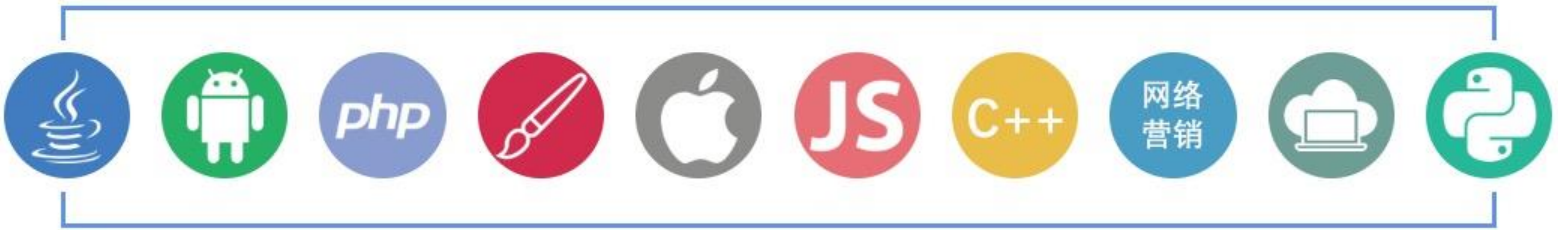
Docker核心技术之Dockerfile



Docker核心技术之Dockerfile - 课程概要

- Dockerfile 简介
- Dockerfile 示例与使用
- Dockerfile 特征
- Dockerfile 命令概述
- 总结

Docker核心技术之Dockerfile

Dockerfile简介

什么是Dockerfile

- Dockerfile其实就是根据特定的语法格式撰写出来的一个普通的文本文件
- 利用docker build命令依次执行在Dockerfile中定义的一系列命令，最终生成一个新的镜像（定制镜像）

Docker核心技术之Dockerfile

Dockerfile 示例与使用

Dockerfile参考示例

```
[root@centos-linux ~]# mkdir df_dir
[root@centos-linux ~]# cd df_dir
[root@centos-linux df_dir]# vi Dockerfile
```

```
# Test
# VERSION               0.0.1

FROM ubuntu
LABEL Description="This image is used to test Dockerfile" Author="Itcast" Version="1.0"
RUN echo 'hello itcast'
CMD ["echo", "this is image created by itcast"]
```

```
[root@centos-linux df_dir]# ls
Dockerfile
[root@centos-linux df_dir]# cat Dockerfile
# Test
# VERSION               0.0.1

FROM ubuntu
LABEL Description="This image is used to test Dockerfile" Author="Itcast" Version="1.0"
RUN echo 'hello itcast'
CMD ["echo", "this is image created by itcast"]
```

Dockerfile使用演示 – docker build

```
[root@centos-linux df_dir]# ls
Dockerfile
[root@centos-linux df_dir]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
redis               latest              bfcb1f6df2db        11 days ago         107MB
ubuntu              latest              452a96d81c30        2 weeks ago         79.6MB
centos              latest              e934aafc2206        5 weeks ago         199MB
[root@centos-linux df_dir]# docker build . -t test-df
Sending build context to Docker daemon  2.048kB
Step 1/4 : FROM ubuntu
 ---> 452a96d81c30
Step 2/4 : LABEL Description="This image is used to test Dockerfile" Author="Itcast" Version="1.0"
 ---> Running in 22c9f76bee25
Removing intermediate container 22c9f76bee25
 ---> a681af4d4745
Step 3/4 : RUN echo 'hello itcast'
 ---> Running in 83aff5229cf6
hello itcast
Removing intermediate container 83aff5229cf6
 ---> d0fc36e866ca
Step 4/4 : CMD ["echo", "this is image created by itcast"]
 ---> Running in 3e727ce0efe3
Removing intermediate container 3e727ce0efe3
 ---> b5b7b22c30c6
Successfully built b5b7b22c30c6
Successfully tagged test-df:latest
[root@centos-linux df_dir]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
test-df             latest              b5b7b22c30c6        2 seconds ago       79.6MB
redis               latest              bfcb1f6df2db        11 days ago         107MB
ubuntu              latest              452a96d81c30        2 weeks ago         79.6MB
centos              latest              e934aafc2206        5 weeks ago         199MB
[root@centos-linux df_dir]#
```

