

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
Институт космических и информационных технологий  
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Г. М. Цибульский  
подпись

«\_\_\_\_\_» \_\_\_\_\_ 2016 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.02.04 «Информационные системы и технологии  
в медиаиндустрии»

Информационная система учета и прогнозирования  
движения товарно-материальных ценностей

Руководитель \_\_\_\_\_ зав. каф. ИС, канд. пед. наук С. А. Виденин  
подпись, дата

Выпускник \_\_\_\_\_ Р. Н. Горянский  
подпись, дата

Нормоконтролер \_\_\_\_\_ М. А. Аникьева  
подпись, дата

Красноярск 2016

## СОДЕРЖАНИЕ

Введение.....	3
1 Обзорно-аналитическая часть.....	5
1.1 Принципы построения OLAP-куба .....	5
1.2 Обзор существующих программных продуктов OLAP-решений .....	7
1.2.1 Аналитическая платформа Deductor (компания Base Group) .....	8
1.2.2 Аналитическая система ВІХ ВІ (компания ВІХ).....	10
1.2.3 Сравнительный анализ продуктов .....	12
1.3 Обзор систем прогнозирования.....	13
1.3.1 Azure Machine Learning от Microsoft .....	14
1.3.2 Линии тренда в программе Microsoft Office Excel.....	17
1.4 Выводы по главе 1 .....	19
2 Выбор среды разработки и инструментария .....	21
2.1 Выбор среды программирования .....	21
2.2 Выбор и обоснование используемой базы данных .....	26
2.3 Выводы по главе 2 .....	28
3 Разработка информационной системы учета и прогнозирования .....	29
3.1 Информационная система учета .....	29
3.2 Проектирование базы данных и системы анализа .....	35
3.3 Проектирование системы прогнозирования .....	44
3.4 Выводы по главе 3 .....	48
Заключение .....	50
Список использованных источников .....	51
Приложение А .....	54

## ВВЕДЕНИЕ

Формирование и развитие рыночных отношений в современной экономике требует значительного повышения эффективности управления деятельностью различных хозяйственных субъектов. При этом повышение эффективности в большой степени зависит от того, насколько корректно ведется управленческий учет и планирование на хозяйственном субъекте. Эффективность управления деятельностью, в свою очередь, оказывает огромное влияние на величину приносимого предприятиями дохода. И соответственно, доход от деятельности подобных экономических субъектов в значительной степени определяет динамику рыночных продаж и соотношение между спросом и предложением товаров. Отсюда вытекает необходимость создания информационных систем учета и прогнозирования, позволяющих управляющему персоналу различных экономических предприятий, являющихся хозяйственными субъектами, более оперативно анализировать различные аспекты своей деятельности и принимать более верные управленческие решения.

К тому же прогнозирование является одним из самых важных аспектов ведения успешного бизнеса. Планирование продаж является одним из элементов в цепи общей стратегии развития предприятия.

Торговая марка «Эмили», производящая питьевые товары, является экономическим объектом, для которого необходимо ведение полноценного учета и оперативного анализа данных. Принимая во внимание факт значительного увеличения продаж данной компании за последние месяцы, также становится очевидна необходимость введения алгоритма прогнозирования, который позволил бы скорректировать деятельность системы складов предприятия. Соответственно, система учета и прогнозирования позволит компании грамотнее планировать пополнение складских помещений товарами, что позволит избежать проблем недостатка товаров при заключении сделок, либо же наоборот, переизбытка изделий на складах. Также следует

уточнить, что менеджеры предприятия на данный момент не обладают достаточным опытом планирования продаж, а используемая в компании система «1С: Предприятие» предлагает лишь линейные алгоритмы, не позволяющие в определенных случаях получить корректные результаты.

Таким образом, проектируемая система должна предоставлять пользователям возможность вести полноразмерный учет собственной продукции и производить оперативный анализ продаж с использованием OLAP-технологий, так как данная технология является одной из самых перспективных. И на основе этого анализа в дальнейшем стало бы возможным планирование взаимодействия с поставщиками и клиентами.

В результате целью данной бакалаврской работы является автоматизация учета и разработка системы прогнозирования движения товарно-материальных ценностей. Задачами проекта являются:

- анализ предметной области;
- разработка системы учета;
- проектирование базы данных;
- разработка системы анализа данных на основе OLAP-технологий;
- разработка системы прогнозирования данных.

## **1 Обзорно-аналитическая часть**

В данной главе произведен обзор существующих на современном рынке систем анализа данных и систем прогнозирования, а также приведена краткая информация об OLAP-технологиях.

### **1.1 Принципы построения OLAP-куба**

Технологии OLAP являются одной из наиболее эффективных средств анализа больших объемов данных. Общее определение данному средству можно дать следующим образом. OLAP (OnLine Analytical Processing — аналитическая обработка данных в реальном времени) — подход к аналитической обработке данных, базирующийся на их многомерном иерархическом представлении [1].

С точки зрения пользователя, OLAP-системы предоставляют функции гибкого просмотра информации, автоматического получения агрегированных данных, выполнения аналитических операций свёртки, детализации, сравнения во времени.

Ключевые требования, предъявляемые к OLAP-системам — скорость, предоставляющая пользователям возможность интерактивной работы с информацией, и многомерность. В этом смысле OLAP-системы отличаются от, во-первых, традиционных реляционных систем управления базами данных (далее — РСУБД), результаты запросов из которых, как правило, довольно затратны по времени ожидания и загрузке, что приводит к снижению эффективности интерактивной работы с подобными базами (в частности, при наличии больших массивов информации) [2]. Во-вторых, OLAP-системы отличаются от плоскофайлового представления данных, например, в виде обычных традиционных электронных таблиц, представление многомерных данных в которых сложно.

Для описания концепции OLAP в настоящее время используется так называемый тест FASMI (Fast Analysis of Shared Multidimensional Information — быстрый анализ доступной многомерной информации), довольно точно характеризующий требования к OLAP-системам [3]. Соответственно, необходимо подробнее рассмотреть все требования.

Fast (быстрый) — данный пункт представляет требования пользователей OLAP-систем к скорости реакции системы. Промежутки с момента создания различных запросов до момента их выполнения должны измеряться секундами. Значимость данного требования увеличивается при использовании подобных систем в качестве средства оперативного представления данных, так как большие по длительности интервалы ожидания могут оказать негативное воздействие на цепочку рассуждений пользователя;

Analysis (анализ) — данный пункт определяет тот факт, что система должна быть способна производить любой статистический анализ, который является характерным для поставленных задач, а также гарантировать его сохранение в виде, доступном для конечного пользователя;

Shared (доступность, общедоступность) — данный пункт отражает требование к предоставлению системой возможности многопользовательского одновременного доступа;

Multidimensional (многомерность) — данный пункт является ключевым требованием концепции OLAP-технологий. Он отражает тот факт, что система должна предоставлять полную поддержку многомерного иерархического представления информации как наиболее эффективного пути анализа данных. Многомерность в рамках OLAP-технологий предполагает концептуальное представление данных в виде многомерной структуры, называемой гиперкуб. Рёбрами в подобном кубе выступают измерения (dimension), а данные (measures) расположены на пересечении осей измерений. При этом измерение обычно представляет собой плоский или иерархический список.

Information (информация) — данный пункт отражает требования к наличию в системе всех относящихся к целям пользователя данных, при этом

наличие «лишних» данных снижает эффективность, которую обеспечивает скорость реакции системы.

На рисунке 1 представлен пример OLAP-куба.

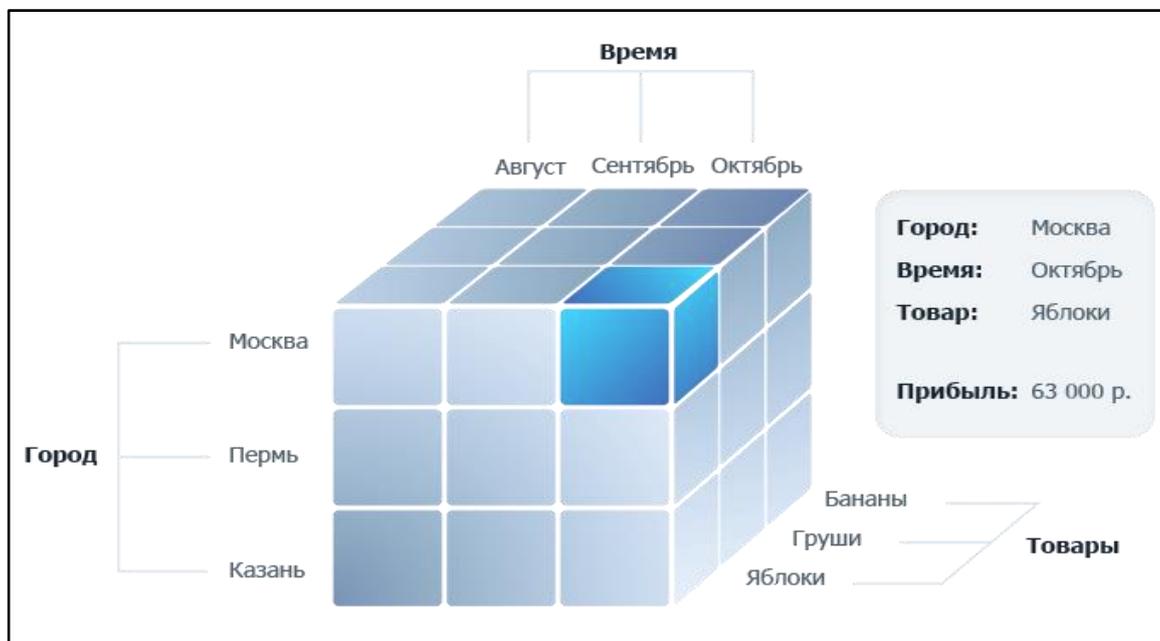


Рисунок 1 — Пример OLAP-куба

Как правило, OLAP-кубы в большинстве случаев содержат различные бизнес-показатели, используемые для анализа и принятия каких-либо управленческих решений, таких как: собственные и заемные средства, прибыль, рентабельность продукции.

Подобные показатели хранятся в кубах в виде разрезов, которые представляют собой основные категории деятельности организации: товары, склады, клиенты, даты сделок.

## 1.2 Обзор существующих программных продуктов OLAP-решений

Разработка различных (в том числе и серверных) OLAP-систем преимущественно происходит в несколько этапов.

На первом этапе данные извлекаются из базы данных экономического объекта (предприятия) в так называемое SQL-хранилище (база данных SQL

Server) , и затем модифицируются в необходимый для анализа вид. Зачастую на данной стадии возникают определенные проблемы, связанные с тем, что многие программные продукты (например, программная платформа 1С) используют систему автоматических названий объектов, поэтому при переносе в базу данных SQL их структура может быть изменена.

На втором этапе по данным, хранящимся в хранилище, проектируются OLAP-кубы.

На третьем этапе происходит настройка пользовательских средств доступа к аналитическим базам для клиентов, настройка отчетности и т.д.

Разработка OLAP-решений для различных предприятий ведется, чаще всего, фирмами, специализирующимися в области обработки данных.

Рассмотрим некоторые продукты OLAP-решений, представленных на рынке в настоящий момент.

### **1.2.1 Аналитическая платформа Deductor (компания Base Group)**

Аналитическая платформа Deductor ориентирована на решение различных задач с помощью методов Data Mining. Данный продукт в качестве основы использует собственные программные и аналитические разработки. OLAP-анализ не является приоритетным направлением данного продукта, но тоже присутствует. К тому же важной особенностью является возможность многоступенчатого преобразования информации, которая описывается в режиме конструктора. Тем не менее, в большей степени продукт ориентирован на использование математических методов для решений таких задач, как прогнозирование продаж, анализ клиентской базы, т.е. задач бизнес-анализа [4].

На данный момент платформа Deductor способна решать задачи аналитики в разнообразных областях, таких как:

- Банки и финансы;
- Розничная торговля;
- Оптовая торговля;

- Телекоммуникации;
- Промышленность;
- Информационные технологии;
- Медицина;
- Наука и образование.

Основные компоненты платформы Deductor:

- Deductor Warehouse — это многомерное хранилище данных, производящее всю требуемую для анализа информацию. Использование единого хранилища позволяет обеспечить пользователям централизованное хранение данных, а также автоматически обеспечивает всю необходимую поддержку процесса анализа.

- Deductor Studio — программа, реализующая функции импорта, обработки, визуализации и экспорта данных. Она может функционировать и без хранилища, получая информацию из любых других источников, но наиболее оптимальным является их совместное использование. В Deductor Studio включен полный набор механизмов, позволяющий получить информацию из произвольного источника данных, провести весь процесс обработки и отобразить полученные результаты наиболее удобным образом;

- Deductor Viewer — программа, ориентированная на конечного пользователя и предназначенная для просмотра подготовленных при помощи Deductor Studio отчетов;

- Deductor Server — служба, которая обеспечивает удаленную аналитическую обработку данных;

- Deductor Client — клиент доступа к службе Deductor Server. Он обеспечивает доступ к серверу из сторонних приложений и управление его работой.

Среди главных преимуществ платформы Deductor можно выделить:

- встроенная интеграция со множеством различных источников данных;
- высокая скорость работы с данными;
- поддержка большинства современных технологий анализа

структурированных данных.

Общая схема архитектуры аналитической платформы Deductor показана на рисунке 2.

