Automation Testing

Maven and ANT

A Day in the Life of a Full Stack Developer

John writes long scripts for testing and developing projects and downloads many dependencies for the same.

He performs manual upgrades for new software stacks for his project. Often, this creates confusion during the execution of a test.

John is looking for a solution to the above problems. Eventually, he discovered the Maven and ANT tools, which addressed all issues related to dependencies and maintained documentation for the entire life cycle of a project.

Let us go through the lesson to learn more about Maven and ANT.

Learning Objectives

By the end of this lesson, you will be able to:

- Define Maven
- Discuss ANT
- Integrate TestNG with Maven and ANT
- Integrate TestNG with Selenium

TECHNOLOGY

Introduction to Maven

Introduction to Maven

Maven is a popular open-source build tool developed by the Apache Group to build, publish, and deploy several projects at once for better project management.

The tool allows developers to build and document the project life cycle framework.

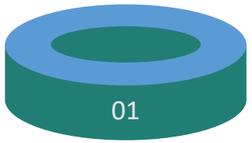
Build Tool

A build tool automates the process of creating applications from source code. It compiles and packages the code into an executable form.

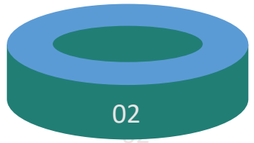
01 The compiled codes are packaged into JAR files.

02 Users must install the packaged code on a server or in a local and central repository.

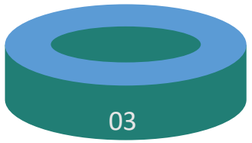
Objectives of Maven



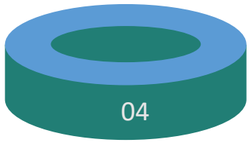
It provides a uniform and easy-to-use build system.



It manages project dependencies and maintains quality documentation.



The framework follows the Project Object Model and facilitates clear, transparent, and easy migration to new features.



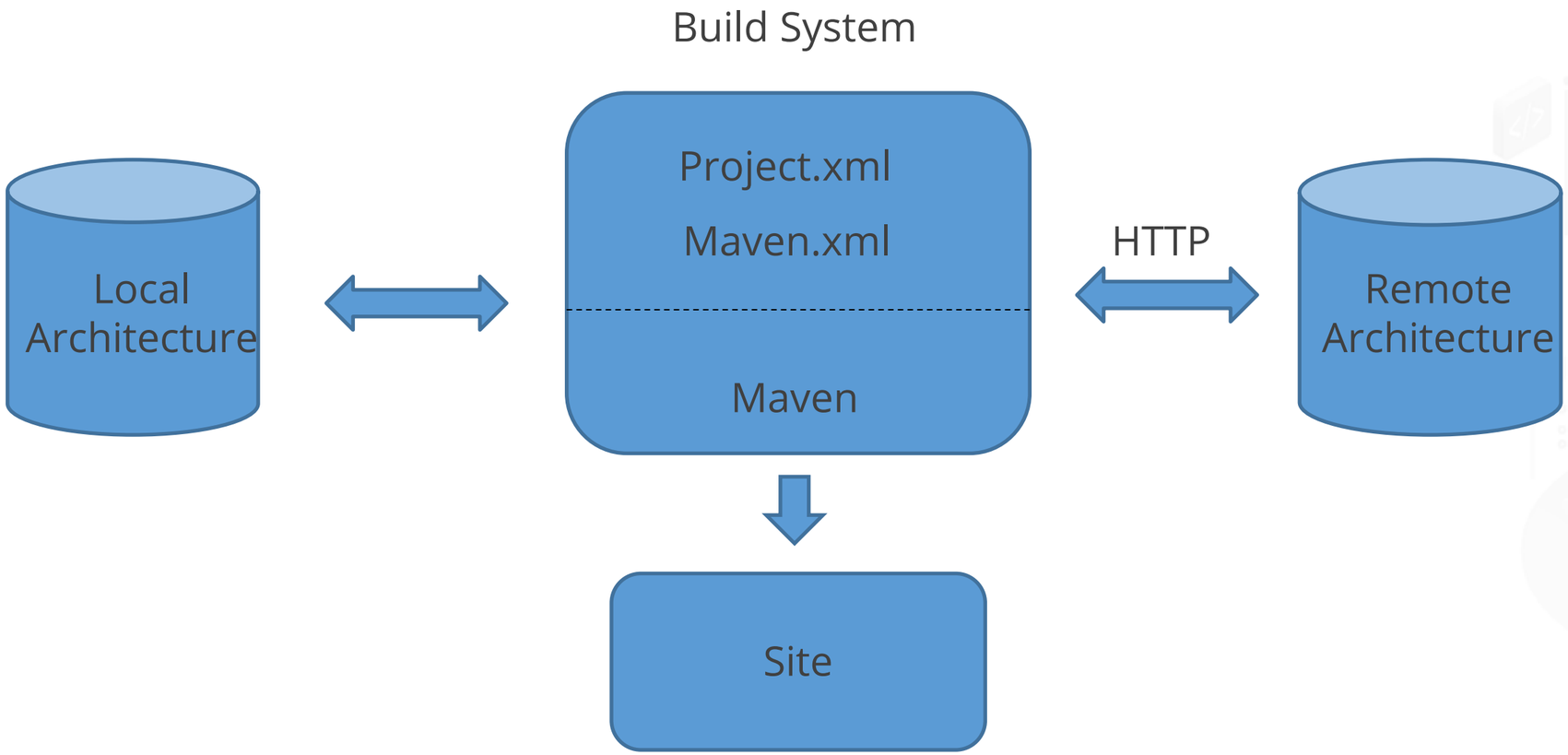
It automatically downloads dependency JAR files from Maven central repositories.

