

Кластеры. Ресурсы под управлением облачного провайдера

Елисей Ильин



План занятия

1. [Databases](#)
2. [Containers](#)
3. [Итоги](#)
4. [Домашнее задание](#)



Databases

Типы баз данных

On Line Transaction Processing (**OLTP**) — учёт операций

On Line Analytical Processing (**OLAP**) — анализ данных

Типы:

- Реляционные БД, SQL
- Нереляционные, NoSQL
- Хранилище данных (DWH) и аналитика

Реляционные базы данных AWS

Relational Database Service (RDS) доступен в виде инстансов базы данных нескольких типов:

- оптимизированные для работы с памятью
- для высокой производительности
- выполнения операций ввода-вывода

RDS запускается на виртуальных машинах и не является serverless. Доступ на эти VM отсутствует. Патчинг и администрирование — зона ответственности AWS



Реляционные базы данных

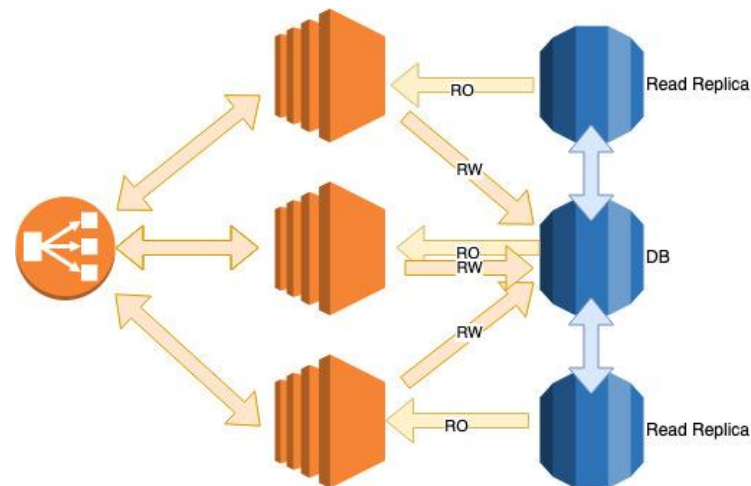
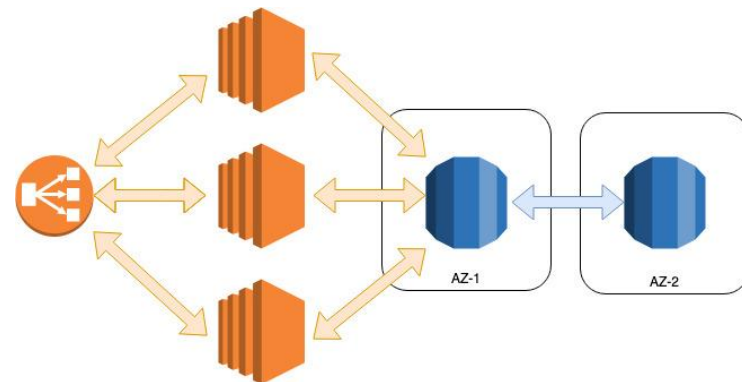
- **Воскуп.** Хранение в S3. Возможно шифрование, в том числе с помощью KMS
 - **Автоматический** с ротацией 1–35 дней (по умолчанию включён). Удаляется при удалении БД
 - **Ручные snapshot** — не удаляется автоматически при удалении БД

При восстановлении dns-имя не меняется автоматически



Реляционные базы данных

- **Multi AZ.** Синхронная репликация на другую AZ. В случае отключения основной БД реплика получает dns-имя основной автоматически. Используется для DR
- **Read Replica.** Асинхронная реплика в копии и доступ на чтение. Используется для масштабирования. Возможно до пяти копий, а также создание read replica самой реплики



Amazon Aurora

Совместимая с MySQL и PostgreSQL реляционная база данных

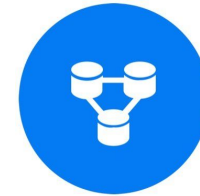
- Работает в 5 раз быстрее, чем MySQL, и в 3 раза быстрее, чем PostgreSQL
- От 10 ГБ с автомасштабированием до 64 ТБ с шагом 10 ГБ
- Вычислительные мощности — до 32 vCPU и 244 ГБ RAM
- Две копии данных на минимум 3 AZ (6 копий данных)
- До 15 Read replica
- Совместимость с MySQL и PostgreSQL

Реляционные базы данных Yandex Cloud

В Yandex Cloud два окружения: **Prestable** и **Production**

- Мажорные и минорные обновления СУБД, а также обновления, касающиеся работы сервиса управляемых БД, сначала попадают в окружение **Prestable**
- В **Production** попадают только проверенные и стабильные обновления

Реляционные базы данных Yandex Cloud



В Yandex Cloud представлены в виде отдельных сервисов Каждый сервис представляет собой кластер **классов хостов**:

- **s* — Standart** — стандартные конфигурации, RAM к количеству vCPU = (4:1)
- **m* — Memory optimised** — конфигурации с увеличенным RAM (8:1)
- **b* — Burstable** — конфигурации с гарантированной долей vCPU ниже 100% (5, 20, 50)

Yandex Database



Распределённая отказоустойчивая NewSQL СУБД, разработанная в Яндексе

- Обеспечивает хранение большого объёма данных с автоматической репликацией
- Поддержка ACID-транзакций между таблицами с нагрузкой, неравномерной по времени
- Горизонтальное масштабирование

Yandex Database



Yandex Database доступна для использования в двух режимах:

- бессерверных вычислений — оплата операций, выполняемых с данными
- с выделенными инстансами — оплата вычислительных ресурсов на почасовой основе

Нереляционные базы данных (NoSQL) AWS

Созданы для определённых моделей данных и обладают гибкими схемами:

- Ключ-значение — (Amazon DynamoDB)
- Документ — (MongoDB, Amazon DocumentDB)
- Графовые БД — (Neptune)
- БД в памяти (ElasticCache) — Redis
- Поисковые БД — (Amazon ElasticSearch Service)
- Big Data — (Redshift)

Amazon DynamoDB

База данных NoSQL, которая поддерживает пары «ключ-значение» и документные модели данных

- Выполняются на SSD
- Кластер на нескольких, географически разнесённых ДЦ
- Чтение потенциально непротиворечивых данных — модель по умолчанию. Увеличивает пропускную способность
- Чтение строго непротиворечивых данных. Чтение меньше 1 сек
- Транзакции ACID

Нереляционные базы данных (NoSQL) YC

Созданы для определённых моделей данных и обладают гибкими схемами:

- Колоночное — (Clickhouse)
- Документ — (MongoDB)
- БД в памяти — (Redis)
- Поисковые БД — (ElasticSearch)
- Big Data — (DataProc, GreenPlum)



Containers

Container services

- Amazon Elastic Container Registry (ECR)
- Amazon Elastic Container Service (ECS)
- Amazon Elastic Kubernetes Service (EKS)

- Yandex Container Registry
- Yandex Serverless Containers
- Yandex Managed Service for Kubernetes

Amazon ECS

Amazon Elastic Container Service — это автоматизированный сервис координирования контейнеров, обеспечивающий развёртывание, администрирование и масштабирование упакованных в контейнер приложений:

- поддержка Docker
- поддерживает API, Scheduling
- поддерживает EC2-модель или AWS Fargate* (по умолчанию)

*AWS Fargate — это ядро для бессерверных вычислений на базе контейнеров. Работает с ECS и EKS

EC2 vs Fargate

Разница между движками на базе EC2 и Fargate

EC2	Fargate
Ответственность за EC2 (control plane, worker nodes)	Serverless (нет ОС)
Установка в своём VPC	Подключение к VPC через ENI
Ценовая модель EC2	Ценовая модель по ресурсам
Несколько контейнеров на одном хосте	Изолированное окружение
Использование на долгий срок	Использование на коротком промежутке времени



Yandex Serverless Containers

- Сервис, позволяющий запускать контейнеризированные приложения в безопасном, отказоустойчивом и масштабируемом окружении **без создания и обслуживания виртуальных машин**
- Сервис выделяет ресурсы, которые необходимы для выполнения приложения
- При увеличении количества вызовов контейнера он автоматически масштабируется

Amazon EKS

Amazon Elastic Kubernetes Service — это управляемый сервис, который позволяет запускать Kubernetes на AWS без необходимости установки и обслуживания собственной плоскости управления Kubernetes или рабочих узлов

- Поддержка экосистемы K8S (Helm, Istio, Envoy, Prometheus, Grafana, Jaeger и т. д.), работает с pod
- Control plane на управлении AWS (scale, security, single tenant)
- Интегрирован с ELB
- Поддерживает EC2-модель или AWS Fargate* (по умолчанию)

[Источник](#)



Yandex Managed Service for Kubernetes

- При использовании создаётся **кластер** и **группы узлов**
- Мастер-ноды, пространство имён, сервис DNS и другие необходимые элементы развёртываются автоматически
- У Yandex Managed Kubernetes есть свой графический интерфейс. Дополнительные расширения не требуются
- Интеграция с Yandex Container Registry



Итоги

Итоги

Сегодня на занятии мы:

- поговорили про различные БД
- изучили инструменты контейнеризации, разобрались в их отличиях друг от друга

Домашнее задание

Ваше домашнее задание можно посмотреть [по ссылке](#)

- Вопросы по домашней работе задавайте **в чате** учебной группы
- Задачи можно сдавать **по частям**
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**

**Задавайте вопросы и
пишите отзыв о лекции!**

Елисей Ильин