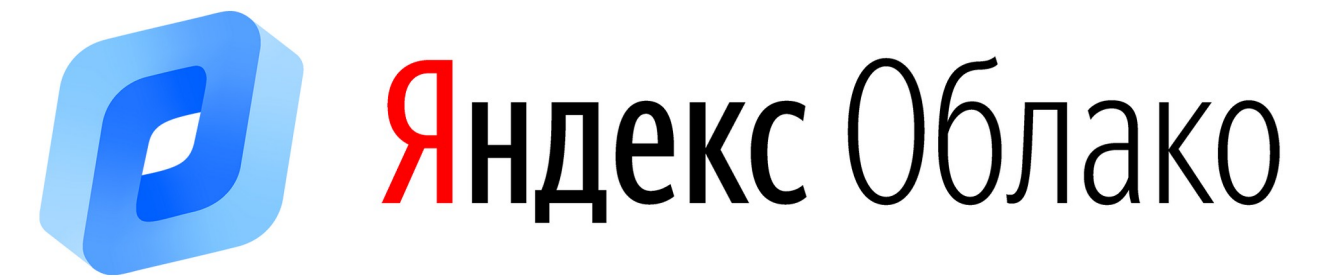


**Я**ндекс



# Вебинар: Yandex Database

Олег Бондарь

# Содержание

- 1 | Что такое Yandex Database (YDB)
- 2 | Как выглядит Yandex Database
- 3 | Примеры использования YDB в Яндексе
- 4 | Тестирование производительности (YCSB)

Что такое Yandex Database

# История развития хранилищ данных



# Ограничения классических СУБД



- Пропускная способность ограничена производительностью одной машины
- Вертикальная масштабируемость
- Нет гибкости расширения

# Ограничения классических СУБД



- Ручное шардирование
- Переключение мастера
- Компромиссы консистентности

# Компромиссы распределённых NoSQL СУБД



- Урезанная функциональность
- Упрощенная модель данных (KV)
- Множество проприетарных API



# Компромиссы распределённых NoSQL СУБД



- Компромисс между доступностью и консистентностью
- Компромисс между производительностью и консистентностью
- Отсутствие распределенных транзакций

# Как мы пришли к созданию Yandex Database?



- Высокие требования внутри Яндекса к отказоустойчивости сервисов
- Сервисы Яндекса должны переживать выход из строя одного из ДЦ
- Прогнозируемые задержки и высокая пропускная способность
- Eventual consistency усложняет код приложения

# Как мы пришли к созданию Yandex Database?



- › Надёжность, репликация данных
- › Доступность, отсутствие единой точки отказа
- › Горизонтальная масштабируемость
- › Использование обычных серверов
- › Гибкое изменение ресурсов

# Решаемые задачи



## > Online-нагрузка (OLTP)

- Чтение / запись
- Большой TPS, прогнозируемые задержки

## > ACID транзакции

## > Диалект SQL для написания запросов

## > Табличная модель

РСУБД

## > Отказоустойчивость

## > Высокая доступность

NoSQL

## > Масштабируемость по нагрузке

Как выглядит YDB



PREVIEW

# Yandex Database

Yandex Database (YDB) — это распределённая отказоустойчивая NewSQL СУБД. YDB обеспечивает высокую доступность и масштабируемость, и, в то же время, строгую консистентность и поддержку ACID-транзакций. Для запросов используется декларативный SQL-подобный язык запросов YQL.

YDB может являться альтернативой имеющимся решениям в следующих случаях:


- В системах, которым необходима поддержка ACID-транзакций между таблицами с нагрузкой, неравномерной по времени.
- В системах, требующих обработки и хранения большого объёма данных.
- При разработке нового продукта, для которого уже на этапе дизайна с высокой вероятностью ожидается большая нагрузка (тысячи запросов в секунду к БД) и требование строгой консистентности.
- В качестве бэкенда для баз данных с плохо предсказуемой нагрузкой.

[Запросить доступ](#)[Документация](#)


# Запросите доступ

[cloud.yandex.ru/services/ydb](https://cloud.yandex.ru/services/ydb)

< Folder

Yandex Database Service 

 Databases

 Operations

Documentation

[Getting started with YDB \(Ru\)](#)

[YDB concepts \(Ru\)](#)

[YDB best practices \(Ru\)](#)

[YQL tutorials \(Ru\)](#)

[YQL reference \(Ru\)](#)



## ← New database

Name 

### Computing resources

<b>s1.large</b>	<b>s2.large</b>
vCPU 16 cores	vCPU 16 cores
Memory 64 GB	Memory 64 GB

Number  1 6

### Storage groups

Disk type   HDD  SSD


Number of groups   1 6 x 100 GB


### Network

Cloud network   or

### Subnets

ru-central1-a   or

ru-central1-b   or

ru-central1-c   or



< Folder

### Yandex Database



Service

 Databases

 Operations

Documentation

[Getting started with YDB \(Ru\)](#)

[YDB concepts \(Ru\)](#)

[YDB best practices \(Ru\)](#)

[YQL tutorials \(Ru\)](#)

[YQL reference \(Ru\)](#)



## Databases

Create a database

Filter by name

All statuses

Name

Status

ydb-s2-3nodes

 Running

ydb-3s1-3ssd

 Running



Yandex Database

ydb-3s1-3ssd

Overview

Navigation

Monitoring

Root folder

Filter by name Create

Name	Owner
episodes	ajegq762fp6sr4lpaha0@as
my_table	ajegq762fp6sr4lpaha0@as
seasons	ajegq762fp6sr4lpaha0@as
series	ajegq762fp6sr4lpaha0@as
usertable	ajegq762fp6sr4lpaha0@as

usertable Table

Schema

#	Name	Key	Charset
0	Id	P	
1	field0		Utf8
2	field1		Utf8
3	field2		Utf8

- Copy path
- Paste path to a query
- Use as template
- Create table
- Alter table
- Select data
- Insert data
- Delete

Query

```

1 SELECT      -- выбрать данные
2 *          -- выбрать все колонки из таблицы
3 FROM episodes; -- таблица, из которой нужно выбрать данные
4 COMMIT;

```

Run Completed

Result #1

#	air_date	episode_id	season_id	series_id	title
0	13182	1	1	1	"Yesterday's Jam"
1	13182	2	1	1	"Calamity Jen"
2	13189	3	1	1	"Fifty-Fifty"
3	13196	4	1	1	"The Red Door"
4	13203	5	1	1	"The Haunting of Bill Crouse"
5	13210	6	1	1	"Aunt Irma Visits"
6	13384	1	2	1	"The Work Outing"

Documentation

Getting started with YDB (Ru)

YDB concepts (Ru)

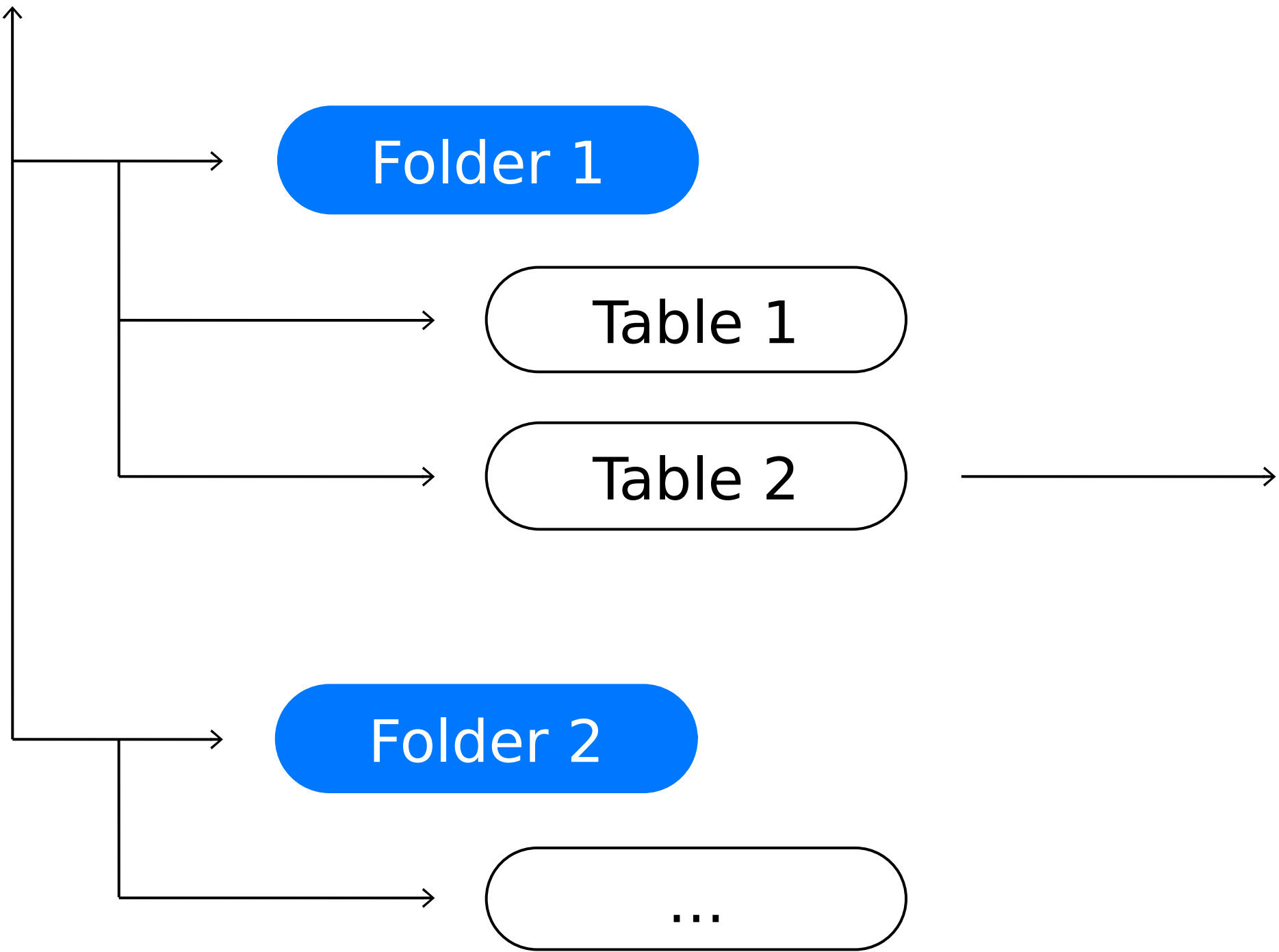
YDB best practices (Ru)