Maven vs. TestNG

Maven	TestNG
Maven is a project management tool.	TestNG is a framework for automated testing.
It is based on Project Object Model.	It is inspired by JUnit and NUnit.
It is used for Java projects.	It overcomes JUnit issues.
It is an automation tool.	It is an open-source automation framework.

Maven Architecture

Maven's architecture follows a POM (project object model) file to manage a project's build, documentation, and dependencies.



Maven Repository

Maven repositories contain packaged JAR files that consist of metadata. Maven uses this metadata to download dependencies.

Local Repository

Remote Repository

Centre Repository

Build Life Cycle

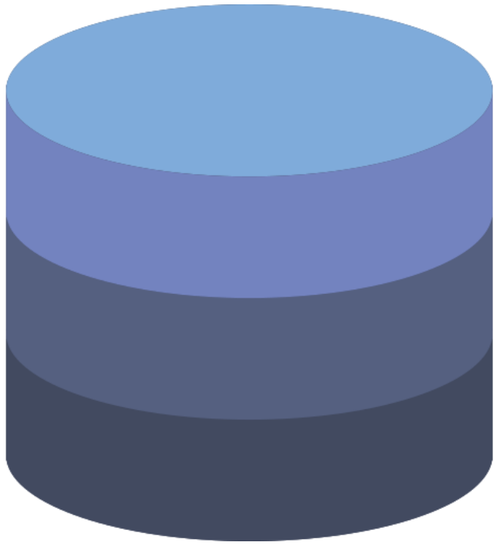
Each build phase consists of a series of goals.

Each goal is assigned a specific task.

All phases and their plugins are compiled during execution.

Build Profiles

To build a project, a set of configuration values is required, and build profiles are added to the POM files to create builds for different environments.



