

# Docker

## Быстрый вход в контейнеризацию

Клоков Станислав

# Работа с приложением

# О работе с запуском кода

## Говорим пару слов о devops

- В целом, имея рабочий код перед глазами, нам уже довольно несложно хоть как то его запустить
- Но есть колоссальное количество проблем с тем, как поддерживать каждый новый запуск работоспособным и устойчивым к изменениям, как стандартизировать решение по запуску внутри какого либо механизма разработки, как научиться рестартить систему, сделать ее масштабируемой и работоспособной

# Решение проблем

Для того, чтобы избежать сложностей с операционной частью существует профессия devops

- Такие инженеры решают проблемы развертки кода, поступательной доставки в работоспособное окружение новых изменений, работы с нагрузкой, помошью в диагностике точек отказа и многие другие.
- Контейнеризация, оркестрация, маршрутизация, балансировка, CI/CD - задачи операционных инженеров, реализующих общие практики, практики SRE или какие-либо внутренние сценарии, удобные командам.

# Пути запуска приложения

# Введение

Перед тем как перейдем к docker скажем пару слов о контейнеризации в целом

- Локальный запуск
- Виртуализация
- Контейнеризация

# Локальный запуск

## Простейший путь

- Есть плюсы - нам просто все запустить
- Но есть и минусы - нам на так уж и просто не получить конфликтов по ресурсам и по доступу к ресурсам
- Консистентность страдает



# Виртуализация

## А что если в виртуалку?

- Все становится лучше, мы учимся управлять ресурсами и никому не мешаем
- Но все таки машина уже сама по себе ест больше ресурсов, больше требует внимания
- Все работает довольно безопасно
- Но внутри машины все же могут быть те же самые проблемы



# Контейнеризация

## Как сделать удобно

- Контейнеризация изолирует запуск одного приложения
- Она дает возможность гибко настроить необходимый пулл инструментов
- Она дает много вариантов для запуска большого числа приложений на одной машине



# Docker

# План по docker

## О чём поговорим

- Docker
- Docker engine
- Docker daemon
- Docker image
- Docker container
- Docker volume
- Docker compose

