



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ  
УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

---

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И  
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**  
**Кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль) «Технологии разработки программного обеспечения»  
форма обучения – очная

## **АНАЛИТИЧЕСКИЙ ОТЧЕТ**

по теме «Технологии баз данных (Database Engineering)»

Обучающегося 4 курса  
Петелина Ивана Андреевича

Научный руководитель:  
кандидат физико-математических наук,  
доцент кафедры ИТиЭО  
Власов Дмитрий Викторович

Санкт-Петербург  
2025

## ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ	2
ВВЕДЕНИЕ	3
ТЕОРЕТИЧЕСКИЕ ОСНОВЫ БАЗ ДАННЫХ	5
1.1 Понятие и назначение баз данных	5
1.2 Эволюция технологий хранения данных	6
МОДЕЛИ И АРХИТЕКТУРЫ БАЗ ДАННЫХ	8
2.1 Реляционные базы данных и SQL	8
2.2 NoSQL и нереляционные базы данных	9
2.3 Графовые базы данных	9
СОВРЕМЕННЫЕ ТЕХНОЛОГИИ И ПОДХОДЫ В DATABASE ENGINEERING	11
3.1 Распределенные и облачные базы данных	11
3.2 Архитектурные подходы хранения данных	11
3.3 Интеграция БД с прикладными системами	12
ЗАКЛЮЧЕНИЕ	14
СПИСОК ЛИТЕРАТУРЫ	15

## ВВЕДЕНИЕ

В современных условиях цифровой трансформации ключевую роль в функционировании информационных систем играют технологии баз данных. Практически все программные продукты, от корпоративных информационных систем и веб-приложений до аналитических платформ и решений в области больших данных - опираются на механизмы хранения, обработки и извлечения информации. Термин база данных охватывает множество структур и подходов к организации данных, включая реляционные, графовые, документно-ориентированные и другие типы хранилищ, которые классифицируются по модели данных и принципам хранения информации. Существует более 50 разновидностей баз данных по различным критериям классификации, что подчеркивает многообразие подходов в данной области .

Рост объёмов данных, генерируемых пользователями, устройствами интернета вещей и цифровыми сервисами, приводит к постоянному усложнению задач хранения и управления информацией. В таких условиях традиционные реляционные базы данных с фиксированными схемами оказываются недостаточно гибкими для современных сценариев: большие данные, многопользовательские системы и распределенные приложения требуют иных архитектур и методов обработки информации. Это обусловило появление и развитие NoSQL-технологий, класса систем управления базами данных, позволяющих работать с неструктурированными и распределенными данными, обладающих гибкими схемами хранения и высокой масштабируемостью [1].

Database engineering как прикладная область знаний охватывает задачи проектирования, реализации и оптимизации систем управления базами данных, а также обеспечение их надежности, доступности и безопасности. Данная область включает выбор модели данных, разработку схем, организацию доступа, обеспечение согласованности транзакций и интеграцию СУБД с прикладными приложениями. Современные исследования подчеркивают, что инженерные подходы к базам данных должны учитывать не только функциональные требования, но и нефункциональные характеристики, такие как производительность, масштабируемость и надежность.

Эволюция технологий баз данных отражает развитие вычислительных систем: от ранних иерархических и сетевых моделей к доминированию реляционной модели, основанной на теории множеств и поддерживающей язык SQL, которая долгое время оставалась стандартом в отрасли. В реляционных СУБД данные хранятся в таблицах, и

специфические свойства данных формализуются перед началом работы, что обеспечивает строгую структуру, но ограничивает масштабируемость [2].

С ростом распределённых вычислений, облачных решений и потребности обработки больших объёмов данных возникли модели NoSQL, которые предлагают гибкие схемы данных и способны эффективно масштабироваться горизонтально через распределение нагрузки между узлами. NoSQL-технологии включают документно-ориентированные, key-value, колоночные и графовые базы данных, каждая из которых оптимизирована под определенные сценарии. Эти подходы дополняют возможности классических реляционных систем и находятся в фокусе современных исследований и разработок в области database engineering [3].

Актуальность данного исследования обусловлена также возрастающими требованиями к обеспечению целостности, доступности и безопасности данных в условиях сложных вычислительных сред, включая распределённые и облачные системы. Масштабируемость обработки данных, а также обеспечение согласованности и отказоустойчивости становятся ключевыми задачами для современных информационных систем, что делает изучение технологий баз данных не только теоретически значимым, но и практически востребованным [4].

