

В Dockerfile каждая инструкция создаёт новый слой в образе Docker. Слои представляют собой набор изменений файловой системы и метаданных, и они накладываются друг на друга в определенной последовательности. Слои помогают оптимизировать процесс сборки и распределения образов Docker.

Вот несколько ключевых моментов:

1. **Кэширование**: Если слой уже был создан ранее и не изменился (то есть инструкции в Dockerfile до этого слоя остались неизменными), Docker будет использовать кэшированный слой, что ускорит процесс сборки.
2. **Разделение**: Разделение образа на слои позволяет более эффективно использовать пространство и ресурсы. Например, если у вас есть несколько образов, которые все базируются на одной и той же ОС, этот базовый слой (ОС) может быть загружен из кэша для всех образов, экономя тем самым пространство.
3. **Наследование**: Вы можете создать новые Docker-образы на основе существующих слоев, добавляя к ним только те слои, которые изменились. Это делает процесс быстрым и эффективным.
4. **Неизменяемость**: Каждый слой является неизменяемым. Это значит, что как только слой создан, его нельзя изменить. Любые изменения приведут к созданию нового слоя.

Вот простой пример Dockerfile с комментариями, объясняющими, какие слои создаются:

```
``Dockerfile
# Этот слой устанавливает базовый образ
FROM ubuntu:18.04

# Установка переменных окружения — создается новый слой
ENV MY_ENV_VAR=my-value

# Установка пакетов — создается новый слой
RUN apt-get update && apt-get install -y curl

# Копирование файлов в образ — создается новый слой
COPY ./my-app /my-app

# Установка рабочего каталога — создается новый слой
WORKDIR /my-app

# Указание команды для запуска контейнера — создается новый слой
CMD ["/run-my-app"]
````
```

Каждая из этих инструкций создаёт новый слой в итоговом образе Docker.

Слои в Docker можно сравнить с процессом строительства многоэтажного дома:

1. **Фундамент (Base Image)**: Всё начинается с фундамента — это ваш базовый образ (например, операционная система). Он служит основой для всего остального.

2. **Каркас (RUN, COPY)**: После фундамента строители возводят каркас здания. В Docker, это слои, которые добавляются при установке программ, копировании файлов и так далее.

3. **Оформление (ENV, WORKDIR)**: Когда каркас готов, приходит время заниматься внутренними работами: проводкой, окраской стен и так далее. В Docker это соответствует установке переменных окружения или рабочего каталога.

4. **Мебель (CMD)**: Последний этап — это меблировка. В Docker это инструкция `CMD`, которая определяет, какое приложение или сервис будет запущен внутри контейнера.

Каждый этап строительства — это отдельный слой. Если вы решите построить еще один дом с таким же фундаментом и каркасом, но с другой окраской и мебелью, вам не нужно начинать всё сначала. Вы можете взять уже готовые части (слои) и добавить только то, что новое. Это экономит время и ресурсы, аналогично тому, как Docker использует кэширование слоев.