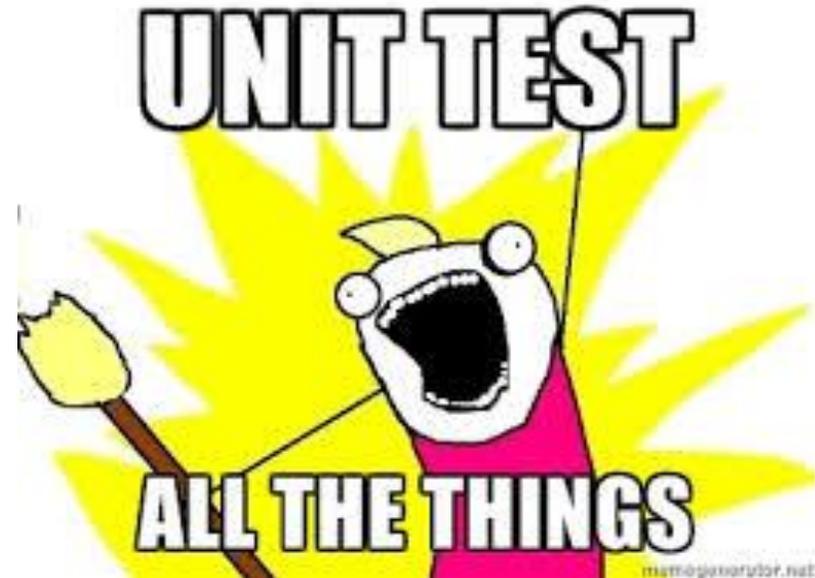
용이한 유지보수

적은 결합 코드

간단한 통합이 가능

품질 보증

지속적인 배포 가능



Unit Test – Junit



Unit Test – JUnit



문자/GUI 기반 -> 사용편리, 결과확인 용이(성공/실패)

예외 테스트 가능

ex) ArithmeticException -> negative unit test

Method에 annotation을 붙여준다

-> 가독성, 원하는 부분 테스트가능

Unit Test – Junit

@BeforeClass

테스트 실행하기 전에 초기화 및 자원 할당

@Before

테스트 실행하기 전에 실행되어야 하는 method 정의

@Test

유닛 테스트를 하기 위한 method 정의

@After

테스트 실행한 후에 실행되어야 하는 method 정의

@AfterClass

테스트 실행한 후에 초기화 및 자원 정리

주요 API

<code>assertEquals(x,y)</code>	x와 y가 같으면 테스트 통과
<code>assertTrue(x)</code>	x가 true이면 테스트 통과
<code>assertFalse(x)</code>	x가 false이면 테스트 통과
<code>assertSame(x,y)</code>	x와 y가 같은 객체를 참조시 통과
<code>assertNotSame(x,y)</code>	x와 y가 같은 객체를 참조하지 않으면 통과
<code>assertfail</code>	테스트를 실패 처리

Unit Test – Junit

```
1 public class JunitTest {
2     private String name;
3     private int age;
4
5     public JunitTest(String name) {
6         this.name = name;
7     }
8
9     public JunitTest(String name, int age) {
10        this.name = name;
11        this.age = age;
12    }
13
14    public String getName() {
15        return name;
16    }
17
18    public int getAge() {
19        return age;
20    }
21
22    public void setName(String name) {
23        this.name = name;
24    }
25
26    public void setAge(int age) {
27        this.age = age;
28    }
29 }
```

Java 예제 코드

Unit Test – Junit

```
1 import org.junit.*;
2
3 import static org.junit.Assert.assertEquals;
4
5 public class JunitTestTest {
6     public static JunitTest junitTest;
7
8     @BeforeClass
9     public static void makeInstance() throws Exception {
10         junitTest = new JunitTest( name: "홍길동", age: 20);
11     }
12
13     @Test
14     public void setAge() throws Exception {
15         junitTest.setAge(29);
16     }
17
18     @Test
19     public void getAge2() throws Exception{
20         assertEquals( expected: 29, junitTest.getAge())
21     }
22 }
23
```

JUnitTestTest

Tests passed: 2 of 2 tests – 3 ms

Process finished with exit code 0

테스트결과(성공)