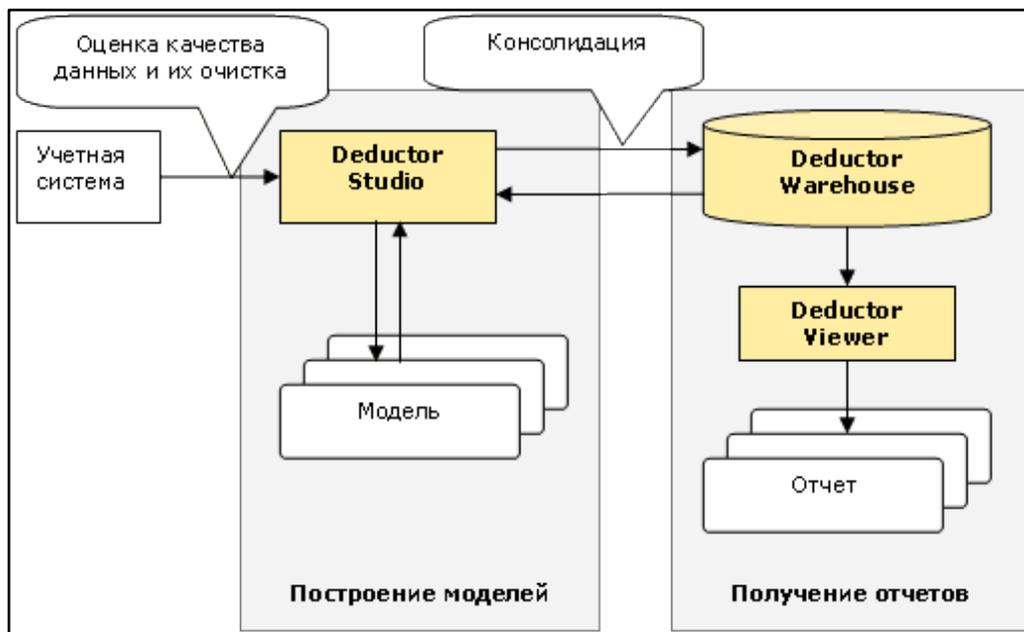


Рисунок 2 — Архитектура платформы Deductor (Base Group)

Как было упомянуто выше, данная платформа использует собственное программное обеспечение для построения отчетов и аналитических вычислений.

### 1.2.2 Аналитическая система ВIX ВI (компания ВIX)

Фирма ВIX предоставляет комплексное решение на платформе Microsoft SQL Server для анализа данных с использованием OLAP-технологий. Для работы с OLAP-кубами можно использовать программу Excel или какой-либо другой OLAP-клиент. Кроме построения кубов, также существует возможность воспользоваться средствами MS SQL Reporting Service для генерации отчетности [5].

Основные компоненты системы ВIX ВI:

- Корпоративное хранилище данных Microsoft SQL Server — данный модуль предоставляет высокопроизводительную оптимизированную базу данных, поддерживающую процедуры загрузки, очистки и преобразования данных.

- Система корпоративной отчетности Reporting Services — данный модуль предоставляет пользователю графические средства создания новых отчетов. Также присутствует поддержка автоматической рассылки отчетов по почте.

- Многомерные витрины данных Microsoft Analysis Services — данный модуль предоставляет аналитические данные, применяемые в бизнес-отчетах и клиентских приложениях, таких как Excel, отчеты служб Reporting Services. Также с помощью данного модуля возможно построение различных аналитических моделей на основе каких-либо табличных данных.

- Интранет-портал Sharepoint — данная платформа предоставляет различным пользователям возможность совместного доступа к данным, или возможность совместной работы.

Общая схема архитектуры системы BIX BI показана на рисунке 3.

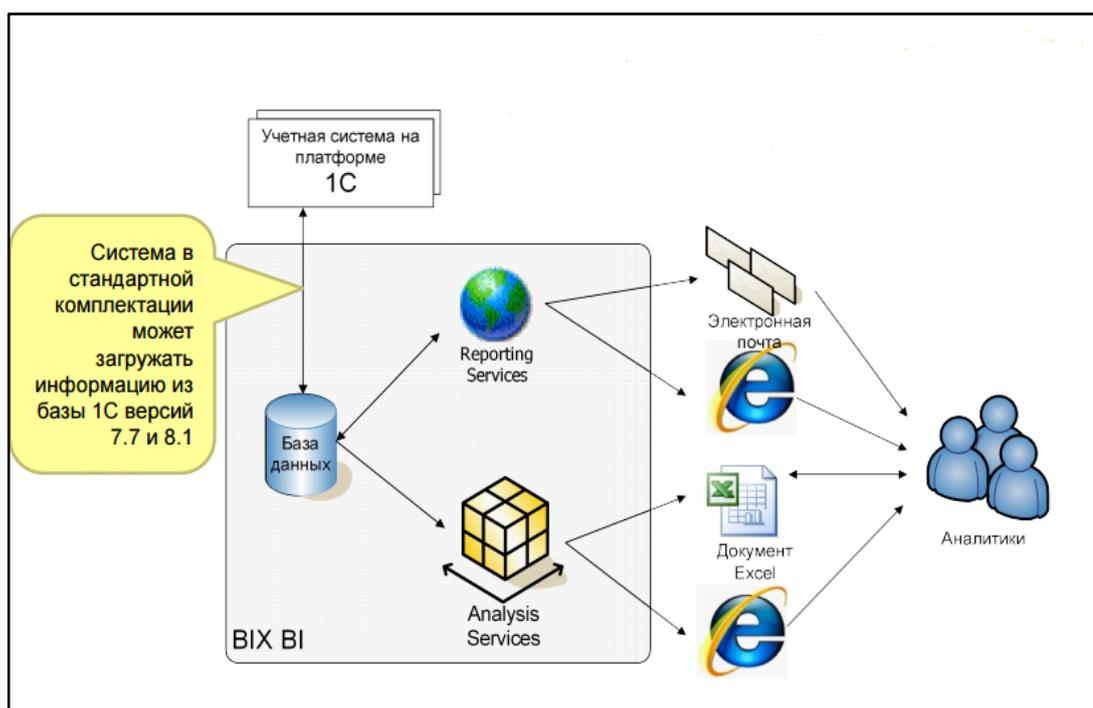


Рисунок 3 — Архитектура продукта BIX BI

Главные преимущества системы ВІХ ВІ перед другими системами анализа данных:

- при построении разных срезов данных в рамках созданных кубов учетная система не нагружается;
- для повторной развертки кубов, как правило, требуется всего несколько часов вне зависимости от времени изменения учетной системы;
- знакомый интерфейс программы Excel и механизм сводных таблиц и сводных диаграмм.

### 1.2.3 Сравнительный анализ продуктов

Подводя итог, следует отметить, что рассмотренные выше существующие системы обработки и хранения информации являются достаточно мощными инструментами для анализа данных, которые имеют превосходную интеграцию с системой 1С, используемой на предприятии «Эмили». Тем не менее, в приведенных ниже таблицах предоставлен результат стоимостного анализа аналитических продуктов, представленных на сегодняшнем рынке. Как видно из таблиц, оба предлагаемых продукта достаточно дороги для внедрения.

В таблице 1 приведена стоимость (в рублях) компонентов аналитической платформы Deductor (Base Group).

Таблица 1 — Стоимость компонентов Deductor (Base Group)

Компонент	Стоимость, рублей
Deductor Studio Enterprise	57820
Deductor Enterprise Suite Analytic Server	578200
Deductor Enterprise Suite Warehouse	бесплатно
Deductor Enterprise Suite Integration Server	578200
Deductor Enterprise Suite Viewer	13570

В таблице 2 приведена стоимость (также в рублях) компонентов программного продукта ВІХ ВІ от компании ВІХ.

Таблица 2 — Стоимость компонентов ВІХ ВІ

Компонент	Стоимость, рублей
Сервер класса C2Duo 6550/ DDR2 2048x2/ 250HSwap	10000–100000
Windows Server	10000–20000
SQL Server	10000–20000
Analysis Services	бесплатно
Reporting Services	бесплатно
Windows Sharepoint Services	бесплатно
Настройка системы	100000

Соответственно, ни один из рассмотренных вариантов для проектирования системы учета и прогнозирования товарно-материальных ценностей не подходит.

### 1.3 Обзор систем прогнозирования

Прогнозирование в любом его виде, как правило, должно основываться на определенных математических зависимостях категорий данных друг от друга. Применительно к предприятиям, занимающимся продажей и приобретением товаров, в большинстве случаев зависимость продаж зависит от календарного периода, что определяется различными величинами, такими как уровень спроса на товар в текущем месяце, регион, в котором происходит продажа продукции и т.д. Соответственно, проектируемая система прогнозирования должна определять уровень зависимости количества продаж от месяца.

Среди существующих на данный момент на рынке систем прогнозирования необходимо рассмотреть две: информационную систему Azure Machine Learning от компании Microsoft как одну из наиболее новых и перспективных, и программу Microsoft Office Excel, в частности, функцию создания линий тренда. Изначально в качестве варианта также рассматривалась

система Yandex Data Factory (далее YDT) от компании Yandex. Однако на данный момент система YDF работает только с крупными компаниями и не ориентирован на широкий круг пользователей, поэтому данный вариант был исключен.

### **1.3.1 Azure Machine Learning от Microsoft**

Azure Machine Learning — облачный сервис от компании Microsoft, предоставляющий пользователям возможность решения различных задач прогнозирования данных. Сервис представлен двумя компонентами: Azure ML Studio – средой разработки, доступной через web-интерфейс, и веб-сервисами Azure ML [6].

Стоит рассмотреть подробно каждый из основных пунктов алгоритма работы сервиса Azure Machine Learning.

Необходимо отметить, что проекты в среде разработки Azure ML Studio называются экспериментами.

Первый этап — получение данных. На данном этапе система позволяет загружать как подготовленные (структурированные) наборы данных, так и неподготовленные. Пользователи могут загружать информацию как из реляционных систем управления базами данных, так и данных из нереляционных источников: NoSQL, OData-сервисов, а также загружать различные текстовые документы. При этом возможен и ручной ввод данных. Для целей конвертации данных различных форматов служат элементы управления из раздела системы с названием Data Format Conversation. Доступны следующие форматы выходных данных: CSV, TSV, ARFF.

Второй этап — подготовка данных. В общем случае может возникнуть ситуация с неполными данными. Однако сервис Machine Learning позволяет пользователям удалять строки/столбцы, содержащие пропущенные данные, заменять пропущенные значения на другие показатели. Возможны также ситуации, что набор содержит повторяющиеся данные, которые, в свою

очередь, могут заметно снизить точность прогнозирования проектируемой модели. Система позволяет удалять дублирующиеся значения.

Третий этап — исследование данных. На данном этапе происходит несколько операций: модификация и очистка данных, определение структуры набора данных.

Модификация данных – данный пункт требует большого объема ручной работы пользователя, особенно, если данные для обучающего набора данных взяты из различных источников. Одно из преимуществ рассматриваемой системы: после загрузки данных пользователь не сталкивается с проблемами унифицированного доступа к различным источникам данным, а работает с информацией, полученной из различных источников, единообразно. Модификация подразумевает операции проецирования и группировки данных, добавления и удаления столбцов.

Определение структуры набора данных — система позволяет явно указать тип данных (string, integer, timestamp), содержащихся в определенных столбцах, отнести содержимое столбца либо к данным, на основе которых выполняется прогнозирование, либо к ответам.

Четвертый этап — деление данных. При использовании алгоритмов обучения с учителем как минимум один раз за эксперимент пользователю придется делить набор данных на два поднабора: обучающую выборку и тестовую. Для создания точной модели прогнозирования очень важно, чтобы обучающая выборка содержала максимально широкий спектр значений, который могут принимать переменные.

Пятый этап — построение модели. Одна из важнейших ступеней при работе системы. На данном этапе осуществляется выбор предикторов — данных, на основе анализа которых и строится прогноз. Данный этап имеет огромное влияние на точность полученной модели. Для корректного выбора предикторов пользователь должен обладать определенными знаниями в области математической статистики. На заключительном этапе уточнения

модели выбор предикторов часто осуществляется на основе экспертного мнения в исследуемой области.

Шестой этап — применение алгоритма машинного обучения. Он состоит из следующих стадий: инициализация модели, использующей определенный алгоритм машинного обучения, обучение модели, оценка полученной модели для обучающей и тестовой выборки, оценка полученного алгоритма. Среди представленных в сервисе Azure Machine Learning алгоритмов стоит выделить регрессию, классификацию, кластеризацию.

Седьмой этап — оценка модели. Результат оценки модели доступен в виде гистограммы, а также в виде различных статистических показателей, самым популярным среди которых является показатель точности предсказания, который рассчитывается как отношение всех успешных предсказаний к полному количеству предсказаний.

Восьмой этап — публикация модели. Модели в среде разработки Azure ML Studio можно опубликовать для их дальнейшего использования другими пользователями.

Подводя итоги, можно сказать, что Azure ML является мощной системой прогнозирования данных. Среди преимуществ данного облачного сервиса можно выделить возможность написания собственных программных инструментов (на языках программирования R и Python).

Недостатки данной системы:

- облачная технология — в данном случае необходим постоянный доступ к интернету. Помимо этого программный код данной технологии анализа данных может изменяться без ведома пользователя, что в ряде случаев может оказать негативный эффект;

- платность услуг — пользователям на выбор предлагается два режима использования системы: Free и Standart. Первый режим является практически бесплатным, но имеет огромное количество функциональных ограничений, которые не позволят производить все необходимые манипуляции с данными, в

то время как доступ ко второму режиму требует внесения определенных денежных средств;

- отсутствие русской локализации, что значительно затрудняет работу обычных пользователей с данной системой.