Целью данной работы является проведение анализа состояния развития технологий баз данных на основе изучения научной литературы, учебных материалов и профессиональных информационных ресурсов. В рамках этого анализа предполагается обобщить современные подходы к организации хранения данных, рассмотреть основные модели и архитектуры СУБД, выявить ключевые проблемы и обозначить перспективные направления развития database engineering.

Для достижения поставленной цели в работе решаются следующие задачи:

- анализ теоретических основ технологий баз данных
- рассмотрение эволюции моделей хранения данных
- изучение современных архитектурных решений и подходов к проектированию СУБД

Полученные результаты могут служить основой для дальнейших исследований и практических разработок в области управления данными.

# ТЕОРЕТИЧЕСКИЕ ОСНОВЫ БАЗ ДАННЫХ

## 1.1 Понятие и назначение баз данных

База данных (БД) — это упорядоченная совокупность данных, организованных таким образом, чтобы обеспечить эффективное хранение, поиск и обработку информации. В широком смысле база данных представляет собой структурированное хранилище фактов и показателей, которые описывают элементы предметной области и взаимоотношения между ними. При этом данные в базе независимы от конкретных программ, использующих их для решения прикладных задач, что отличает БД от обычных наборов файлов или локальных таблиц данных.

Профильные источники подчеркивают, что БД обладает следующими ключевыми особенностями: структурированность информации, возможность поиска по заданным критериям, обеспечение целостности данных и их независимость от прикладных программ. Совокупность таких характеристик делает базу данных важным компонентом информационных систем любой сложности [5].

С развитием вычислительной техники появилась специализированная категория программного обеспечения — система управления базами данных (СУБД). СУБД представляет собой комплекс языковых, программных и технических средств, предназначенных для создания, управления, администрирования и обеспечения доступа к базам данных. Такая система позволяет пользователям и прикладным приложениям эффективно выполнять операции по вставке, обновлению, удалению и выборке данных, а также обеспечивает контроль целостности, безопасность и согласованность информации.

База данных и СУБД обычно рассматриваются вместе как единая информационная система управления данными: база данных обеспечивает структуру и содержание информации, а СУБД - механизм управления этими данными. Например, международная энциклопедия Britannica определяет СУБД как программный комплекс, который управляет хранением и доступом к данным, решая задачи безопасности, консистентности и производительности [6].

Назначение баз данных выходит за рамки простого хранения информации. Их основной задачей является обеспечение удобного и эффективного доступа к данным для широкого круга пользователей и приложений, поддержка одновременной обработки запросов в многопользовательской среде, а также поддержание целостности и

согласованности информации в условиях регулярного обновления. Кроме того, современные СУБД обеспечивают механизмы резервного копирования, восстановления после сбоев, контроля доступа и управления транзакциями, что критически важно для устойчивой работы бизнес-приложений и распределенных систем.

Таким образом, базы данных и СУБД являются фундаментальными компонентами информационных систем: базы данных формируют структурированное представление данных, а системы управления обеспечивают эффективную работу с этими данными на протяжении всего жизненного цикла информационной системы.

## **1.2 Эволюция технологий хранения данных**

Развитие технологий баз данных отражает изменение потребностей информационных систем и вычислительной техники в целом. На ранних этапах информатизации, в 1960-х - 1970-х годах, для обработки данных широко использовались иерархические и сетевые модели данных. Эти модели ориентировались на жестко структурированные форматы, в которых данные хранились в виде иерархий или сложных сетевых связей, а доступ к ним осуществлялся через заранее определенные пути навигации. Однако такие подходы оказались малоэффективными для сложных многопользовательских систем, требующих гибкого доступа к информации и независимости данных от программного кода [7].

С появлением реляционной модели, предложенной на основе работ таких ученых как Э. Кодд и развиваемой в последующем, база данных стала рассматриваться как совокупность таблиц с четко определёнными отношениями между ними. Реляционные СУБД получили широкое распространение благодаря способности формализовать данные при помощи таблиц, поддерживать нормализацию, обеспечивать согласованность транзакций (ACID) и использовать стандартизованный язык запросов SQL для манипулирования данными. Эти свойства сделали реляционные системы доминирующей технологией в области хранения и управления данными на протяжении нескольких десятилетий.

