Continuous Integration refers to a process that builds and test code on a frequent basis

Life Cycle

* Monitor
  + Source Code
    - GIT
    - SVN
    - CVS
    - GIT
    - Clear Case
* Build
  + Compile
    - Maven
    - Ant
* Run Test
  + Unit Testing
    - JUnit
    - NUnit
    - HtmlUnit
* Report
  + Notify
    - Email
    - IM
    - SMS
    - RSS

--

Maven

Jenkins

Sonar

--

**What is Maven?**

* Project management and comprehension tool
* Provides developers a complete build lifecycle framework
* Maven can manage a projects build reporting and documentation from a central price of integration

**What arePrerequisites?**

* Introduction to the java stack
* Basic Java and XML skill

**Maven Objective and Mindset**

Objective

-Making build process easy

-Providing uniform build system

-Providing quality project information

-Reduce the development effort to make builds

All build systems are essentially the same

Compile Source Code

Copy Resource

Compile and Run Tests

Package Project

Deploy Project

Cleanup

Describe the Project and configure the build

-You don’t script a build

-Maven has no concept of a condition

-Plugins are configured

--

Build Tools

Without Build Tools

//TODO

Ant

* Another Neat Tool
* Ant’s first official version was released in 2000.
* Multiple targets can be chained to combine single units of work into full workflows
* Ant doesn’t give any guidance on how to structure your project
* External libraries required by your project are usually checked into version control because there is no authenticated mechanism to pull them from a central location.

Maven

* Maven 2, released in October 2005
* It provided a standardized project and direct structure, as well as dependency management.
* Projects consisting of multiple modules could define their dependencies on each other.

Gradle

* Gradle build scripts are declarative, readable and certainly express their intension.
* Writing code in Groovy instead of XML, significantly cuts down the size of a build script and is far more readable

--

Difference with ANT and Maven

|  |  |
| --- | --- |
| ANT | Maven |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Problems without Maven

--

Advantages and Disadvantages of Maven

Advantages

Disadvantages

--

Maven Features

--

Maven POM

Project Object Model

Describes a project

Name and Version

Artifact Type

Source Code Locations

Dependencies

Plugins

Profiles

Uses XML

--

Maven Project Name (GAV)

Maven uniquely identifies a project using:

groupID:Arbitrary project grouping identifier (no spaces or colons)

Usually loosely based on java packages

artifactId: Arbitary name of the project(no spaces or colons)

version: Version of the project

Format{Major}.{Minor}.{Maintance}

Add ‘-SNAPSHOT’ to identify in development

Gav Syntax groupId:atrifactId:version

<>

<>

<>

<>

<>

--

**Packaging**

Build type identified using the packaging element

Tells Maven how to build the project

Pom, jar, war, ear, custom,

Default is jar

--

**Maven folder Structure**

Maven is maintained its own project structure

src:

src/main

src/main/java

src/main/resources

src/test/java

src/test/resources

target

Maven Build Lifecycle

* Compile
* Test
* Packages
* Integration-test
* Install
* Deploy

Maven Clean Lifecycle

* TODO

Maven Goals

* mvn install
* mvn clean
* mvn clean compile
* mvn compile install
* mvn test clean

**How to install Maven?**

* Download maven zip
* Extract maven zip into local machine
* Setting path and M2\_HOME

**How to Change the maven Default Repository folder?**

settings.xml

<localRepository>C:/Users/user/.m2/repository</localRepository>

**How to create maven project from CLI?**

mvnarchetype:generate

-DgroupId=domain.host.com

-DartifactId=my-application

-DarchetypeArtifactId=maven-archetype-quickstart

-DinteractiveMode=false

**How to Integrate Maven with Eclipse IDE?**

Window- Preferences-maven-installations-add-directory

How to create maven Project from Eclipse?

File->new-> project -> Maven ->Maven Project

How to Import Maven project in to Eclipse?

File->import-> search as maven-> select Existing Maven Projects-> next ->Browse->select the project->finish

**What is local Repository?**

C:/Users/user/.m2/repository

**What is Remote Repository?**

<https://mvnrepository.com/repos/central>

**How to add dependencies to the Maven Project from the Eclipse?**

FAQs

What is diff between local, remote and central repos?