### **1.3.2 Линии тренда в программе Microsoft Office Excel**

Линии тренда в программе Microsoft Office Excel позволяют графически отображать различные тенденции данных и прогнозировать данные. Подобное выявление тенденций очень часто называется регрессионный анализ. Регрессионный анализ — это вид статистического анализа, применяемый для прогнозирования. Регрессионный анализ позволяет оценить уровень связей между различными переменными, и предлагает пользователю средства вычисления предполагаемого значения на основе имеющихся данных (и связей между ними). Соответственно, используя регрессионный анализ, можно продлить линию тренда за границы реальных данных для предсказания будущих значений.

Линиями тренда можно применять к наборам данных, представленных на ненормированных плоских диаграммах с областями, линейчатых диаграммах, гистограммах, графиках, биржевых, точечных и пузырьковых диаграммах. При этом нельзя добавлять линии тренда на объемные диаграммы, нормированные диаграммы, лепестковые диаграммы, круговые и кольцевые диаграммах. При замене корректного типа диаграммы на один из вышеперечисленных — например, изменении типа диаграммы на объемную — линии тренда, соответствующие данным, будут потеряны.

В зависимости от особенностей вводимых пользователем данных, стоит выбрать один из представленных вариантов, далее представлено описание видов линии тренда:

- экспоненциальная аппроксимация. Данная линия тренда используется, если у значений в наборе данных, по которым строится прогноз, происходит

непрерывное возрастание скорости изменения. В таком случае именно данная линия будет наиболее корректна. Однако если же данные, которые были введены в набор, содержат нулевые или же отрицательные характеристики, данный тип линии тренда неприемлем.

- линейная аппроксимация. Данная линия применяется, если величина значений данных в наборе увеличивается или уменьшается в относительном постоянстве. Соответственно, линия тренда в данном случае представляет собой прямую;

- логарифмическая аппроксимация. Данная линия тренда применяется, если величина значений в наборе данных сначала быстро увеличивается или уменьшается, а затем стабилизируется;

- полиномиальная аппроксимация. Характеристики, свойственные для данной линии тренда — переменное возрастание и убывание значений данных в наборе. Стоит отметить, степень самих полиномов (многочленов) определяется количеством максимумов и минимумом значений данных;

- степенная аппроксимация. Данная линия тренда характеризует монотонное возрастание и убывание величины значений данных, но использование ее невозможно в том случае, если данные имеют отрицательные и/или нулевые значения;

- скользящее среднее. Алгоритм данной линии используется, чтобы наглядно показать прямую зависимость одного значения от другого, путем сглаживания всех точек колебания. Это достигается путем выделения среднего значения между двумя соседними точками. Таким образом, график усредняется, а количество точек сокращается до значения, которое было определено пользователем.

Стоит отметить тот факт, что использование линий тренда весьма эффективно для аналитиков компаний, занимающихся продажей различных товаров

Пример линий тренда показан на рисунке 4.

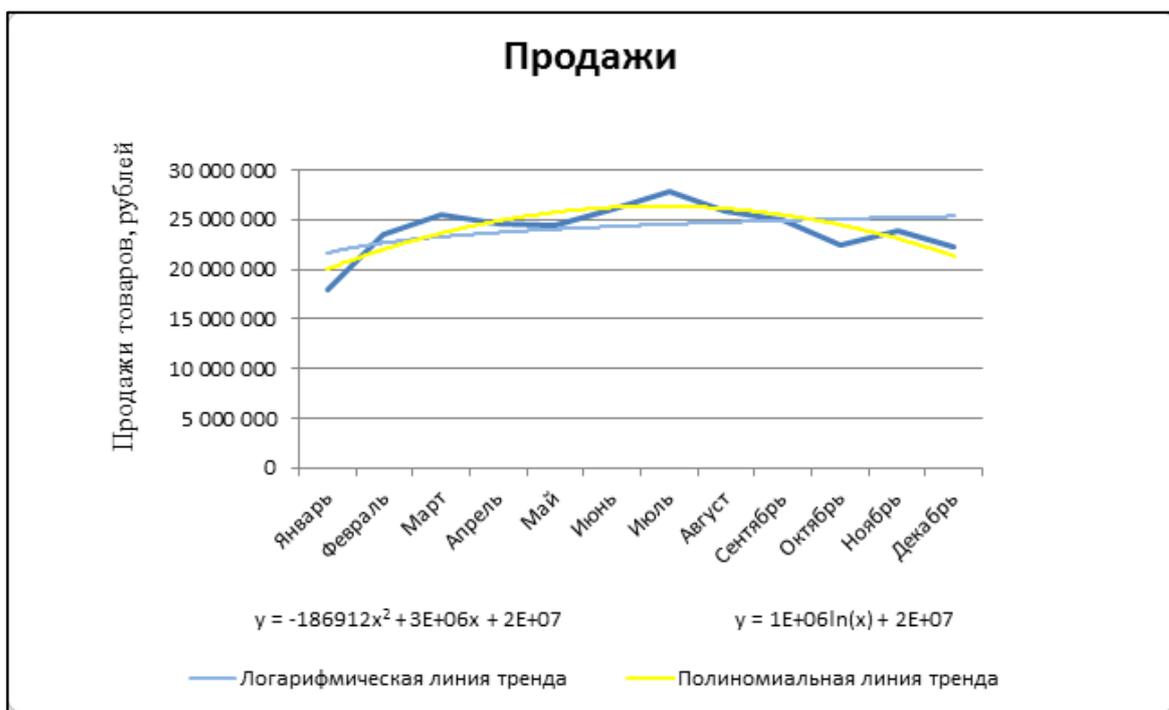


Рисунок 4 — Примеры линий тренда

Проведя сравнительный анализ перечисленных доступных способов прогнозирования данных, можно сделать определенные выводы. Сервис Azure Machine Learning от компании Microsoft обладает обширным выбором алгоритмов прогнозирования, но при этом имеет больше искусственных ограничений, и при этом является платной. Средства программы Microsoft Office Excel предоставляют меньшее количество алгоритмов, при этом обладая более понятным для обычного пользователя интерфейсом. Принимая во внимание тот факт, что разрабатываемый в рамках проекта OLAP-куб предусматривает функцию импорта в Excel-файлы, логичным выводом является тот факт, что использование программного продукта Excel в рамках проектируемой системы более корректно.

#### 1.4 Выводы по главе 1

В данной главе была поставлена цель и сформулированы задачи дипломного проекта. Была рассмотрена актуальность проекта, основные

правила и принципы OLAP-технологий, произведен обзор и анализ программных продуктов, используемых для построения OLAP-кубов, а также обзор существующих систем прогнозирования данных. В результате было принято решение создавать OLAP-куб вручную. В связи с этим, в качестве OLAP-сервера будет использоваться Microsoft SQL Server, в качестве клиента — Microsoft Excel. В качестве системы прогнозирования также были выбраны средства Microsoft Office Excel.

## **2 Выбор среды разработки и инструментария**

В данной главе произведен обзор существующих сред разработки программного обеспечения и инструментов для создания информационной системы учета и прогнозирования товарно-материальных ценностей.

### **2.1 Выбор среды программирования**

Среда разработки программного обеспечения — совокупность программных средств, используемая программистами для разработки программного обеспечения. Чаще всего среда разработки включает в себя компилятор и/или интерпретатор, редактор текста, отладчик и средства сборки. В том случае, если данные компоненты собраны в единый программный комплекс, подобная совокупность называется интегрированной средой разработки (Integrated Development Environment — IDE). В состав комплекса кроме перечисленных выше компонент могут входить средства управления проектами, система управления версиями, разнообразные инструменты для упрощения разработки интерфейса пользователя, стандартные заготовки («мастера»), упрощающие разработку стандартных задач.

Технология программирования во многом определяется языком программирования, на котором пишутся программы. В языке могут быть заложены средства, влияющие на технологичность и архитектуру разрабатываемой системы (например, объектно-ориентированность, модульность и т.п.). Соответственно, обычно выбирают ту модель разработки и те языки программирования, которые хорошо знакомы разработчику программного продукта.

Необходимо рассмотреть две приоритетные среды разработки: Java Developer Kit (далее — JDK) и Microsoft Visual Studio (далее — MS Visual Studio).

Среда разработки JDK — бесплатно распространяемый фирмой Sun комплект разработчика приложений на языке Java, включающий в себя компилятор Java (javac), стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java [7]. В состав среды JDK не входит интегрированная среда разработки, поэтому разработчики должны использовать любой внешний текстовый редактор и компилировать собственные программы, используя утилиты командной строки. Все современные интегрированные среды разработки на языке Java, такие, как NetBeans, Sun Java Studio Creator, IntelliJ IDEA, Borland JBuilder, Eclipse, опираются на сервисы, предоставляемые комплектом JDK. Поэтому эти среды разработки либо включают в собственные поставки одну из версий комплекта разработчика JDK, либо требуют для своей работы предварительной инсталляции комплекта на машине разработчика.

Среди основных преимуществ среды JDK, а также языка программирования Java стоит выделить:

- независимость от платформы (код различных программ можно выполнять на различных операционных системах, таких как Linux, Solaris, FreeBSD и т.д.);
- объектно-ориентированный подход к разработке;
- исключение для разработчиков возможности явного выделения и освобождения памяти, так как память в языке Java освобождается автоматически с помощью механизма сборки мусора).

При этом среди недостатков можно выделить:

- отсутствие наглядного визуального представления о структуре разрабатываемого проекта;
- относительно низкое быстродействие (по сравнению с другими языками программирования);
- достаточно высокие требования к объему оперативной памяти.

Среда разработки MS Visual Studio является продуктом компании Microsoft, который позволяет создавать приложения, работающие на

современной платформе .NET. Особенность этой платформы заключается в широком наборе сервисов, которые доступны в различных языках программирования [8]. При этом сервисы реализуются в виде промежуточного кода, который не зависит от базовой архитектуры. Едва ли не главной целью создания такой платформы было оснащение разработчиков программного обеспечения специальными сервисно-ориентированными приложениями, которые могли бы работать на любой платформе, начиная от персонального компьютера и заканчивая мобильным устройством.

На сегодняшний день данная среда разработки является предпочтительным выбором очень многих разработчиков, работающих на платформе Windows. Она позволяет эффективно и оперативно создавать сложные проекты за короткий промежуток времени. Также стоит отметить, что рассматриваемая среда разработки поддерживает огромное количество языков программирования, таких как C#, C++, Visual Basic, F#, TypeScript, JavaScript, Python.

Главные особенности среды MS Visual Studio:

- поддержка большого количества языков программирования;
- высокая скорость разработки;
- наличие инструментов отладки;
- поддержка объектно-ориентированного программирования;
- большой набор потенциальных архитектурных типов построения приложений;
- возможность настройки отдельного экземпляра среды Visual Studio для конкретного пользователя (поддержка нескольких мониторов, изменение четкости текста и т.д.);

Необходимо отметить, что среда MS Visual Studio обладает также некоторыми недостатками, такими как:

- высокие требования к вычислительным возможностям платформы, на которой ведется разработка;
- отсутствие кросс-платформенности.

Сравнивая рассмотренные среды разработки, логичным выводом является тот факт, что продукт компании Microsoft имеет значительно больше преимуществ перед продуктом от компании Sun, главным из которых является большое количество поддерживаемых языков программирования. В соответствии с этим, средой разработки для информационной системы учета и прогнозирования товарно-материальных ценностей целесообразно выбрать Microsoft Visual Studio. Также, в качестве языка программирования необходимо использовать Visual C#, так как данный язык обладает высокой скоростью разработки проектов (по сравнению с другими языками), простым и лаконичным синтаксисом, а также объектно-ориентирован.

Среда разработки MS Visual Studio с использованием языка программирования C# предоставляют разработчикам большой выбор типов проектов. К ним можно отнести создание консольных приложений, приложений Windows Forms (далее — WF), WPF (Windows Presentation Foundation) приложений и т.д. Однако принимая во внимание тот факт, что разрабатываемая в рамках дипломного проекта система должна предоставлять пользователю интуитивно понятный интерфейс и возможность интерактивного взаимодействия, подробному рассмотрению подлежат лишь приложения Windows Forms и веб-приложения ASP.NET (Active Server Pages для .NET).

Необходимо рассмотреть подробнее тип проекта Windows Forms. Приложения Windows Forms используются для создания настольных приложений, снабженных графическим интерфейсом. Основываются они на библиотеках класса .NET Framework и имеют намного более совершенную и удобную в работе модель программирования, чем, к примеру, программные интерфейсы Win32 API (Application Programming Interface) или MFC (Microsoft Foundation Classes).

В связи с этим, проекты Windows Forms рассматриваются как замена вышеупомянутой библиотеке MFC, изначально написанной на языке C++. С другой стороны, WF не предлагает парадигму, сравнимую с парадигмой MVC

(Model-View-Controller). Для исправления этой ситуации и реализации данной функциональности в WF существуют сторонние библиотеки.