Тем не менее, рост объемов обрабатываемых данных, повсеместное распространение распределенных приложений и появление веб-платформ с высокими требованиями к масштабируемости привели к необходимости поиска новых архитектур. В

начале 2000-х годов начала формироваться концепция NoSQL-баз данных — класса СУБД, которые не ориентируются на традиционную реляционную модель, а предлагают альтернативные модели хранения данных, такие как key-value, документно-ориентированная структура, колоночное хранение и графовые структуры. Основное преимущество NoSQL технологий заключается в высокой гибкости схемы данных и возможности горизонтального масштабирования для обработки больших объемов данных, что особенно важно для веб-сервисов и распределенных систем.

NoSQL изначально не позиционировался как замена реляционным СУБД, а скорее как дополнение, обеспечивающее эффективную работу с массивами данных, характерными для социальных сетей, потоковых сервисов и больших данных. Несмотря на это, такие решения стали широко использоваться в промышленности и коммерческих проектах, где требования к гибкости и масштабируемости превышают возможности классической реляционной модели.

Кроме того, развитие технологий баз данных не ограничивается только SQL и NoSQL. На пересечении этих подходов возникли системы NewSQL, стремящиеся объединить преимущества традиционных реляционных СУБД (транзакционность и согласованность) с масштабируемостью и производительностью NoSQL-решений. Появление гибридных архитектур отражает стремление индустрии найти баланс между структурированной моделью данных и требованиями современных распределенных приложений.

Таким образом, эволюция технологий хранения данных представляет собой движение от жёстко структурированных моделей к более гибким и распределенным архитектурам, что обусловлено возрастанием объёмов данных, разнообразием сценариев их использования и потребностью в высокой масштабируемости и производительности современных информационных систем.

# МОДЕЛИ И АРХИТЕКТУРЫ БАЗ ДАННЫХ

## 2.1 Реляционные базы данных и SQL

Реляционные базы данных являются одним из наиболее распространенных и фундаментальных подходов к организации хранения данных в информационных системах. В основе реляционной модели лежит представление данных в виде взаимосвязанных таблиц, состоящих из строк и столбцов, где каждая таблица описывает отдельную сущность предметной области. Такой подход обеспечивает наглядность структуры данных и упрощает их логическую организацию.

Ключевым преимуществом реляционных баз данных является строгая формализация структуры информации [8]. Использование первичных и внешних ключей позволяет поддерживать связи между таблицами и обеспечивать целостность данных. В учебной и научной литературе подчеркивается, что реляционная модель хорошо подходит для задач, требующих высокой согласованности информации и чётко определённых бизнес-правил, таких как финансовые, учетные и корпоративные информационные системы.

Для работы с реляционными базами данных применяется стандартизированный язык SQL (Structured Query Language), который используется для создания таблиц, управления данными и выполнения запросов. SQL позволяет описывать операции выборки, добавления, обновления и удаления данных в декларативной форме, что повышает удобство разработки и сопровождения программных систем. Благодаря стандартизации SQL реляционные СУБД обеспечивают высокий уровень переносимости и совместимости между различными платформами.

Важной особенностью реляционных СУБД является поддержка транзакций и соблюдение принципов ACID, гарантирующих атомарность, согласованность, изолированность и долговечность операций с данными. Эти свойства обеспечивают надежную работу системы в многопользовательской среде и при возникновении сбоев, что делает реляционные базы данных актуальными для критически важных приложений.

Несмотря на появление альтернативных моделей хранения данных, реляционные базы данных продолжают широко использоваться в современных информационных



системах благодаря своей надёжности, зрелости технологий и развитой экосистеме инструментов.

## **2.2 NoSQL и нереляционные базы данных**

С развитием веб-приложений, распределенных систем и технологий больших данных возникла необходимость в более гибких подходах к хранению информации, что привело к появлению NoSQL-баз данных. В отличие от реляционных СУБД, NoSQL-решения не требуют строгой схемы данных и ориентированы на обработку больших объемов разнородной информации в распределенной среде.

К основным типам NoSQL-баз данных относятся хранилища типа «ключ-значение», документно-ориентированные, колоночные и графовые базы данных. Каждая из этих моделей оптимизирована под определённые сценарии использования, такие как хранение полуструктурированных данных, обработка потоков информации или анализ сложных связей между объектами [9].

