

Модели данных

Выбор

- Это решение определяет, насколько система будет производительной, масштабируемой, гибкой и в конечном итоге — успешной.
- Неверный выбор модели данных подобен выбору неправильного фундамента для здания

Реляционная

Особенности

- Организует данные в виде двумерных таблиц.
- Используют структурированный язык запросов (SQL) для запросов.
- ACID - свойства

Реляционная

Преимущества

- Благодаря механизму ключей, ограничений и транзакций обеспечивается целостность и непротиворечивость данных.
- Использование SQL как стандарта облегчает разработку и поддержку приложений, а также миграцию данных между системами.
- Процесс нормализации позволяет устранить избыточность данных и аномалии

Реляционная

Недостатки

- Изменение структуры таблиц на работающей системе может быть сложной и дорогой операцией.
- Горизонтальное масштабирование реляционных баз данных зачастую сложнее реализовать по сравнению с некоторыми NoSQL решениями
- Сложные JOIN-запросы на огромных таблицах могут выполняться медленно.
- Неэффективна для хранения иерархических (древовидных) или графовых данных.

Реляционная

Применение

- OLTP-системы
- Системы с высокими требованиями к целостности
- Данные четко структурированы

Реляционная Реализации

- Postgres
- MySQL
- SQLite
- MariaDB
- Microsoft SQL Server
- Oracle DB
- YDB (Распределенная СУБД)

Колоночная

Особенности

- Вместо того чтобы хранить каждую запись целиком, данные организуются по столбцам.
- Относятся к семейству NoSQL

Колоночная

Преимущества

- Высокая производительность аналитических запросов
- Эффективное сжатие данных
- Улучшенная производительность ввода-вывода
- При горизонтальном масштабировании почти линейное увеличение производительности

Колоночная

Недостатки

- Операции, характерные для OLTP-систем, работают менее эффективно
- При частых изменениях данных могут возникать дополнительные сложности в синхронизации и обеспечении целостности данных между столбцами.

Колоночная Реализации

- Cassandra
- ScyllaDB
- Google BigQuery
- ClickHouse (колоночная реляционная СУБД)

Документориентированная

Преимущества

- Возможность изменения структуры документов без необходимости модифицировать общую схему базы данных
- Документы в формате JSON или BSON естественным образом отображаются на объекты в ЯП
- Простота горизонтальной масштабируемости

Документоориентированная

Недостатки

- Ограничения целостности данных
- Сложность запросов
- Потенциальная избыточность

Документоориентированная Применение

- Хранения логов
- OLTP-системы
- Разработка MVP
- Данные неструктурированы
- Обработка больших объемов данных

Графовая

Особенности

- Вместо традиционных таблиц, как в реляционных базах данных, здесь основными объектами являются: узлы, ребра, свойства
- Относятся к семейству NoSQL

Графовая

Преимущества

- Запросы, направленные на поиск путей, обход графа или анализ связей, выполняются очень быстро.
- Нет жестко фиксированной схемы
- Легко добавлять новые типы связей и узлов без необходимости пересмотра всей структуры данных

Ключ-значение

Особенности

- Данные организованы в виде пар «ключ-значение»
- Значение может быть произвольного типа

Ключ-значение

Преимущества

- Операции получения, вставки и удаления данных осуществляются за $O(1)$
- Быстрые и требуют мало ресурсов

Ключ-значение

Недостатки

- Не подходят для сложных запросов, агрегаций или выборок по значениям внутри хранимых данных
- Не поддерживается сложное моделирование связей между данными

Ключ-значение

Применение

- Кэширование данных
- Хранение сессионных данных
- Реализация очередей
- Хранение конфигурационных данных

Ключ-значение

Реализации

- Redis
- Memcached
- Etcd
- Kafka (Брокер сообщений)

Временные ряды

Преимущества

- Оптимизация на высокую скорость записи
- Позволяет легко делать агрегация и интерполяция
- Хранение больших объемов данных
- Быстрый доступ к данным за определённые временные интервалы

Временные ряды

Применение

- Мониторинг и логирование
- Финансовые рынки
- Интернет вещей

Векторная

Особенности

- Данные сохраняются в виде векторов вещественных чисел

Векторная

Преимущества

- Позволяют быстро выполнять запросы поиска похожих векторов благодаря специальным индексам и алгоритмам (напр. Hierarchical Navigable Small World)

Векторная

Применение

- Рекомендательные системы
- Поиск по смыслу
- Кластеризация

Векторная Реализации

- Pinecone
- Milvus

Хранилище данных

Особенности

- предназначены для работы с файлами, такими как документы, изображения, видео, аудиофайлы.
- располагаются в каталогах и подкаталогах
- Хранит помимо файла - метаданные
- Доступ к файлам осуществляется по их именам и путям в файловой системе

Хранилище данных

Преимущества

- Интеграция с операционными системами и стандартными программами для работы с файлами
- Позволяет выстраивать сложные модели доступа
- Позволяет разворачивать CDN (Content Delivery Network)

Хранилище данных

Недостатки

- Поиск данных осуществляется по именам файлов и метаданным, а не по содержимому

Хранилище данных

Применение

- Хранение слабоструктурированной информации (картинки, видео, документы, чертежи, аудио)

Хранилище данных

Реализации

- Amazon S3
- MinIO