



Apache Maven

Beginner to Guru

Why Use Apache Maven?



Why Use Apache Maven?

- Maven has ~70% of the build tool market for Java applications
 - ~20% of marketshare : Gradle
 - ~10% of marketshare : Ant
- Very common in large companies
- Started in 2002, became top level Apache project in 2004
- Supported by all major IDEs
- Commonly used by large Open Source Projects (ie Spring Framework, Spring Boot, etc)





Maven - So Stable Its Boring





Maven Advantages

- Quick Project Setup
 - Maven brings conventions over configuration, thus reducing setup time
- Projects are modular
- Mature Dependency Management
- Mature Project Build Lifecycle
- Robust plugin community





Apache Maven - De Facto Standard

- Apache Maven has established 'standards' used by other build tools
 - **Maven Standard Directory Layout** - In most part adapted by other tools such as Gradle
 - **Artifact Naming** - Apache Maven helped establish how Java artifacts are named.
 - **Artifact Repositories** - Apache Maven established the structure of artifact repositories
- Prior to Maven these 'standards' did not exist
- New build tools are compatible with these 'standards'





Maven Disadvantages

- Projects are described in an XML document, constrained by an XML Schema
 - Some consider XML 'dated'
 - Gradle uses a Groovy DSL, which can offer greater build flexibility
 - Rare the additional flexibility is needed