Основным преимуществом NoSQL-технологий является высокая масштабируемость и гибкость, что делает их востребованными в современных информационных системах. При этом NoSQL-решения, как правило, дополняют реляционные базы данных, а не полностью заменяют их, формируя комбинированные архитектуры хранения данных в рамках современных программных систем.

## **2.3 Графовые базы данных**

Графовые базы данных представляют собой специализированный тип нереляционных хранилищ, ориентированный на хранение и обработку данных с большим количеством взаимосвязей. В основе графовой модели лежит представление информации в виде вершин (узлов) и рёбер, которые отражают объекты предметной области и связи между ними. Такой подход позволяет наглядно и эффективно описывать сложные структуры данных.

Основным преимуществом графовых баз данных является высокая производительность при выполнении запросов, связанных с анализом отношений и зависимостей между объектами [10]. В отличие от реляционных баз данных, где сложные связи требуют использования множественных соединений таблиц, графовые СУБД обеспечивают прямой доступ к связанным данным, что снижает вычислительные затраты при обработке таких запросов.

Графовые базы данных находят применение в системах управления знаниями, социальных сетях, рекомендательных сервисах и аналитических платформах. В современных информационных системах графовые СУБД, как правило, используются в сочетании с другими моделями хранения данных, дополняя реляционные и документно-ориентированные базы данных в рамках многоуровневых архитектур.

# **СОВРЕМЕННЫЕ ТЕХНОЛОГИИ И ПОДХОДЫ В DATABASE ENGINEERING**

## **3.1 Распределенные и облачные базы данных**

Современные информационные системы всё чаще используют распределенные и облачные базы данных, что обусловлено ростом объемов данных и требованиями к высокой доступности сервисов. В таких системах данные размещаются на нескольких узлах, что позволяет повысить отказоустойчивость и обеспечить масштабируемость за счет распределения нагрузки.

Распределенные базы данных используют механизмы репликации и фрагментации данных, обеспечивая доступность информации даже при отказе отдельных компонентов системы. Облачные базы данных, в свою очередь, предоставляют доступ к СУБД как к сервису, снижая затраты на администрирование и упрощая развертывание информационных систем.

Использование распределенных и облачных решений является важным направлением развития технологий баз данных и позволяет адаптировать системы хранения данных к требованиям современных приложений, работающих в условиях высокой нагрузки и динамически изменяющихся объемов информации.

## **3.2 Архитектурные подходы хранения данных**

Развитие технологий баз данных тесно связано с эволюцией архитектур программных систем. В традиционных монолитных приложениях, как правило, использовалась единая централизованная база данных, обслуживающая все функциональные компоненты системы. Такой подход отличается простотой реализации, однако ограничивает масштабируемость и гибкость системы.

В современных информационных системах всё более широкое распространение получают микросервисные архитектуры, в рамках которых каждый сервис может использовать собственное хранилище данных. Данный подход позволяет выбирать тип базы данных с учетом специфики задач конкретного сервиса и повышает устойчивость системы к сбоям отдельных компонентов.

Использование нескольких типов баз данных в рамках одной системы получило название *polyglot persistence*. Такой архитектурный подход позволяет сочетать преимущества реляционных, нереляционных и графовых баз данных, обеспечивая более эффективное хранение и обработку информации в сложных программных системах.

### **3.3 Интеграция БД с прикладными системами**

Интеграция баз данных с прикладными системами является ключевым аспектом обеспечения функциональности современных информационных решений. Прикладные приложения используют базы данных для хранения и обработки информации, а эффективная интеграция обеспечивает оперативный обмен данными, согласованность информации и поддержку бизнес-процессов.

Современные подходы к интеграции включают использование API, веб-сервисов (REST, SOAP), а также специализированных промежуточных слоев (middleware), которые обеспечивают взаимодействие между СУБД и приложениями. Важно, чтобы интеграция поддерживала управление транзакциями, согласованность данных и безопасность, особенно в распределенных и облачных системах.

В контексте полиглот-персистентности интеграция может предусматривать одновременное использование нескольких типов баз данных в рамках одного приложения. Например, реляционные СУБД обеспечивают транзакционную обработку, графовые базы данных работу с сетями и связями объектов, а документно-ориентированные решения хранение полуструктурированных данных. Такой подход позволяет приложениям использовать оптимальные средства хранения и обработки информации под конкретные задачи.

