Title: What is the best practice to get the file name that was defined in the pom.xml, from within the docker file?

Post Body:

I am creating a docker file for my Spring Boot application.

Here is the file:

```
FROM openjdk:8-jdk-alpine COPY eureka/target/eureka-1.0.jar app.jar ENTRYPOINT ['java','-Djava.security.egd=file:/dev/./urando
```

The name of the jar file (it is eureka-1.0.jar in this case) is defined in the pom.xml

```
<artifactId>eureka</artifactId> <version>1.0</version> <packaging>jar</packaging>
```

My question is; when I change the file name in the pom, I am having to reflect the change to the docker file manually, which is ruining the automatic deployment process.

What is the best practice to get the file name that was defined in the pom.xml, from within the docker file?

(In case it matters for best practices: there are multiple docker files all similar to this one and they are used by a docker-compose.yml)

Accepted Answer: None

Highest Rated Answer:

Usually you define the `<finalName>` in your pom file to keep the name static..

The default for the final name is defined like this:

```
 <build>     <finalName>${project.artifactId}-${project.version}</finalName>     ..  </build>
```

This means if you release your artifact etc. you have to change the `Dockerfile`...The simplest solution is to change the definition in your pom ilke this:

```
 <build>     <finalName>${project.artifactId}</finalName>     ..  </build>
```

Than you need to change the `Dockerfile` only if you change your artifactId which usually does not happen very often...

Update

What you could do is to provide arguments to your `Dockerfile` like:

```
#!/bin/bash POM_VERSION=$(mvn -q help:evaluate -Dexpression=project.version -DforceStdout=true) echo 'POM Version: $POM_VERSIO
```

One word about the line: `POM_VERSION=..` Starting with maven-help-plugin version 3.1.0 it is possible to extract things from the pom file like this in particular without any grep/awk vodoo.

The Dockerfile can look like this:

```
# FROM alpine:3.6 (plus Open JDK?) FROM openjdk:8u131-jre-alpine ARG APPVERSION RUN echo 'Building ${APPVERSION}' RUN mkdir /u
```

The problem here is simply that `CMD` does not support ENV,ARGS expanding which means you need to do the copy by using a version as above. You could use the `ARG` at several points but not at all locations...