Таким образом, Windows Forms — это набор различных управляемых библиотек, с помощью которых разработчики могут выполнить все необходимые для оконного приложения действия, начиная от обмена сообщениями с операционной системой для отслеживания любых событий клиентского окна, заканчивая диалоговыми системами, связью с другими компьютерами по сети и многими другими возможностями. Под формой понимается видимая поверхность окна, включающая информацию для конечного пользователя, а также содержащую в себе набор инструментов (элементов управления) для работы с представленными данными или взаимодействия с пользователем.

Среди особенностей архитектуры приложений Windows Forms следует отметить, что они являются событийно-ориентированными. Это означает, что, в отличие от обычных пакетных программ, большая часть времени тратится на ожидание от пользователя каких-либо действий, таких как ввод текста в текстовое поле или клика кнопкой мыши по какой-либо кнопке.

Также следует рассмотреть веб-приложения ASP.NET. Данная технология является способом создания веб-приложений и веб-сервисов от компании Майкрософт. Она является составной частью платформы Microsoft .NET и развитием более старой технологии от Microsoft Active Server Pages (ASP) [9]. Последней версией этой технологии является ASP.NET 5.

Технология ASP.NET внешне во многом сохраняет схожесть с более старой технологией ASP, что позволяет разработчикам относительно легко перейти на ASP.NET. В то же время внутреннее устройство ASP.NET существенно отличается от ASP, поскольку она основана на платформе .NET и, следовательно, использует все новые возможности, предоставляемые этой платформой.

Разработчики могут писать код для приложений ASP.NET, используя практически любые языки программирования, входящие в комплект

платформы .NET Framework (C#, Visual Basic.NET и JScript .NET). Технология ASP.NET имеет преимущество в скорости по сравнению со скриптовыми технологиями, так как при первом обращении код компилируется и помещается в специальный кэш, и впоследствии только исполняется, не требуя затрат времени на оптимизацию и т. д.

Особенности приложений технологии ASP.NET:

- компилируемый код выполняется быстрее, большинство ошибок отлавливается ещё на стадии разработки;
- улучшенная обработка ошибок во время выполнения запущенной готовой программы;
- пользовательские элементы управления позволяют выделять часто используемые шаблоны, такие как меню сайта;
- возможность кэширования всей страницы, или какой-либо её части для увеличения производительности;
- возможность кэширования данных, используемых на странице.

При выборе правильного типа проекта, необходимо принять во внимание тот факт, что разрабатываемая в рамках дипломного проекта система предназначена для ведения учета товаров, следовательно, будет применяться на складах, где отсутствует постоянный доступ к сети Интернет. При учете особенностей приложений Windows Forms и веб-приложений ASP.NET, настольное приложение WF в большей степени соответствует функциональным характеристикам разрабатываемой системы.

## **2.2 Выбор и обоснование используемой базы данных**

База данных (далее БД) — именованная совокупность данных, обладающая определенной структурой, компонентами которой могут быть любые структурные единицы данных (элементы, группы, записи, файлы), связанные между собой определенным образом [10]. Причем эти данные могут

использоваться как в одной, так и в нескольких задачах, одним или несколькими пользователями.

Отличительными признаками любой базы данных являются:

- модельность (БД должна моделировать определенные объекты реального мира);
- актуальность (БД должна отражать текущее состояние объектов реального мира, а также периодически обновляться с учетом изменений);
- непротиворечивость (данные в БД не должны противоречить друг другу);
- целостность (БД должна наиболее полно моделировать предметы окружающего мира);
- надежность (данные в БД должны быть защищены от любого искажения).

На данный момент существует несколько различных классификаций баз данных, таких как классификация по модели данных (сетевые, реляционные и т.д.), по среде физического хранения (оперативной памяти, вторичной памяти и т.д.), по содержанию (научные, исторические и т.д.) и другие. Рассмотрим более подробно классификацию по модели данных.

Определяющим понятием в данной классификации является понятие модели. Модель данных — это абстракция, которая, будучи приложенной к конкретным данным, позволяет пользователям и разработчикам трактовать их уже как информацию, то есть сведения, содержащие не только данные, но и взаимосвязь между ними. В соответствии с этой классификацией различают:

- иерархические БД (такие базы могут быть представлены в виде дерева, которое состоит из объектов разных уровней);
- сетевые БД (данные базы являются расширением иерархического подхода: разница состоит в том, что в сетевой модели объект, расположенный на каком-либо нижнем уровне может иметь более одного родительского объекта);

- реляционные БД (данные в подобных базах данных обычно представлены в виде набора отношений);

- объектные БД (информация в таких базах представлена в виде объектов);

- объектно-реляционные БД (подобные БД являются так называемым гибридом объектного и реляционного подходов).

Исходя из того, что хранилище данных для проектируемой системы должно состоять из взаимосвязанных таблиц использование реляционной модели данных является оптимальным вариантом.

Соответственно, в качестве используемой системы управления базами данных (далее — СУБД) следует использовать программный продукт Microsoft SQL Server [11]. Данная СУБД обладает следующими функциональными особенностями [12]:

- поддержка функций построения отчетов и анализа данных;
- высокая производительность;
- высокие показатели надежности, стабильности, масштабируемости и безопасности;
- поддержка интеграции с облачными системами.

### **2.3 Выводы по главе 2**

В данной главе был произведен обзор и анализ существующих на данный момент сред разработки программного обеспечения, а также существующих видов баз данных. Также был сделан выбор используемого при проектировании разрабатываемой информационной системы типа проекта. В результате было принято решение о принятии в качестве среды разработки платформы Microsoft Visual Studio, главным типом проекта было выбрано настольное приложение Windows Forms. Система управления базами данных Microsoft SQL Server выбрана в качестве используемой базы данных.

### **3 Разработка информационной системы учета и прогнозирования**

В данной главе рассматривается разработка систем учета и прогнозирования товарно-материальных ценностей.

#### **3.1 Информационная система учета**

Информационная система (далее — ИС) — система, предназначенная для хранения, поиска и обработки информации и соответствующие организационные ресурсы (человеческие, технические, финансовые и т. д.), которые обеспечивают и распространяют. Информационная система предназначена для своевременного обеспечения надлежащих людей надлежащей информацией, то есть для удовлетворения конкретных информационных потребностей в рамках определенной предметной области, при этом результатом функционирования информационных систем является информационная продукция — документы, информационные массивы, базы данных и информационные услуги [13].

С учетом функциональных особенностей разрабатываемой системы, необходимо ясное определение типа системы с точки зрения распределенности. По степени распределенности различают настольные (все компоненты находятся на одном компьютере) и распределенные ИС (компоненты распределены на нескольких компьютерах). Учитывая, что разрабатываемая ИС с наибольшей долей вероятности будет использоваться на одном из складов компании сотрудником-аналитиком, использование настольной ИС более корректно.

Также необходимо отметить, что по типу хранимых и используемых данных разрабатываемая система относится к фактографическим ИС. Подобные системы используются для хранения и взаимодействия со структурированными данными, которые чаще всего представлены в виде чисел

и какого-либо текста. Соответственно, с подобными данными очень легко выполнять различные операции.

Отличительной особенностью фактографических ИС является тот факт, что в основном подобные системы работают с фактическими сведениями. На базе подобных систем возможно создание справочников, систем анализа и управления предприятиями, отчетных систем и т.п. Фактические записи отражают конкретные значения данных (атрибутов) об объектах реального мира.

Наиболее полно функциональные возможности разрабатываемой ИС представлены с помощью диаграммы прецедентов на рисунке 5.

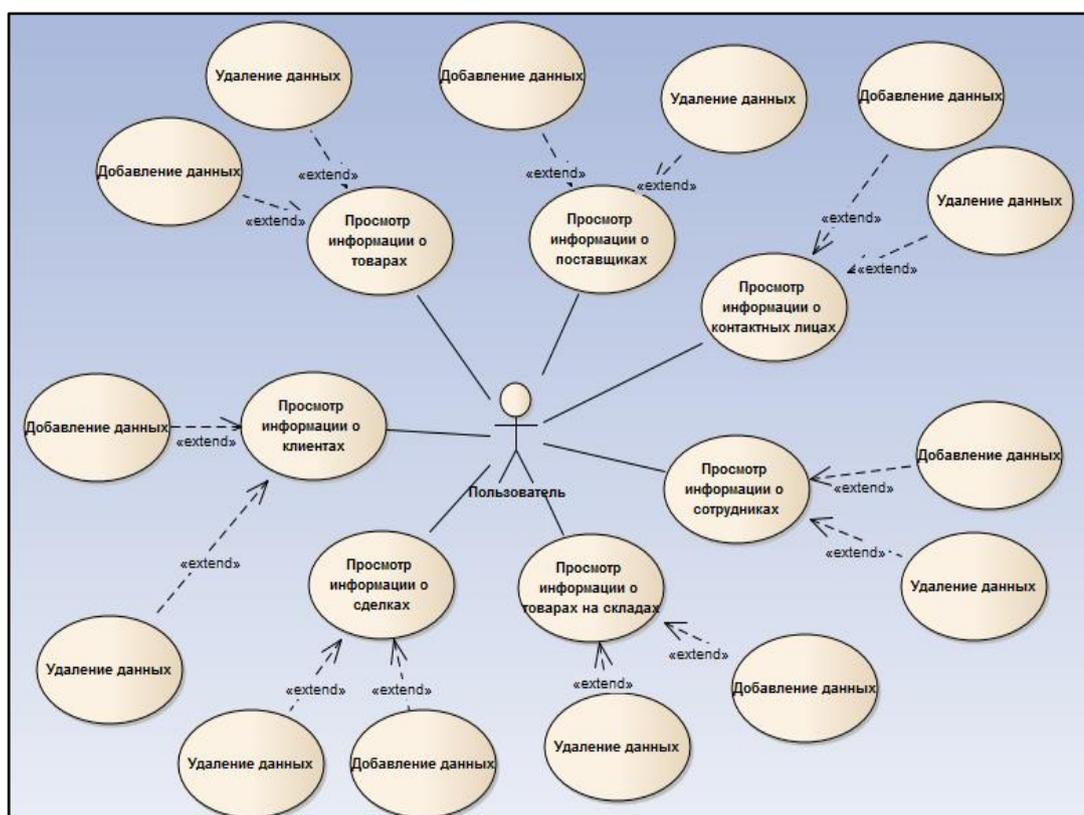


Рисунок 5 — Диаграмма прецедентов

При этом на основании диаграммы прецедентов необходимо создать диаграмму классов для визуального представления классов системы, их атрибутов и взаимосвязей между ними. Диаграмма классов представлена на рисунке 6.

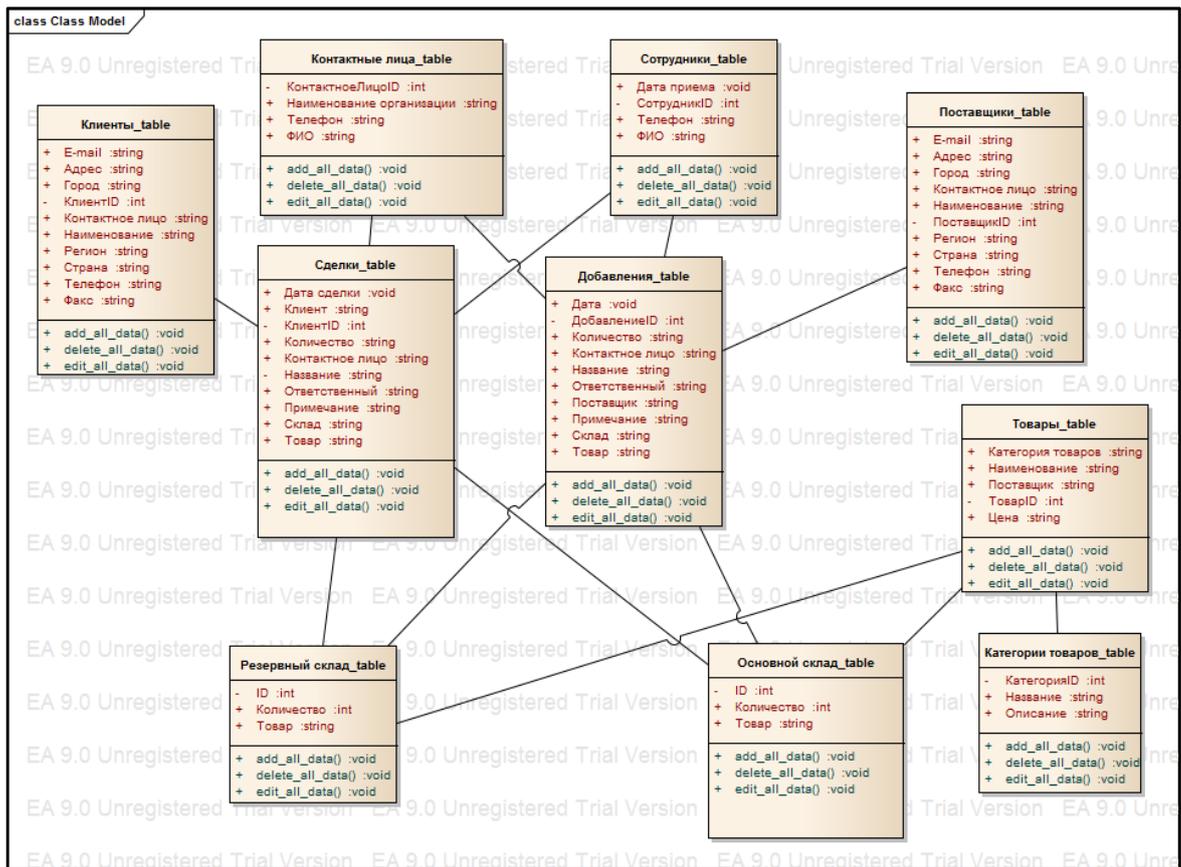


Рисунок 6 — Диаграмма классов

Рассмотрим структуру и интерфейс разрабатываемой ИС.