Unit Test – Junit

```
1 import org.junit.*;
2
3 import static org.junit.Assert.assertEquals;
4
5 public class JunitTestTest {
6     public static JunitTest junitTest;
7
8     @BeforeClass
9     public static void makeInstance() {
10         junitTest = new JunitTest(1);
11     }
12
13     @Test
14     public void setAge() throws Exception {
15         junitTest.setAge(29);
16     }
17
18     @Test
19     public void getAge2() throws Exception {
20         assertEquals("expected: 20, junitTest.getAge()");
21     }
22 }
23
```

Comparison Failure

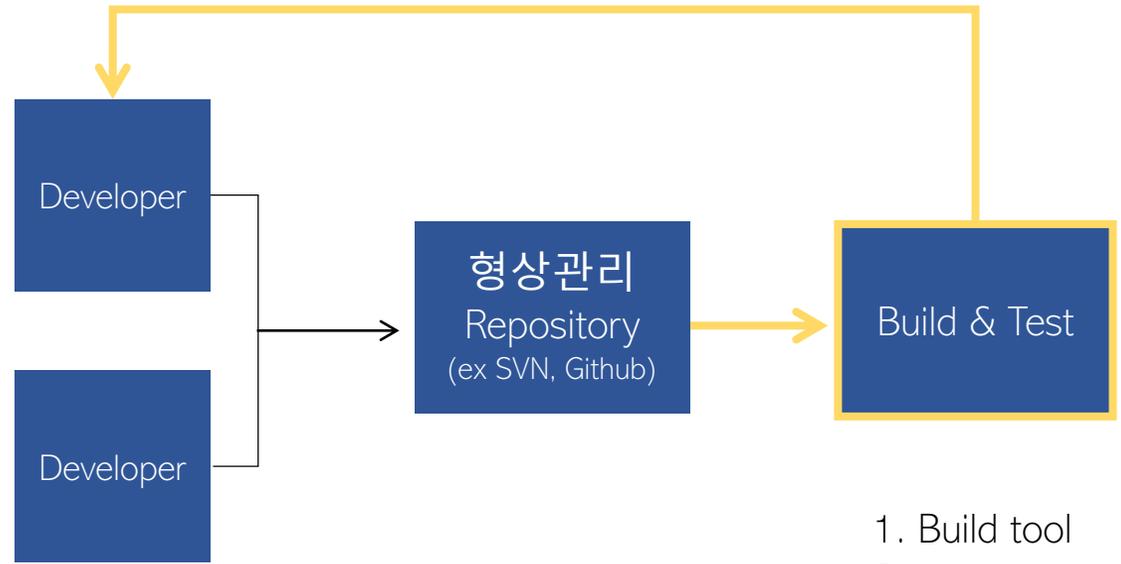
Side-by-side viewer | Do not ignore | Highlight words | 1 difference

Expected	Actual
20	29

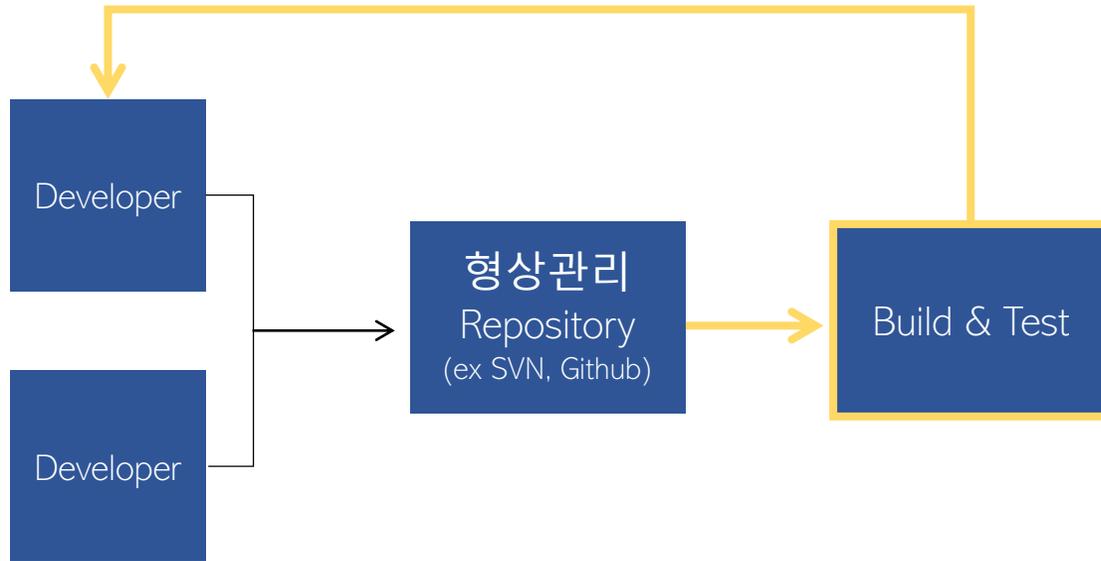
Tests failed: 1, passed: 1 of 2 tests – 14 ms

```
14 ms D:\Java\jdk-10.0.2\bin\java.exe -ea -Didea.test.cyclic.buffer.size=1048576 "-javaagent
0 ms
14 ms java.lang.AssertionError:
Expected :20
Actual   :29
<Click to see difference>
```

