

# Yandex Database



Яндекс Облако

# Yandex Database

— это распределённая отказоустойчивая СУБД с поддержкой распределённых ACID-транзакций.

В основу дизайна системы заложены принципы и подходы, позволившие взять лучшее от традиционных РСУБД — транзакционность и SQL, а также от так называемых NoSQL-решений — геораспределённость, высокую доступность, большую пропускную способность и горизонтальную масштабируемость.

# От реляционных СУБД к NewSQL

Десять лет назад основным массовым решением для хранения и обработки данных были классические РСУБД, такие как MS SQL, Oracle, PostgreSQL или MySQL. С одной стороны, эти решения имеют ряд преимуществ: годами проверенные технологии, поддержка транзакций (ACID) и стандартов SQL, огромное количество всевозможных инструментов и надстроек, устоявшаяся экосистема. С другой стороны, РСУБД часто были ограничены производительностью одной машины, а масштабирование представляло собой сложную и дорогостоящую задачу, решение которой обычно предполагало использование высокопроизводительных серверов.

Поэтому с ростом объёма обрабатываемых данных появился ряд систем, которые часто обобщают термином NoSQL. Эти системы не предъявляют высоких требований к оборудованию и легко масштабируются с использованием серверов среднего ценового сегмента.

Самые известные и доступные представители семейства NoSQL — Cassandra, HBase и MongoDB. Особенности этих решений: высокая доступность, высокая пропускная способность, горизонтальная масштабируемость и поддержка нереляционных данных, например, JSON-документов. К основным недостаткам NoSQL систем можно отнести ограниченную поддержку транзакций, часто присущую таким системам ослабленную консистентность данных, отсутствие поддержки OLAP и декларативного языка запросов.

Перечисленные недостатки NoSQL-систем иногда приводят к усложнению логики клиентских приложений. Поэтому

следующим логичным шагом стало совмещение преимуществ, предлагаемых классическими РСУБД и NoSQL-решениями, что привело к появлению нового класса систем, которые иногда объединяют при помощи термина NewSQL.

К NewSQL относится широкий спектр решений — начиная с простого добавления к NoSQL-системам декларативного языка запросов с похожим на SQL синтаксисом, как CQL в Cassandra, заканчивая системами, которые предоставляют строгую консистентность и поддержку транзакций.

В Яндексе таким решением — базой данных нового поколения, предоставляющей масштабируемость NoSQL, но со строгой консистентностью, поддержкой транзакций и мощностью декларативного языка запросов — является Yandex Database.

# Почему Yandex Database

Исторически многие приложения в Яндексе использовали широкий спектр систем хранения и обработки данных — от традиционных РСУБД и некоторых NoSQL решений до NoSQL-систем для управления большими данными собственной разработки.

Разработчики приложений, использующие классические РСУБД, были всем довольны, пока их приложения не становились «большими». В этом случае им требовалось решение, в котором сохранятся основные привычные свойства РСУБД и будет обеспечена возможность масштабирования.

В Яндексе на NoSQL-системах построено большое количество критически важных приложений. Разработчики некоторых из них находили, что код приложений существенно усложняется из-за отсутствия в NoSQL некоторых возможностей: схематизации, строгой консистентности совместно с географической распределённостью, транзакций между записями из разных таблиц или декларативного языка запросов.

Как правило, выбор СУБД для проекта связан с компромиссом между масштабируемостью и консистентностью. Yandex Database позволяет избежать такого выбора, так как даёт возможность получить лучшее от реляционных баз данных с их таблицами, транзакциями, привычным SQL, строгой схемой и совместить это с возможностью горизонтального масштабирования NoSQL-решений. В дополнение к полученному сочетанию Yandex DB предлагает такие возможности как ACID-транзакции между разными таблицами, строгую консистентность и распределённость между разными дата-центрами или регионами.

Yandex Database горизонтально масштабируется и позволяет обрабатывать данные с низкими задержками. При этом обеспечивается высокая доступность, необходимая сервисам Яндекса (99,999 %). С Yandex DB вы можете создать хороший задел на будущее для своего приложения и избежать головной боли, которая обычно появляется и усиливается с ростом популярности приложения и сложностями масштабирования остальных решений под нагрузкой. В числе прочего, с ней не придётся переписывать много кода приложения для миграции с одной СУБД на другую.

# Обзор Yandex Database

Yandex Database — это горизонтально масштабируемая распределённая СУБД, разработанная и развёрнутая в Яндексе. Система развивается в течение последних четырёх лет и используется в ряде критически важных сервисов Яндекса в качестве высокопроизводительной OLTP СУБД.