ИС учета товаров разработана в виде настолько приложения, состоящего из нескольких форм. Доступ ко всем формам предоставлен в главном окне системы. Общее число форм системы равно семи.

Форма «Товары» позволяет пользователю просмотреть, какие товары на данный момент продает компания, кто является поставщиком этих товаров, к какой категории они относятся, а также просмотреть точную денежную стоимость единичного экземпляра товара. Помимо этого пользователь может выполнять поиск по наименованию конкретного товара, добавлять данные о товарах в базу данных, заполнив именованные поля формы и нажав программную кнопку «Добавление». Выделив определенную строку данных, при нажатии кнопки «Удаление» происходит удаление выделенной строки. Все манипуляции с данными сохраняются при нажатии кнопки «Сохранить». Интерфейс формы «Товары» представлен на рисунке 7.

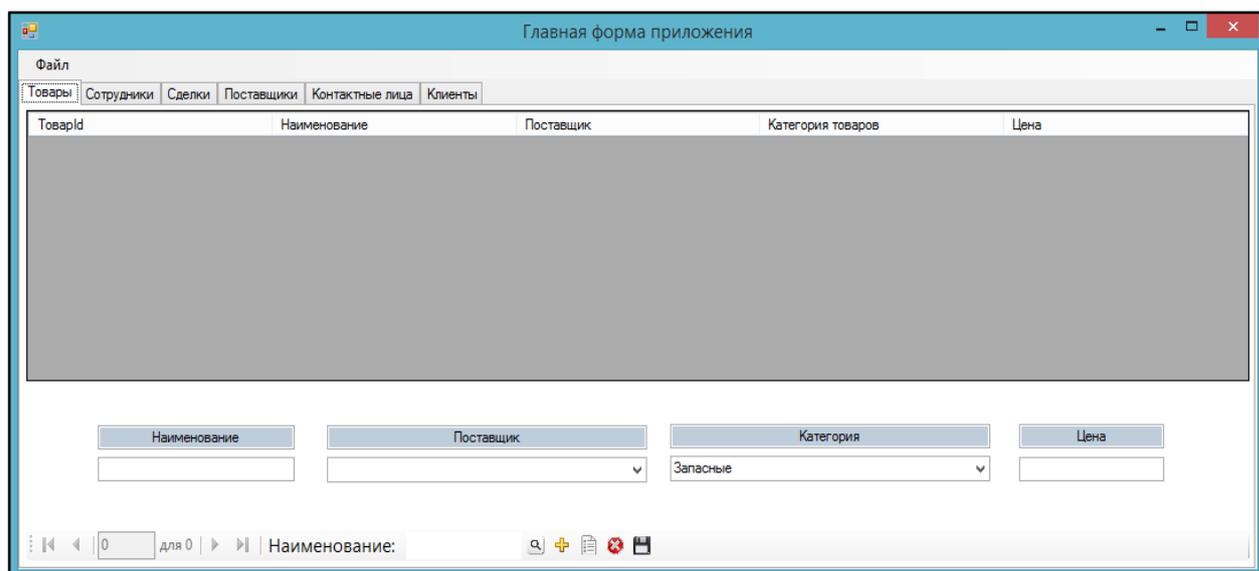


Рисунок 7 — Интерфейс формы «Товары»

Стоит отметить, что в дальнейшем при модификации системы возможно введение дополнительных атрибутов товаров. К числу подобных атрибутов можно отнести валюту и единицы измерения.

Форма «Сотрудники» предоставляет краткую информацию о сотрудниках компании. Пользователь может узнать фамилию, имя и отчество каждого сотрудника, а также дату приема сотрудника на работу в компанию. Соответственно, сотрудники, информация о которых находится в базе данных, занимают должности, связанные с продажей и покупкой товаров. Также можно просмотреть информацию о телефонных номерах сотрудников. В будущем, при возможной дальнейшей модификации системы учета данная форма даст возможность пользователю добавлять и удалять дополнительную информацию, такую как адрес проживания, дату рождения сотрудника, фотографию, а также персональный номер счета.

Функции поиска, добавления и удаления данных также доступны для данной формы. Интерфейс представлен на рисунке 8. Потенциально, данная форма и связанная с ней таблица в базе данных может стать основой для создания дополнительного измерения в проектируемом в рамках дипломной работы OLAP-кубе.

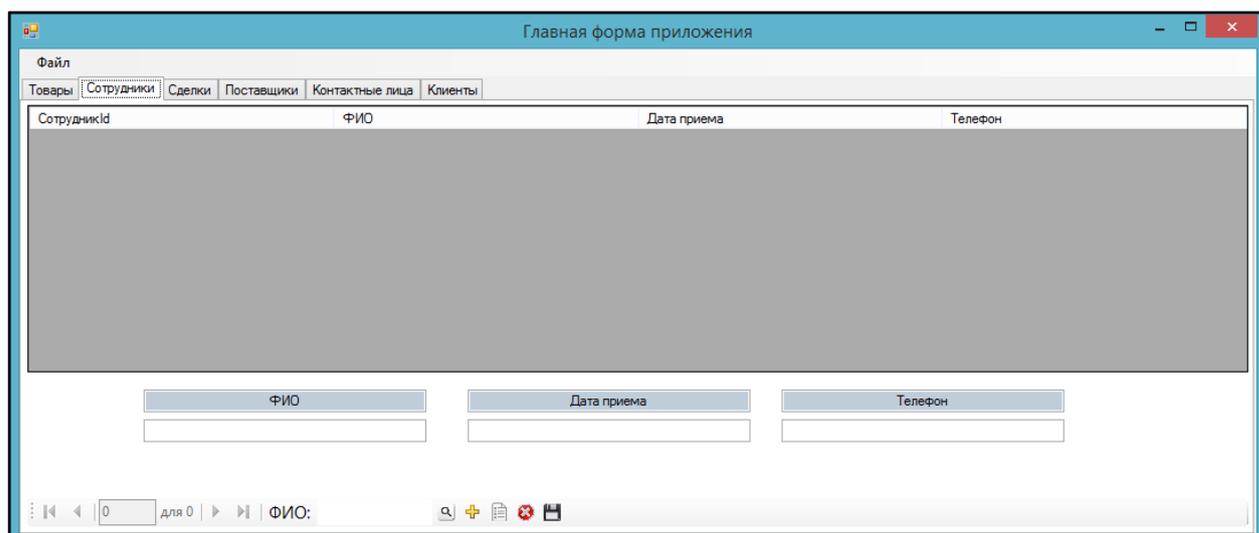


Рисунок 8 — Интерфейс формы «Сотрудники»

Форма «Сделки» является главной формой приложения. Пользователь может просмотреть данные обо всех сделках, произведенных компанией за все время. В число этих данных входит наименование клиента, которому был продан товар, количество товара, склад, с которого товар был отправлен клиенту, а также сотрудника, который произвел сделку. Также как на предыдущих формах, есть возможность поиска, добавления и удаления данных. Интерфейс формы представлен на рисунке 9.

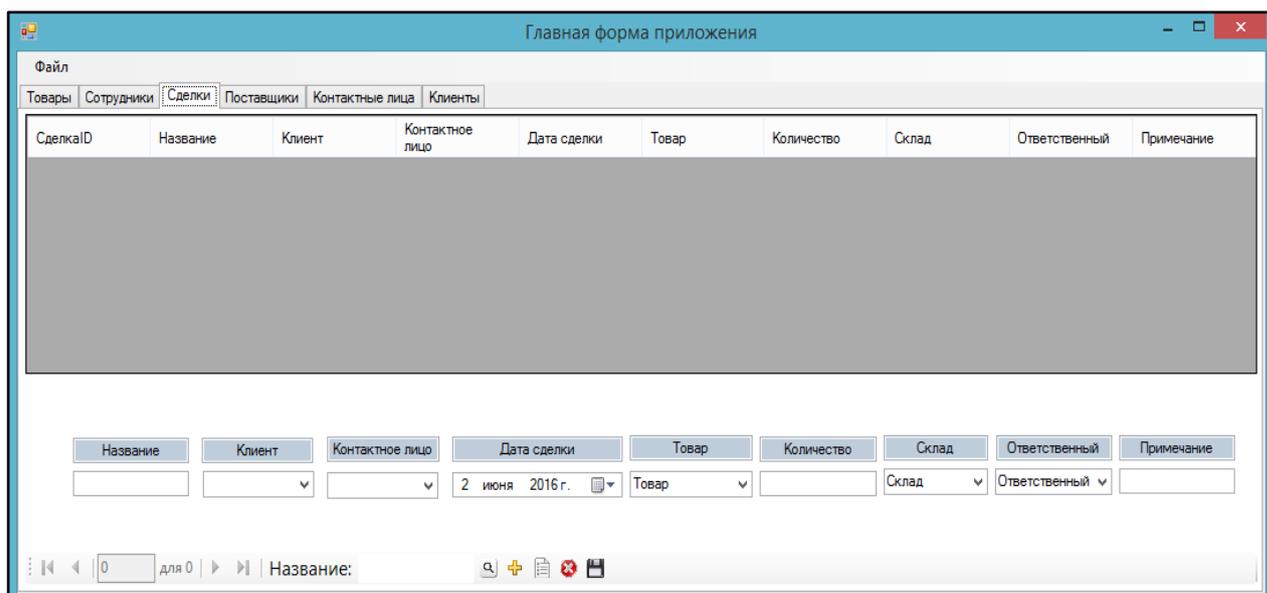


Рисунок 9 — Интерфейс формы «Сделки»

Форма «Поставщики» предоставляет пользователю подробную информацию о зарегистрированных поставщиках товаров, доставляющих товары на склады компании. Информация в базе указывает: полное наименование компании-поставщика, список контактных лиц, геолокационные данные, адрес электронной почты и т.п. Манипуляции с данными (удаление, добавление и поиск) также возможны, как и в других формах. Интерфейс формы представлен на рисунке 10.

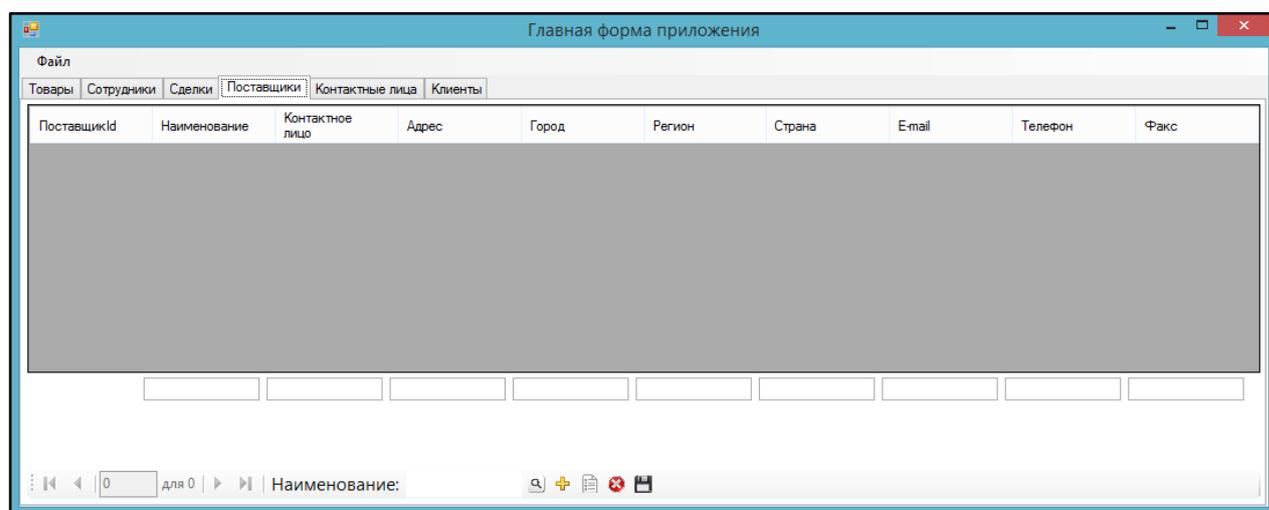


Рисунок 10 — Интерфейс формы «Поставщики»

Форма «Контактные лица» содержит информацию о контактных лицах клиентов и поставщиков компании. Данная информация включает фамилию, имя, отчество, компанию, в которой данное контактное лицо занимает должность. Операции поиска, редактирования и удаления данных также возможны при использовании соответствующих кнопок. Интерфейс формы «Контактные лица» предоставлен на рисунке 11.

В дальнейшем, при возможной модификации системы учета по аналогии с формой «Сотрудники» возможно добавление дополнительной информации о контактных лицах клиентов и поставщиков. К числу такой информации относятся адрес проживания контактного лица, электронный адрес, факс. Также планируется внедрение в систему информационного атрибута «Статус», который позволит определять наиболее активных контактных лиц.