테스트결과(실패)



- 1. Build tool
- 2. Unit test tool
- 3. CI server

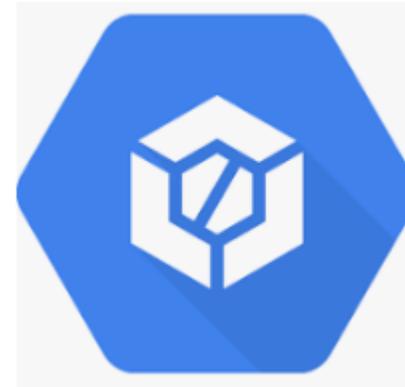


github에서 최신버전 받아서 merge..
빌드 버전올리기..
iOS 빌드..
안드로이드 빌드..
서버에 빌드 올리기..
새로운 빌드 공유 메일 보내기..
test 결과 통보..

자동화

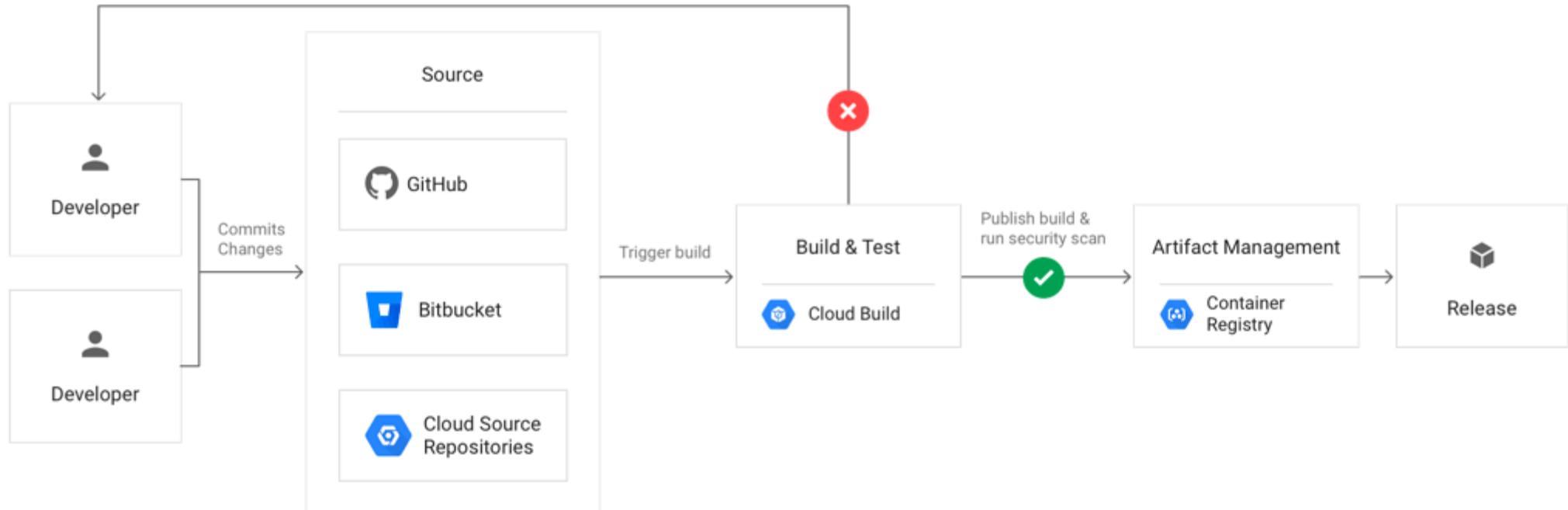


