

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической  
кибернетики и компьютерных наук

**СОЗДАНИЕ ПРИЛОЖЕНИЯ ОПЕРАТИВНОЙ АНАЛИТИЧЕСКОЙ  
ОБРАБОТКИ ДАННЫХ ДЛЯ ИНТЕРНЕТ МАГАЗИНА**

**АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

Студентки 4 курса 411 группы  
направления 02.03.02 — Фундаментальная информатика и информационные  
технологии  
факультета КНиИТ  
Долговой Полины Юрьевны

Научный руководитель  
старший преподаватель

\_\_\_\_\_

М. И. Сафрончик

Заведующий кафедрой  
к. ф.-м. н.

\_\_\_\_\_

С. В. Миронов

Саратов 2018

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1 Теоретические основы многомерного анализа данных .....	4
1.1 Системы бизнес-аналитики .....	4
1.2 Многомерная модель данных .....	4
1.3 12 требований Кодда .....	5
1.4 Архитектура OLAP-системы .....	6
1.5 OLAP-кубы в Microsoft SQL Server .....	7
1.6 Язык MDX .....	7
2 Реализация приложения .....	8
2.1 Создание OLAP-куба .....	8
2.2 Получение доступа к службам Analysis Services .....	8
2.3 Разработка приложения .....	9
2.4 Работа с данными .....	9
2.5 Веб-интерфейс .....	10
2.5.1 Режим «Table Editor» .....	12
2.5.2 Режим «Query Editor» .....	12
2.5.3 Режим «Diagram Editor» .....	12
2.6 Контроллер приложения .....	12
ЗАКЛЮЧЕНИЕ .....	14
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	15

## ВВЕДЕНИЕ

Повсеместное использование баз данных для ведения бизнеса и накопление огромного объема информации привело к пониманию важности задач, связанных с ее анализом, с целью получения новых знаний и выработки эффективных стратегий дальнейшего развития. Возникла потребность в программных инструментах, позволяющих удобным для пользователя образом извлекать из накопленных данных значимую для бизнеса информацию, на ее основе проводить анализ, вырабатывать стратегию и принимать различного рода решения, позволяющие сохранять предприятиям их конкурентоспособность на рынке.

На сегодняшний день именно многомерный подход к реализации систем управления базами данных вызывает особый интерес.

В связи с развитием интернета всё большую популярность получают веб-приложения. Эта тенденция также коснулась и средств для работы с OLAP. Появившиеся динамические технологии представляют собой прекрасную альтернативу традиционным клиент/серверным OLAP-приложениям.

Целью работы является создание веб-приложения по оперативной аналитической обработке данных, для реализации которой были поставлены следующие задачи:

1. анализ предметной области,
2. проектирование OLAP-куба,
3. реализация OLAP-куба в среде Microsoft Visual Studio 2015 с использованием служб Microsoft SQL Server Analysis Services,
4. написание приложения для связи с данными, предоставляемыми службами Microsoft SQL Server Analysis Services,
5. написание пользовательского веб-интерфейса.

В первой главе рассматриваются теоретические основы многомерного анализа данных, основные концепции оперативной обработки, требования к системам. Вторая глава содержит описание созданного веб-приложения и используемых технологий, их особенности и возможности.

Для разработки приложения был выбран язык Java, а также фреймворки Spring Boot и Spring MVC. Работа с многомерными базами данных осуществлялась средствами Olap4j. Веб-интерфейс реализован с помощью фреймворка AngularJS и библиотеки ChartJS.

# **1 Теоретические основы многомерного анализа данных**

## **1.1 Системы бизнес-аналитики**

Система бизнес-анализа (англ. Business Intelligence — BI) — это методы и инструменты для перевода необработанной информации в осмысленную, удобную форму для последующего бизнес-анализа. Технологии BI обрабатывают большие объёмы неструктурированных данных, чтобы найти стратегические возможности для бизнеса, необходимые для увеличения прибыли и развития.

Целью BI является интерпретировать большое количество данных, заостряя внимание только на ключевых факторах эффективности, моделируя исход различных вариантов действий и отслеживая результаты принятия решений.

Выделяют три основных типа таких инструментальных средств:

1. средства многомерного анализа OLAP (On-Line Analytical Processing) — программное обеспечение, которое дает пользователю возможность наблюдать данные в различных измерениях, направлениях или сечениях;
2. инструментальные средства запросов (Query Tools) — программное обеспечение, позволяющее формировать запросы к данным по содержанию или образцу;
3. инструментальные средства поиска скрытых знаний в данных (Data Mining Tools) — программное обеспечение, которое осуществляет автоматический поиск важных образцов (моделей), или зависимостей в данных.