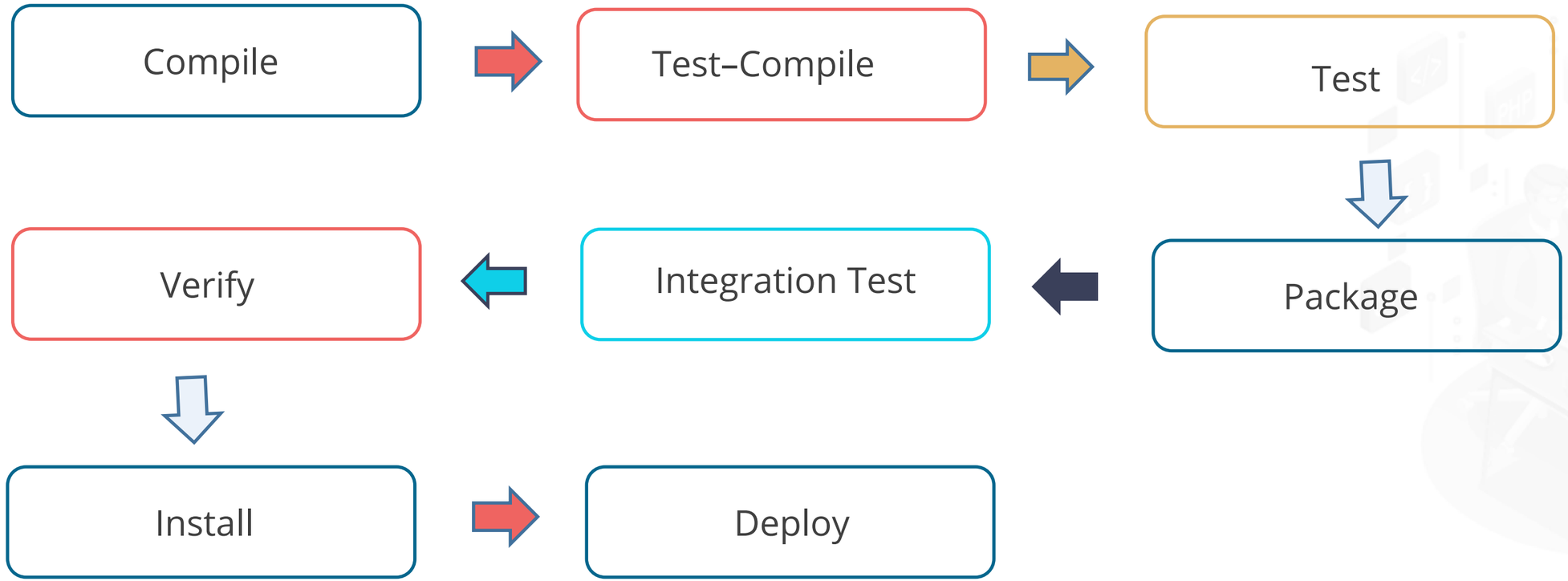
Build Plugins

The Maven plugin is a collection of goals that may or may not be in the same phase, and each plugin performs a specific task.

Maven has its own standard plugins.

Maven Build Life Cycle

A Maven Build Life Cycle is a sequence of steps that must be followed to build a project and execute goals.



Maven With TestNG

Maven projects execute test scripts or suites, and their dependencies are managed in POM.xml. However, the user cannot select a specific test suite from a list.

TestNG does not allow users to manage their dependencies, but the user can select and execute certain test scripts or suites.

Integration of Maven With TestNG

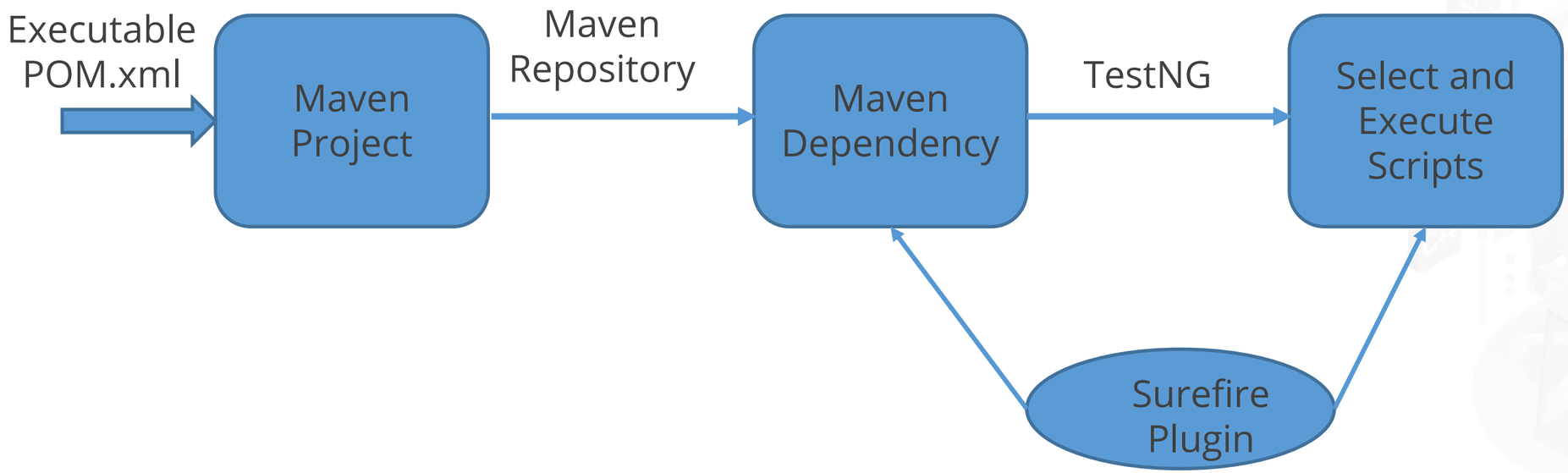
The following methods can be used to integrate Maven:

01 Users can integrate applications using Maven architecture.

02 They can integrate Maven and TestNG with Maven Surefire Plugin.

Configuration Using Maven Surefire Plugin

Maven Surefire plugin executes unit tests and generates HTML reports for applications.



Execute Test Script Using Maven Surefire Plugin

Maven Surefire plugin executes unit tests and generates HTML reports for applications.