В частности, сервисы Яндекс.Облака Yandex Object Storage и Yandex Block Storage используют Yandex DB для хранения данных и базируются на её компонентах.

Yandex DB может работать как однодатацентровая или как геораспределённая (кросс-ДЦ) база данных. Обеспечивает автоматическую репликацию данных и высокую доступность с автоматической обработкой отказов вычислительных узлов и дата-центров. Также Yandex DB поддерживает автоматическое перешардирование данных при увеличении их количества.

Для пользователя Yandex DB выглядит как традиционный сервер баз данных, взаимодействие с которым построено на базе протокола gRPC. При написании своего приложения можно использовать клиентские библиотеки или SDK, доступные для популярных языков программирования (C++, java, Python). Спецификация протокола общения открыта. Это позволяет реализовать SDK для любого языка программирования. Для удобства построения веб-сервисов в Yandex DB предоставляет RESTful API.

Yandex DB поддерживает привычную для пользователей РСУБД модель данных и оперирует таблицами

с predetermined схемой. Для удобства организации таблиц поддерживается создание директорий по аналогии с файловой системой.

Основное средство формирования команд к базе данных — декларативный язык запросов YQL, представляющий собой один из диалектов SQL. Таким образом, пользователю предлагается мощный и, в то же время, привычный способ взаимодействия с системой.

В Yandex DB поддерживаются высокопроизводительные распределенные ACID-транзакции, которые могут затрагивать несколько записей из разных таблиц. Предоставляется самый строгий уровень изоляции транзакций — serializable. Также имеется возможность ослабления уровня изоляции в пользу увеличения производительности.

В дизайн Yandex DB заложена поддержка разных сценариев нагрузки, таких как OLTP и OLAP. В текущей реализации поддержка аналитических запросов ограничена. Поэтому можно говорить, в первую очередь, что Yandex DB — это OLTP-база данных.

## Сравнение Yandex DB с РСУБД и NoSQL

	Yandex Database	NoSQL	Реляционные
Поддержка схемы	✓ Да	—	✓ Да
Auto failover	✓ Да	✓ Да	Настраиваемая
Auto repair	✓ Да	✓ Да	—
Распределённые транзакции	✓ Да	—	—
ACID-транзакции	✓ На уровне БД	Построчно	✓ На уровне БД
Консистентность	✓ Strong	Eventual	✓ Strong
Поддержка SQL	✓ YQL	—	✓ Да
Масштабируемость	✓ Автоматическая Горизонтальная	✓ Горизонтальная	Вертикальная
Репликация	✓ Автоматическая	✓ Автоматическая	Настраиваемая



# Для каких задач подойдёт Yandex Database

Yandex Database совмещает в одном продукте подходы и технологии, привычные разработчикам баз данных и разработчикам высоконагруженных распределённых систем. От баз данных Yandex DB унаследовала SQL-подобный язык запросов, распределённые ACID-транзакции, схематизацию данных. От распределённых систем — автоматическое горизонтальное масштабирование, автоматическое шардирование, автоматическое восстановление после сбоев.

Yandex Database целесообразно рассматривать как альтернативу имеющимся решениям для следующих задач:

- Поддержка ACID-транзакций между таблицами с нагрузкой, неравномерной по времени. Например, обслуживание распродаж (чёрная пятница, киберпонедельник) для крупных проектов или плохо прогнозируемые всплески трафика по новостным поводам.
- Обработка и хранение большого объёма данных. Например, задача по кэшированию результатов поисков авиабилетов, туров, которая встречается у онлайн-турагентств или агрегаторов туров, или задача по пересчёту цен на сезон, с которой сталкиваются туристические операторы.
- Разработка нового продукта, для которого уже на этапе дизайна с высокой вероятностью ожидается большая нагрузка (тысячи запросов в секунду к БД) и требование строгой консистентности.

- Бэкенд для хранения и обработки данных, собираемых с непредсказуемого количества устройств с плохо предсказуемой нагрузкой (различные сценарии IoT).
- Значительная переработка существующего продукта, связанная с тем, что применённая РСУБД или NoSQL решение перестали удовлетворять требованиям по нагрузке, доступности или консистентности, и есть планы по замене БД.
- Реализация плана по disaster recovery, в котором предусмотрено размещение существующей БД в нескольких ДЦ с автоматическим переключением. В некоторых случаях такая задача может потребовать сильного изменения логики приложения, сравнимого с переездом на новую СУБД.