

Title: Maven clean install: Failed to execute goal org.apache.maven.plugins:maven-resources-plugin:3.2.0:resources

Post Body:

I don't know what's the problem. Trying to install maven for a college project. I already installed maven as instructed [here](#). [Picture Environmental Variables, maven version in cmd](#) And I also tried to delete the .m2 file. When I try to install maven over the plugin in IntelliJ, I get the message: "process terminated" with no extra info. Then I tried to install over the terminal with mvn clean install and i got this.

```
[INFO] Scanning for projects... [WARNING] [WARNING] Some problems were encountered while building the effective model for com.
```

Then I ran "mvn clean install -X" for more Information and it gives out this in addition.

```
at org.apache.maven.lifecycle.internal.MojoExecutor.execute (MojoExecutor.java:215) at org.apache.maven.lifecycle.inte
```

I honestly have no clue what it is telling me. Is it because there is something wrong with my pom.xml? Pls help me! Thank you in advance.

```
<?xml version="1.0" encoding="UTF-8"?> <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XM
```

Edit: I used the spring boot initializer to create the project.

Accepted Answer:

It might be a new bug coming with some latest release of *Spring-boot* (and then : *Spring*).

I had it. The workaround is to put in your `pom.xml` this corrected dependency :

```
<!-- Pour contourner un bug à la génération par Spring-boot 2.4.x : Failed to execute goal org.apache.maven.plugins:maven-res
```

But this is a temporary workaround to 'save you' at the time the trouble fall upon you.

The way to correct the trouble correctly is to ensure that your `properties` files are now correctly encoded in UTF-8.

Because the underlying cause that makes this problem appearing is the presence of "junk chars" replacing accented characters when ISO-8859-1 (or other) properties files are considered as UTF-8 ones by *Spring*, while they aren't.

Highest Rated Answer:

The root cause of the exception is that some resources contain resources which should not filtered (aka binary). The correct solution is to define an exclude for file extensions which should not filtered [like the following](#):

```
<project> ... <build> <plugins> <plugin> <groupId>org.apache.maven.plugins</groupId> <artifactId>
```