

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра Математического и компьютерного моделирования

Сравнение реляционной и нереляционной моделей данных

на примере реализации ИС «Абитуриент»

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 451 группы

направление 38.03.05 — Бизнес-информатика

механико-математического факультета

Аксеновой Евгении Олеговны

Научный руководитель  
доцент, к.ф. - м.н.

О. М. Ромакина

Зав. кафедрой  
зав. каф., д.ф.-м.н., доцент

Ю.А. Блинков

Саратов 2019

**Введение.** В современном обществе количество информации, получаемой по компьютерным сетям, стремительно растет. Для того чтобы правильно организовать, хранить и обрабатывать большие объемы данных необходимо обеспечить грамотный подход к выбору технологий в практической разработке информационной системы. К таким технологиям можно отнести модели данных, базы данных, а также системы управления базами данных.

Модели данных обеспечивают логическую структуризацию используемых данных и являются главным критерием того, как будет работать и управлять информацией приложение.

Как правило, выделяют два основных подхода к построению информационных систем – реляционные и нереляционные модели данных. На сегодняшний день, реляционная модель данных, принципы построения которой были предложены в конце 1960-х годов, считается наиболее популярной и хорошо зарекомендовавшей себя среди пользователей.

Однако за последнее десятилетие заметное распространение получили и другие модели данных. Для обозначения появившихся классов БД и моделей данных был введен термин «NoSQL», синонимом которого выступает термин «нереляционный». Актуальность данной работы заключается в том, что выбор подхода, на основе которого будет разрабатываться ИС, является одним из важнейших этапов при ее разработке и представляет собой сложную многопараметрическую задачу. Выбранные решения должны удовлетворять как текущим, так и будущим потребностям информационной системы.

Предмет исследования - информационная система «Абитуриент», которая обладает заранее определенной структурой и схемой. Данные, хранящиеся в информационной системе, нормализованы.

Подразумевается, что информационная система «Абитуриент» с каждым годом будет расти в масштабах в смысле объемов хранимой информации. Также хранимая или вновь вносимая информация не всегда будет являться однородной. При работе с ИС «Абитуриент», время от времени возникает потребность в модификации структуры данных, что может вызвать необходимость в перепроектировании БД и модификации интерфейса системы. Важно, чтобы это обстоятельство минимально отражалось на работе с ИС:

никакие изменения в структуре базы данных не должны вызывать трудности в работе с приложением.

На основании этого была определена цель бакалаврской работы: обоснование выбора модели данных, наиболее полно соответствующей требованиям ИС «Абитуриент».

Задачами работы являются:

1. Изучение теоретических аспектов информационных систем;
2. Изучение реляционного и нереляционного подходов к построению информационных систем;
3. Обзор систем управления базами данных PostgreSQL и MongoDB;
4. Сравнительный анализ рассматриваемых подходов, а также функциональных и нефункциональных характеристик СУБД PostgreSQL и MongoDB на примере ИС «Абитуриент»;
5. Оценка производительности по времени выполнения запросов в сравниваемых СУБД;
6. Выбор технологий разработки информационной системы «Абитуриент» и его обоснование.

**Основная часть.** Основная часть работы содержит 4 раздела:

1. Теоретические основы;
2. Реляционный и нереляционный подходы к построению информационных систем;
3. Сравнительный анализ СУБД PostgreSQL и MongoDB на примере ИС «Абитуриент».

В первом разделе рассматривается понятие «Информационная система». Приведены основные функции, свойства и компоненты информационной системы.

Информационная система представляет собой взаимосвязанную совокупность информационных, технических, программных, математических, организационных, правовых, технологических и других средств, а также персонала. Информационная система должна удовлетворять конкретные информационные потребности в рамках определенной предметной области, при этом результатом функционирования информационных систем является информа-

ционная продукция — документы, информационные массивы, базы данных и информационные услуги.

*Функциями информационных систем являются:*

- Сбор и регистрация информационных ресурсов. Обеспечивается «фиксирование» информации о состоянии предметной области, формирование и поддержка на этой основе модели предметной области экстенсивного уровня;
- Хранение информационных ресурсов. Данная функция связана с необходимостью управления двумя видами ресурсов — ресурсами хранимых данных и ресурсами памяти. Требования к этим функциям различаются в разных классах информационных систем;
- Актуализация информационных ресурсов. Назначение информационной системы состоит в поддержке динамической информационной модели ее предметной области. Для того чтобы эта модель была практически полезной, необходимо своевременно отображать в ней изменения состояния предметной области;
- Обработка информационных ресурсов и их предоставление пользователям.