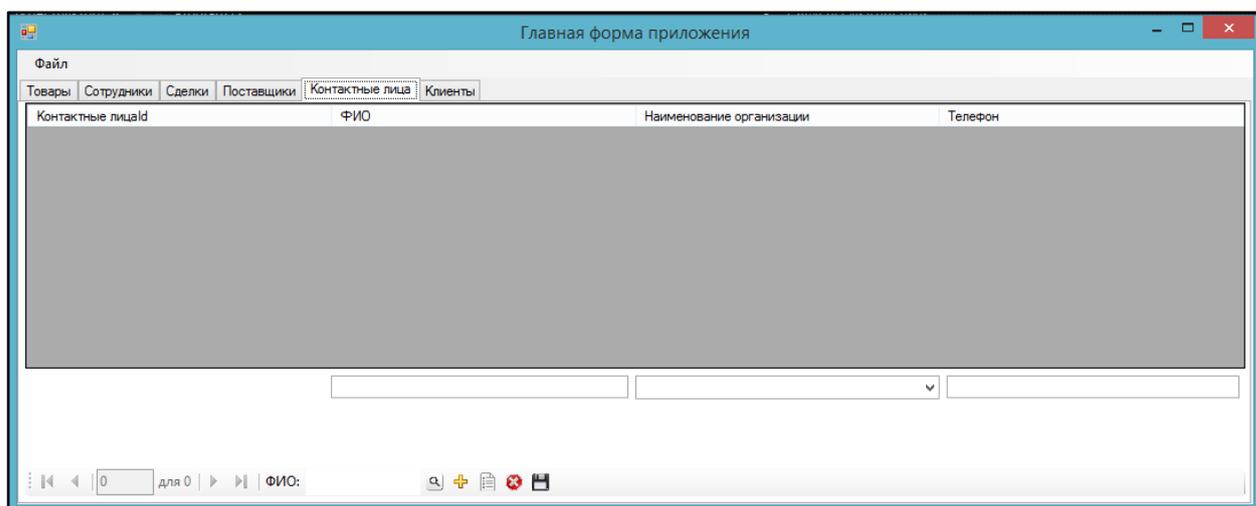


Рисунок 11 — Интерфейс формы «Контактные лица»

Стоит отметить, что функциональность и интерфейс формы «Добавление товаров» идентичны таковым у формы «Сделки». Форма «Клиенты» аналогична форме «Поставщики».

Таким образом, были рассмотрены функциональность и интерфейс системы учета данных.

### 3.2 Проектирование базы данных и системы анализа

Информационные системы различных предприятий очень часто содержат приложения, предназначенные для комплексного многомерного анализ данных, динамики данных и т.д. В конечном счете, такой анализ должен содействовать принятию управленческих решений. Соответственно, очень часто такие системы носят название — системы поддержки принятия решений.

Принять любое управленческое решение не представляется возможным, если не обладать для этого необходимой информацией, в большинстве случаев количественной [14]. Соответственно, создание базы данных является необходимым атрибутом создания ИС учета и прогнозирования.

При этом стоит рассмотреть понятие хранилища данных, очень тесно связанное с системами анализа данных. Авторы концепции хранилищ данных, описывают хранилище данных как «место, где люди могут получить доступ к

своим данным» [15]. Ими же были и сформулированы основные требования к хранилищам данных:

- присутствие высокой скорости получения данных из хранилища;
- поддержка внутренней непротиворечивости данных;
- возможность получения и сравнения так называемых срезов данных;
- наличие удобных утилит просмотра данных в хранилище;
- полнота и достоверность хранимых данных;
- поддержка качественного процесса пополнения данных.

Тем не менее, удовлетворять всем вышеперечисленным требованиям зачастую не удастся в рамках проектирования какого-либо программного продукта. Поэтому очень часто для реализации хранилищ данных используется несколько продуктов. Одни представляют собой средства хранения информации, другие — средства их извлечения и просмотра, а третьи — средства их пополнения и т.д.

При этом, обычное хранилище данных, как правило, отличается от типичной реляционной базы данных. Во-первых, обычные базы данных предназначены для того, чтобы помогать пользователям выполнять повседневную работу, тогда как хранилища данных предназначены для принятия решений. Например, продажа товара и выписка со счета производится с использованием базы данных, предназначенной для обработки транзакций, а анализ продаж на несколько лет, позволяющий спланировать работу с поставщиками — с помощью хранилища данных.

Во-вторых, обычные базы данных зачастую подвержены постоянным изменениям в процессе работы пользователей, а хранилища данных относительно стабильно: данные в нем обычно обновляются согласно расписанию. В идеальном варианте использования, процесс пополнения представляет собой просто добавление новых данных за определенный период времени без изменения прежней информации, уже находящейся в хранилище.

В-третьих, обычные базы данных чаще всего являются источником данных, попадающих в хранилище. Кроме того, хранилище может пополняться за счет внешних источников.

Однако с учетом функциональных особенностей разрабатываемой в рамках дипломного проекта системы, в качестве источника данных для системы OLAP-анализа данных корректнее использование обычной реляционной базы. Причина заключается в том, что пользователю системы необходимо производить анализ собственных продаж товаров, которые находятся в режиме постоянного изменения, соответственно, проектирование хранилища данных в нашем случае неэффективно.

Конечной целью использования OLAP-технологий является анализ данных и представление результатов этого анализа в виде, удобном для восприятия и принятия решений. Основная идея OLAP-систем заключается в построении многомерных кубов, которые будут доступны для пользовательских запросов [16]. При этом исходные данные чаще всего хранятся в вышеупомянутых реляционных базах данных.

В разрабатываемой базе данных должна храниться информация о совершенных сделках (таблица «Сделки»), о добавлениях товаров на склады (таблица «Добавления»), о продаваемых товарах (таблица «Товары»), о текущем состоянии складов (таблицы «Основной склад» и «Резервный склад»), о категориях товаров (таблица «Категории товаров»), о сотрудниках компании (таблица «Сотрудники»), о клиентах компании (таблица «Клиенты»), о поставщиках товаров (таблица «Поставщики»), а также о контактных лицах клиентов и поставщиков компании (таблица «Контактные лица»).

Стоит отметить, что в будущем используемая база данных может измениться. Данный факт связан с тем, что в дальнейшем в систему учета потенциально необходимо будет добавлять дополнительную информацию в виде изображений и видеофайлов. Также в дальнейшем возможно введение двухуровневой структуры базы данных.

Структура разрабатываемой базы данных представлена на рисунке 12.

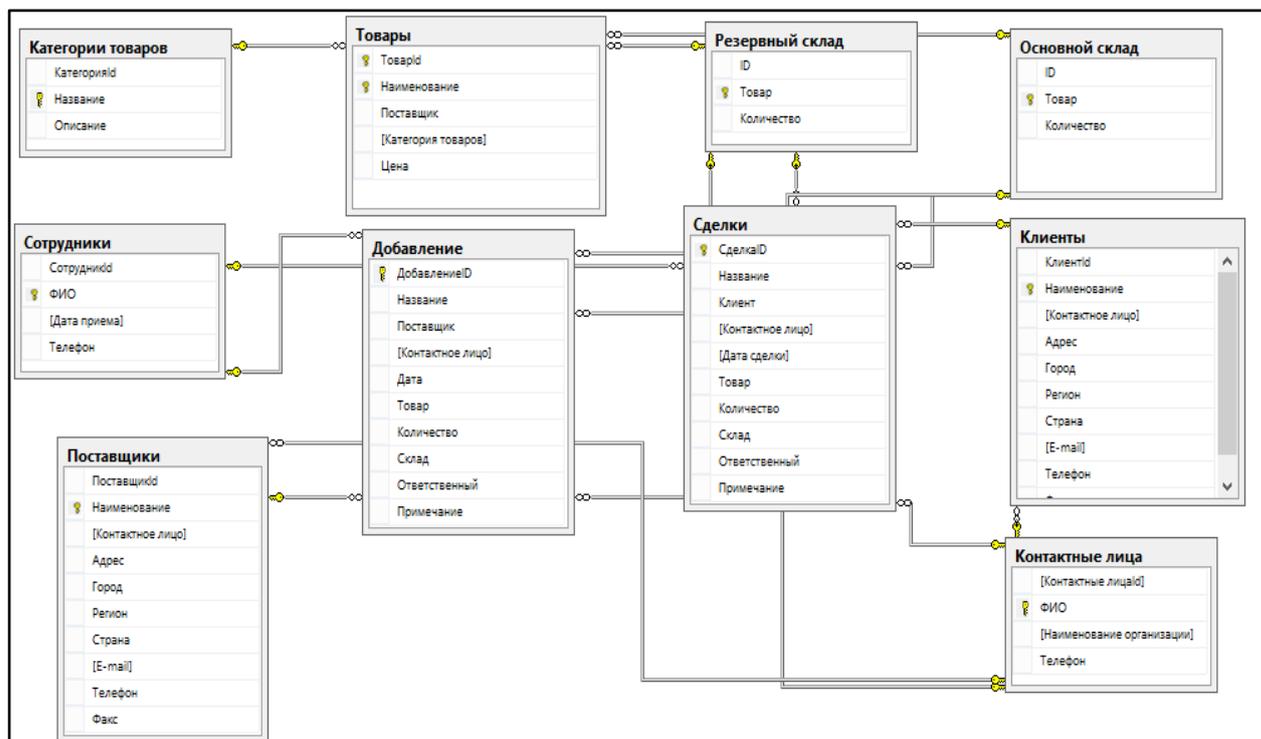


Рисунок 12 — Структура разрабатываемой базы данных

Ниже приведено описание всех полей таблиц.

Таблица «Сделки» содержит поля:

- СделкаID — уникальный идентификатор каждой сохраненной и произведённой сделки;
- Название — название каждой сделки;
- Клиент — наименование компании, с которой заключена сделка;
- Контактное лицо — Ф.И.О контактного лица клиента, с которым заключается сделка;
- Дата сделки — точная дата заключения сделки (вплоть до указания часов и минут);
- Товар — наименование товара, который должен быть доставлен клиенту;
- Количество — количество товара, который должен быть доставлен клиенту;
- Склад — наименование склада, с которого товар должен быть продан клиенту;

- Ответственный — Ф.И.О сотрудника компании, ответственного за выполнение сделки;

- Примечание — различная информация о сделке, которая сохраняется в базу данных.

Таблица «Добавление» содержит поля:

- ДобавлениеID — уникальный идентификатор каждого сохраненного и произведенного добавления товаров;

- Название — название каждого добавления;

- Поставщик — наименование поставщика товаров, который выполняет пополнение склада;

- Контактное лицо — Ф.И.О контактного лица поставщика, осуществляющего поставку;

- Дата — дата поставки товара на склад;

- Товар — наименование поступаемого товара;

- Количество — количество товара, поступающего на склад в результате добавления;

- Склад — наименование склада, на который происходит добавление товара;

- Ответственный — Ф.И.О сотрудника, ответственного за пополнение склада;

- Примечание — различная информация о добавлении товара.

Таблица «Клиенты» содержит поля:

- КлиентID — идентификатор каждого клиента;

- Наименование — полное наименование организации или физического лица, пользующегося услугами фирмы «Эмили» (является уникальным значением);

- Контактное лицо — Ф.И.О контактных лиц клиента;

- Адрес — физический адрес клиента;

- Регион — действующий регион месторасположения клиента;

- Страна — страна, на территории которой зарегистрирован клиент;

- E-mail — официальный электронный адрес клиента;
- Телефон — официальный номер телефона клиента;
- Факс – действующий факс клиента.

Таблица «Поставщики» содержит поля:

- ПоставщикID — идентификатор каждого поставщика;
- Наименование — полное наименование организации или физического лица, предоставляющего услуги по поставке товаров фирме «Эмили» (является уникальным значением);

- Контактное лицо — Ф.И.О контактных лиц поставщика;
- Адрес — физический адрес поставщика;
- Регион — действующий регион месторасположения поставщика;
- Страна — страна, на территории которой зарегистрирован поставщик;
- E-mail — официальный электронный адрес поставщика;
- Телефон — официальный номер телефона поставщика;
- Факс – действующий факс поставщика.

Таблица «Контактные лица» содержит поля:

- КонтактноелицоID — идентификатор каждого контактного лица;
- ФИО — фамилия, имя и отчество каждого контактного лица (является уникальным значением);

- Наименование организации — полное наименование организации, в которой работает контактное лицо;

- Телефон — личный или мобильный телефон контактного лица;

Таблица «Сотрудники» содержит поля:

- СотрудникID — идентификатор каждого сотрудника;
- ФИО — фамилия, имя и отчество сотрудников компании «Эмили» (является уникальным значением);

- Дата приема — точная дата, когда сотрудник компании получил свою должность;

- Телефон — личный или мобильный телефон сотрудника.

Таблица «Товары» содержит поля:

- ТоварID — уникальный идентификатор каждого товара;
- Наименование — полная номенклатура товара (также уникальное значение);
- Поставщик — полное наименование поставщика данного товара;
- Категория товаров — полное название категории, к которой относится товар;
- Цена — стоимость одного экземпляра товара.

Таблица «Категории товаров» содержит поля:

- КатегорияID — идентификатор каждой категории товаров;
- Название — полное наименование категории;
- Описание — текстовое описание категории.

Таблица «Основной склад» содержит поля:

- ID — идентификатор каждой записи данной таблицы;
- Товар — полное наименование товара, хранящегося на данном складе;
- Количество — общее количество экземпляров товара, хранящегося на складе.

Таблица «Резервный склад» имеет идентичную с таблицей «Основной склад» структуру.

Рассмотрим реализацию построения OLAP-куба. Для этого необходимо дать определение понятиям таблицы фактов и таблицы измерений.