YQL tutorials (Ru)

YQL reference (Ru)

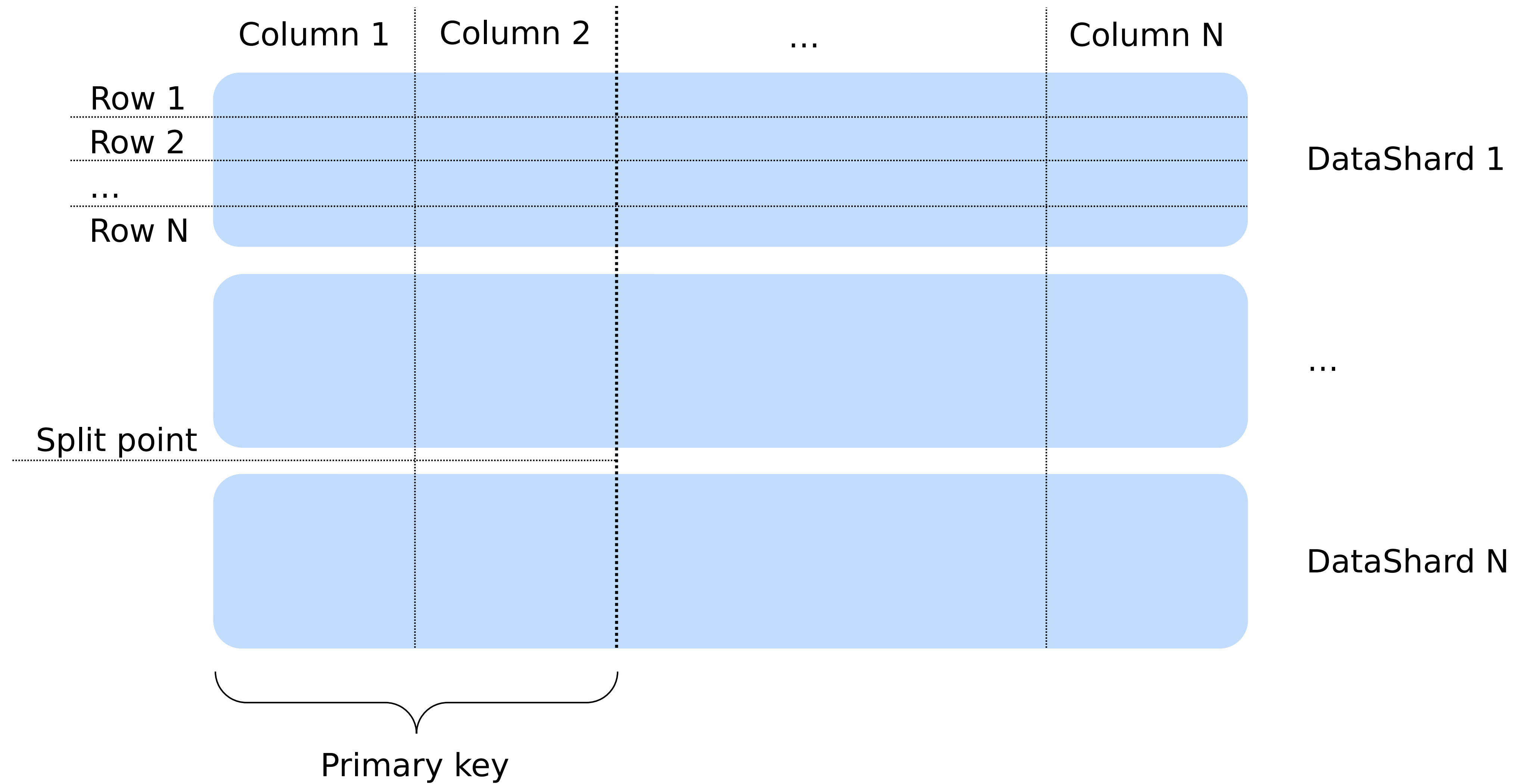
# Схема базы данных

Database

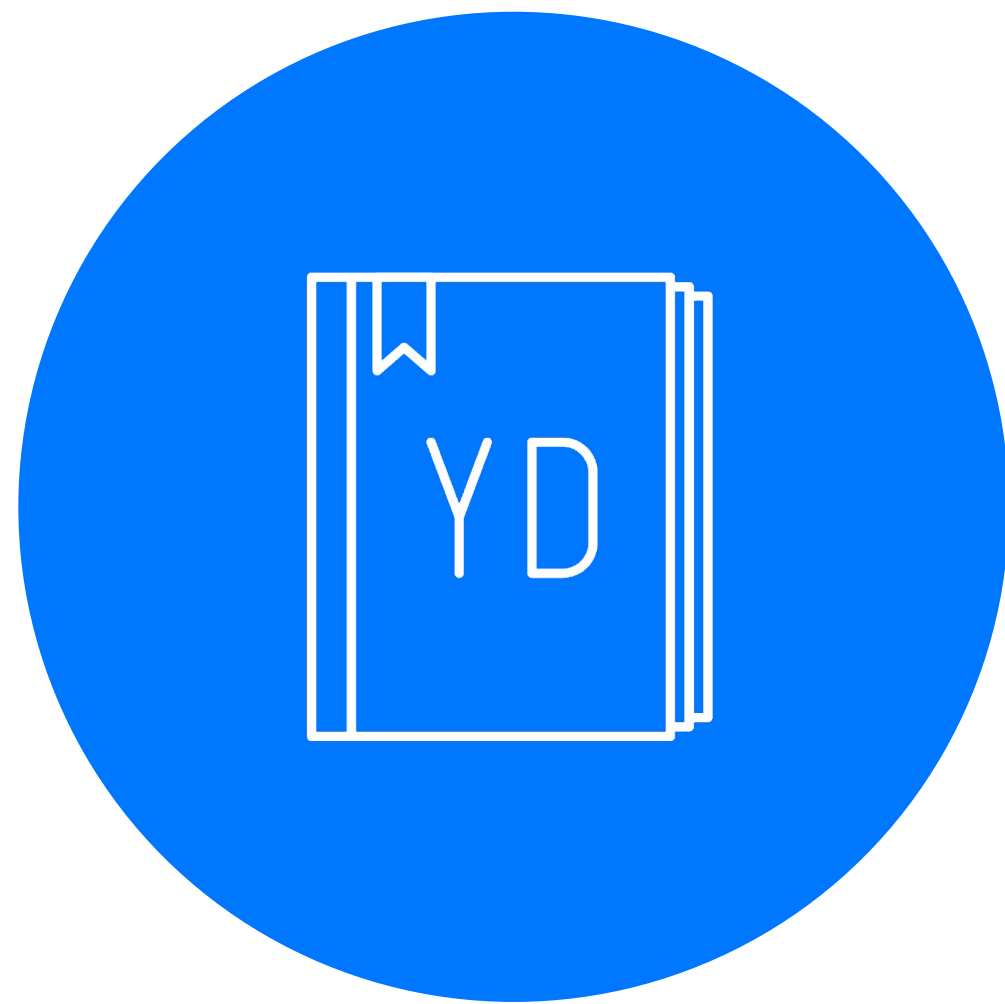


- Transactional DB
- > ACL
  - > Access time
  - > Size
  
  - > Columns
  - > Primary key
  - > Shards
  - > Statistics
  - > Configuration

# Таблицы

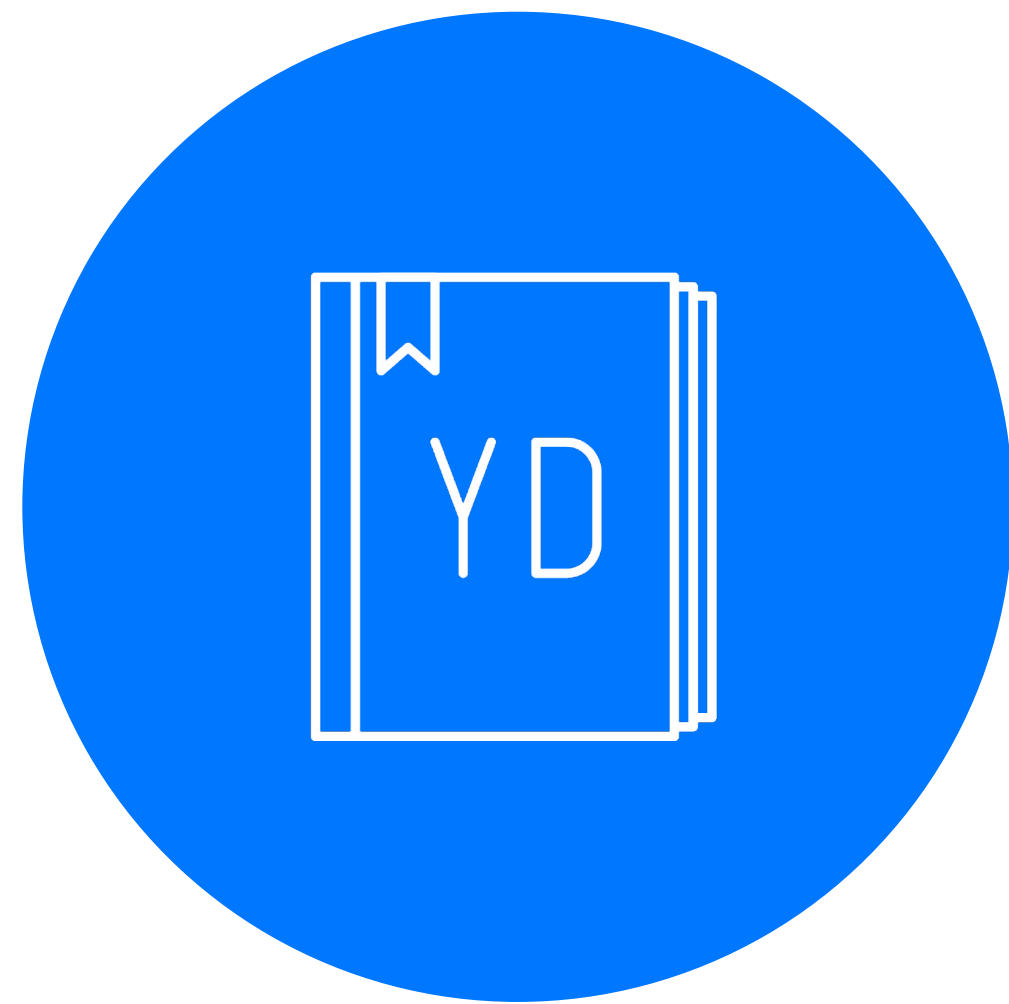


# Клиентское взаимодействие



- Yandex Database — сервис Яндекс.Облака
- › Клиентские библиотеки для популярных языков программирования
  - Python
  - Go
  - Java