*К свойствам информационных систем относятся:*

- Динамичность - информационные системы с течением времени меняют свою структуру и состояние элементов;
- Интегрируемость - система должна взаимодействовать с вновь подключаемыми компонентами или подсистемами;
- Масштабируемость – возможность расширения системных ресурсов и производительной мощности;
- Адаптивность - приспособляемость системы к условиям конкретной предметной области;
- Обоснованность - получение обоснованных результатов в ходе выполнения прикладных программ;
- Безопасность - возможность предотвращения дезорганизации системы в результате несанкционированного доступа;
- Системный подход - предполагает наличие и учет большого числа связанных между собой системных элементов;

- Информационная система должна обладать гибкой управляемостью;
- В процессе реализации функций управления имеет место постоянное взаимодействие пользователей и технических средств.

*Компоненты информационных систем включают:*

- Базу данных;
- Концептуальную схему;
- Информационный процессор.

В совокупности перечисленные компоненты образуют систему хранения и манипулирования данными.

База данных - совокупность данных, организованных по определенным правилам, устанавливающим общие принципы описания, хранения и манипулирования данными. Хранение данных в БД обеспечивает централизованное управление, соблюдение стандартов, безопасность и целостность данных, сокращает избыточность и устраняет противоречивость данных. БД не зависит от прикладных программ. Создание БД и обращение к ней по запросам осуществляется с помощью системы управления базами данных.

Концептуальная схема представляет собой описание структуры всех единиц информации, хранящихся в БД. Под структурой понимается вхождение одних единиц информации в состав других единиц информации.

Информационный процессор - это механизм, который в ответ на получение команды выполняет операции с БД и концептуальной схемой. Информационный процессор состоит из вычислительной системы и системы управления базой данных СУБД.

СУБД - комплекс программ и языковых средств, предназначенных для создания, ведения и использования баз данных. Эти приложения управляют или помогают управлять наборами данных. Поскольку эти данные могут быть разного формата и размера, были созданы разные виды СУБД. СУБД основываются на моделях баз данных - определённых структурах для обработки данных. Каждая СУБД создана для работы с одной из них с учётом особенностей операций над информацией.

Во втором разделе подробно рассматриваются реляционный и нереляционный подходы к построению информационных систем.

Ядром любой базы данных является модель данных. Модель данных описывает некоторый набор родовых понятий и признаков, которыми должны обладать все конкретные СУБД и управляемые ими базы данных, если они основываются на этой модели. Наличие модели данных позволяет сравнивать конкретные реализации, используя один общий язык.

На сегодняшний день существует достаточно много решений, реализующих различные модели баз данных. Однако можно выделить два основных подхода – реляционные и нереляционные модели данных.

Реляционная модель данных. Кодд предложил применение реляционной алгебры в реляционных СУБД для расчленения данных в связанные наборы. Он организовал свою систему БД вокруг концепции, основанной на наборах данных. Данные разбиваются на наборы, которые составляют табличную структуру. Эта структура таблиц состоит из индивидуальных элементов данных, называемых полями. Одиночный набор или группа полей известна как запись.

Нереляционная модель данных. Изначально термин NoSQL являлся названием очередной реляционной системы управления базами данных. Тем не менее, термин подразумевал немного другое - избежание использования стандартов реляционной модели. В последующие несколько лет данная идея получила развитие и термин стал использоваться в отношении нереляционных баз данных. Согласно задумке, NoSQL базы данных не подразумевают внутренних связей. Они не основываются на одной модели, а каждая база данных в зависимости от целей использует различные модели. Эти модели обычно принято разделять на четыре категории: ключ - значение, колоночные хранилища, графовые и документо-ориентированные хранилища.

В бакалаврской работе рассматривается информационная система, построенная на нереляционной модели документо-ориентированного хранилища. Модель данных подобных хранилищ позволяет объединять множество пар ключ-значение в абстракцию, называемую «документ». Документы могут иметь вложенную структуру и объединяться в коллекции. Однако, это скорее удобный способ логического объединения, так как никакой жесткой схемы у документов нет и множества пар ключ-значение, даже в рамках одной коллекции, могут быть абсолютно произвольными. Работа с документа-

ми производится по ключу, однако существуют решения, позволяющие осуществлять запросы по значениям атрибутов. Они работают так же, как и предыдущие системы, но они допускают гораздо большую вложенность и сложность структуры данных.

Реляционная и нереляционная системы управления баз данных. При реляционном подходе построения ИС «Абитуриент» использовалась СУБД – PostgreSQL, при нереляционном - MongoDB.

PostgreSQL – свободно распространяемая объектно-реляционная система управления базами данных. PostgreSQL разрабатывается международным сообществом разработчиков и не контролируется ни одной компанией и ни одним частным лицом. PostgreSQL распространяется по крайне свободной лицензии, которая позволяет включать его в состав коммерческих программных продуктов. Данная технология предоставляет множество различных возможностей, достаточно надежна и имеет хорошие характеристики по производительности.

MongoDB представляет собой открытую систему управления баз данных типа NoSQL, она разработана с учетом требований в крайне высокой производительности и постоянной масштабируемости. Платформа MongoDB поддерживает документно-ориентированную модель данных, динамическую схему, полноценное индексирование, сверхбыстрое обновление с замещением, функциональные динамические запросы, встроенный шардинг и автоматическое восстановление после сбоя.

