

J-TECH
TALK

编写Dockerfile的 最佳实践

嘉宾： Jina AI Engineering Manager 苗兆丰

线上直播分享回顾



CONTENT

01 What is Dockerfile?

02 Why it's important?

03 Principles

04 Action items



扫码添加小助手
加入Jina AI社区

What is Dockerfile?

01

What is Dockerfile?

Dockerfile/Blueprint



```
FROM ubuntu:18.04
LABEL author="zhaofeng.miao@jina.ai"
COPY . /app
RUN touch /app/test
RUN rm -r /app/test
CMD echo 'hello S&T'
```



Build



```
docker build -t hello-world .
```



Use



```
docker run hello-world
```



Why it's important to write a good Dockerfile?

02

Write a good Dockerfile

Why?

1. **Reduce the size of Docker image.**
 - a. **Easier for transmission
(faster push/pull)**
 - b. **Smaller storage
requirements**

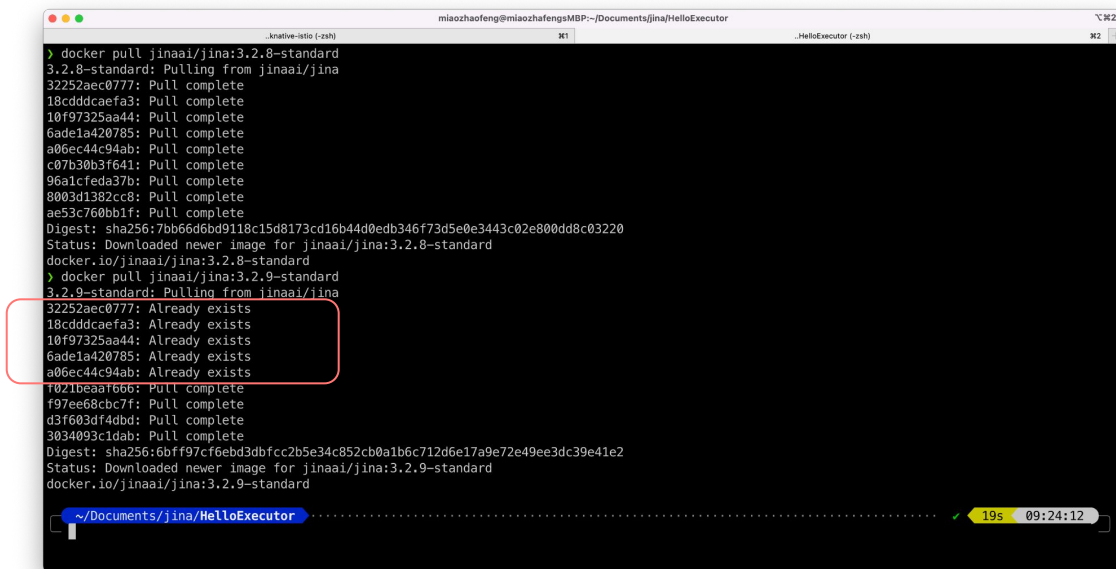


扫码了解更多
Jina Hub

Write a good Dockerfile

Why?

