Project creation "   "

Creating a project from an archetype involves three steps:

- the selection of the archetype,
- the configuration of that archetype,
- the effective creation of the project from the collected information.

Usage

Calling `archetype:generate` the plugin will first ask to choose the archetype from the internal catalog. Just enter the number of the archetype.

It then asks you to enter the values for the groupId, the artifactId and the version of the project to create and the base package for the sources.

It then asks for confirmation of the configuration and performs the creation of the project.

In the following example, we selected the quickstart archetype (default value, which is 15 here but may vary depending on the repository content) and set `groupId` to `com.company`, `artifactId` to `project`, `version` to `1.0` and `package` to `com.company.project`.

```
$ mvn archetype:generate            mvn archetype:generate -Dfilter=quickstart
[INFO] Scanning for projects...
[INFO] Searching repository for plugin with prefix: 'archetype'.
[INFO] ------------------------------------------------------------
---------
[INFO] Building Maven Default Project
[INFO]    task-segment: [archetype:create] (aggregator-style)
[INFO] ------------------------------------------------------------
---------
[INFO] Preparing archetype:generate
[INFO] No goals needed for project - skipping
[INFO] Setting property: classpath.resource.loader.class => 'org.codeh
aus.plexus.velocity.ContextClassLoaderResourceLoader'.
[INFO] Setting property: velocimacro.messages.on => 'false'.
[INFO] Setting property: resource.loader => 'classpath'.
[INFO] Setting property: resource.manager.logwhenfound => 'false'.
[INFO] [archetype:generate]
Choose archetype:            "       "
1: internal -> org.appfuse.archetypes:appfuse-basic-jsf (AppFuse arche
type for creating a web application with Hibernate, Spring and JSF)
2: internal -> org.appfuse.archetypes:appfuse-basic-spring (AppFuse ar
chetype for creating a web application with Hibernate, Spring and Spri
ng MVC)
3: internal -> org.appfuse.archetypes:appfuse-basic-struts (AppFuse ar
chetype for creating a web application with Hibernate, Spring and Stru
ts 2)
4: internal -> org.appfuse.archetypes:appfuse-basic-tapestry (AppFuse
archetype for creating a web application with Hibernate, Spring and Ta
```

```
pestry 4)
5: internal -> org.appfuse.archetypes:appfuse-core (AppFuse archetype
for creating a jar application with Hibernate and Spring and XFire)
6: internal -> org.appfuse.archetypes:appfuse-modular-jsf (AppFuse arc
hetype for creating a modular application with Hibernate, Spring and J
SF)
7: internal -> org.appfuse.archetypes:appfuse-modular-spring (AppFuse
archetype for creating a modular application with Hibernate, Spring an
d Spring MVC)
8: internal -> org.appfuse.archetypes:appfuse-modular-struts (AppFuse
archetype for creating a modular application with Hibernate, Spring an
d Struts 2)
9: internal -> org.appfuse.archetypes:appfuse-modular-tapestry (AppFus
e archetype for creating a modular application with Hibernate, Spring
and Tapestry 4)
10: internal -> org.apache.maven.archetypes:maven-archetype-j2ee-simpl
e (A simple J2EE Java application)
11: internal -> org.apache.maven.archetypes:maven-archetype-marmalade-
mojo (A Maven plugin development project using marmalade)
12: internal -> org.apache.maven.archetypes:maven-archetype-mojo (A Ma
ven Java plugin development project)
13: internal -> org.apache.maven.archetypes:maven-archetype-portlet (A
 simple portlet application)
14: internal -> org.apache.maven.archetypes:maven-archetype-profiles (
)
15: internal -> org.apache.maven.archetypes:maven-archetype-quickstart
 ()
16: internal -> org.apache.maven.archetypes:maven-archetype-site-simpl
e (A simple site generation project)
17: internal -> org.apache.maven.archetypes:maven-archetype-site (A mo
re complex site project)
18: internal -> org.apache.maven.archetypes:maven-archetype-webapp (A
simple Java web application)
19: internal -> org.apache.struts:struts2-archetype-starter (A starter
 Struts 2 application with Sitemesh, DWR, and Spring)
20: internal -> org.apache.struts:struts2-archetype-blank (A minimal S
truts 2 application)
21: internal -> org.apache.struts:struts2-archetype-portlet (A minimal
 Struts 2 application that can be deployed as a portlet)
22: internal -> org.apache.struts:struts2-archetype-dbportlet (A start
er Struts 2 portlet that demonstrates a simple CRUD interface with db
backing)
23: internal -> org.apache.struts:struts2-archetype-plugin (A Struts 2
 plugin)
Choose a number or apply filter (format: [groupId:]artifactId, case se
nsitive contains): 15:    Enter
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:

1: 1.0-alpha-1
2: 1.0-alpha-2
3: 1.0-alpha-3
```

```
4: 1.0-alpha-4
5: 1.0
6: 1.1
Choose a number: 6:   Enter
Define value for groupId: : com.company
Define value for artifactId: : project
Define value for version: : 1.0
Define value for package: : com.company.project
Confirm properties configuration:      "         "
groupId: com.company
artifactId: project
version: 1.0
package: com.company.project
 Y: :   Enter
[INFO] ---------------------------------------------------------------
-------------
[INFO] Using following parameters for creating OldArchetype: maven-arc
hetype-quickstart:RELEASE
[INFO] ---------------------------------------------------------------
-------------
[INFO] Parameter: groupId, Value: com.company
[INFO] Parameter: packageName, Value: com.company.project
[INFO] Parameter: package, Value: com.company.project
[INFO] Parameter: artifactId, Value: project
[INFO] Parameter: basedir, Value: /home/local/rafale/projects/tmp
[INFO] Parameter: version, Value: 1.0
[INFO] OldArchetype created in dir: /home/local/rafale/projects/tmp/pr
oject
[INFO] ---------------------------------------------------------------
---------
[INFO] BUILD SUCCESSFUL
[INFO] ---------------------------------------------------------------
---------
[INFO] Total time: 54 seconds
[INFO] Finished at: Fri Aug 26 23:01:01 GMT 2011
[INFO] Final Memory: 10M/25M
[INFO] ---------------------------------------------------------------
---------
```

Here's the resulting tree of the created project

```
$ tree project
project
|-- pom.xml
`-- src
    |-- main
    |   `-- java
    |        `-- com
    |             `-- company
    |                  `-- project
    |                       `-- App.java
```

```
    `-- test
        `-- java
            `-- com
                `-- company
                    `-- project
                        `-- AppTest.java


11 directories, 3 files
```

Filtering to reduce archetype list

As of version 2.1, you can reduce the list of displayed archetypes. The filter use the following format: `[groupId:]artifactId`. If you use single word without `:`, only artifactId will be checked. The filtering applied is a case sensitive contains on the artifactId (and groupId if set).

Two options

1. With a mojo parameter:

```
$ mvn archetype:generate -Dfilter=org.apache:struts
```

The displayed list will contain only archetypes with a groupId containing `org.apache` AND an artifactId containing `struts`

2. Through the prompt:

```
$ mvn archetype:generate
```

The full list is displayed and in the prompt response, you will be able to answer with a filter.

```
  Choose a number or apply filter (format: [groupId:]artifactId, ca
se sensitive contains): org.apache:struts
```

Notice that if your filter doesn't match any archetype, the previous list will be displayed again.