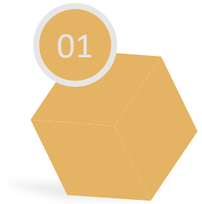
01 Select Script

02 Execute using the **mvn** command

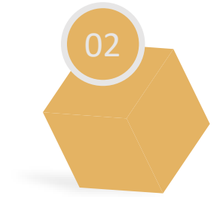
03 Check the results

Benefits of Maven

Maven can be used to achieve the following benefits:



It is a powerful build management tool that helps user to manage their Selenium projects easily.



It plays an important role as a project management tool that facilitates the creation of reports and the management of dependencies.

TECHNOLOGY

Introduction to ANT

Introduction to ANT

Another Neat Tool (ANT) is a Java library and command-line tool used to drive processes described in build files as targets and extension points.

Using ANT, the user can compile, assemble, test, and run Java applications.

Importance of ANT

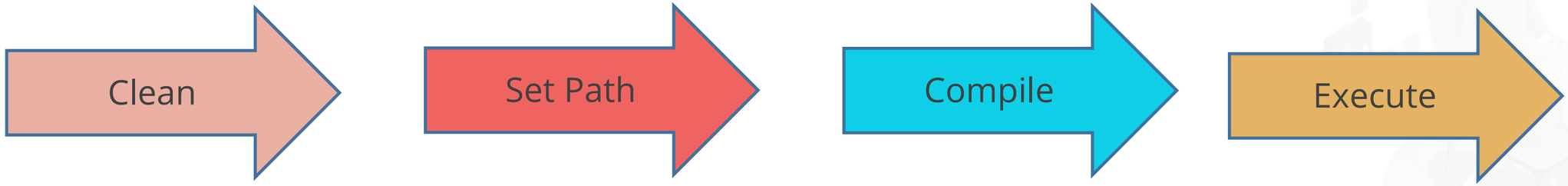
The ANT Build tool manages third-party APIs and classpaths, compiles source code, executes source code, creates reports, and deploys code.

Build.xml

The ANT Build.xml file allows users to complete all tasks in one go.

Build Tool

Build.xml is an ANT configuration file that saves, executes, and automates all processes.



Configure TestNG With ANT

To run TestNG with ANT, the user must follow the given steps:

01 Download the Apache ANT package

02 Set up the ANT environment

03 Download the TestNG Archive

Configure TestNG With ANT

04

Develop a project structure

05

Create the ANT build.xml file

06

Execute the command

Advantages of ANT

ANT can be used to achieve the following benefits:

01

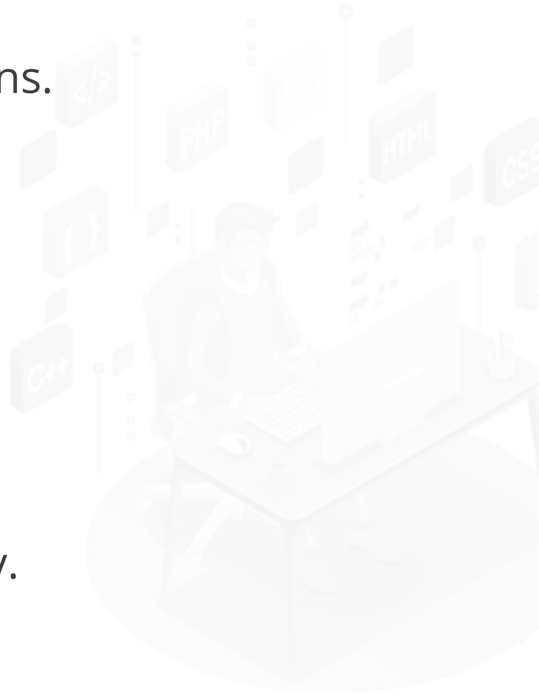
It provides end-to-end deployment and delivery of applications.

02

The user can keep the code clean by separating logic and configuration.

03

It configures third-party API dependencies and JAR files easily.



Integration of Selenium

TestNG with Selenium

To generate test reports with Selenium, users need an external framework, such as TestNG, that will simplify functional testing, regression testing, end-to-end testing, and even multi-site testing.

TestNG can generate reports based on Selenium test scripts.

Configure TestNG With Selenium

To run TestNG with Selenium, the user must follow the given steps:

01

Add the **Selenium API JAR** files to TestNG project

02

Set up the path environment

03

Execute and process the results

Key Takeaways

- Maven can create, publish, and deploy several projects at once for better project management.
- POM (project object model) files manage a project's build, documentation, and dependencies in Maven.
- ANT build tool manages APIs, compiles and executes source code, generates reports, and deploys applications.
- Selenium scripts can be analyzed with TestNG to generate reports.