В третьем разделе произведен сравнительный анализ СУБД PostgreSQL и MongoDB на примере ИС «Абитуриент». Для реализации базы данных «Абитуриент», основанной на реляционной модели данных, необходимо на ранних этапах спроектировать структуру БД. Проектирование структуры обычно осуществляется с помощью концептуальных схем. Концептуальная схема представляет собой единое логическое описание базы данных, определяет все логические элементы базы данных и способы связи между ними. Главным недостатком наличия схемы является трудность поддержания гибкости, которая связана с изменениями в предметной области. Например, при добавлении принципиально новой информации в реляционную БД возникает необходимость модифицировать ее структуру, что влечет за собой дополни-

тельные временные и материальные затраты, также появляется вероятность того, что часть данных, хранящихся в реляционной БД может быть утрачена.

При создании БД, основанной на документо-ориентированном подходе, нет необходимости создавать схему с определенными связями между документами и коллекциями, также отсутствуют ограничения целостности. Коллекции состоят из документов, которые теоретически могут обладать своей структурой. Отсутствие структуры не вызывает трудностей при добавлении нового функционала, позволяет на различных этапах разработки базы данных вносить изменения в её структуру.

При сравнении характеристик PostgreSQL и MongoDB выяснилось, что данные СУБД конкурируют между собой в функциональности и производительности, а также перенимают друг у друга технологические преимущества.

Оценка производительности сравниваемых СУБД проводилась способом измерения времени выполнения запросов. На производительность СУБД оказывают влияние два фактора:

- СУБД, которые следят за соблюдением целостности данных, несут дополнительную нагрузку, которую не испытывают другие программы;
- Производительность прикладных программ сильно зависит от правильного проектирования и построения базы данных.

Сравнение производительности MongoDB и PostgreSQL было проведено с помощью следующих запросов:

- Вывод всех данных;
- Вывод данных удовлетворяющих какому-либо условию;
- Выбор данных по нескольким параметрам;
- Выбор данных лежащих в каком-либо диапазоне;
- Соединение таблиц;
- Вывод данных удовлетворяющих нескольким условиям;
- Сортировка данных.

Деятельность по оценке производительности по времени выполнения запросов в СУБД MongoDB и PostgreSQL показала, что MongoDB затрачивает меньше времени на обработку и вывод данных в ответ на запросы.

**Заключение.** Реляционный подход является наиболее распространённым подходом к организации баз данных в наше время. Именно на нем ос-



новано подавляющее большинство современных коммерческих систем управления базами данных. В реляционных системах информация представляется пользователю в виде таблиц. Такое представление является весьма удобным и достаточно понятным для любого человека. Реляционная модель данных основана на развитом математическом аппарате, который позволяет достаточно лаконично описать основные операции над данными.

Нереляционный подход появился относительно недавно, но он уже успел завоевать высокую популярность. Причинами его появления стали необходимость повышения быстродействия и увеличения возможностей масштабируемости систем. За счет избегания использования стандартов реляционной модели, NoSQL решения достигают своих целей. Каждая из нереляционных систем управления базами данных имеет свою модель данных. В данной работе сравнивались две СУБД – одна представляет реляционный класс – СУБД PostgreSQL, другая – СУБД MongoDB нереляционный класс.

На основании проделанной работы, можно сделать вывод, что нереляционная модель данных и основанная на данной модели СУБД MongoDB является наиболее подходящим решением для информационной системы «Абитуриент» по следующим причинам:

- Информационные системы должны иметь возможность адаптироваться к изменчивости структуры данных. Отсутствие структуры в MongoDB не вызывает трудностей при добавлении нового функционала. Бесструктурная архитектура коллекций в MongoDB не обязывает разработчиков определять и обслуживать схемы на уровне данных, обеспечивая быструю итерацию при разработке продукта;
- Встроенная технология шардинга позволяет с легкостью расширять масштабы информационной системы в смысле объемов хранимой информации;
- MongoDB – относительно новая технология, которая завоевала высокую популярность, поддерживается большим сообществом разработчиков и пользователей, функционал СУБД совершенствуется от версии к версии;
- Важным недостатком нереляционных СУБД является отсутствие поддержки транзакций, удовлетворяющих требованиям ACID. Однако

MongoDB, начиная с версии 4.0, стала поддерживать данные транзакции. Требования ACID обеспечивают надежность и целостность данных;

- Язык запросов в MongoDB пусть и является специфичным, но он практически не уступает в функциональности такому известному и привычному языку запросов, как SQL;
- Деятельность по оценке производительности по времени выполнения запросов в СУБД MongoDB и PostgreSQL показала, что MongoDB затрачивает меньше на обработку и вывод данных в ответ на запросы.