Основываясь на анализ современного рынка, веб-архитектуры быстро вытесняют традиционные клиент/серверные приложения. Это коснулось и рынка корпоративных OLAP-решений. В первую очередь это связано с развитием и распространением веб-технологий. Также в связи с расширением числа пользователей системы разрабатываются для выполнения анализа, который частично отличается от того, что проводят обычными инструментами.

Использование веб-приложения имеет ряд следующих преимуществ: удобство использования, интерактивность, функциональность, доступность и переносимость, а также сниженные затраты на внедрение и администрирование.

## **1.2 Многомерная модель данных**

Согласно [1], в процессе принятия решений пользователю необходимо сгенерировать некоторые гипотезы. Следующим этапом является их проверка

на основании информации об анализируемой предметной области, после чего будут получены уже готовые законченные решения.

С точки зрения бизнес-анализа каждый анализируемый факт удобно рассматривать как функцию от его характеристик. Совокупность значений какого-либо анализируемого параметра называется измерением. Все измерения независимы друг от друга. Если данные можно распределить на некоторое количество измерений, это означает, что их можно представить в виде многомерной модели. Таким образом, по измерениям в полученной модели откладывают параметры, необходимые для дальнейшего анализа. При многомерном анализе одновременно используется несколько измерений.

Мерами называются данные, количественно характеризующий анализируемые факты, например, издержки, объемы продаж, прибыли и др. Она располагается на пересечениях осей измерений.

Таким образом, многомерную модель данных можно представить как гиперкуб, ребрами которого являются измерения, а ячейками — меры.

Над гиперкубом могут выполняться следующие операции:

1. срез (slice) — получение частного случая подкуба, проекции куба.
2. вращение (rotate) — визуализация, выборка и ориентация срезов куба для визуального анализа.
3. консолидация (Drill Up) и детализация (Drill Down) — операции, которые представляют переход вверх по направлению от детального (down) представления данных к агрегированному (up) и наоборот, соответственно.

Оперативный анализ, проводимый средствами OLAP-систем, тесно связан с идеей многомерного анализа данных.

OLAP (On-Line Analytical Processing) — это технология оперативной аналитической обработки данных, использующая методы и средства для сбора, хранения и анализа многомерных данных в целях поддержки процессов принятия решений. OLAP-системы предназначены для поддержки аналитической деятельности, произвольных запросов пользователей-аналитиков.

Таким образом, OLAP-анализ предоставляет возможности для моделирования данных и подтверждения различных теорий.

### **1.3 12 требований Кодда**

На основании информации из [1], у истоков технологии OLAP стоит основоположник реляционного подхода Э. Кодд. В своей статье «OLAP для

пользователей-аналитиков: каким он должен быть» он выделил основные концепции оперативной обработки. Они включают в себя 12 требований, которым должны удовлетворять продукты, чтобы на их основе можно было бы выполнять оперативную аналитическую обработку:

1. концептуальное многомерное представление;
2. прозрачность;
3. доступность;
4. постоянная производительность при разработке отчетов;
5. клиент-серверная архитектура;
6. равноправие измерений;
7. динамическое управление разреженными матрицами;
8. многопользовательская поддержка ;
9. неограниченные перекрестные операции;
10. интуитивная манипуляция данными;
11. гибкие возможности получения отчетов;
12. неограниченная размерность и число уровней агрегации.

#### **1.4 Архитектура OLAP-системы**

По данным из [1], OLAP-система состоит из следующих компонентов:

1. OLAP-сервер — обеспечивает вычисление и хранение агрегатных данных, выполнение над ними необходимых операций и формирование многомерной модели.
2. OLAP-клиент — предоставляет пользователю интерфейс, необходимый для удобного манипулирования данными многомерной модели данных для выполнения задач анализа.

OLAP-серверы формируют гиперкуб, с которым пользователи посредством OLAP-клиента выполняют все необходимые манипуляции, анализируя данные. Производительность и занимаемые ресурсы зависят в большей степени от способа реализации многомерной системы.

Выделяют 3 основных способа реализации:

1. MOLAP (Multidimensional OLAP) — исходные и агрегатные данные хранятся в многомерной базе данных или локальном кубе.
2. ROLAP (Relational OLAP) — исходные данные хранятся в реляционных базах данных или в плоских локальных таблицах на файл-сервере.
3. HOLAP (Hybrid OLAP) — для реализации многомерной модели исполь-

зуют и многомерные, и реляционные базы данных, а именно исходные данные остаются в реляционной базе, а агрегаты размещаются в многомерной.