# Клиентское взаимодействие



- › Клиентские библиотеки реализуют
  - Пул подключений
  - Кеш подготовленных запросов
  - Балансировку соединений на стороне клиента
  - Автоматическое обнаружение серверной топологии
  - Аутентификацию
  - Обработку ошибок

# Yandex Query Language (YQL)



- › Диалект SQL
- › Строгая типизация
- › Именованные подзапросы
- › Явная параметризация

# Yandex Query Language (YQL)



- Стандартные DML-конструкции:
  - SELECT, INSERT, UPDATE, DELETE
  - JOIN, GROUP BY, ORDER BY
  
- Специализированные DML-конструкции:
  - UPSERT / REPLACE
  - UPDATE ON
  - DELETE ON
  
- Привычные DDL-конструкции
  - CREATE TABLE
  - DROP TABLE
  - ALTER TABLE

# Yandex Query Language (YQL)



## > Встроенные функции

- DateTime
- Regexp
- Math
- String
- Unicode
- IP
- Digest



# Yandex Query Language (YQL)

```
DECLARE $name AS String;
```

```
$name = "Foo" || $name;
```

```
SELECT Id FROM Users WHERE Name = $name;
```

```
INSERT INTO Users (Id, Name) VALUES (1, "Bar");
```

# Примеры использования Yandex Database в Яндексе

# Yandex Database — сценарии использования

Яндекс

 Яндекс Услуги



 Яндекс Репетитор



 Яндекс Коллекции

# Yandex Database — сценарии использования

- › Обработка сотен тысяч транзакций в секунду с прогнозируемыми задержками
- › Катастрофоустойчивость с высоким уровнем автоматизации
- › Неравномерная по времени OLTP-нагрузка, требующая возможностей быстрого масштабирования
- › Для систем, достигших предела возможностей РСУБД или NoSQL-решений

# Пример использования: Турбо-страницы поиска Яндекс



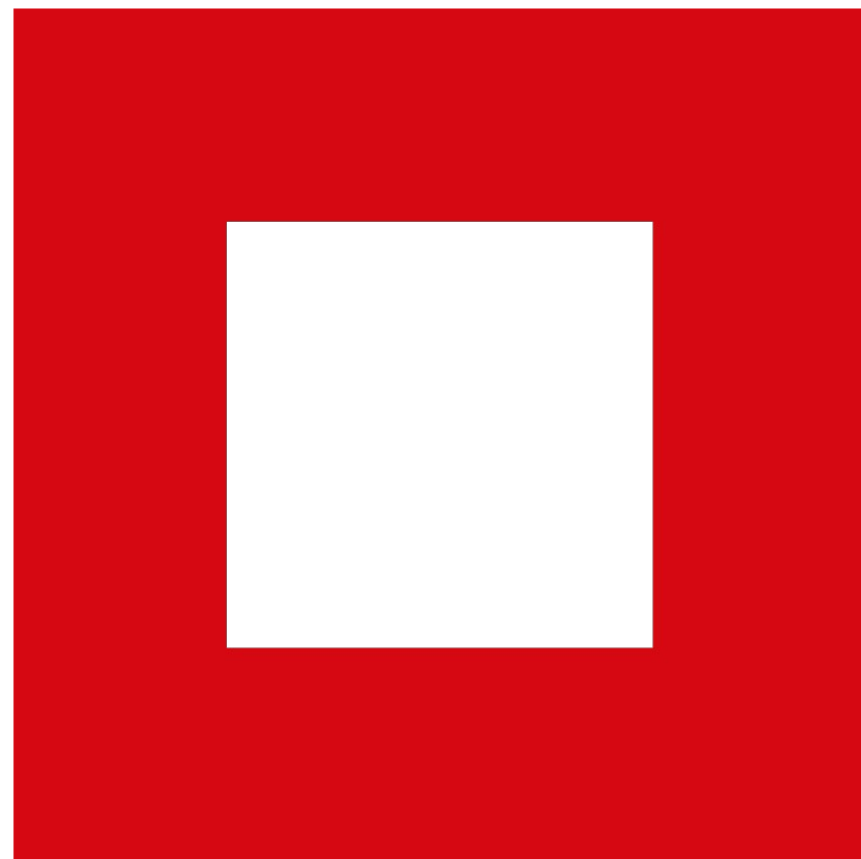
- › Необходимость переживать потерю одного из дата-центров
- › Read-Write транзакции
- › Большие объемы данных
- › Большой TPS
- › Задержки не более 50 ms для 99% запросов

# Пример использования: Яндекс.Коллекции



- › Необходимость переживать потерю одного из дата-центров
- › Read-Write транзакции для получения и обновления истории
- › Большие объемы данных и большой TPS
- › Задержки не более 20 ms для 99% запросов

# Пример использования: Яндекс Репетитор



- Быстрый рост
- Гибкая масштабируемость ресурсов
- Привычные подходы к проектированию схемы СУБД и запросов

# Пример использования: Яндекс Услуги



- Быстрый рост
- Привычные подходы к проектированию схемы СУБД и запросов
- Возможность гибкого расширения базы



# Пример использования: Алиса



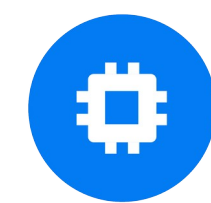
- Переход с MongoDB
- Использование Yandex Database как сервис
- Высокие требования к задержкам
- Большая нагрузка
- Высокие требования к доступности

# Пример использования: IoT



- › Использование Yandex Database как сервис
- › Высокие требования к задержкам
- › Высокие требования к доступности