Dockerfile使用命令 – docker build

作用:

根据dockerfile创建镜像

命令格式:

docker build [OPTIONS] PATH | URL | -

命令参数:

PATH **Dockerfile所在路径(文件夹路径)，文件名必须是Dockerfile**

URL **Dockerfile所在URL地址**

OPTIONS:

-t, --tag list **为镜像设置名称和tag**

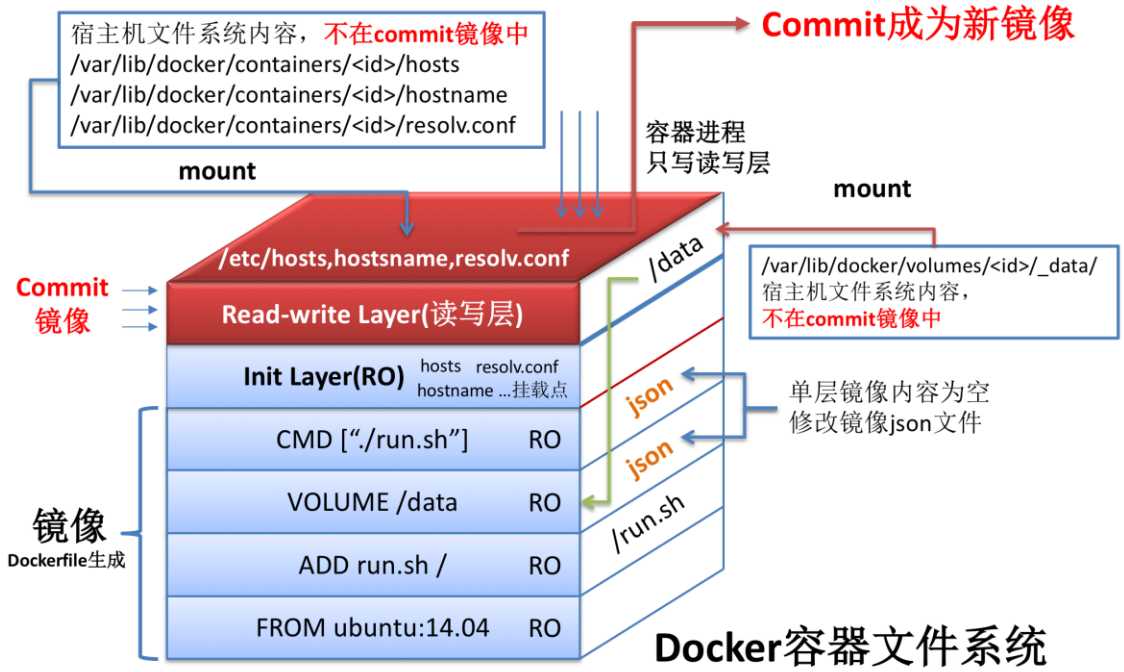
-f, --file string **指定Dockerfile的路径(这是可以使用其他名称命名Dockerfile)**

Docker核心技术之Dockerfile

Dockerfile 特征

Dockerfile 构建特征（一）

- 查看官方的Dockerfile：https://github.com/docker-library/docs



Dockerfile 构建特征（二）

- Dockerfile必须具备一个FROM命令来进行构建
- 每一个Dockerfile命令都会构建一层镜像（本质上是每一层都会启动一个容器，执行完命令后，将容器进行提交后，产生新的镜像层）
- 通过查看下载下来的镜像，发现历史层信息的层ID是missing，其实是因为原本的层id只存在于构建镜像的宿主机上，一旦转移镜像后，历史层消息中将只保留最新一层的ID

Docker核心技术之Dockerfile

Dockerfile 命令概述

Dockerfile 命令概述（一） 查看完整介绍

- FROM: 指定基础镜像
- RUN： 构建镜像过程中需要执行的命令。可以有多条。docker build
- CMD：添加启动容器时需要执行的命令。多条只有最后一条生效。可以在启动容器时被覆盖和修改。
- ENTRYPOINT：同CMD，但这个一定会被执行，不会被覆盖修改。
- ：为镜像添加对应的数据。
- MLABELAINTAINER：表明镜像的作者。将被遗弃，被LABEL代替。
- EXPOSE：设置对外暴露的端口。
- ENV：设置执行命令时的环境变量，并且在构建完成后，仍然生效
- ARG：设置只在构建过程中使用的环境变量，构建完成后，将消失
- ADD：将本地文件或目录拷贝到镜像的文件系统中。能解压特定格式文件，能将URL作为要拷贝的文件
- COPY：将本地文件或目录拷贝到镜像的文件系统中。
- VOLUME：添加数据卷
- USER：指定以哪个用户的名义执行RUN, CMD 和ENTRYPOINT等命令
- WORKDIR：设置工作目录

