Build Tools

Build tools are programs that automate the creation of executable

applications from source code. Building incorporates compiling, linking

and packaging the code into a usable or executable form.

Problems associated with old processes.

* Downloading dependencies.
* Compiling source code into binary code.
* Packaging that binary code.
* Running tests.
* Deployment to production systems.
* Reusing the user created project binary code in another project.
* Flexibility of different actions in different environments.

What is Maven?

Meaning accumulator of knowledge,

began as an attempt to simplify the build processes in the Jakarta Turbine project.

Used for Simplification and standardization of the build and build process.

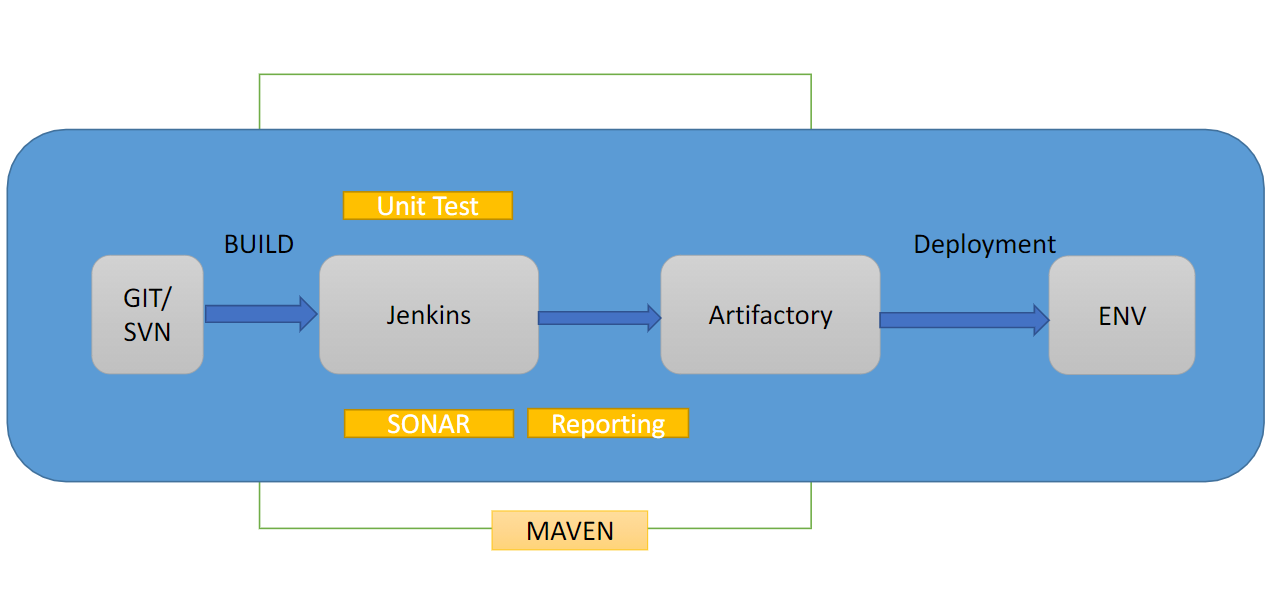
Phases in Maven

Each of these build lifecycles is defined by a different list of build phases, wherein a build phase represents a stage in the life cycle.

For example, the default lifecycle comprises of the following phases (for a complete list of the lifecycle phases, refer to the [Lifecycle Reference](https://maven.apache.org/guides/introduction/introduction-to-the-lifecycle.html#Lifecycle_Reference)):

* validate - validate the project is correct and all necessary information is available
* compile - compile the source code of the project
* test - test the compiled source code using a suitable unit testing framework. These tests should not require the code be packaged or deployed
* package - take the compiled code and package it in its distributable format, such as a JAR.
* verify - run any checks on results of integration tests to ensure quality criteria are met
* install - install the package into the local repository, for use as a dependency in other projects locally
* deploy - done in the build environment, copies the final package to the remote repository for sharing with other developers and projects.

Pipeline:



Problems associated with old processes.

* Downloading dependencies.

(maven local/remote/central repo accumulates all dependencies and can be used later in any new project implementation)

* Compiling source code into binary code.

(no need to compile manually :mvn compile will do the job)

* Packaging that binary code.

(no need to package manually :mvn package will do the job)

* Running tests.

(mvn test : automatically compiles and runs unit test cases on application code)

* Deployment to production systems.

(Jenkins CI/CD process uses maven to build the app)

* Reusing the user created project binary code in another project.

(mvn install and mvn deploy : places packaged products at maven repo.)

* Flexibility of different actions in different environments.

(Plugins & Profiles)

Stages of GIt:

Untracked

Unmodified

Modified nidhi abc = avgf git add nidhi.txt

Staged git commit