# Docker

## В общем об инструменте

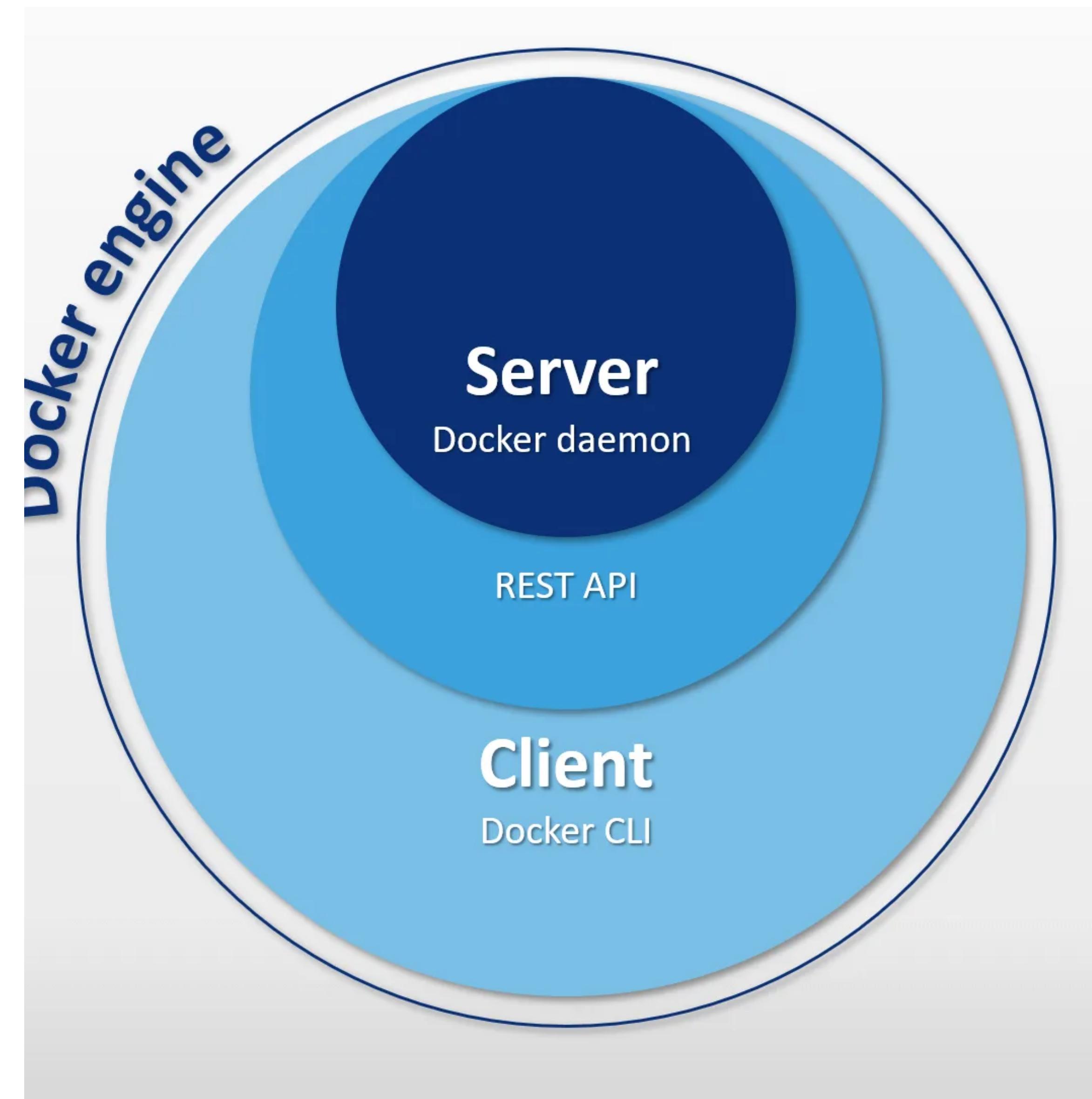
- Контейнеризация — метод, с помощью которого программный код упаковывается в единый исполняемый файл вместе с библиотеками и зависимостями, чтобы обеспечить его корректный запуск. Такие файлы называют контейнерами. Контейнеры можно разворачивать в разных средах и там управлять их работой.
- Docker - инструмент, решающий такую задачу.



# Docker engine

## Портативный движок

- Docker Engine – это программное обеспечение, которое позволяет создавать, разворачивать и администрировать контейнеризированные приложения.
- Составляющие: docker daemon, REST API, docker CLI.



# Docker daemon

## Демон - сервер

- Docker daemon (демон Docker) – это сервер Docker, который ожидает запросов к API Docker.
- Он функционирует как задний плановый процесс на системе-хосте и управляет объектами Docker, включая образы, контейнеры, сети и тома



