

## ОСНОВНЫЕ ТЕРМИНЫ

**Оркестрация контейнеров** - это процесс автоматизации развертывания, масштабирования и управления контейнеризированными приложениями.

Допустим: у вас есть приложение, разбитое на несколько микросервисов, каждый из которых работает в своем контейнере. Как управлять всем этим? Тут и понадобится оркестрация.

- автоматизация развертывания: вместо ручного запуска каждого контейнера, вы просто предоставляете оркестратору описание вашего приложения и он автоматически запускает все необходимые контейнеры на нужных серверах.
- масштабирование: по мере увеличения нагрузки на ваше приложение, оркестратор автоматически запускает больше контейнеров, чтобы обеспечить его стабильную работу.
- управление ресурсами: оркестратор следит за состоянием контейнеров, перезапускает упавшие, распределяет ресурсы между ними и оптимизирует их работу.
- сеть: оркестратор помогает организовать взаимодействие между контейнерами, устанавливая соединения между ними и обеспечивая их доступность извне.
- безопасность: оркестратор может контролировать доступ к контейнерам, изолировать их друг от друга и применять правила безопасности.
- мониторинг: оркестратор собирает информацию о работе контейнеров и предоставляет инструменты для мониторинга их состояния, производительности и логов.

**Kubernetes** - один из самых популярных оркестраторов. **Kubernetes** (часто сокращается до "K8s") - это система с открытым исходным кодом для оркестрации контейнеров. Она позволяет :

- автоматизировать развертывание, масштабирование и управление приложениями в контейнерах. Вместо ручной настройки и управления каждым контейнером, Kubernetes делает это за вас, предоставляя вам инструменты для автоматизации всех этапов жизненного цикла приложения.
- обеспечивать высокую доступность и отказоустойчивость. Kubernetes следит за состоянием ваших приложений и автоматически перезапускает их, если они выходят из строя. Он также позволяет создавать резервные копии и восстанавливать приложения, если произойдет сбой оборудования.
- обеспечивать масштабируемость и гибкость. Kubernetes позволяет вам легко масштабировать ваши приложения в зависимости от нагрузки, а также обеспечивает возможность добавлять и удалять узлы в кластере без остановки работы.
- управлять ресурсами кластера. Kubernetes распределяет ресурсы (CPU, память, сеть) между приложениями, обеспечивая оптимальную производительность и использование ресурсов.

## **Основные компоненты Kubernetes:**

- **кластер** - это совокупность узлов (nodes), работающих вместе, чтобы запускать и управлять контейнеризированными приложениями.
- Узлы (nodes) - это физические или виртуальные машины, которые составляют кластер. Каждый узел может выполнять одну или несколько ролей
- Мастер-узел (Master Node): отвечает за управление кластером, координацию работы узлов, планирование развертывания приложений, управление ресурсами.
- Рабочий узел (Worker Node): выполняет контейнеризированные приложения, которые запускаются в кластере.
- **Pods**: Базовая единица развертывания в Kubernetes. Каждый pod содержит один или несколько контейнеров.
- Службы (Services): Предоставляют доступ к приложениям, запущенным в кластере, извне.
- Контроллеры развертывания (Deployment Controllers): Управляют развертыванием и обновлением приложений, обеспечивая автоматическое масштабирование и перезапуск.
- Namespaces: Разделяют ресурсы кластера на логические группы, что упрощает управление и изоляцию приложений.

## **Основные компоненты кластера Kubernetes:**

- Kubernetes API Server: предоставляет интерфейс для взаимодействия с кластером, обрабатывает запросы от пользователей и инструментов.
- Kubernetes Controller Manager: управляет состоянием кластера, отслеживает изменения и принимает меры для восстановления желаемого состояния.
- Kubernetes Scheduler: планирует размещение приложений на рабочих узлах, выбирая оптимальный вариант для выполнения задач.
- etcd: хранит конфигурацию кластера и информацию о состоянии.

**Pod** — это базовая единица развертывания в Kubernetes, представляющая собой группу одного или нескольких контейнеров, которые работают вместе и разделяют ресурсы.

## **Характеристики Pods:**

- Среда выполнения приложений: Pods — это место, где ваши приложения работают. Они содержат все необходимые для запуска приложения зависимости, включая образы контейнеров, файлы конфигурации и т.д.
- Единица планирования: Kubernetes планирует Pods на узлах кластера, чтобы обеспечить наилучшую производительность и доступность.
- Объекты Kubernetes: Pods являются объектами Kubernetes, которые могут быть созданы, удалены, масштабированы и управляться ими с помощью инструментов Kubernetes.
- Недолговечные: Pods — это временные объекты. Если узел перезагружается, Pods могут быть перепланированы на другой узел.

**Расширение `.yaml` (или `.yml`)** относится к файлам написанным на языке разметки YAML (YAML Ain't Markup Language). YAML — это формат сериализации данных, который используется для представления структурированных данных в удобочитаемом формате, особенно часто применяется в конфигурационных файлах.

Вот небольшой пример, демонстрирующий, как выглядит файл YAML:

```
version: '3.8'
services:
  web:
    image: nginx
    ports:
      - "80:80"
  db:
    image: postgres
    environment:
      POSTGRES_USER: user
      POSTGRES_PASSWORD: password
```

Этот пример представляет собой файл конфигурации, который описывает два сервиса: веб-сервер на базе Nginx и базу данных PostgreSQL.