### **1.5 OLAP-кубы в Microsoft SQL Server**

Согласно [2], службы Microsoft SQL Server Analysis Services обеспечивают интерактивную аналитическую обработку (OLAP) и функции интеллектуального анализа данных для приложений бизнес-аналитики. Для подобных приложений службы Analysis Services позволяют разрабатывать, создавать и наглядно представлять модели интеллектуального анализа данных, построенные на основе других источников данных, используя с этой целью широкий спектр стандартных алгоритмов интеллектуального анализа данных, а также управлять этими структурами.

Полезной функцией куба OLAP является то, что данные в кубе могут содержаться в статистическом («агрегатном») виде. Поскольку куб содержит множество предварительно вычисленных значений, для пользователя это выглядит так, словно в кубе уже заранее есть все необходимые ответы.

### **1.6 Язык MDX**

На основании [3], далеко не всегда необходимые срезы данных можно получить, просто манипулируя имеющимся кубом, поэтому практически во всех промышленных OLAP-средствах, включая и Analysis Services, для получения нестандартных срезов данных требуется отдельный непроцедурный язык для формулирования запросов к многомерным базам данных. Одним из таких языков является MDX, который был создан группой разработчиков SQL Server.

Язык MDX (Multidimensional Expressions) предоставляет разработчикам средства простого и эффективного доступа к многомерным структурам данных и расширяет возможности работы с ними. С его помощью в Microsoft SQL Server Analysis Services формируются запросы, а также описываются алгоритмы получения вычисляемых значений. Несмотря на то, что MDX является не общим стандартом, а только внутренней спецификацией Microsoft, он был принят многими ведущими распространителями технологии OLAP. Более подробная информация представлена в [4].

## **2 Реализация приложения**

### **2.1 Создание OLAP-куба**

В ходе практической работы был построен OLAP-куб в среде Microsoft Visual Studio 2015 с использованием служб Microsoft SQL Server Analysis Services.

В качестве источников данных выступало хранилище данных онлайн-магазина товаров на основе схемы типа «звезда».

Измерениями куба являются таблицы валют (Currency), покупателей (Customer), товары магазина (Product), территории (Territory) и дат (Date). У каждой записи есть свой индивидуальный номер, обеспечивающий уникальность данных. Измерение Product являлось медленно меняющимся, поэтому у каждого товара также есть бизнес-ключ, поле Current, по которому определялась актуальность записи для товара, и поля DateStart и DateEnd, в которых фиксировались даты изменений.

В таблице фактов (Sale) фиксировались номера продажи, товара, покупателя, даты данной продажи, места доставки товара, валюты. В качестве мер выступали количество товара (Quantity), предварительная стоимость покупки без учета доставки, скидок и налогов (Subtotal), стоимость доставки (DeliveryAmt), процентная величина скидки (Discount), процентная величина налога (Tax), а также конечная стоимость заказа (Total).

### **2.2 Получение доступа к службам Analysis Services**

Службы Microsoft SQL Server Analysis Services предоставляют интерактивную аналитическую обработку и функции интеллектуального анализа данных для приложений бизнес-аналитики. Они поддерживают OLAP, позволяя разрабатывать и создавать многомерные структуры, а также управлять и наглядно представлять модели интеллектуального анализа данных. В них содержатся данные, собранные из других источников, таких как реляционные базы данных.

Данные службы используют протокол XML для аналитики (XMLA) — это собственный XML-протокол Microsoft SQL Server службы Analysis Services, основанный на протоколе SOAP, который специально предназначен для обеспечения унифицированного доступа к данным в любом стандартном многомерном источнике данных, находящемся в сети интернет. Он используется



для обеспечения взаимодействия между клиентскими приложениями и экземпляром служб Analysis Services. Для получения доступа к созданному OLAP-кубу и содержащимся в нем данным было настроено соединение по протоколу HTTP по URL-адресу *http : //localhost/OLAP/msmdpump.dll*.

### **2.3 Разработка приложения**

В ходе практики было создано приложение «Olap Viewer», предназначенное для обращения к OLAP-кубам, генерации запросов к ним и визуализации полученных данных, на основе которых проводится последующий анализ. Для начала работы необходимо выбрать нужный каталог и OLAP-куб. Пользователю предоставлено 3 режима работы с данными:

1. «Table Editor» — создание запросов на основе выбранных элементов и последующее получение данных;
2. «Query Editor» — непосредственное написание запросов на языке MDX и последующее получение данных;
3. «Diagram Editor» — создание запросов на основе выбранных элементов и последующее визуальное представление полученных данных.

Более подробно каждый из режимов будет рассматривать ниже.