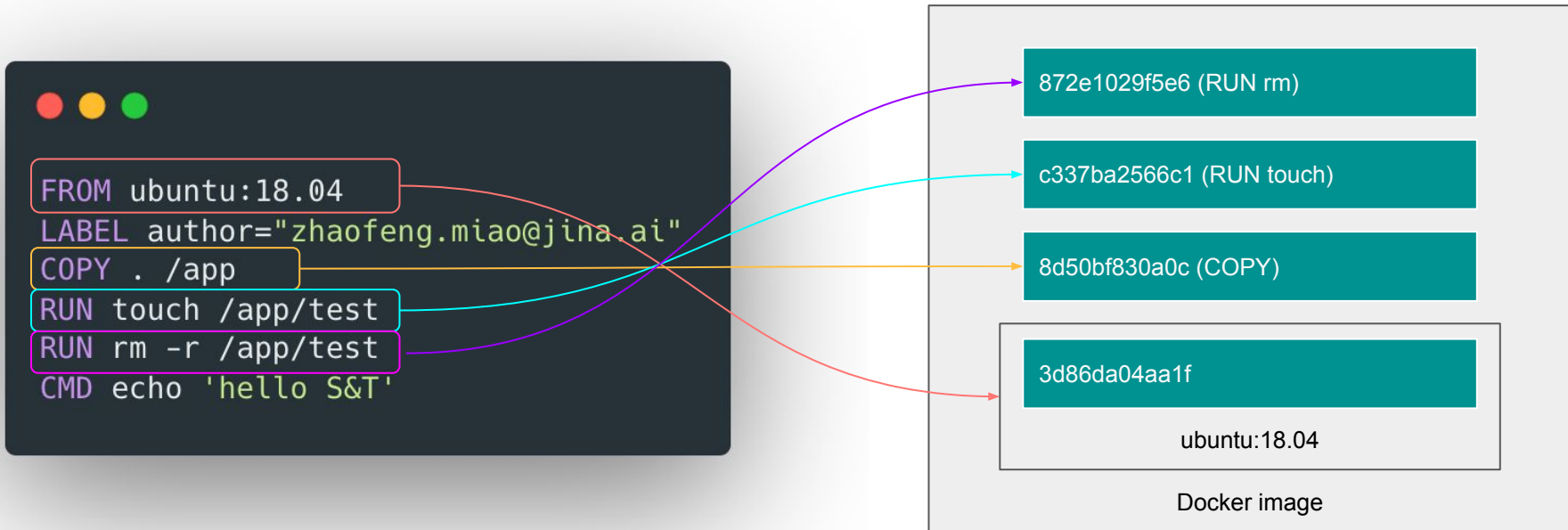
1. Reduce the size of Docker image.
 - a. Easier for transmission (faster push/pull)
 - b. Smaller storage requirements
2. Get more opportunities to take advantage existing cache layer



```
miaozaofeng@miaozaofengMBP:~/Documents/jina>HelloExecutor
> docker pull jinaai/jina:3.2.8-standard
3.2.8-standard: Pulling from jinaai/jina
32252aec0777: Pull complete
18cdddcaefa3: Pull complete
10f97325aa44: Pull complete
6ade1a420785: Pull complete
a06ec44c94ab: Pull complete
c07b30b3f641: Pull complete
96a1cfeda37b: Pull complete
8003d1382cc8: Pull complete
ae53c760bb1f: Pull complete
Digest: sha256:7bb66d6bd9118c15d8173cd16b44d0edb346f73d5e0e3443c02e800dd8c03220
Status: Downloaded newer image for jinaai/jina:3.2.8-standard
docker.io/jinaai/jina:3.2.8-standard
> docker pull jinaai/jina:3.2.9-standard
3.2.9-standard: Pulling from jinaai/jina
32252aec0777: Already exists
18cdddcaefa3: Already exists
10f97325aa44: Already exists
6ade1a420785: Already exists
a06ec44c94ab: Already exists
f021beaa1666: Pull complete
f97ee68cbc7f: Pull complete
d3f603df4dbd: Pull complete
3034093c1dab: Pull complete
Digest: sha256:6bfff97cf6ebd3dbfcc2b5e34c852cb0a1b6c712d6e17a9e72e49ee3dc39e41e2
Status: Downloaded newer image for jinaai/jina:3.2.9-standard
docker.io/jinaai/jina:3.2.9-standard
```

What's a Docker image underneath?

What does “cache layer” means?





What's a Docker image underneath?

What does “cache layer” means?

Linux/Unix:

Is `/var/lib/docker/overlay2`

Mac:

Not easy to see this folder since it's
using Hyperkit (Virtualized)

```
/ # cd /var/lib
/var/lib # ls
apk      docker      ip6tables  iptables  misc      udhcpd
/var/lib # cd docker/
/var/lib/docker # ls
buildkit  containers  network    plugins    swarm      trust
containerd image        overlay2    runtimes    tmp         volumes
/var/lib/docker # cd overlay2/
/var/lib/docker/overlay2 # ls
014c28c6bc19d290b6aa87a833d8e7ad313ac824207033b4ac2d0755f035c9a0
01910d1ba970901d4e199e6162714a097ca433c1bcd49bd244da391937b6de4b
024afb859ae5ae99df689294287c3565c2c6b1a40ee4157763c667dc8b99c871
036b61ea5aec54dce67f39a876968d511615fbee498ff94eb90e96e8b327591e2
03e796aa95e15027ce6baaa704a57458f9c73752f1f9f82e206ad9709766fa9
0496cb06c38f33556acaf2d0b16b299ee96a815fe9ff0298200682a166cf72
0520739c7b5484caa814353dd890be1e56501374379cfb047a091654deecb801
058c327ab7d0a02b663f0c92e589e68fbdcb2a3bf4413b13bbec5e3e652959b3
061f6e6600d959b3138cd0f7e7e4c32d3e43c99ae66be99fa04b92ae556af
06682845cbad19310b8e301d7d08138b1fac0a2a09acf929d39090e2fdad99b5
06948b78068769c7388054389cb1081ef99ae0144846f2c53aef23b16b10c104
06e7f41e02b115838c702c156914ea2e201d52587039da2df876952d715ba2b5
06f1cb621007a597da327c61695b512e764d23dfd9bc4dbbf049b162e7981a9b
0708c718e30ece68518f7ad42c538a549777555d44bac14b88afe2c8aaf9e153
0795bd9e71bc2ecc2ea277088da7acb0c1ecf9f79a1e4f4b12acaa4de2dfc5f65
07ab9d26691396c7aee3d4b400bf6e47f525945be05ff643ea0a37a548790028
0a65e149df3ef351b308a4ebc299d669f97058a05bca2d2b6c39064f8a7adb
0aac9a00f78562f99b201d8a7f7fb9f575a6c02f2e95f75fca386a94b1323e58
0c4be128433b543410a622eab2edea5adf222680b133f45c7453cc432273191
0ce053e40693b922060d49106eae355c77aa242f2eee904d9bdfb59e767ff4040
0dl87002b6fecb5d5cae00f2b8443592c88a4882cea8b4bd591db74a60b990a61
0ee813e359a874cf92112465f07d56b49713c4822726d7fd100a7c2ccd0a247
0f97a30f09c7ef3d4c85cc8cd47ef369602bfdc995da4cf7fedb13660a277f2
0ffb6d6d5f338d03e9f502b7df10e183c6606b2f288f56369e950d74dd46642a
1019a0a5ecbce661dec50c859f3656d3f319493e4a168ea750a381c7a64b013
10409c3948c2877fac9846c83cea098b399d741157521cafe5c8be68ee4272be
1184c6457d7b88bd4c57f038a91f90bc452cea6d214f9c48fbcld5d066cc8ff1
131a7efc7bbb8e18ac0af3f3afb868e0eab39f845deb87c2ce0f151a9b0ad13
13e81a6eec3586b750182053d9e38956ceccc7b25956bd89676b45b34dd41768
1737759bce7c6ccea0124b3188c671b212aff88a8b68a7b3fbcd7a8596546111
173cc2eac451ed2bb4e112a9b222993bf9e2024548245862d3aa7a23efc8f11
```

Principles

03

Principles for writing Dockerfile

1. Only contains minimum things

Smaller, faster and easier to share



扫码添加小助手
加入Jina AI社区

Principles for writing Dockerfile

2. Always remember you are building several layers instead of one image

Understand the actual way to store is important for writing a good Dockerfile



扫码添加小助手
加入Jina AI社区

Action items

04

Action items for writing Dockerfile

1. Use popular image as base image

For example: **alpine**, **busybox**, **python**...

Not just because they are more reliable, **but also there is more opportunities to take advantage of existing layer in users' machine.**



alpine Official Image

Updated a day ago

1B+ 8.6K
Downloads Stars



busybox Official Image

Updated 18 days ago

1B+ 2.5K
Downloads Stars



python Official Image

Updated 2 hours ago

1B+ 7.2K
Downloads Stars

Python is an interpreted, interactive, object-oriented,...