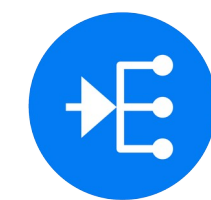
# Пример использования: Яндекс.Облако



Compute Cloud, NBS — виртуальные машины и диски



Monitoring — сервис для сбора и визуализации метрик приложений



Load Balancer — сетевые балансировщики нагрузки

# Пример использования: Яндекс.Облако

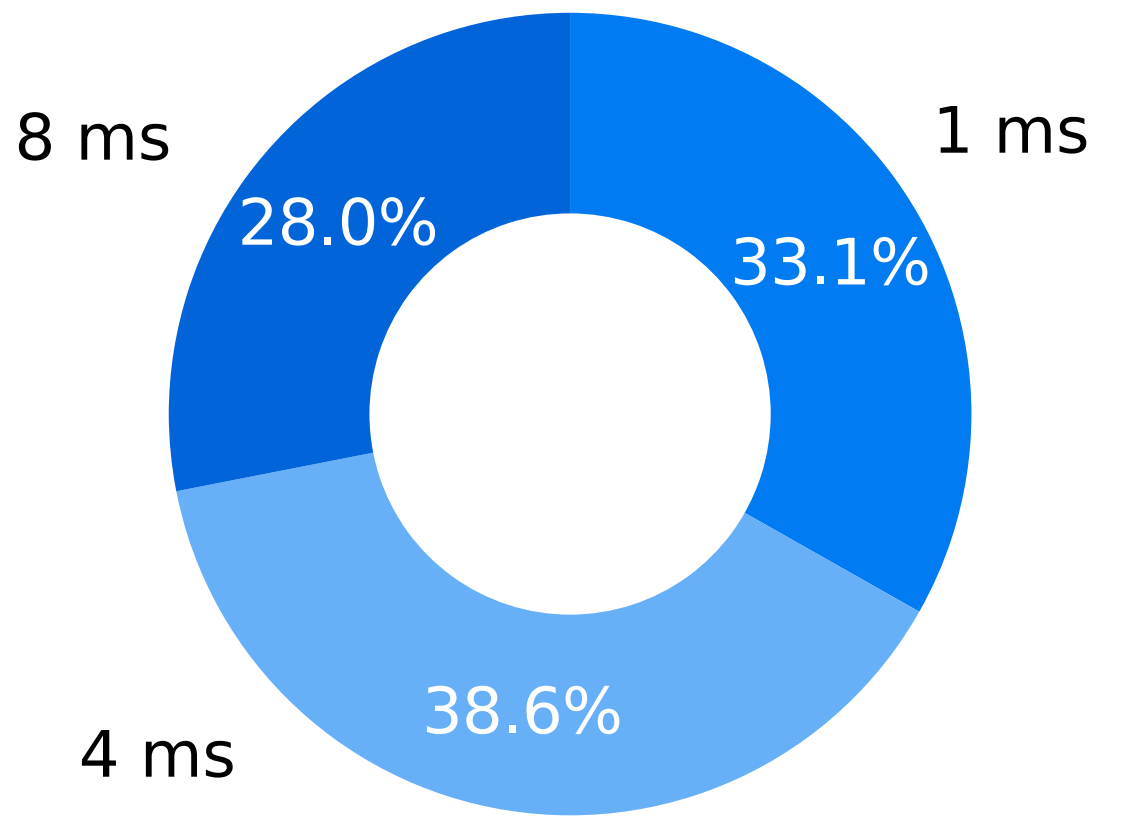
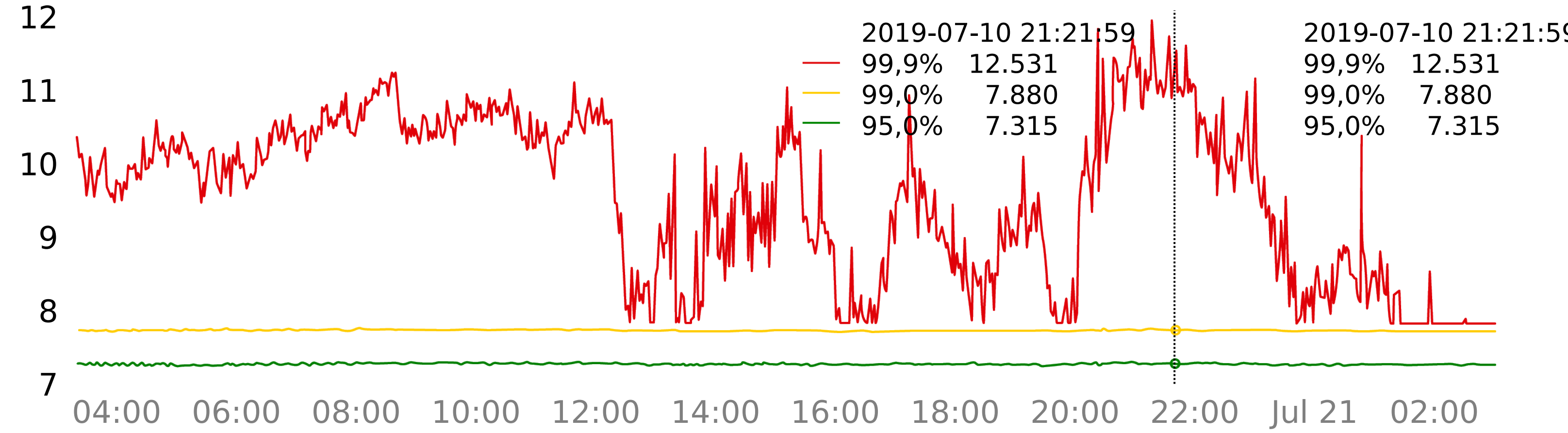
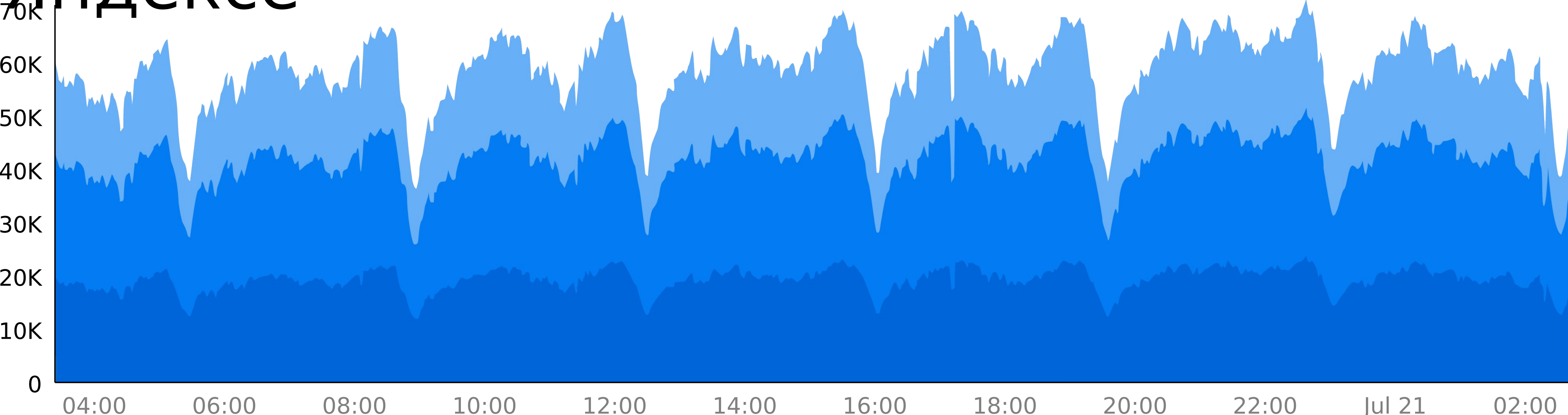


