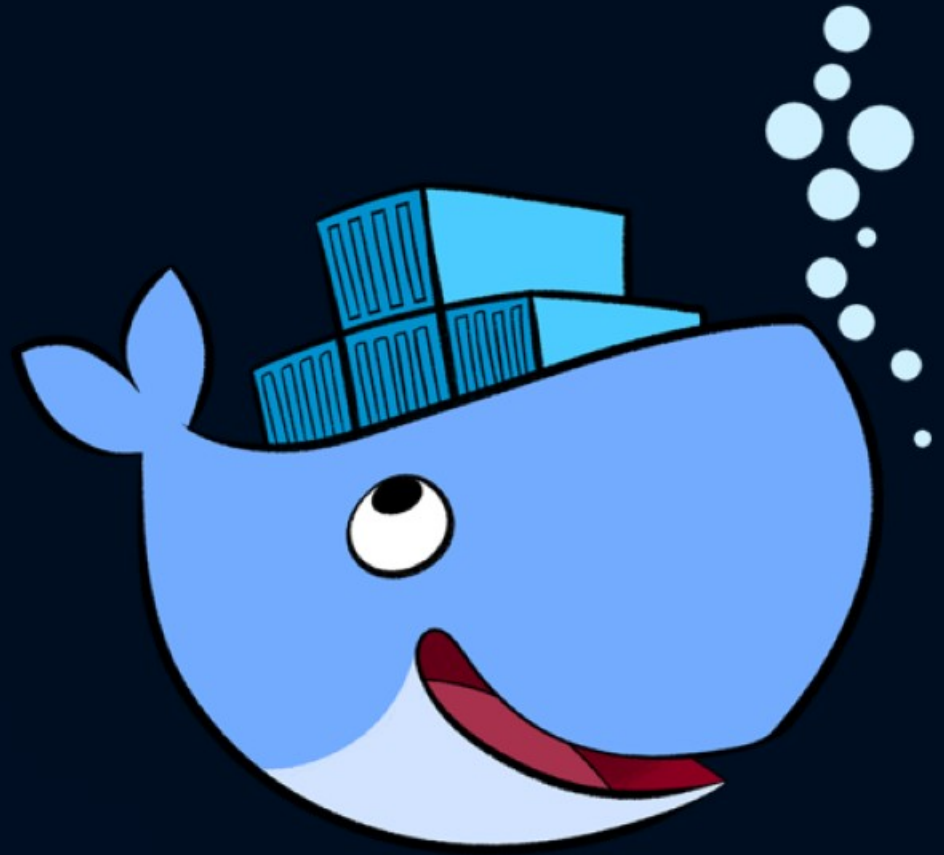




Docker

Что такое Docker?

- **Docker** — программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации. Позволяет «упаковать» приложение со всем его окружением и зависимостями в контейнер.



Основные термины

Docker-образ (Docker-image) — файл, включающий зависимости, сведения, конфигурацию для дальнейшего развертывания и инициализации контейнера.

Docker-файл (Docker-file) — описание правил по сборке образа, в котором первая строка указывает на базовый образ. Последующие команды выполняют копирование файлов и установку программ для создания определенной среды для разработки.

Docker-контейнер (Docker-container) — это легкий, автономный исполняемый пакет программного обеспечения, который включает в себя все необходимое для запуска приложения: код, среду выполнения, системные инструменты, системные библиотеки и настройки.

Том (Volume) — эмуляция файловой системы для осуществления операций чтения и записи. Она создается автоматически с контейнером, поскольку некоторые приложения осуществляют сохранение данных.

