

Title: docker-compose build download few pom dependencies each time

Post Body:

I am using [Heroku Java Docker Image](#) and [docker-compose](#) to run dropwizard java based web service locally.

When I run `docker-compose build web` command to build code, It goes to download few dependencies each time. So build process turnaround time increased.

My project docker file is only single line : `FROM heroku/java`

Here are build log:

```
[INFO] ----- [INFO] Building generator-app-server 0.0.2-SNAPSHOT
```

`docker-compose build` not using cached dependencies for above libs. How to forcefully use cached dependencies?

Googled a lot but no luck. Share if anyone had faced and fixed.

#### Update:

```
Dockerfile FROM heroku/java docker-compose.yml web: build: . command: 'bash -c ''java $JAVA_OPTS -jar target/generator-ap
```

Accepted Answer: None

Highest Rated Answer:

My previous answer didn't make sense (I mistakenly thought that Maven dependencies were handled at run time and they are handled at `ONBUILD` instructions instead). But I'll try to give a second explanation of why these specific dependencies are not cached.

The downloaded dependencies listed in the question are from the builtin plugins for Maven `clean`. In the parent `Dockerfile` we download the project dependencies at <https://github.com/heroku/docker-java/blob/master/Dockerfile#L14> (everything in your POM). Then, later on in the `Dockerfile` it runs the `clean` at <https://github.com/heroku/docker-java/blob/master/Dockerfile#L18>. Since the `maven-clean-plugin` isn't part of your POM, it gets downloaded automatically at the `clean` step (it isn't cached from the previous `Dockerfile` instructions).

So, if you wanted to cache the `maven-clean-plugin` as well, you may need to add it as a dependency in your POM (you might be able to get away with just `<scope>import</scope>`).