# Docker image

<https://hub.docker.com/>

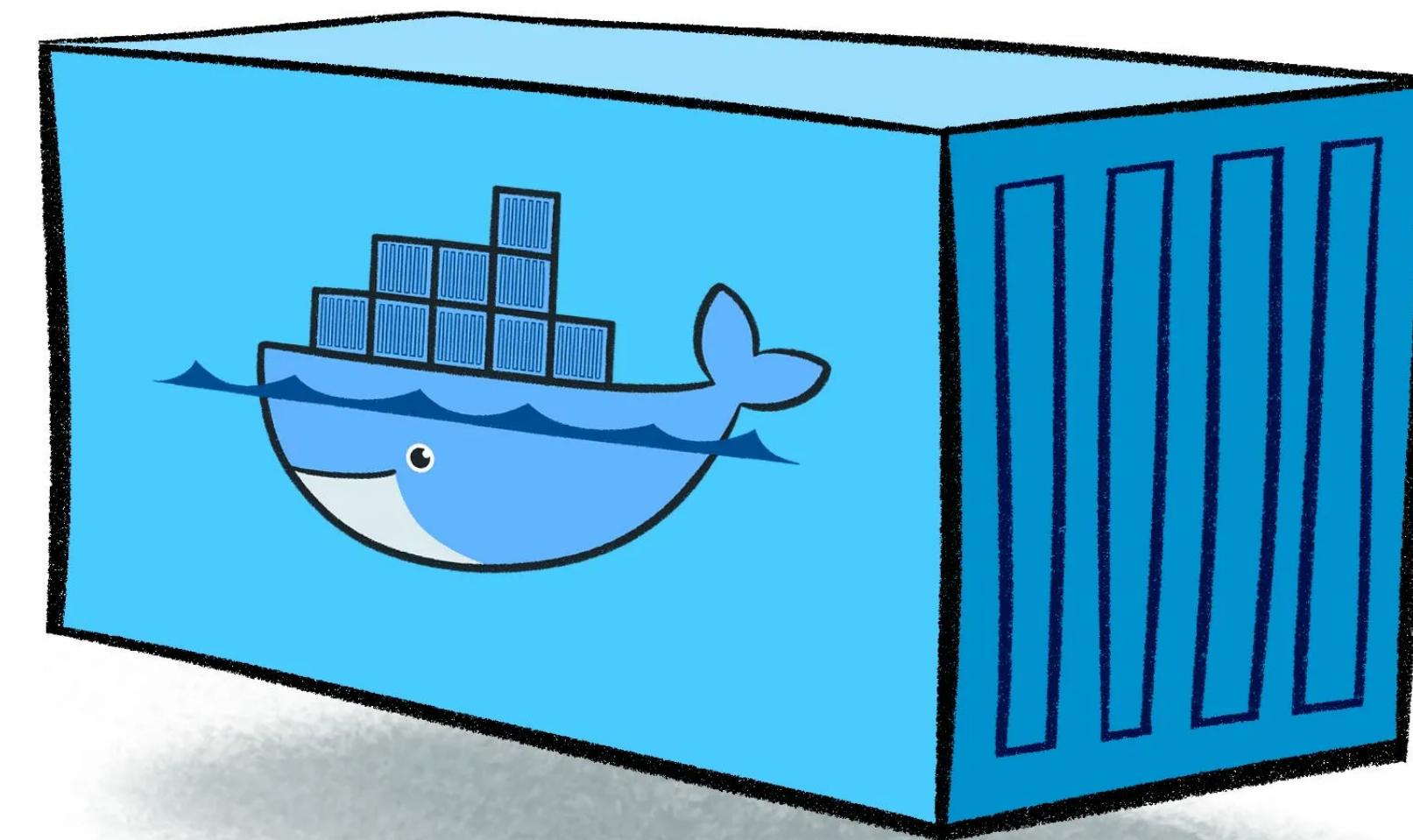
- Docker-образ – шаблон для создания Docker-контейнеров. Представляет собой исполняемый пакет, содержащий всё необходимое для запуска приложения: код, среду выполнения, библиотеки, переменные окружения и файлы конфигурации



# Docker container

## Сами контейнеры

- Docker container – это уже развёрнутое из образа и работающее приложение
- Контейнеры Docker представляют собой компактные портативные хосты, в которых можно быстро и легко запустить приложение. Они работают автономно, изолированно от основной системы и других контейнеров, и потому ошибка в одном из них не влияет на другие работающие контейнеры, а также поддерживающий их сервер