Таблица фактов является основной таблицей базы данных. Как правило, она содержит информацию об объектах или событиях, совокупность которых будет в дальнейшем анализироваться. В целом, обычно говорят о четырех наиболее встречающихся типах фактов [17]:

- факты, связанные с транзакциями. Подобные факты основаны на отдельных определенных событиях (типичными примерами которых являются телефонный звонок или снятие денег со счета при помощи банкомата);
- факты, связанные с «моментальными снимками». Основаны на состоянии объекта (например, банковского счета) в конкретные моменты

времени, такие как конец года, месяца или квартала. Стандартным примером таких фактов является объем продаж за день;

- факты, связанные с элементами документа. Основаны на том, или ином документе (например, счете за товар или услугу) и содержат подробную информацию об элементах этого документа;

- факты, связанные с событиями или состояниями объекта. Представляют возникновение события без подробностей о нем (например, просто факт продажи).

Таблица фактов, как правило, должна содержать уникальный составной ключ, объединяющий уникальные первичные ключи таблиц измерений [18]. С учетом того, что таблица фактов может содержать сотни тысяч или миллионы записей, и хранить повторяющиеся текстовые описания, что чаще всего невыгодно, уникальные первичные ключи — это целочисленные значения типа либо значения типа «дата/время». При этом как ключевые, так и определенные неключевые поля должны соответствовать будущим измерениям OLAP-куба. Также таблица фактов содержит одно или несколько числовых полей, на основании которых будут получены агрегатные данные.

Таблицы измерений, в свою очередь, содержат неизменяемые, либо редко изменяемые данные [19]. В большинстве случаев эти данные представляют собой по одной записи для каждого члена нижнего уровня иерархии в измерении. Таблицы измерений также содержат как минимум одно описательное поле (зачастую с именем члена измерения), и целочисленное ключевое поле (обычно это суррогатный ключ) для однозначной идентификации члена измерения. Если будущее измерение, основанное на данной таблице измерений, содержит иерархию, то таблица измерений также может содержать поля, указывающие на родительский объект данного члена в этой иерархии. Также таблица измерений содержит и поля, указывающие на «прародителей», и иных «предков» в данной иерархии, а также дополнительные атрибуты членов измерений, содержащиеся в исходной оперативной базе данных (например, адреса и телефоны клиентов). Каждая

таблица измерений должна находиться в отношении один ко многим с таблицей фактов.

Реализация OLAP-куба подразумевает под собой создание многомерного куба, с помощью которого аналитики на предприятии смогут осуществлять анализ продаж. Для этого первоначально была определена и построена структура базы данных (рисунок 2.2), в которой будет храниться вся информация для построения куба. Основой базы является таблица фактов под названием «Сделки», в которой собрана вся необходимая информация о произведенных сделках предприятия. С данной таблицей фактов связаны несколько таблиц измерений.

В дальнейшем были определены меры OLAP-куба. В качестве меры, показанной на рисунке 13, взят показатель «Количество» проданного материала.

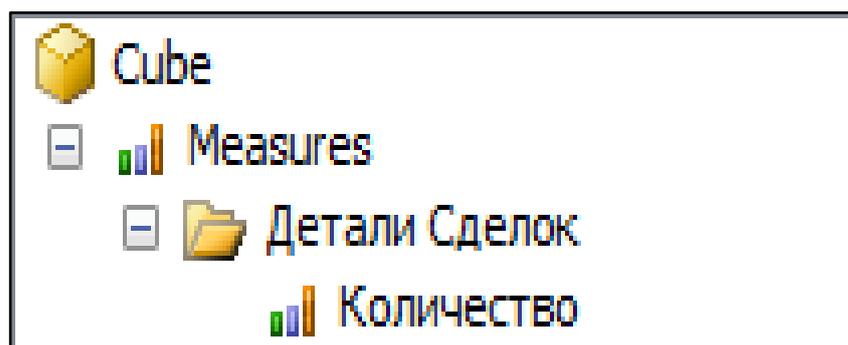


Рисунок 13 — Меры OLAP-куба

Также определены измерения куба. В качестве измерений взяты показатели: дата сделки и товары. Оба измерения имеют двухуровневую иерархию. В измерении «Товары» верхним уровнем является показатель категории товаров, а нижним уровнем — показатель наименования товаров. В измерении «Дата сделки» верхним уровнем является год сделки, а нижнем уровнем — месяц, в котором была произведена сделка.

При дальнейшей разработке системы анализа возможно добавление новых измерений.

На рисунке 14 представлены измерения созданного OLAP-куба.

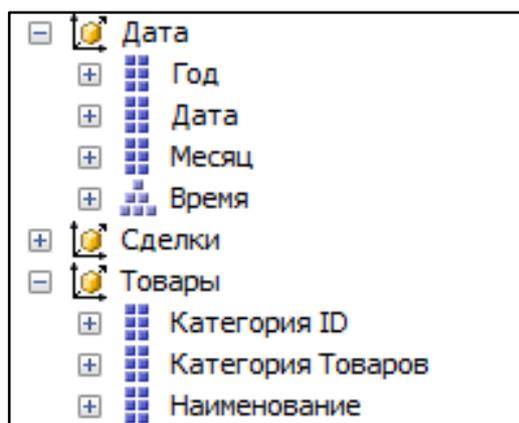


Рисунок 14 — Измерения OLAP-куба

Полный развернутый OLAP-куб представлен в приложении А. Также следует отметить, что куб был импортирован в приложение Microsoft Office Excel для разработки системы прогнозирования.

### 3.3 Проектирование системы прогнозирования

Функциональность разрабатываемой системы прогнозирования движения товарно-материальных ценностей должна предоставлять пользователям возможность строить графики продаж на основе проанализированных данных, а затем при использовании определенных статистических и математических алгоритмов вычислять ожидаемое количество продаж на ближайшие периоды (в рамках дипломной работы — месяцы).

Как описано в первой главе, пользователю в программе Microsoft Office Excel предлагается выбрать одну из шести различных видов линий тренда. Выбор зависит от особенностей каждого отдельного набора данных, на основе которых и строится прогнозирование. Для описания алгоритма построения прогноза необходимо создать тестовое прогнозирование данных продаж.

На рисунке 15 представлены проанализированные данные из созданного ранее OLAP-куба. Соответственно, в качестве тестового прогноза использовались показатели продаж категории товаров «Кондитерские изделия (Confections)» за 1997 год.

Названия строк	⊕ Beverages	⊕ Condiments	⊕ Confections	⊕ Dairy Products
⊕ 1996	1842	962	1357	2086
⊖ 1997	3996	2895	4137	4374
1	330	271	623	448
2	220	393	425	248
3	471	150	265	311
4	268	259	394	231
5	322	300	291	494
6	313	95	107	405
7	327	300	210	374
8	334	206	335	256
9	174	222	371	435
10	491	270	390	422
11	248	172	255	344
12	498	257	471	406
⊕ 1998	3694	1441	2412	2689

Рисунок 15 — Тестовые показатели продаж

В основе любого прогноза лежит выявление закономерностей, повторяющихся в течение определенного периода времени. Поэтому при прогнозировании продаж производятся вычисления, позволяющие выявить периоды времени, характерные для тех или иных показателей продаж товаров. Такие периоды данных называются периодичностью. На рисунке 16 в тестовом прогнозе в качестве периодов служат все двенадцать месяцев 1997 года.

⊖ 1997	3996	2895	4137
1	330	271	623
2	220	393	425
3	471	150	265
4	268	259	394
5	322	300	291
6	313	95	107
7	327	300	210
8	334	206	335
9	174	222	371
10	491	270	390
11	248	172	255
12	498	257	471

Рисунок 16 — Соответствие между периодами и показателями

Следующим шагом является определение корректной линии тренда. Для осуществления выбора линии, необходимо предварительно построить график зависимости объема продаж от месяца, т.е. периода. Затем на построенный график добавляется линия тренда, наиболее точно совпадающая с реальными

данными. Определение достоверности производится в настройках линии тренда при добавлении на график, необходимо лишь отметить пункт «Поместить на диаграмму величину достоверности аппроксимации ( $R^2$ )». Данный пункт показывает, насколько точно линия повторяет реальные данные. Также в настройках линии тренда необходимо отметить пункт «Показывать уравнение на диаграмме». Используемая для тестового набора данных линия тренда показана на рисунке 17.

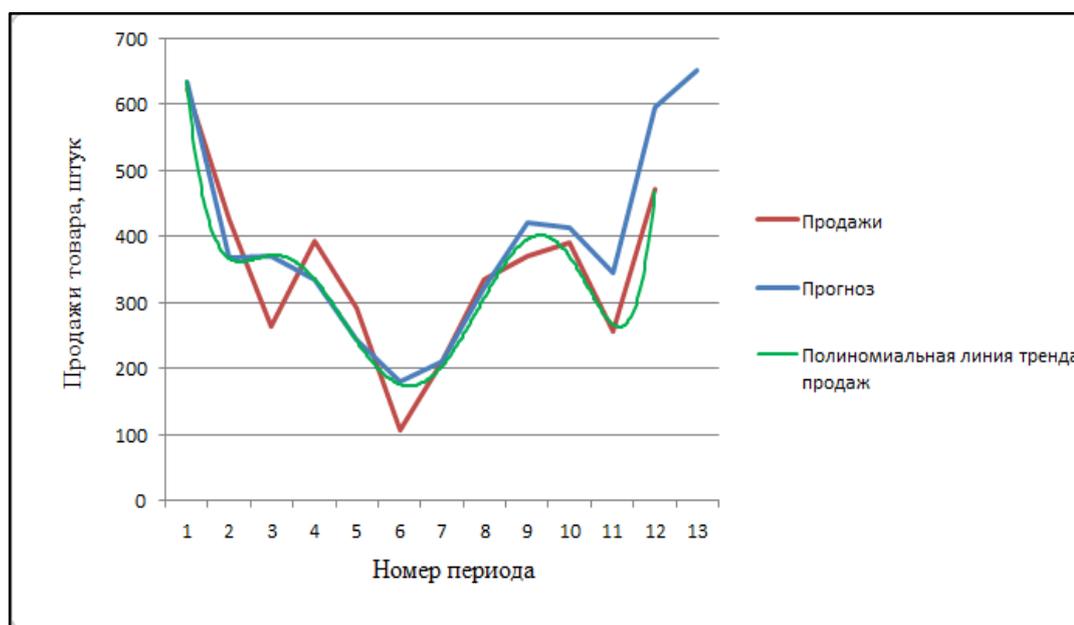


Рисунок 17 — Линия тренда для тестового прогноза

Соответственно, для тестового прогноза наиболее корректной является полиномиальная линия 6-й степени, на 86% совпадающая с реальными данными. Формула данной линии тренда:

$$y = 0,0916x^6 - 3,5201x^5 + 52,299x^4 - 378,4x^3 + 1388,5x^2 - 2465,7x + 2041,2 \quad (1)$$

где  $y$  — прогнозируемое значение показателя;

$x$  — номер периода, за который создается прогноз.

Следующим шагом является определение коэффициентов полинома, необходимых для прогноза, из уравнения. Выполнив вышеуказанные действия, с помощью графика выявляем значения, представленные на рисунке 18.

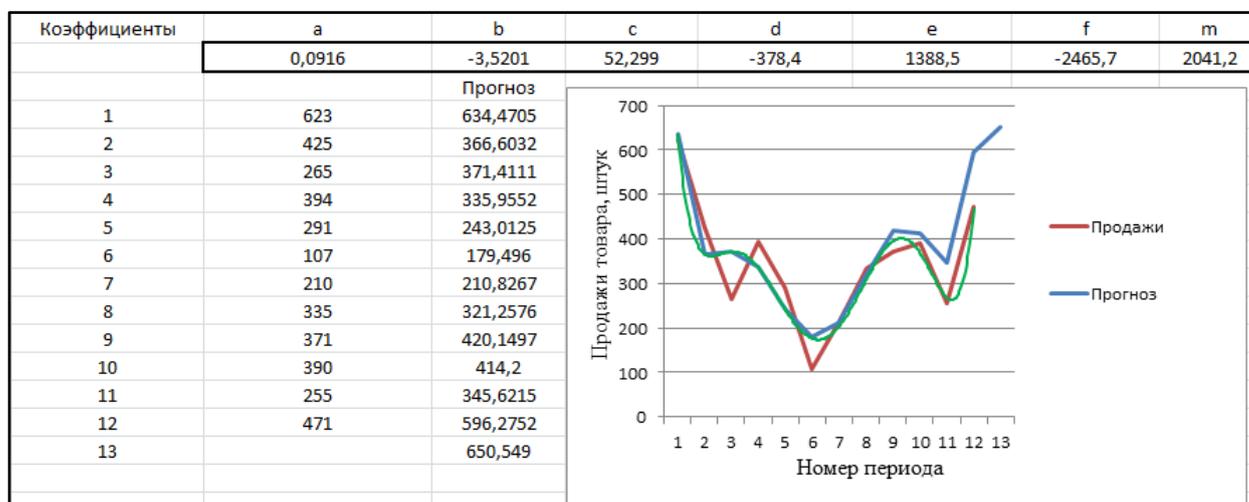


Рисунок 18 — Определение коэффициентов полинома

Финальный этап включает в себя построение теоретико-прогнозируемых данных с помощью определенных выше коэффициентов. Для выявления этих данных для каждого периода времени, в общем уравнении полиномиальной линии тренда 6-й степени необходимо использовать значения периода, начиная с первого и заканчивая тринадцатым периодом. Полученные значения являются теоретико-прогнозируемыми, т.к. они построены на основе формулы полинома, а также было получено значение продаж в 13 периоде, т. е. первом месяце 1998 года.