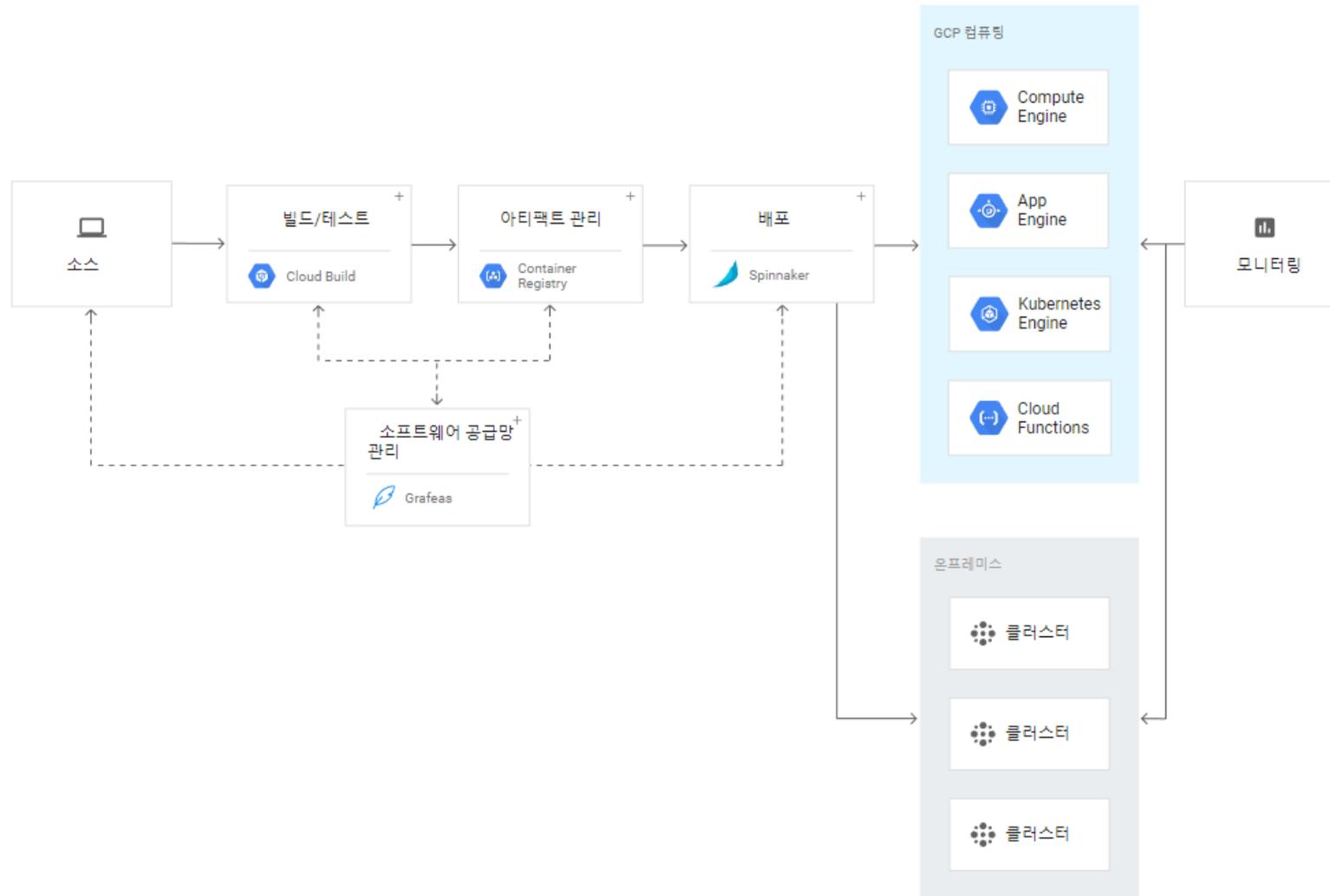
Jenkins

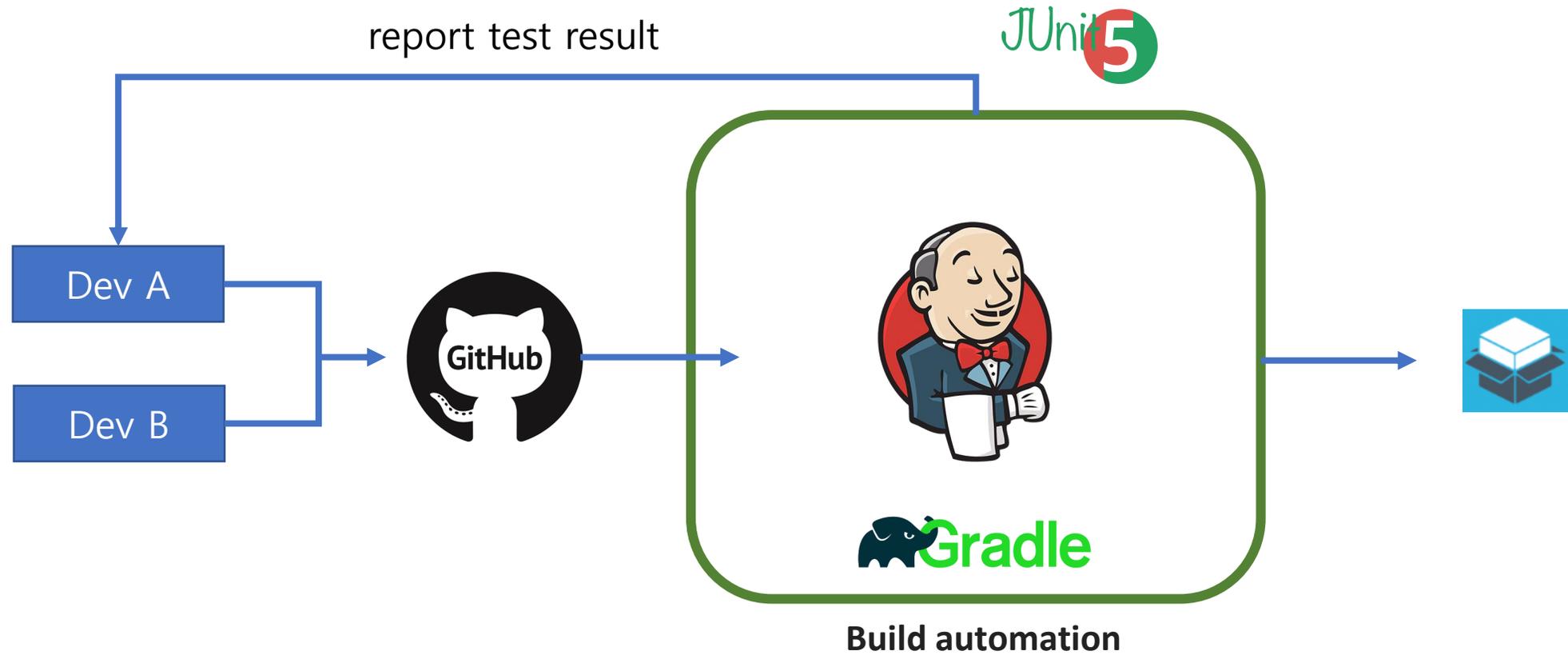


Google cloud platform
+ cloud build (CI)

CI server









Jenkins

- 무료
- 다양한 플러그인 제공
- 널리 사용되고있어 정보수집 용이

Hudson [?](#)

Hudson » [Dependency Graph](#) [ENABLE AUTO REFRESH](#)

- [New Job](#)
- [Manage Hudson](#)
- [People](#)
- [Build History](#)
- [Project Relationship](#)
- [Check File Fingerprint](#)
- [Dependency Graph](#)**

Build Queue
No builds in the queue.