Yandex Object Storage —  
масштабируемое хранилище данных



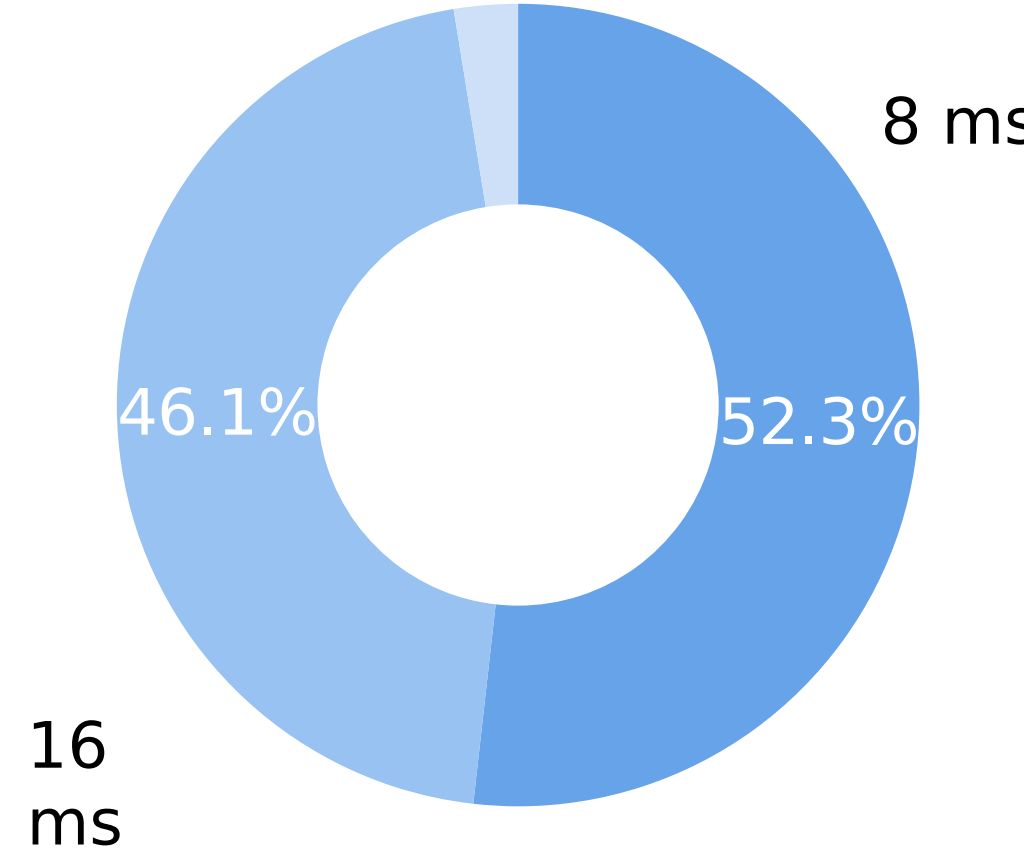
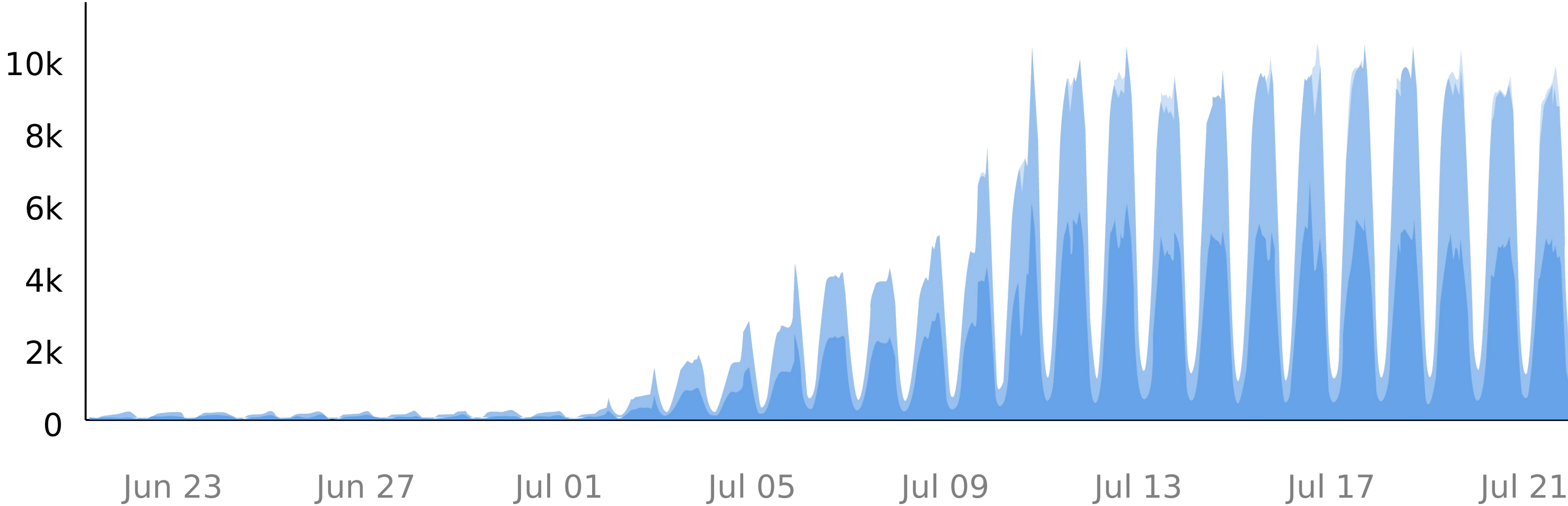
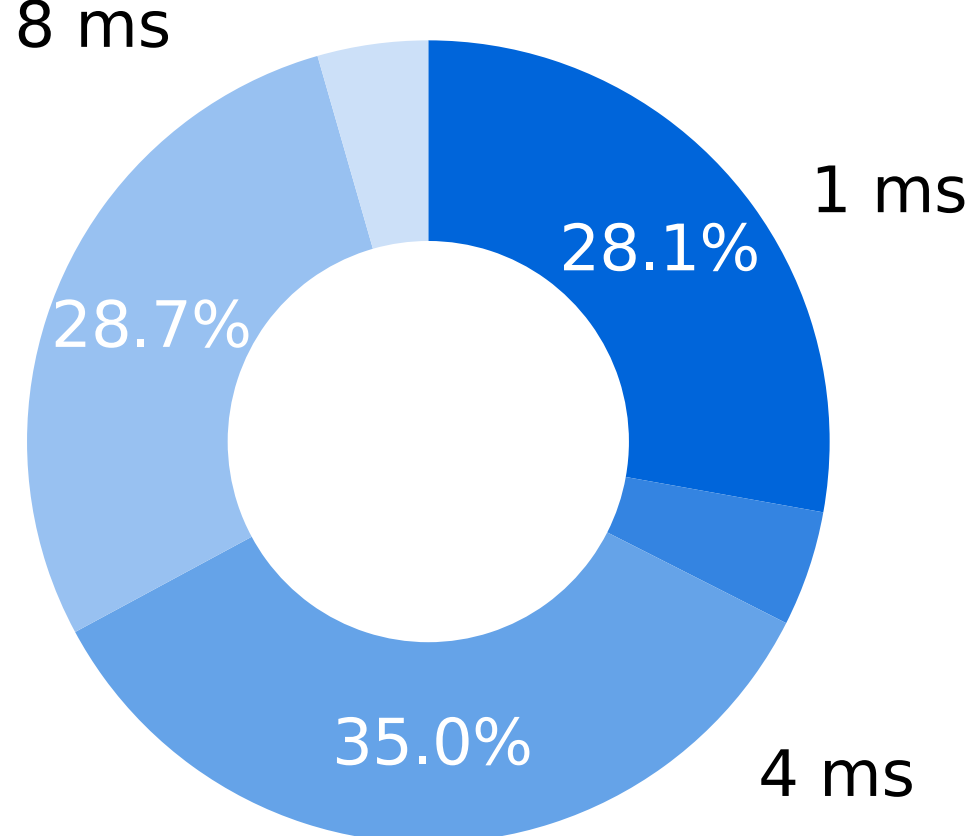
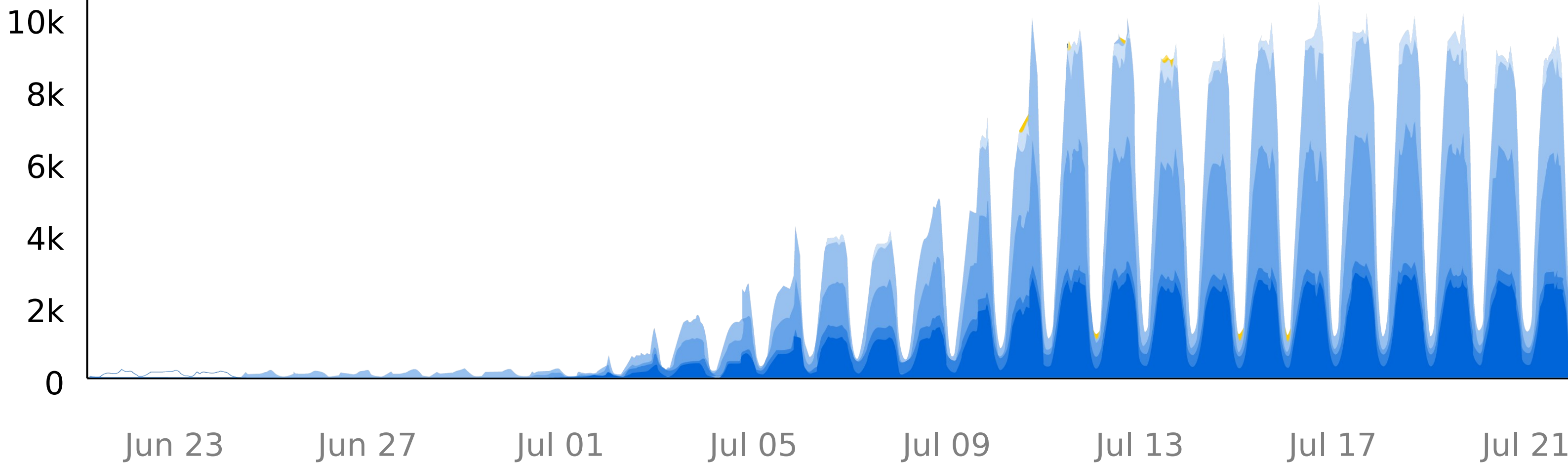
Yandex Message Queue — очереди для  
обмена сообщениями между  
приложениями

# Yandex Database — использование в Яндексе





# Yandex Database — использование в Яндексе



# Тестирование производительности (YCSB)

# Yahoo! Cloud Serving Benchmark (YCSB)



- Общепризнанный стандарт для тестирования NoSQL-нагрузки
- Opensource спецификация и инструменты для бенчмарков
- Простые и одновременно распространенные паттерны нагрузки
- Выпущен в 2010 году и поддерживает 50+ различных систем
- Замеры задержек и пропускной способности на клиенте



# Особенности запуска YCSB



- › Версия 0.17 от 4 июня 2019 без каких-либо патчей
- › Драйвер для Yandex Database разработан на основе SDK, остальные из стандартной поставки YCSB
- › Никакого тюнинга специально под бенчмарк
- › СУБД используются в режиме наиболее строгой консистентности
- › По 10 миллионов и записей в таблицах, и операций в запуске
- › 48 потоков на клиенте

# Рассматриваемые СУБД



- Yandex Database
- MongoDB 4.0  
(Yandex Managed Service for MongoDB)
- Redis 5.0  
(Yandex Managed Service for Redis™)

# Используемое оборудование (характеристики)



- Виртуальные машины в Яндекс.Облаке на Intel Cascade Lake
- Клиент: 48 ядер, 96Гб RAM
- Серверная часть с СУБД:
  - 3 ноды в разных зонах доступности
  - 16 ядер\*, 64Гб RAM, network NVME диск\*\*

- \* Кроме Redis
- \*\* SSD диск для YDB

# Паттерны нагрузки



## > Workload A:

- 50% read, 50% update
- Пример: хранилище сессий, записывающих последние действия пользователей

## > Workload B:

- 95% read, 5% update
- Пример: разметка любых объектов тегами

## > Workload C:

- 100 % read
- Пример: кеш профилей пользователей, которые строятся в другой системе

# Паттерны нагрузки



## > Workload D:

- Чтение недавно вставленных данных
- Пример: обновления статусов пользователей социальных сетей

## > Workload E:

- Чтение коротких диапазонов
- Пример: ветки постов / комментариев на форумах

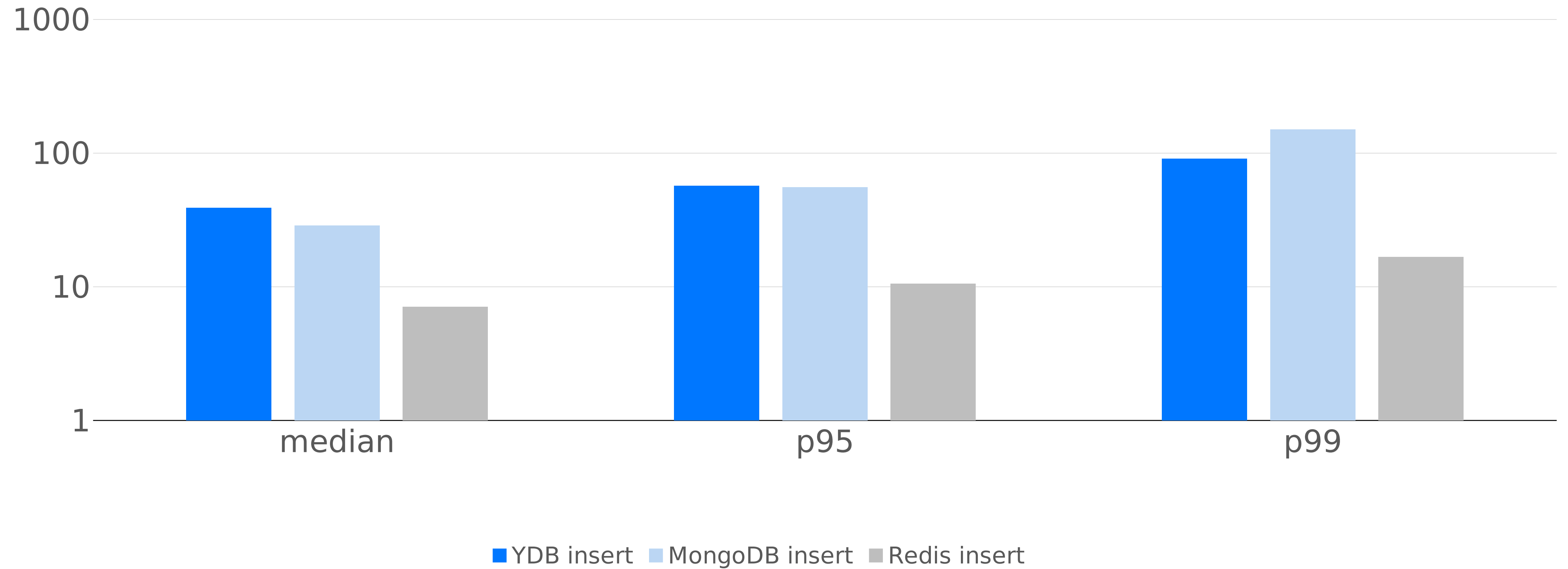
## > Workload F:

- Read-modify-write
- Пример: база данных пользователей с возможностью редактировать свой профиль из интерфейса

# Workload A:

50% read, 50% update

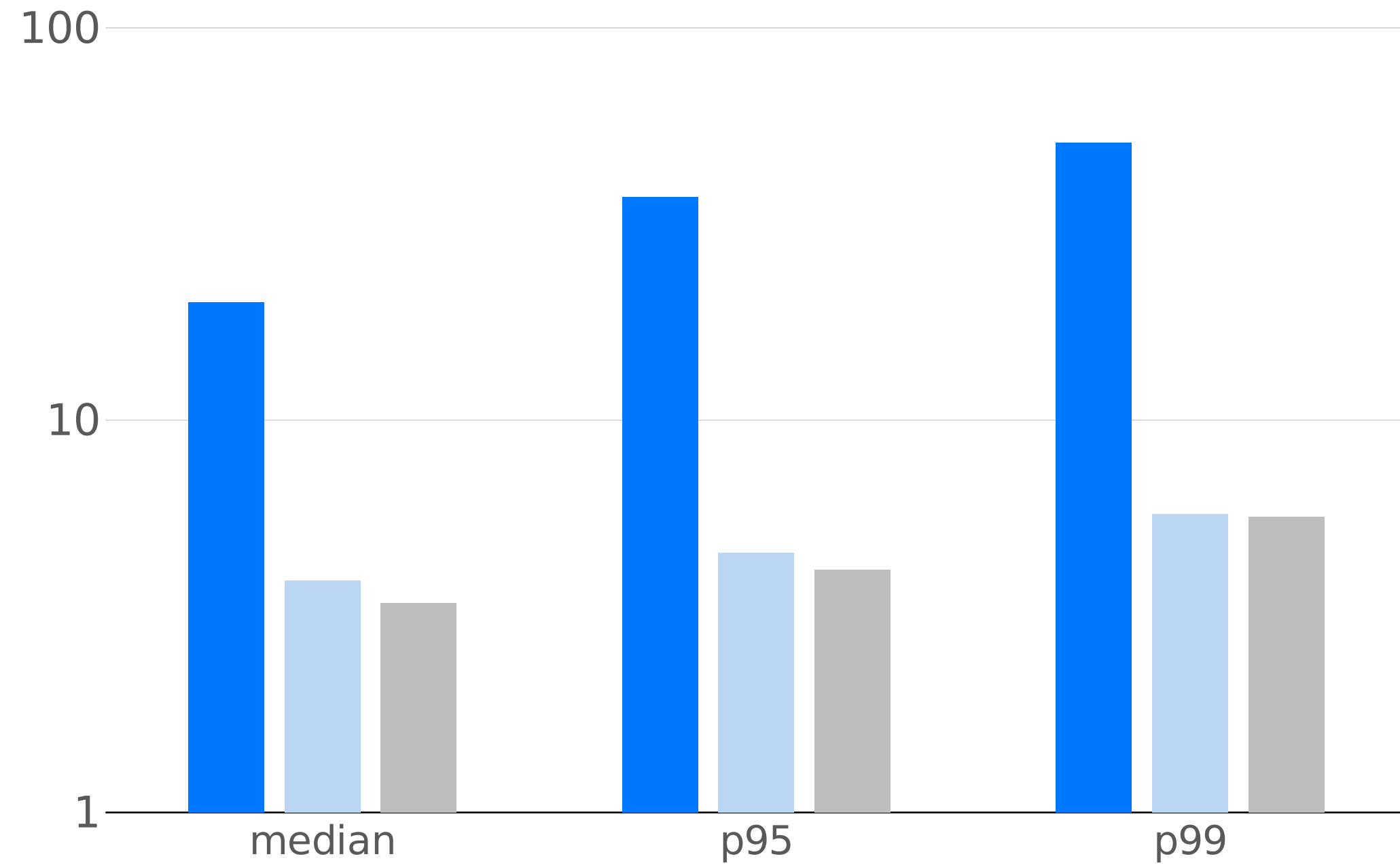
Insert



# Workload A:

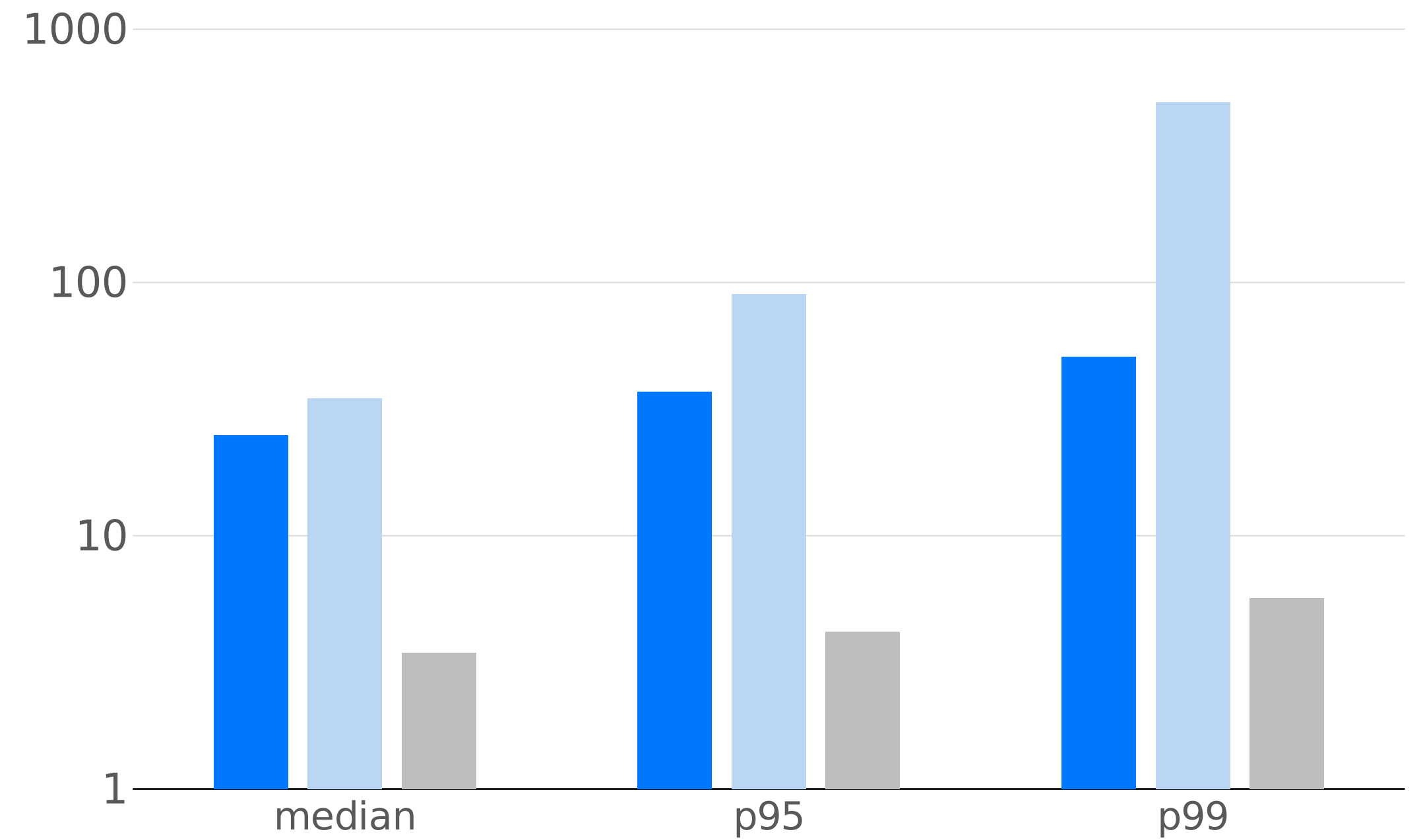
50% read, 50% update

## Read



■ YDB read ■ MongoDB read ■ Redis read

## Update

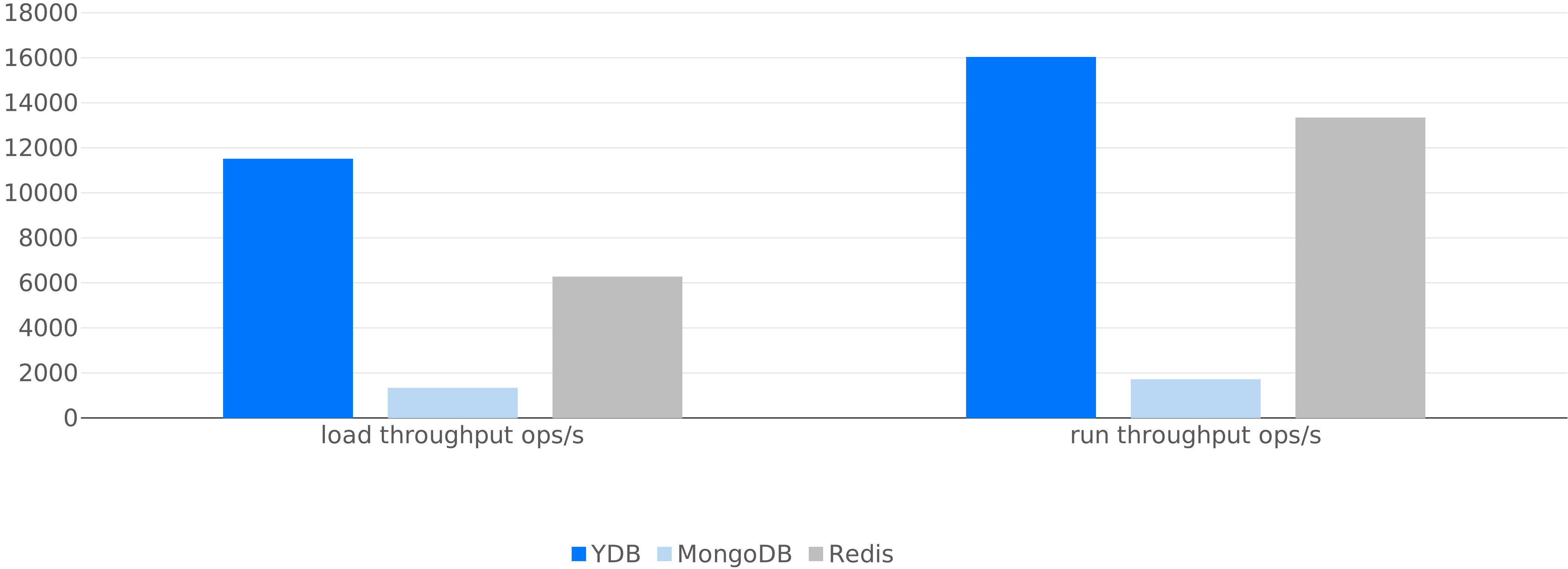


■ YDB Update ■ MongoDB update ■ Redis update

# Workload A:

50% read, 50% update

### Throughput

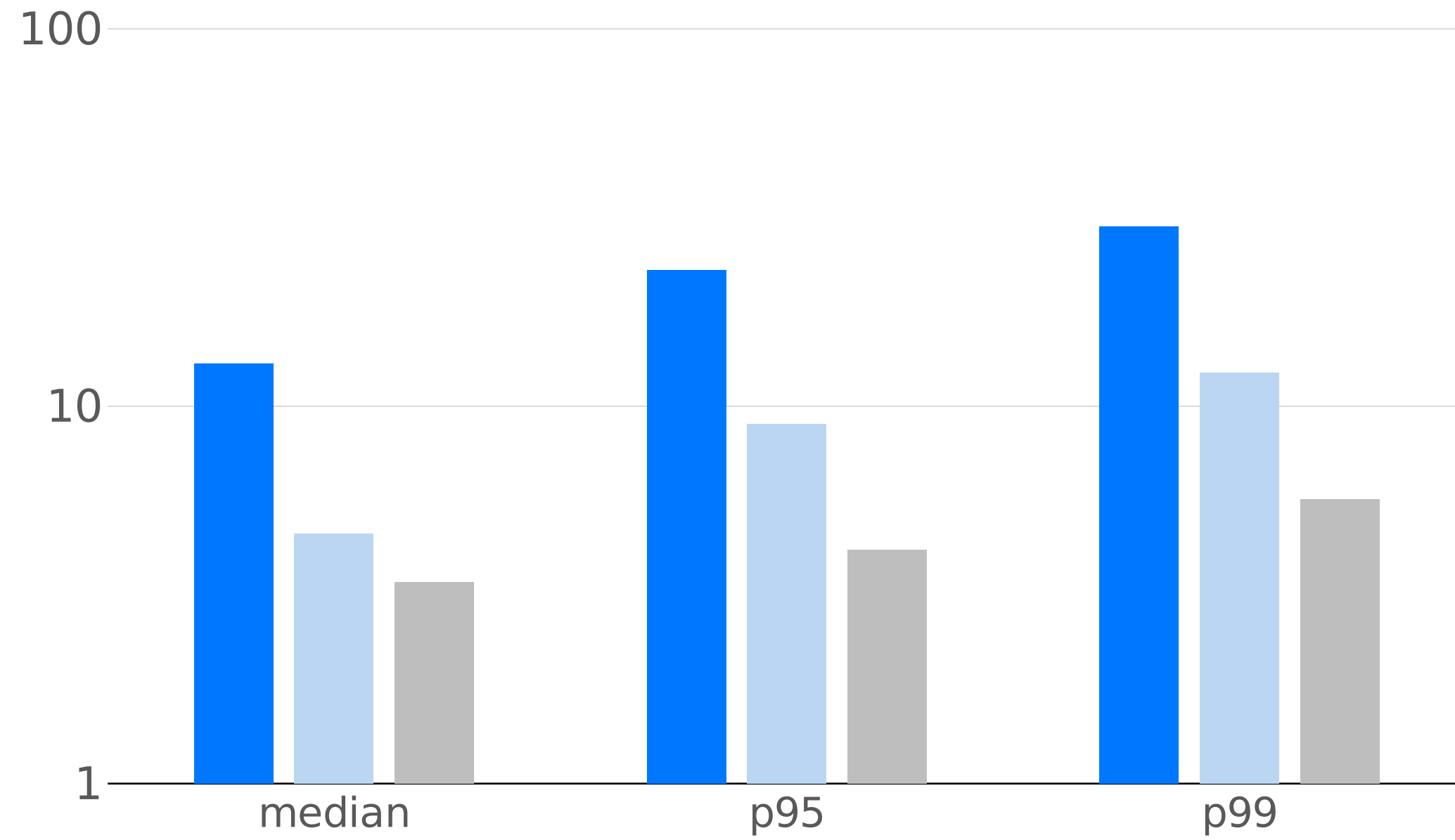




# Workload B:

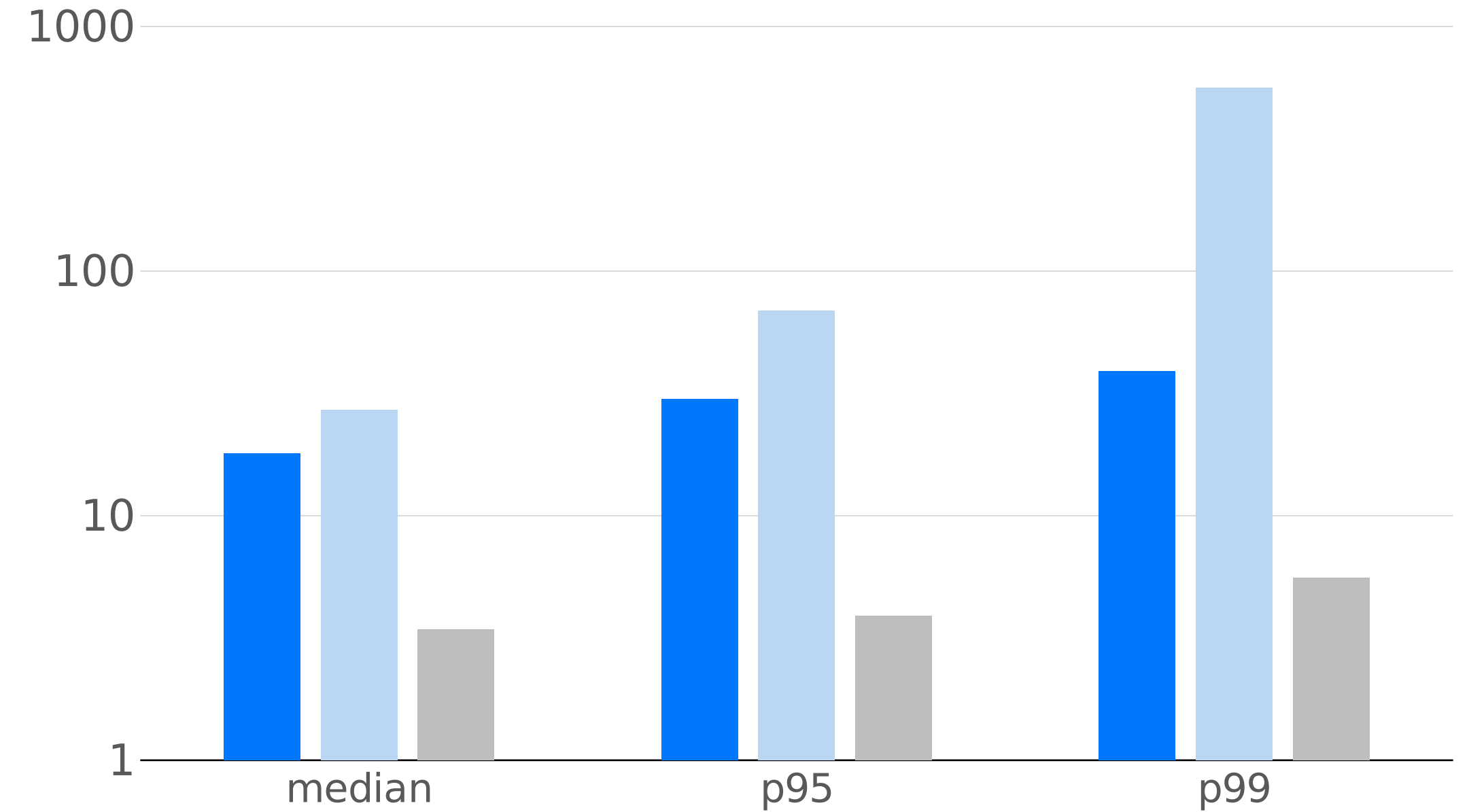
95% read, 5% update

## Read



■ YDB read ■ MongoDB read ■ Redis read

## Update

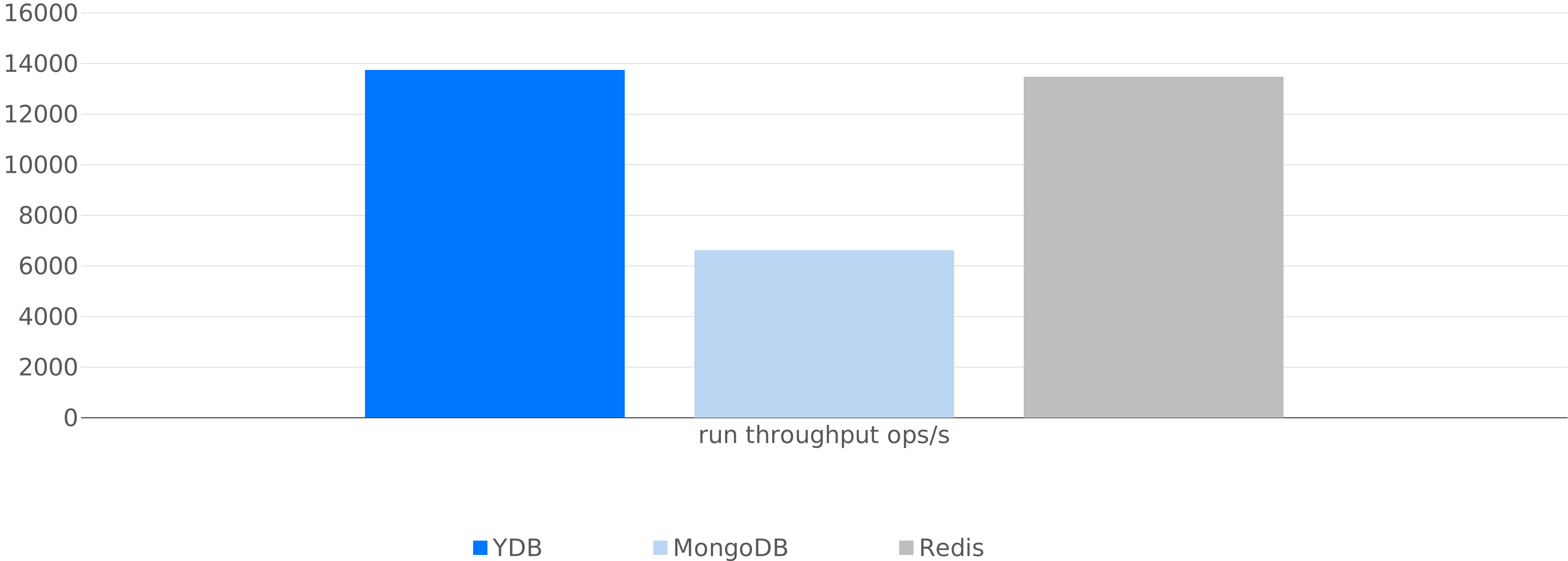


■ YDB Update ■ MongoDB update ■ Redis update