Для получения, обработки и подготовки данных было реализовано приложение на языке программирования Java. Для его разработки был использован фреймворк Spring Boot. Он позволяет создавать полноценные приложения на основе Spring Framework и его компонентов, а также упрощает разработку веб-приложений и увеличивает их производительность.

Для проектирования приложения был выбран паттерн «модель-представление-контроллер» («model-view-controller»). Он заключается в разделении данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо. Подобную структуру в данном приложении обеспечивает фреймворк Spring MVC.

### **2.4 Работа с данными**

Для получения и обработки данных использовался программный интерфейс Olap4j. Это программное обеспечение с открытым кодом, написанное на языке программирования Java. Оно предназначено для разработки аналитиче-

ских приложений для любого вида серверов, поддерживающих OLAP. Olap4j предоставляет широкий набор инструментов для работы с многомерными базами данных. Написанные приложения взаимодействуют с OLAP-серверами с помощью запросов, написанных на языке MDX. Операторы определяются в терминах метаданных и проверяются в соответствии с системой типов. Более подробно со полным списком возможностей данного программного обеспечения можно ознакомиться в [5].

Для установки соединения, работы с полученными данными и их преобразования были созданы классы CubeService и DBService. Это необходимо для того, чтобы организовать сервис-слой приложения. Данный слой предназначен для определения границ приложений, набора допустимых операций с точки зрения взаимодействия клиента с источником данных и исполнения бизнес-логики.

Так сервис-класс DBService имеет доступ ко всем каталогам многомерных баз данных и предоставляет необходимую информацию, такую как наличие каталогов и доступных для работы OLAP-кубов.

Сервис-класс CubeService в свою очередь предназначен для:

1. работы непосредственно с конкретным OLAP-кубом,
2. выполнения MDX-запросов,
3. обработки полученных данных,
4. преобразования их в необходимый для представления вид.

Также для генерации MDX-запросов в соответствии с выбранными клиентом данными был реализован сервис-класс QueryService и единственный в нем метод для генерации запроса:

Также были созданы классы-сущности, соответствующие терминам баз данных: Row (строка таблицы), Cell (ячейка), Dimension (измерение), Measure (меры) и OlapCube (OLAP-куб). Именно они лежат в основе работы приложения, ими оперируют все методы и к ним преобразуются полученные данные из базы. Это необходимо для реализации бизнес-логики приложения и удобства работы.

## **2.5 Веб-интерфейс**

Приложение «Olap Viewer» является одностраничным приложением. Одностраничное приложение— это веб-приложение, использующее единственный HTML-документ как оболочку для всех веб-страниц и организующее

взаимодействие с пользователем через динамически подгружаемые HTML, CSS, JavaScript файлы. В их основе лежит AJAX (Asynchronous Javascript and XML) — это подход к построению интерактивных пользовательских интерфейсов, заключающийся в «фоновом» обмене данными браузера с веб-сервером. В результате, при обновлении данных веб-страница не перезагружается полностью, что обеспечивает более быструю и удобную работу.

Данное приложение написано с помощью JavaScript-фреймворка с открытым исходным кодом AngularJS, предназначенный для разработки одностраничных приложений [6].

Таким образом, в основе веб-интерфейса лежит единственный HTML-документ `index.html`. Для организации навигации между различными частями приложения, а именно переход из одного режима в другой, используется библиотека AngularJS UI Router. Для этого в отдельном файле `app.js` необходимо описать, как устроен переход.

Для отображения новых элементов на странице в основном файле приложения необходимо указать с помощью тегов `<div ui-view></div>`, куда выводить информацию из отдельных файлов, указанных при переходе из одной части приложения в другую. Ниже представлен фрагмент кода файла `index.html`:

Связующим звеном между HTML-документом и описываемой моделью является контроллер. Его основная цель — получение данных из модели и вывод их пользователю, или же, в другом направлении, получение данных из представления и отправка их в модель.

Для построения графиков и диаграмм использовалась библиотека ChartJS. Она позволяет создавать графики и адаптивные диаграммы любой сложности на основе HTML5 Canvas. Для того, чтобы использовать в приложении инструменты данной библиотеки, необходимо подключить соответствующие файлы в основной странице `index.html`.

Для начала работы в веб-приложении «Olap Viewer» пользователю необходимо выбрать нужный каталог, а также OLAP-куб для дальнейшей работы с ним.

После этого пользователю может работать в любом режиме приложения:

1. «Table Editor» — создание запросов на основе выбранных элементов и последующее получение данных;

2. «Query Editor» — непосредственное написание запросов на языке MDX и последующее получение данных;
3. «Diagram Editor» — создание запросов на основе выбранных элементов и последующее визуальное представление полученных данных.

