Build automation

Cosa imparerai

- Cosa si intende per «build automation»
- Cos'è un «build automation tool»
- Quali sono i principali strumenti di «build automation»

Cosa si intende per «build automation»

Nell'ambito dell'informatica ed in particolare dello sviluppo del software, il termine «**build**» si riferisce al processo che trasforma file e altre risorse in un software utilizzabile (un eseguibile, un jar, un war, ecc…).

La build include diversi step, a seconda del livello di profondità del processo:

- validazione del codice sorgente
- compilazione dei file sorgente
- esecuzione di unit test sul codice
- raggruppamento (è il «*packaging*») dei file compilati in file in formato compresso (jar, war,..)
- esecuzione di test di integrazione tra i pacchetti realizzati
- copia dei file compressi su server o altro ambiente in cui verranno utilizzati o eseguiti (è il «*deploy*»)

paolopreite

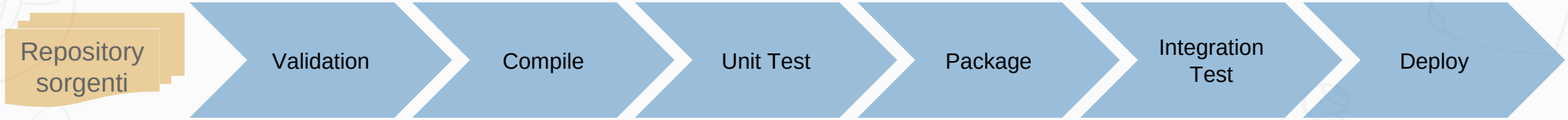
Cosa si intende per «build automation»

Quando si lavora in team o comunque si realizzano prodotti, è importante utilizzare strumenti per la gestione delle versioni dei file (si dice «**versioning**»).

Attualmente **il principale strumento di versioning dei sorgenti è GIT** e GITHUB è la principale piattaforma (https://github.com/). Un altro strumento molto utilizzato fino a qualche anno fa è SVN (e prima ancora CVS…).

A seconda dell'ambito in cui si sta lavorando (in proprio per scopi didattici, in team, …), il codice sorgente utilizzato nel processo di «build» cambia.

- **Scopo didattico**: il codice sorgente è sul mio computer (repository locale)
- **Sviluppo di un prodotto in proprio**: il codice sorgente è su un repository (GIT, SVN,…) o sul mio computer
- **Sviluppo di un prodotto in team**: il codice sorgente è su un repository (GIT, SVN,…)



NOTA: il termine «**repository**» indica l'ambiente in cui vengono gestiti i file sorgente.

paolopreite

Cosa si intende per «build automation»

Il processo di build può essere manuale o automatico.

- La **build manuale** richiede l'intervento dello sviluppatore che deve eseguire tutti i passi (validazione, compilazione, test, packaging, …) manualmente.
- La **build automatica** non richiede alcun intervento umano diretto al netto dell'avvio del processo e può essere eseguita in qualsiasi momento.

ATTENZIONE: **la build automatica non deve essere confusa con l'integrazione continua** (o Continous Integration). Quest'ultima consiste nell'esecuzione del processo di build automatica frequentemente (idealmente ogni volta che viene trovata una modifica del codice nel repository dei sorgenti).

paolopreite

Cosa si intende per «build automation»

Benefici della «**build automation**»:

- Migliora la qualità del software prodotto
- Velocizza il processo di compilazione e creazione dei pacchetti da rilasciare (eseguibili, jar, war,...)
- Elimina le attività ridondanti
- Riduci al minimo i rilasci errati
- Consente di gestire la cronologia delle build e dei rilasci, per esaminare eventuali problemi
- Risparmiare tempo e soldi (grazie ai precedenti vantaggi)

La build automation, è la condizione necessaria affinché si abbia l'integrazione continua.

paolopreite

Cos’è un «build automation tool»

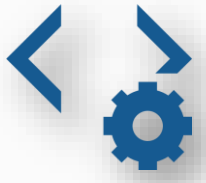
I «**build automation tool**» sono software che consentono di implementare il processo di build in maniera automatica.

In base al «build automation tool» utilizzato si crea un file chiamato «**artefatto**» (in formato XML, JSON, …) che definisce tutti gli step che dovranno essere eseguiti (recupero sorgenti dal repository, compilazione, unit test, packaging, …).

Attraverso appositi comandi che variano in base al «build automation tool», questo file verrà letto e verranno eseguite tutte le operazioni definite.

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.
http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
    <parent>
        <groupId>it.my</groupId>
        <artifactId>My</artifactId>
        <version>1.0.0-SNAPSHOT</version>
        <relativePath>../pom.xml</relativePath>
    </parent>
    <modelVersion>4.0.0</modelVersion>
    <artifactId>My-portlet-service</artifactId>
    <packaging>jar</packaging>
    <name>My Portlet Service</name>
    <build>
        <plugins>
            <plugin>
                <groupId>com.my.maven.plugins</groupId>
                <artifactId>my-maven-plugin</artifactId>
                <configuration>
                    <webappBaseDir>${basedir}/../My-portlet</webappBaseDir>
                </configuration>
            </plugin>
        </plugins>
    </build>
    <dependencies>
        <dependency>
            <groupId>com.my.portal</groupId>
            <artifactId>portal-service</artifactId>
            <scope>provided</scope>
        </dependency>
```

Esempio di «artefatto» Maven



paolopreite

Cos’è un «build automation tool»

Gli strumenti di build automation offrono una serie di funzionalità, tra cui:

- Numerosi plugin per l’esecuzione dei task (ad es. recupero sorgenti da GIT, unit test, …)
- Compatibilità con i principali IDE (Eclipse, Intellij, NetBeans, …)
- Analisi delle build
- Gestione delle dipendenze tra i vari progetti o da librerie esterne
- Strumenti di debug per analizzare nel dettaglio l’esecuzione della build
- …

paolopreite

Quali sono i principali strumenti di «build automation»

I principali strumenti di build automation sono:

- Make
- Rake
- Cmake
- MSBuild
- Ant
- Maven
- Gradle
- Grunt
- Gulp

paolopreite

Di cosa abbiamo parlato in questa lezione

- Cosa si intende per «build automation»
- Cos'è un «build automation tool»
- Quali sono i principali strumenti di «build automation»