Кроме того, современные инструменты интеграции включают ETL-процессы (Extract, Transform, Load), конвейеры потоковой обработки данных и платформы для работы с большими данными. Это позволяет прикладным системам эффективно обмениваться информацией с различными источниками данных, обеспечивать аналитические и отчетные функции, а также поддерживать оперативное принятие решений.

Таким образом, интеграция баз данных с прикладными системами является неотъемлемой частью проектирования и эксплуатации информационных систем, обеспечивая согласованность, масштабируемость и гибкость работы приложений в современных вычислительных средах.

## ЗАКЛЮЧЕНИЕ

В данной работе был проведен анализ современных технологий баз данных, рассмотрены их модели, архитектуры и подходы к интеграции с прикладными системами. Изучение эволюции БД показало переход от жёстко структурированных иерархических и реляционных моделей к гибким и распределенным решениям, включая NoSQL и графовые базы данных. Такие подходы обеспечивают масштабируемость, высокую производительность и возможность эффективного хранения больших объемов разнородных данных.

Особое внимание уделено современным архитектурным решениям, включая распределенные, облачные и микро сервисные системы, а также полиглот-персистентность, которая позволяет комбинировать различные типы БД в рамках одной информационной системы для оптимального решения прикладных задач. Рассмотрены ключевые методы интеграции баз данных с прикладными системами, включая API, веб-сервисы, middleware и ETL-процессы, что обеспечивает согласованность данных, безопасность и гибкость работы приложений.

Полученные результаты подчеркивают, что развитие технологий баз данных напрямую связано с потребностями современных информационных систем: увеличением объемов данных, требованиями к масштабируемости, отказоустойчивости и эффективности обработки информации. Знания, полученные в ходе исследования, могут быть использованы для проектирования и внедрения современных программных продуктов, ориентированных на работу с большими и разнообразными данными, а также для дальнейших исследований в области database engineering.

Таким образом, работа подтверждает актуальность изучения технологий баз данных и интеграции их с прикладными системами как основы для создания эффективных, надежных и масштабируемых информационных систем в условиях современной цифровой среды.

## СПИСОК ЛИТЕРАТУРЫ

1. NoSQL: что это за базы данных, для чего они нужны и как работают. — Текст : электронный // Skillbox : [сайт]. — URL: <https://skillbox.ru/media/code/nosql-cto-eto-za-bazy-dannykh-dlya-chego-oni-nuzhny-i-kak-rabotayut/> (дата обращения: 29.12.2025).
2. Удахина, С. В. Базы данных : учебное пособие / С. В. Удахина. — Санкт-Петербург : СПбГУ ГА им. А.А. Новикова, 2025. — 143 с. — ISBN 978-5-907860-09-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/482711> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
3. Основные типы баз данных: от реляционных до NoSQL. — Текст : электронный // Skypro : [сайт]. — URL: <https://sky.pro/wiki/sql/tipy-i-klassifikaciya-baz-dannyh/> (дата обращения: 29.12.2025).
4. Базы данных : учебное пособие / составители Т. Ж. Базаржапова [и др.]. — Улан-Удэ : Бурятская ГСХА им. В.Р. Филиппова, 2022. — 84 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/284240> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
5. database management system. — Текст : электронный // Britannica : [сайт]. — URL: <https://www.britannica.com/technology/database-management-system> (дата обращения: 29.12.2025).
6. Развитие баз данных. — Текст : электронный // Хабр : [сайт]. — URL: <https://habr.com/ru/articles/803247/> (дата обращения: 29.12.2025).
7. Базы данных – что это такое, примеры использования. — Текст : электронный // Сравни : [сайт]. — URL: <https://edu.sravni.ru/kursy/info/bazy-dannyh/> (дата обращения: 29.12.2025).
8. Базы данных и базы знаний : учебно-методическое пособие / составители М. В. Юрчишина [и др.]. — Сургут : СурГУ, 2022. — 68 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/337898> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
9. Примеры нереляционных баз данных для новичков. — Текст : электронный // Skyeng : [сайт]. — URL:

<https://skyeng.ru/it-industry/it/primery-nerelyatsionnykh-baz-dannykh-dlya-nachinayushchikh/> (дата обращения: 29.12.2025).

10. What is Graph Database - Introduction. — Текст : электронный // GeeksForGeeks : [сайт]. — URL: <https://www.geeksforgeeks.org/dbms/what-is-graph-database/> (дата обращения: 29.12.2025).