Dockerfile 命令概述（二）

- ONBUILD：如果制作的镜像被另一个Dockerfile使用，将在那里被执行Docekrfile命令
- STOPSIGNAL：设置容器退出时发出的关闭信号。
- HEALTHCHECK：设置容器状态检查。
- SHELL：更改执行shell命令的程序。Linux的默认shell是[“/bin/sh”，“-c”]，Windows的是[“cmd”，“/S”，“/C”]。

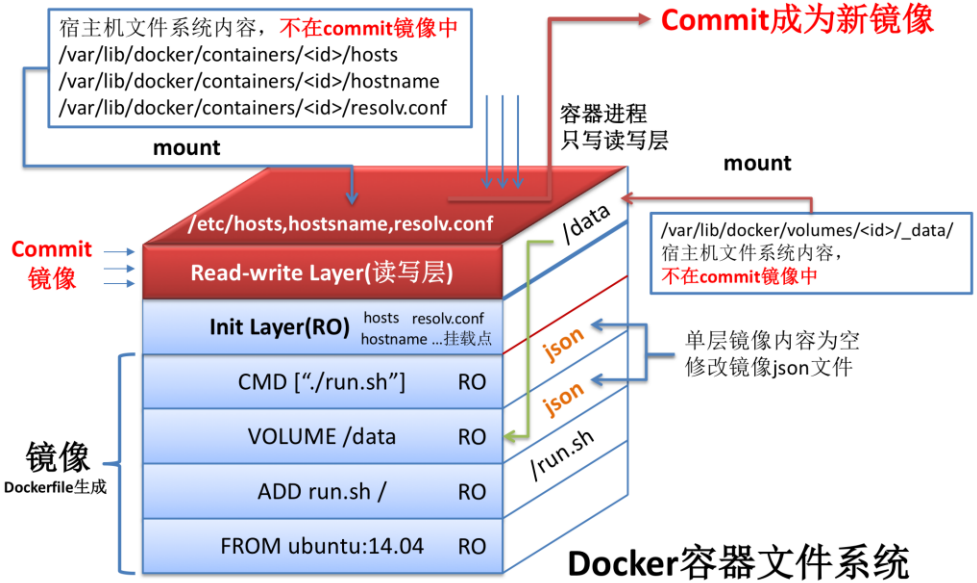
Docker核心技术之Dockerfile

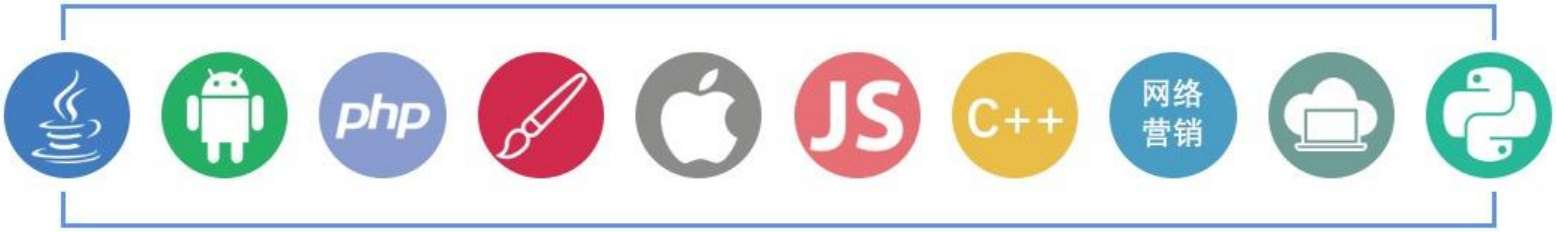
总结

Docker核心技术之Dockerfile - 总结

重点掌握：

- Docker 容器与镜像之间的关系（尤其commit命令的作用和效果）
- Dockerfile 的书写规则和使用规则





Thank you!

改变中国IT教育，我们正在行动