# Docker volume

## Где хранить данные

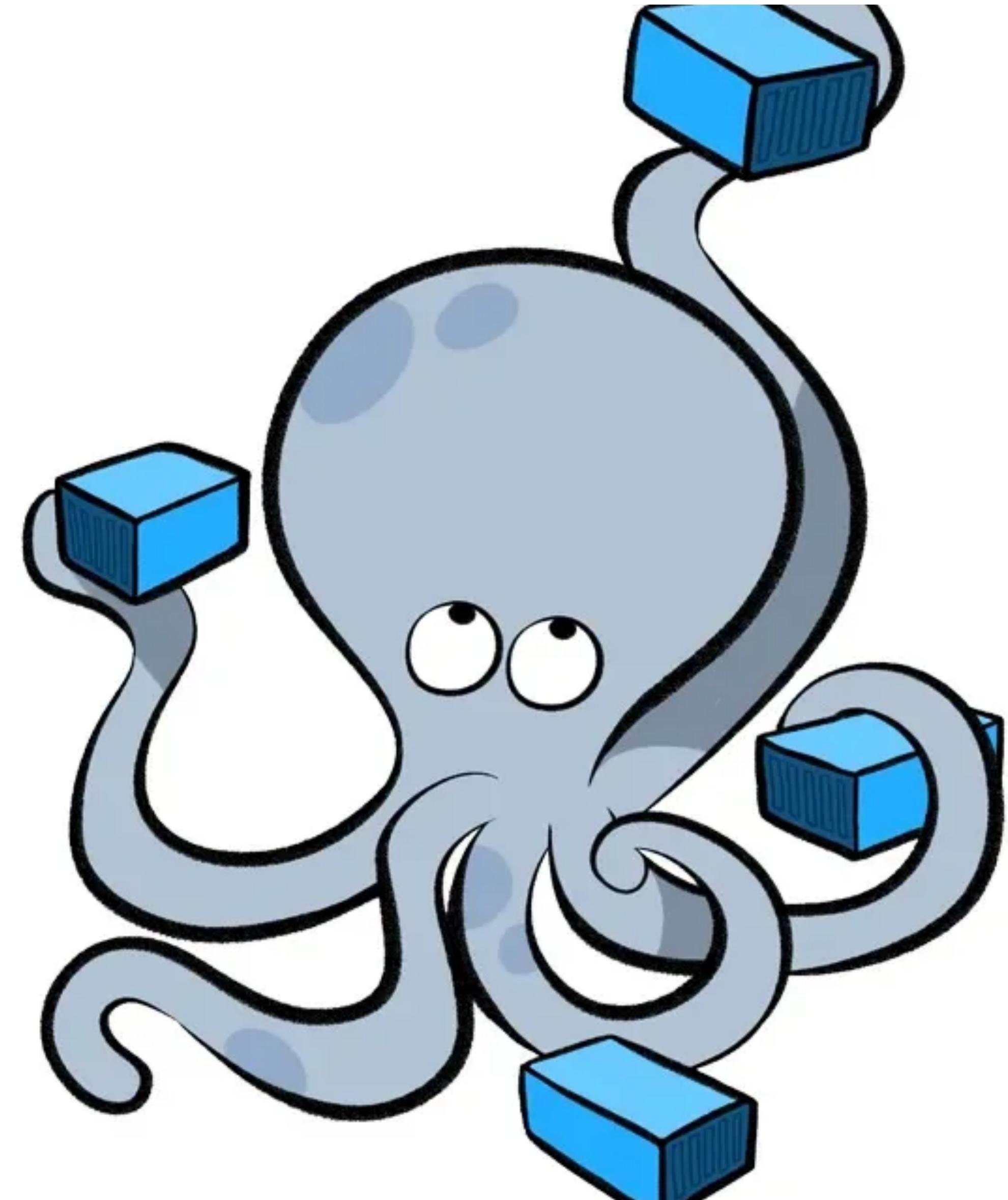
- Docker volume (тома Docker) – это способ создания постоянного хранилища для контейнеров Docker. Они не привязаны к времени жизни контейнера, поэтому сделанные в них записи не исчезнут.



# Docker compose

## Запустить сразу все

- Запускаем много сервисов сразу
- С помощью компоуза можно запустить сразу несколько контейнеров через одну точку входа
- Можно создать общий вольюм или сделать что то еще полезное



**Дополнительно**

# Немного об оркестрации

## Что делать дальше и как масштабироваться?

- Существует инструмент такой как docker-swarm  
<https://docs.docker.com/reference/cli/docker/swarm/>
- Существует прекрасное решение k8s  
<https://kubernetes.io/>
- Для управления этим существует helm  
<https://helm.sh/>