# Workload B:

95% read, 5% update

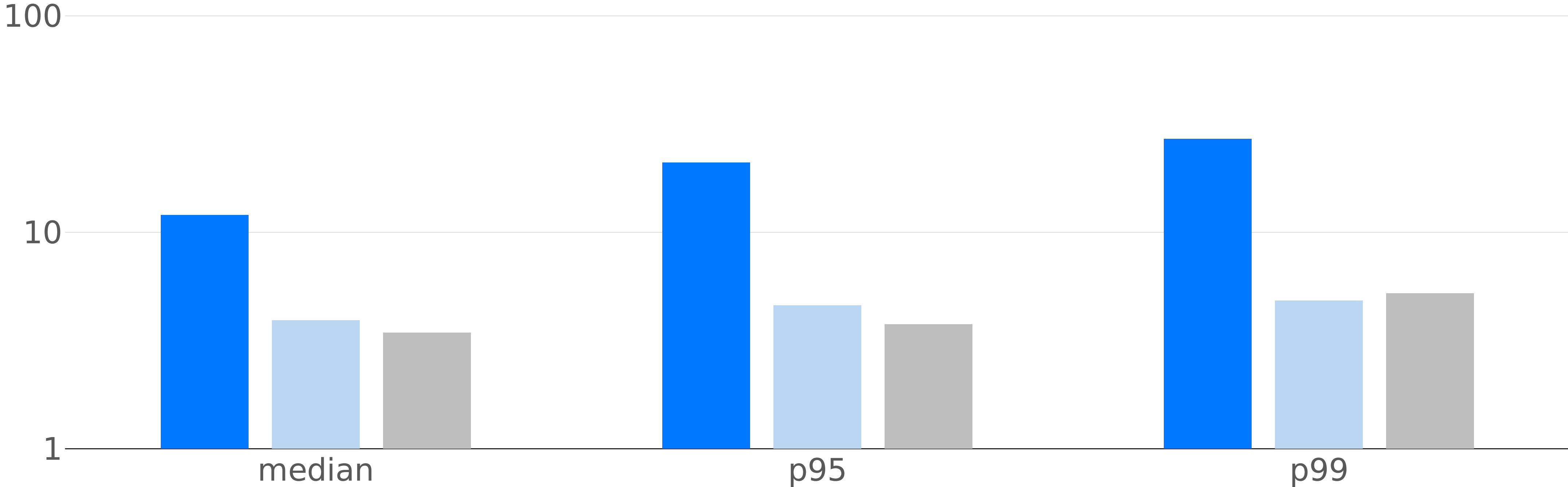
Throughput



# Workload C:

100% read

Read

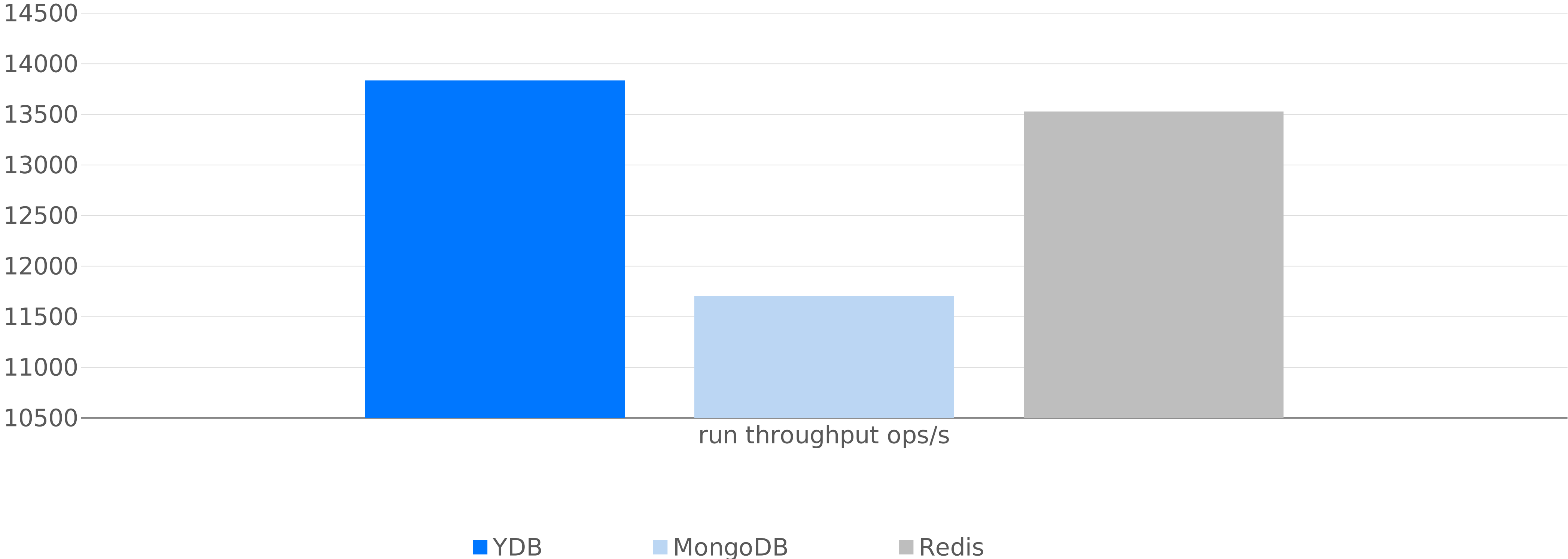


■ YDB read ■ MongoDB read ■ Redis read

# Workload C:

100% read

### Throughput



Что мы получили в итоге  
внутри YDB?

# Заключение



Yandex Database — геораспределённая отказоустойчивая база данных с механизмом строго консистентных транзакций

- Направлена на решение задач, требующих надёжного хранения данных и высокой доступности
- Предоставляет SQL-like интерфейс для выполнения запросов
- Горизонтально масштабируется по нагрузке
- Предоставляется как сервис в Яндекс.Облаке

# Yandex Database



- › Надёжное хранение данных с автоматической репликацией