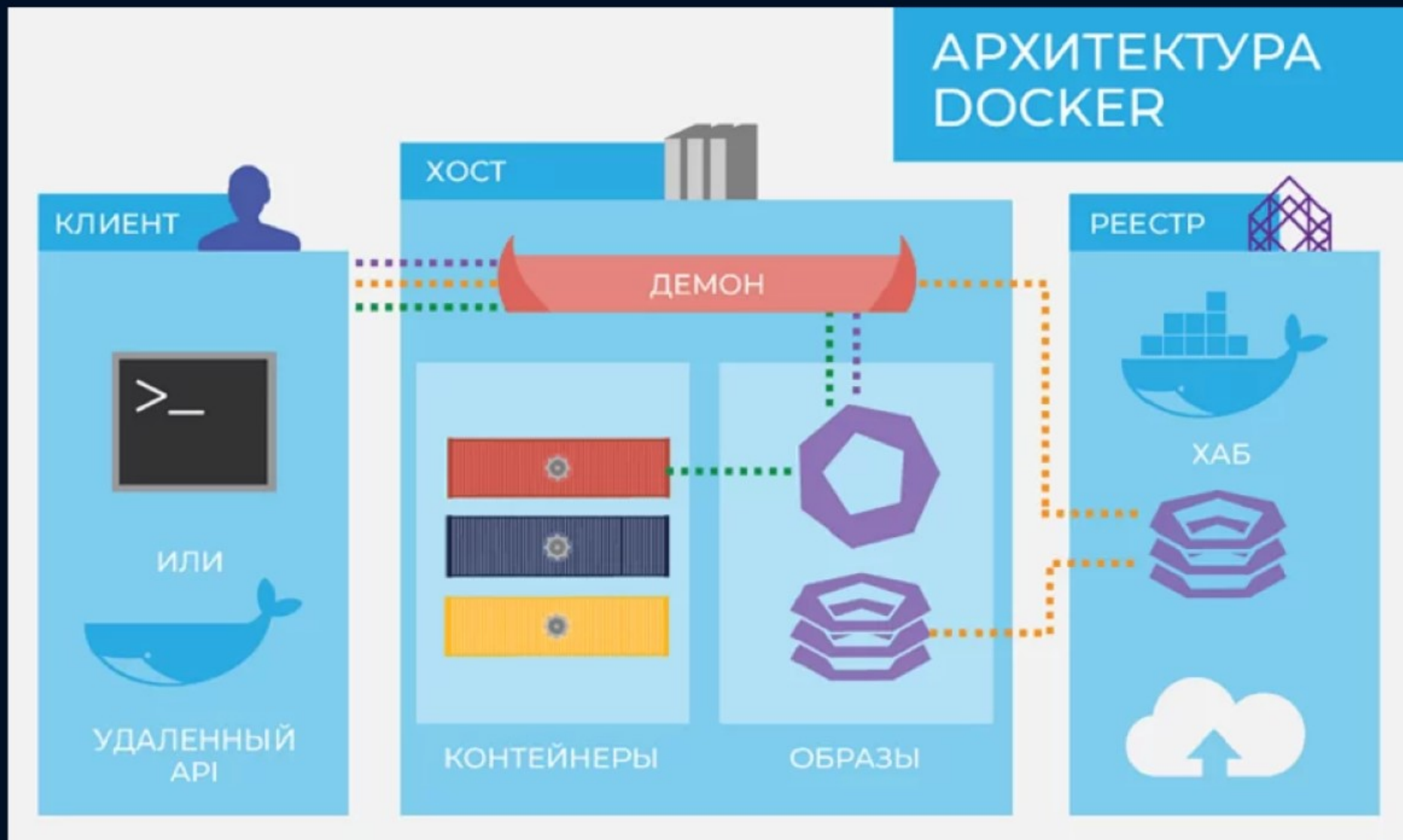
Реестр (Docker-registry) — зарезервированный сервер, используемый для хранения docker-образов.

Stateless vs Stateful

Stateless-приложения. Для примера возьмем два приложения — «корзину» в интернет-магазине и блог на WordPress. Когда мы добавляем товары в корзину (но не переходим к оплате заказа), она сохраняет все данные в cookies на нашей стороне. По завершении сессии приложение сбрасывает все данные и не хранит ничего на сервере. Когда мы снова заходим в интернет-магазин, информация о товарах считывается из нашей системы и снова отображается в корзине. Чтобы работать с покупателем, корзине не нужно получать никакую информацию с бэкенда — роль хранилища выполняет система покупателя. Такие приложения называются *stateless*, поскольку они не хранят информацию о своем состоянии.

Stateful-приложения. Теперь посмотрим на блог. Когда мы создаем новый пост или оставляем комментарий, нужно чтобы любой пользователь в любое время мог получить к ним доступ. Поэтому все данные должны храниться на стороне сервера. Когда мы переходим на ту или иную страницу, движок блога загружает имеющуюся информацию из базы данных, а если мы что-то меняем на странице — записывает изменения на сервер. Такие приложения, которые должны хранить информацию о своем состоянии для корректной работы, называются *stateful*.

Как работает Docker



Что происходит при запуске контейнера

- Происходит запуск образа (Docker-image). Docker Engine проверяет существование образа. Если образ уже существует локально, Docker использует его для нового контейнера. При его отсутствии выполняется скачивание с Docker Hub.
- Создание контейнера из образа.
- Разметка файловой системы и добавление слоя для записи.
- Создание сетевого интерфейса.
- Поиск и присвоение IP-адреса.
- Запуск указанного процесса.
- Захват ввода/вывода приложения.