Container	Windows	Linux	386	mips64le
PowerPC 64 LE	IBM Z	ARM	x86-64	ARM 64
Programming Languages				



扫码了解更多
Jina Hub

Action items for writing Dockerfile

2. Use multi-stage builds as possible as you can (recommend)

Keep in mind that sometimes to leverage cache layers, this might not be the right way to go.

```
# syntax=docker/dockerfile:1
FROM golang:1.16-alpine AS build

# Install tools required for project
# Run `docker build --no-cache .` to
update dependencies
RUN apk add --no-cache git
RUN go get
github.com/golang/dep/cmd/dep

# List project dependencies with
Gopkg.toml and Gopkg.lock
# These layers are only re-built when
Gopkg files are updated
COPY Gopkg.lock Gopkg.toml
/go/src/project/
WORKDIR /go/src/project/
# Install library dependencies
RUN dep ensure -vendor-only

# Copy the entire project and build
it
# This layer is rebuilt when a file
changes in the project directory
COPY . /go/src/project/
RUN go build -o /bin/project

# This results in a single layer
image
FROM scratch
COPY --from=build /bin/project
/bin/project
ENTRYPOINT ["/bin/project"]
CMD ["--help"]
```

3. Batch multi command that have same purpose into a single RUN

```
RUN apt-get update && apt-get install -y \  
    bzip \  
    cvs \br/>    git \br/>    mercurial \br/>    subversion \  
    && rm -rf /var/lib/apt/lists/*
```



扫码添加小助手
加入Jina AI社区

4. Remove useless stuff

```
RUN apt-get update && apt-get install -y \  
    bzip2 \  
    cvs \  
    git \  
    mercurial \  
    subversion \  
    && rm -rf /var/lib/apt/lists/*
```

Action items for writing Dockerfile

5. Use .dockerignore

Exclude files not relevant to the build, such as .gitignore to git



扫码了解更多
Jina Hub

Action items for writing Dockerfile

6. Add proper labels

The [OCI Containers Specification](#) defines several conventional labels that encapsulate common use cases for container images.

- `org.opencontainers.image.created`
- `org.opencontainers.image.title`
- etc.

Pre-Defined Annotation Keys

This specification defines the following annotation keys, intended for but not limited to [image index](#) and [image manifest](#) authors:

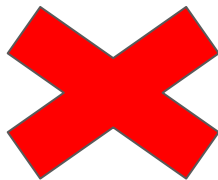
- **org.opencontainers.image.created** date and time on which the image was built (string, date-time as defined by [RFC 3339](#)).
- **org.opencontainers.image.authors** contact details of the people or organization responsible for the image (freeform string)
- **org.opencontainers.image.url** URL to find more information on the image (string)
- **org.opencontainers.image.documentation** URL to get documentation on the image (string)
- **org.opencontainers.image.source** URL to get source code for building the image (string)
- **org.opencontainers.image.version** version of the packaged software
 - The version MAY match a label or tag in the source code repository
 - version MAY be [Semantic versioning-compatible](#)
- **org.opencontainers.image.revision** Source control revision identifier for the packaged software.
- **org.opencontainers.image.vendor** Name of the distributing entity, organization or individual.
- **org.opencontainers.image.licenses** License(s) under which contained software is distributed as an [SPDX License Expression](#).
- **org.opencontainers.image.ref.name** Name of the reference for a target (string).
 - SHOULD only be considered valid when on descriptors on `index.json` within [image layout](#).
 - Character set of the value SHOULD conform to alphanum of `A-Za-z0-9` and separator set of `[-._:@/+]`
 - The reference must match the following [grammar](#):

```
ref      ::= component ("/" component)*
component ::= alphanum (separator alphanum)*
alphanum  ::= [A-Za-z0-9]+
separator ::= [-._:@+] | "--"
```

Action items for writing Dockerfile

7. Assume only have one cpu (Application)

Leave the multi-cores leveraging to orchestrator.



Daemon process
Worker process



扫码了解更多
Jina Hub

Action items for writing Dockerfile

8. Assume can stop anytime (Application)

- Stateless
- Graceful stop
- etc.



扫码添加小助手
加入Jina AI社区

A useful tool for debugging docker image – **dive**

Sum up

- **Docker image consists of layers**
- **2 Principles**
- **Several action items**

References:

- [Best practices for writing Dockerfile](#)
- [About storage drivers](#)
- [GitHub Repo: Dive](#)

欢迎加入 **Jina AI** 开源社区 解锁数据应用的可能性

 官方网站

<https://jina.ai/>

 GitHub

<https://github.com/jina-ai>

 加入全球社区

<https://slack.jina.ai/>