#### 2.5.1 Режим «Table Editor»

В режиме «Table Editor» после выбора каталога и OLAP-куба на экран выводится информация о кубе. Пользователю необходимо отметить интересные его измерения, атрибуты для них, а также меры. После отправки выбранных элементов на сервере генерируется и выполняется соответствующий MDX-запрос, после чего результаты выполнения выводятся в виде таблицы. Над таблицей также расположен получившийся запрос.

#### 2.5.2 Режим «Query Editor»

Данный режим предоставляет возможность выполнения конкретного запроса. Пользователю необходимо выбрать каталог и OLAP-куб, после чего в появившемся окне ввести уже готовый MDX-запрос. Результат выполнения также будет выведен в табличном виде.

#### 2.5.3 Режим «Diagram Editor»

Режим «Diagram Editor» предназначен для визуального представления данных в виде графиков и диаграмм. Пользователю также необходимо сначала выбрать каталог и OLAP-куб, после чего на экране появятся доступные измерения и меры данного куба. Затем на основе выбранных элементов на сервере генерируется соответствующий MDX-запрос, а полученные данные отображаются в основной части сайта. Справа от графика пользователь может выбрать интересующий его вид визуализации: простой график, круговая диаграмма, столбчатая диаграмма и т. д.

### 2.6 Контроллер приложения

Контроллер управляет запросами пользователя, т. е. интерпретирует его действия, оповещая модель о необходимых изменениях. Когда пользователь нажимает на те или иные элементы интерфейса, он отправляет запрос для выполнения какого-либо действия. Чаще всего это HTTP-запросы POST или GET. Метод запроса POST предназначен для запроса, при котором веб-сервер

принимает данные, заключённые в тело сообщения, для хранения. Он часто используется для загрузки файла или представления заполненной веб-формы. В отличие от него, метод HTTP GET предназначен для получения информации от сервера.

Для каждого метода контроллера веб-приложения, описанного в файле `app.js`, существует соответствующий метод на стороне сервера приложения.

## ЗАКЛЮЧЕНИЕ

Системы бизнес-анализа решают большой спектр задач, связанных с принятием бизнес-решений. Они обеспечивают оценку эффективности различных направлений, деятельности, достижимости поставленных целей.

Использование OLAP-систем обеспечивает быструю работу с данными, собранных в одном месте, в процессе интерактивной работы аналитика, предоставляя удобные средства работы с информацией. На сегодняшний день разработано множество продуктов, реализующих технологию оперативной аналитической обработки данных OLAP, но наибольшую популярность набирают веб-приложения.

В ходе дипломной работы было создано веб-приложение для работы с многомерными кубами, хранящимися на OLAP-сервере Microsoft Analysis Services, позволяющее просматривать данные куба, создавать различные визуализации в виде графиков и диаграмм и выполнять набор стандартных операций с многомерным кубом. Приложение разработано на языке программирования Java. Веб-интерфейс реализован средствами фреймворка AngularJS и библиотеки ChartJS. Также были применены различные технологии, предназначенные для написания веб-приложения на языке Java, такие как Spring Boot, Spring MVC, а также для работы с многомерными базами данных Olap4j.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 *Барсегян, А. А.* Методы и модели анализа данных: OLAP и Data Mining / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. — СПб.: БВХ-Петербург.
- 2 Общее представление о кубах OLAP [Электронный ресурс]. — URL: [https://msdn.microsoft.com/ru-ru/library/hh916543\(v=sc.12\).aspx](https://msdn.microsoft.com/ru-ru/library/hh916543(v=sc.12).aspx) (Дата обращения 30.05.2018). Загл. с экр. Яз. рус.
- 3 Введение в OLAP, часть 8. Обзор MDX [Электронный ресурс]. — URL: [http://www.olap.ru/basic/olap\\_intro8.asp](http://www.olap.ru/basic/olap_intro8.asp) (Дата обращения 30.05.2018). Загл. с экр. Яз. рус.
- 4 *Куинн, С.* SQL Server Analysis Services 2005 и MDX для профессионалов / С. Куинн, С. Харинатх. — М.: WROX.
- 5 olap4j Specification [Электронный ресурс]. — URL: [http://www.olap4j.org/olap4j\\_fs.html](http://www.olap4j.org/olap4j_fs.html) (Дата обращения 30.05.2018). Загл. с экр. Яз. рус.
- 6 10 преимуществ использования фреймворка Angular.js при разработке веб-приложений [Электронный ресурс]. — URL: <https://stfalcon.com/ru/blog/post/why-use-angularjs-for-webapps> (Дата обращения 30.05.2018). Загл. с экр. Яз. англ.