Title: gradle continuous build trick doesn't work in docker container

Post Body:

Hi I'm trying to use the trick described here to allow continuous building inside docker containers. The trick works fine when I open two separate terminals on my host machine, but fails when used in docker containers.

docker-compose.yml

```
  build_server:     image: gradle:6.3.0-jdk8     working_dir: /home/gradle/server     volumes:       - ./server:/home/gradle/s
```

The error message I got from server container:

```
server_1         | FAILURE: Build failed with an exception. server_1         | server_1         | * What went wrong: server_1
```

It looks like gradle has added locks on local cache files and prevents `bootRun` task from being run in the other container. However, the trick works fine when I run the tasks in two terminals on my host machine, or when I only run the `build_server` container and run `bootRun` on host terminal. I wonder why it doesn't work inside docker containers. Thanks for helping in advance!

Accepted Answer: None

Highest Rated Answer:

Found a workaround by setting a different project cache dir for the server container. i.e. replace the command with the following

```
command: ["gradle", "bootRun", "--project-cache-dir=/tmp/cache"]
```

Might not be the best solution but it does circumvent the problem caused by gradle's lock.