Основные команды

- **docker run** – создание и первый запуск контейнера из образа
- **docker ps** – показывает список запущенных контейнеров. Ключ **-a** показывает список всех контейнеров на хосте.
- **docker logs “dockername/hash”** – показывает логи контейнера. Ключ **-f** отображает обновления логов в режиме реального времени.
- **docker start/stop “dockername/hash”** – запуск существующего контейнера
- **docker rm** – удаление контейнера
- **docker rmi** – удаление образа

Основные команды

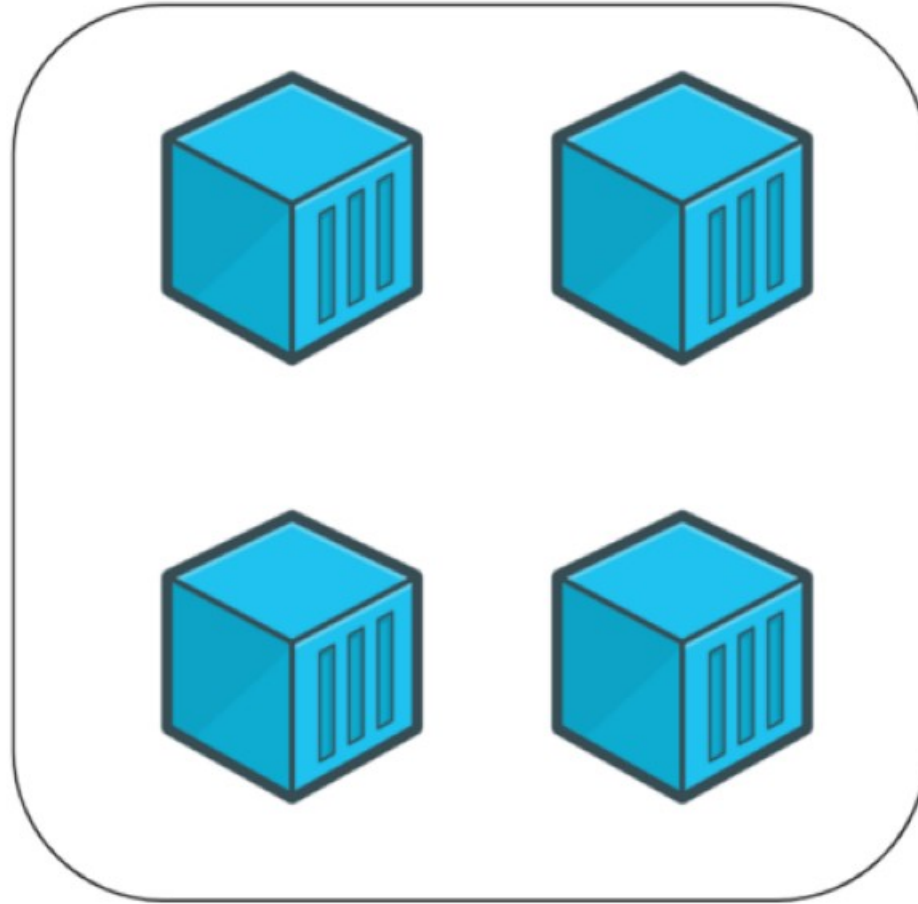
- **docker build** – собрать образ из Dockerfile
- **docker pull/push** – аналогично системе git, получить или отправить образ в хранилище
- **docker cp** – скопировать файл/директорию в контейнер или наоборот
- **docker exec** – запуск команды в запущенном контейнере
- **docker inspect "dockername/hash"** – показывает детальную информацию о запущенном контейнере.

Docker-compose

Технология Docker Compose предназначена для конфигурирования многоконтейнерных приложений.



Docker



Docker-Compose

Portainer

Portainer – это проект с открытым исходным кодом, предоставляющий лёгкий образ Web-интерфейса для управления Docker

portainer.io

Home

LOCAL

Dashboard

App Templates

Stacks

Containers

Images

Networks

Volumes

Events

Host

SETTINGS

Users

Endpoints

Registries

Settings

Container list

Containers

admin

my account

log out

Containers

Columns

Settings

Start

Stop

Kill

Restart




















Pause

Resume

Remove

Add container

Search...

<input type="checkbox"/>	Name	State Filter	Quick actions	Stack	Image	Created	Published Ports	Ownership
<input type="checkbox"/>	mysql	running	   	-	mysql:latest	2021-02-24 11:39:06	 49156:3306	 administrators
<input type="checkbox"/>	postgres	running	   	-	postgres:latest	2021-02-24 11:37:52	 49155:5436	 administrators
<input type="checkbox"/>	portainer	running	   	-	portainer/portainer-ce	2021-02-24 11:24:14	 8000:8000  9000:9000	 administrators

Items per page 10

Полезные ссылки

1. Материалы лекции - <https://github.com/Stenlav/docker-lecture>
2. Docker install - <https://docs.docker.com/get-docker/>
3. Portainer - <https://documentation.portainer.io/v2.0/deploy/ceinstalldocker/>