

Naming Conventions

Naming Conventions

In the Maven world, a project product (jar, war, ...) is called an artifact. Every project generates **one and only one primary artifact** (secondary artifacts will be covered in more advanced trails). An artifact is always identified by two ids:

- **groupid** It identifies your project as a whole (it will make more sense when you start using multi-modules build) and should be unique. Maven enforces a naming schema following the Java package name rules. Hence, the groupid has to start by at least a domain name you control followed by any subgroups you want. For further informations, look at [The package names in the Java language](#). [**characters restrictions**]

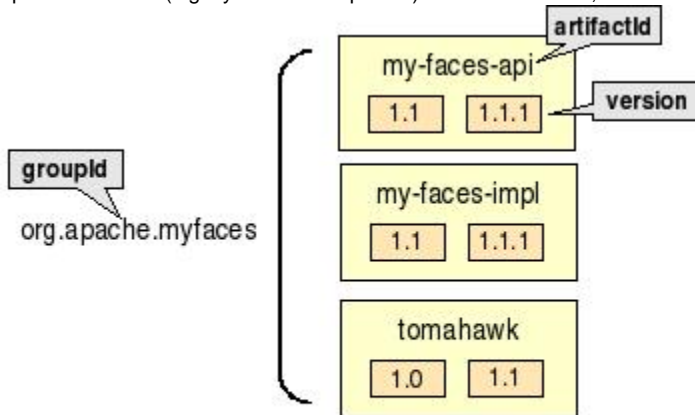
An example: `org.apache.maven`

- **artifactId** It identifies precisely the current project inside the group identified by the groupid. [**characters restrictions**]

An example: **groupid**: `org.apache.maven.doxia`, **artifactId**: `doxia`

and a version:

- **version**: The version number allows you to distinguish your different project releases. There are two kinds of versions depending of the release type it is associated with:
 - Official releases : 1.0, 2.2, ...
 - Specific releases (nightly build or snapshots) : 1.0-SNAPSHOT, 2.1.2-SNAPSHOT [**characters restrictions**]



The apache myfaces artifacts

Having precise naming formats allows Maven to manage efficiently for you the dependencies and the plugins your project depends upon. We'll see what that means in the next lessons.