- › Отказоустойчивость, автоматическое восстановление от сбоев
- › Распределённые ACID-транзакции с serializable-уровнем изоляции транзакций
- › Высокая пропускная способность при прогнозируемом времени отклика
- › Горизонтальная масштабируемость до тысяч нод

# ССЫЛКИ

Yandex Database Private Preview

[cloud.yandex.ru/services/ydb](https://cloud.yandex.ru/services/ydb)

Python SDK

[github.com/yandex-cloud/ydb-python-sdk](https://github.com/yandex-cloud/ydb-python-sdk)

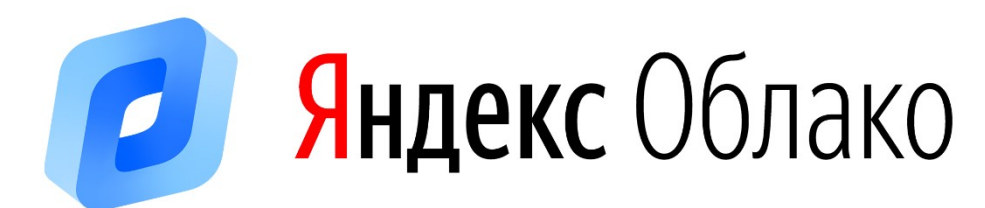
Go SDK

[github.com/yandex-cloud/ydb-go-sdk](https://github.com/yandex-cloud/ydb-go-sdk)

Java SDK

[github.com/yandex-cloud/ydb-java-sdk](https://github.com/yandex-cloud/ydb-java-sdk)





# Спасибо!

Олег Бондарь

 olegbondar@yandex-team.ru