

DATE : 18-03-2025

Maven and AEM Project Structure Guide

1. Maven Life Cycle

Maven follows a structured build life cycle consisting of phases:

- **Clean:** Removes target directory and compiled files.
- **Validate:** Checks the correctness of the project.
- **Compile:** Compiles the source code.
- **Test:** Runs unit tests.
- **Package:** Bundles the project into a JAR/WAR.
- **Verify:** Ensures the integrity of the package.
- **Install:** Installs the package in the local repository.
- **Deploy:** Deploys the package to a remote repository.

Command to Run Maven Life Cycle:

```
mvn clean install
```

2. What is pom.xml File and Why Do We Use It?

POM (Project Object Model) XML is the core configuration file in a Maven project. It defines:

- Project Information
- Dependencies (libraries required for the project)
- Plugins (to perform tasks like compilation, packaging, deployment)
- Build Configuration

Example pom.xml:

```
<project xmlns="http://maven.apache.org/POM/4.0.0" ...>
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.myTraining</groupId>
  <artifactId>aem-project</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>pom</packaging>
</project>
```

3. How Dependencies Work?

Dependencies define external libraries required for a project. Maven fetches these from a central repository.

Example Dependency in pom.xml:

```
<dependencies>
  <dependency>
    <groupId>org.apache.sling</groupId>
    <artifactId>org.apache.sling.api</artifactId>
    <version>2.20.0</version>
  </dependency>
</dependencies>
```

4. Check the Maven Repository

Maven repositories are used to store dependencies.

- Local Repository (~/.m2/repository/)
- Central Repository (Maven's default repository)
- Remote Repository (hosted by organizations)

Command to Check Local Repo:

ls ~/.m2/repository/

5. How All Modules Build Using Maven?

AEM projects use a multi-module structure, and all modules can be built together using:

`mvn clean install`

This will compile and package all modules in the project.

6. Can We Build a Specific Module?

Yes, use the `-pl` (project list) flag to build a specific module.

Example: Build only core module:

`mvn clean install -pl core -am`

7. Role of ui.apps, ui.content, and ui.frontend Folder

- `ui.apps`: Stores AEM components, templates, and Java code.
 - `ui.content`: Contains sample content like pages and assets.
 - `ui.frontend`: Holds frontend assets (CSS, JS, client-side libraries).
-

8. Why Are We Using Run Mode?

Run modes allow different configurations for different environments (dev, stage, prod).

Example Command to Run AEM in Specific Mode:

`java -jar aem-author-p4502.jar -r author,dev`

9. What is Publish Environment?

The publish environment (port 4503) serves content to end-users. Content is replicated from the author environment (4502) to the publish instance.

10. Why Are We Using Dispatcher?

The Dispatcher is used for:

- **Caching**: Improves performance.
 - **Security**: Filters requests before reaching AEM.
 - **Load Balancing**: Distributes traffic across multiple instances.
-

11. From Where Can We Access crx/de?

The CRX/DE (Content Repository Extreme Developer Edition) is accessed via:

`http://localhost:4502/crx/de`

This allows developers to view and modify AEM content structures.