Build Executor Status

#	Master
1	Idle
2	Idle
white (offline)	
1	Offline

Dependency Graph

```
graph TD; TP[test_parametrized_trigger] --> P1[Project1]; TP --> N[noop]; P1 --> OJ[on_join]; N --> OJ; TB[Test_Builds] --> J1[job1]; TB --> J2[job2]; O1[01_Build_Test_Project] --> O2[02_Unittest_Test_Project]; O2 --> O3[03_Integrationtest_Test_Project]; O3 --> O4[04_Create_Test_Reports]; O3 --> O5[05_Publish_Test_Artifacts]; O5 --> O6[06_Deploy_Project]; TW[test_windows]; TM[test_matrix];
```

Hudson

Hudson » GroovyTest ENABLE AUTO REFRESH

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

Project GroovyTest

[Workspace](#)

[Recent Changes](#)

Permalinks

- [Last build \(#6\), 29 seconds ago](#)
- [Last stable build \(#6\), 29 seconds ago](#)
- [Last successful build \(#6\), 29 seconds ago](#)
- [Last failed build \(#4\), 5 minutes 24 seconds ago](#)

Disk Usage Trend

[add description](#)

Disk Usage: Workspace 7KB, Builds 25KB

Build #	Build Usage (KB)	Workspace Usage (KB)
#1	4.5	4.0
#2	4.5	4.0
#3	4.5	4.0
#4	4.5	4.0
#5	4.5	4.0
#6	7.0	7.0

Build History [\(trend\)](#)

#6	Jun 30, 2008 2:20:49 PM	7KB
#4	Jun 30, 2008 2:15:54 PM	5KB
#3	Jun 30, 2008 12:33:48 PM	4KB
#2	Jun 30, 2008 12:33:24 PM	4KB
#1	Jun 30, 2008 12:32:55 PM	4KB

[for all](#) [for failures](#)

빌드 후 조치

- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report
- Record fingerprints of files to track usage
- Git Publisher
- E-mail Notification
- Editable Email Notification
- Set GitHub commit status (universal)
- Set build status on GitHub commit [deprecated]
- Delete workspace when build is done

빌드 후 조치 추가 ▾

Jenkins search

Jenkins > junit Example > #14 > Test Results > CustomerContact [ENABLE AUTO REFRESH](#)

[Back to Project](#) [Status](#) [Changes](#) [Console Output](#) [Edit Build Information](#) [History](#) [Test Result](#) [Previous Build](#)

Test Result : CustomerContact

1 failures

7 test Took 0 ms [add description](#)

All Failed Tests

Test Name	Duration	Age
>>> CustomerContact.Address.Zip	0 ms	1

All Tests

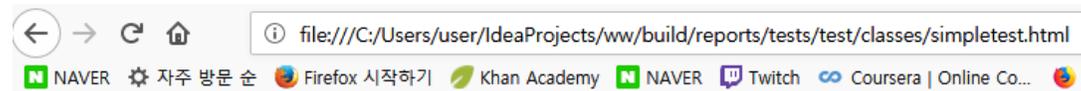
Class	Duration	Fail	(diff)	Skip	(diff)	Total	(diff)
Address	0 ms	1	+1	0		4	+4
Phone	0 ms	0		0		3	+3

Chapter 6

Jenkins

```
sdd.wv x simpletest.java x gradlew x
1 import org.junit.Test;
2 import static org.junit.Assert.*;
3
4 public class simpletest {
5     @Test
6     public void testAdd() {
7         assertEquals( expected: 42, Integer.sum( a: 19, b: 23));
8     }
9
10    @Test
11    public void testFail(){
12        assertEquals( unexpected: 43, Integer.sum( a: 19, b: 23));
13    }
14 }
15
```

```
C:\Users\user\IdeaProjects\ww>gradlew test
BUILD SUCCESSFUL in 2s
2 actionable tasks: 2 executed
```



Class simpletest

all > default-package > simpletest

2	0	0	0.001s
tests	failures	ignored	duration

100%
successful

Tests

Test	Duration	Result
testAdd	0.001s	passed
testFail	0s	passed

~~로컬 환경에서 build, unit test by Junit~~

~~github 계정생성~~

~~Jetkins 설치 및 환경구축 (gradle, junit, github 연동)~~

github 에 workspace 구축

github workspace에서 빌드

After unit test, report the result

Q & A

Thank you