Поскольку прогнозирование дает лишь теоретическое представление о возможных значениях продаж на будущие периоды времени, необходимо отметить, что определенные тестовые значения продаж имеют значительное расхождение с теоретико-прогнозируемыми результатами. Также данный обуславливается тем, что тестовые данные представляют собой лишь теоретический набор значений.

Вычисленные значения представлены на рисунке 19.



Рисунок 19 — Спрогнозированные данные продаж

Таким образом, общий алгоритм создания прогнозов состоит из следующих этапов:

- определение прогнозируемого набора данных и построение на основе этого графика данных;
- определение корректной линии тренда;
- определение коэффициентов прогноза;
- вычисление прогнозируемых данных.

Таким образом, функциональность системы прогнозирования состоит в том, чтобы автоматизировать выявление коэффициентов для создания прогноза и определение прогнозируемых значений. Данные функции были реализованы с помощью средств программы Microsoft Office Excel (макросы).

### 3.4 Выводы по главе 3

В данной главе были спроектированы и разработаны информационная система учета и база данных. ИС позволяет пользователю загружать, изменять и модифицировать данные. Таким образом реализованы все основные функции системы учета.

Также был разработан и построен OLAP-куб, позволяющий пользователям анализировать данные из ИС. Был представлен алгоритм разработки прогнозов продаж и разработана система прогнозирования.

## ЗАКЛЮЧЕНИЕ

В результате бакалаврской работы выполнены поставленные задачи. Был произведен анализ существующих программных решений, применяемых для анализа и прогнозирования данных. Также была рассмотрена актуальность разработки системы учета и прогнозирования движения товарно-материальных ценностей.

Разработанная система учета товаров позволит пользователям вести полноразмерный учет своей продукции. Созданный на её основе OLAP-куб позволяет проанализировать продажи с точки зрения ключевых показателей.

Таким образом, использование современных OLAP-технологий приводит к положительному экономическому эффекту (отражаемому в увеличении доходности предприятия).

Спроектированная система прогнозирования позволит пользователям на основе проанализированных данных вычислить возможные будущие показатели продаж, что приведет к увеличению эффективности взаимодействия с поставщиками и клиентами компании.

В будущем планируется улучшить разработанную в рамках дипломной работы информационную систему, в частности ввести двухуровневую организацию базы данных.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1) Методы и модели анализа данных: OLAP и Data Mining : учебное пособие / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. – Санкт-Петербург : БХВ-Петербург, 2004. – 336 с.

2) Елманова, Н. З. Введение в OLAP-технологии Microsoft : учебное пособие / Н. З. Елманова, А. И. Федоров. – Москва : Диалог-МИФИ, 2002. – 272 с.

3) Тест FASMI [Электронный ресурс] : Что следует понимать под термином OLAP. – Режим доступа: <http://www.olap.ru/basic/fasmi.asp>

4) Аналитическая платформа Deductor [Электронный ресурс] : Продвинутая аналитика без программирования. – Режим доступа: <https://basegroup.ru/>

5) ВIX ВI [Электронный ресурс]: ВIX — высокотехнологичная российская компания, присутствующая на рынке с 2000 года. – Режим доступа: <http://www.bix.ru/>

6) Облачный сервис Azure Machine Learning [Электронный ресурс] : Машинное обучение от компании Microsoft. // Официальный интернет-портал корпорации Microsoft. – Режим доступа: <https://azure.microsoft.com/ru-ru/services/machine-learning/>

7) Среда Java Development Kit [Электронный ресурс] : Решения Oracle. // Официальный интернет портал компании Oracle Corporation. – Режим доступа: <http://www.oracle.com/technetwork/ru/index.html>

8) Среда Microsoft Visual Studio [Электронный ресурс] : Ресурсы и средства разработки приложений. // Официальный интернет-портал корпорации Microsoft. – Режим доступа: <https://msdn.microsoft.com/ru-ru/library/dd831853.aspx>

9) Технология ASP.NET и среда Visual Studio [Электронный ресурс]: ASP.NET и Visual Studio для веб-разработки. // Официальный интернет-портал

корпорации Microsoft. – Режим доступа: [msdn.microsoft.com/ru-ru/library/dd566231.aspx](http://msdn.microsoft.com/ru-ru/library/dd566231.aspx)

10) Базы данных [Электронный ресурс] : Введение в базы данных. – Режим доступа: <http://www.sql.ru/articles/mssql/2006/.html>

11) Мутина Е. И. Повышение эффективности принятия решений в распределенной системе на основе гибридного подхода в анализе данных : дис. канд. техн. наук : 05.13.01 / Мутина Елена Игоревна. – Москва, 2008. – 136 с.

12) Система управления базами данных Microsoft SQL Server [Электронный ресурс]: Новые возможности SQL Server и платформы данных Microsoft. // Официальный интернет-портал корпорации Microsoft. – Режим доступа: <https://microsoft.com/ru-ru/server-cloud/products/sql-server/>

13) Соловьев, С.В. Технология разработки прикладного программного обеспечения : учебное пособие / С. В.Соловьев, Л. С.Гринкруг, Р. И.Цой. – Москва : Академия естествознания, 2011. – 256 с.

14) Войтовский И. А. Математическое моделирование процессов принятия решений в сетевых системах управления запасами : дис. канд. техн. наук : 05.13.18 / Войтовский Илья Андреевич. – Воронеж, 2012. – 152 с.

15) Многомерные базы данных [Электронный ресурс] : Технология многомерных баз данных. – Режим доступа: [http://www.olap.ru/basic/multi\\_dim\\_DWH.asp](http://www.olap.ru/basic/multi_dim_DWH.asp)

16) Туманов В. Е. Проектирование хранилищ данных для систем бизнес-аналитики : учебное пособие / В. Е. Туманов. – Москва : Интернет-Университет Информационных Технологий, 2010. – 616 с.

17) Бакулева М. А. Модели и алгоритмы автоматизации проектирования структур хранилищ данных для аналитической обработки числовых показателей : дис. канд. техн. наук : 05.13.12 / Бакулева Марина Алексеева. – Рязань, 2007. – 147 с.

18) Бутенко И. В. Разработка моделей и методов построения и автоматизированного наполнения системы метаданных : дис. канд. техн. наук : 05.13.11 / Бутенко Игорь Всеволодович. – Санкт-Петербург, 2011. – 147 с.

19) Зубрилина Т. В. Базы данных. Проектирование реляционных баз и хранилищ данных с использованием CASE-технологий : учебное пособие / Т. В. Зубрилина, В.Н. Юрьев. – Санкт-Петербург : Издательство Политехнического института, 2007. – 44 с.

20) СТО 4.2-07-2014 : Стандарт организации. – Общие требования к построению, изложению и оформлению документов учебной деятельности. – Система управления СФУ. – Красноярск, 2014. – 60с.

## ПРИЛОЖЕНИЕ А

### Плакаты презентации



Рисунок А.1 — Плакат презентации №1

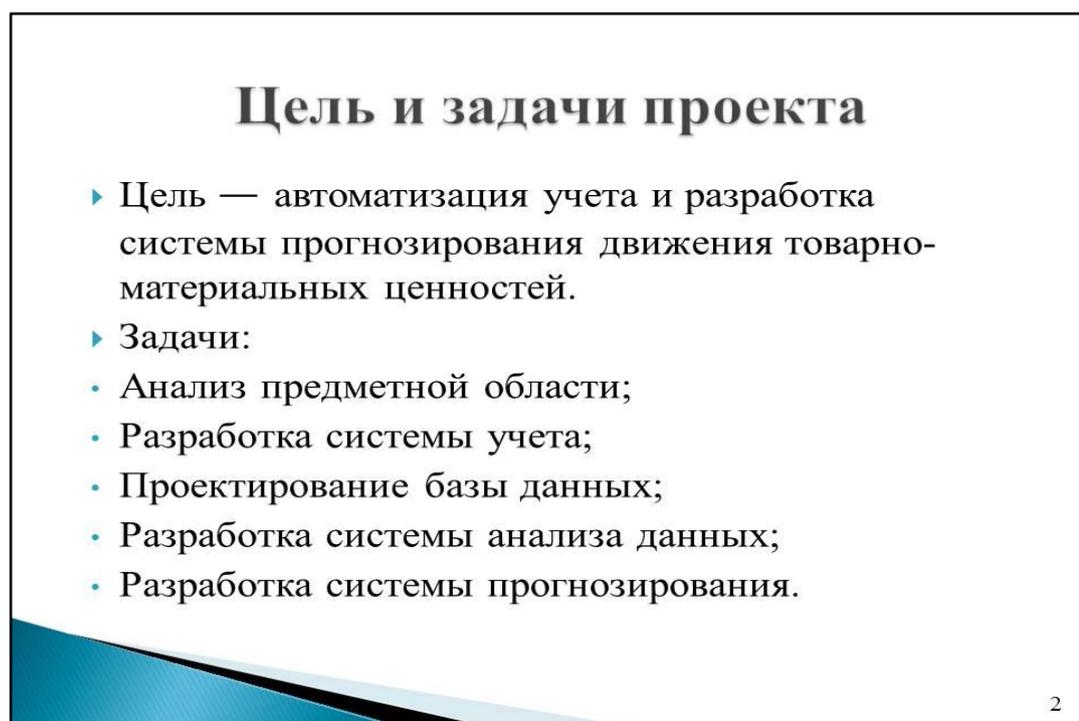


Рисунок А.2 — Плакат презентации №2

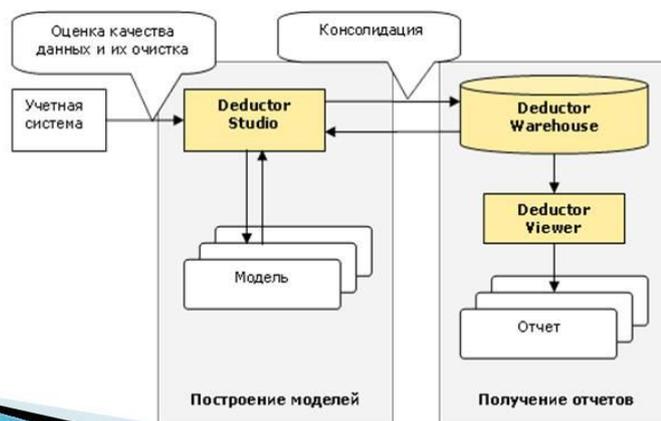
## Актуальность

- ▶ Системы анализа и прогнозирования повышают эффективность управления экономическим объектом
- ▶ Повышение эффективности оказывает влияние на величину дохода объекта
- ▶ Прогнозирование позволяет грамотнее планировать взаимодействие с клиентами

3

Рисунок А.3 — Плакат презентации №3

## Системы анализа данных Платформа Deductor (Base Group)

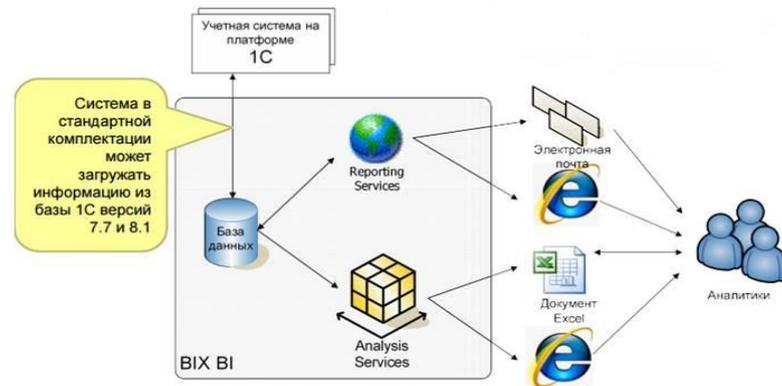


4

Рисунок А.4 — Плакат презентации №4

# Системы анализа данных

## Система ВІХ ВІ

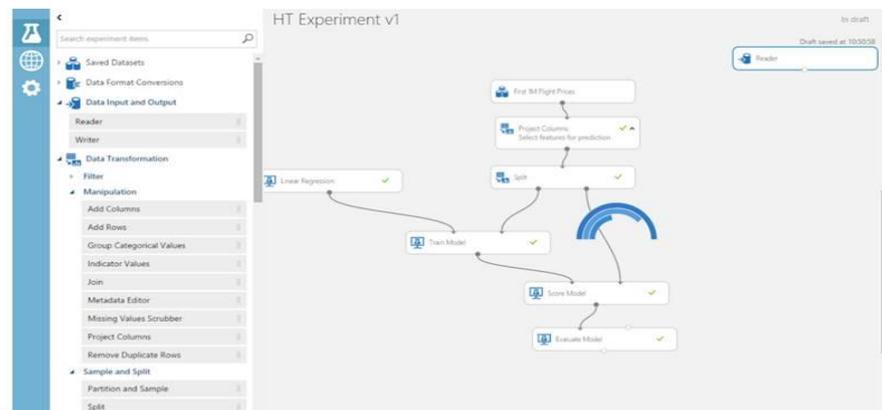


5

Рисунок А.5 — Плакат презентации №5

# Системы прогнозирования данных:

## Сервис Microsoft Azure Machine Learning

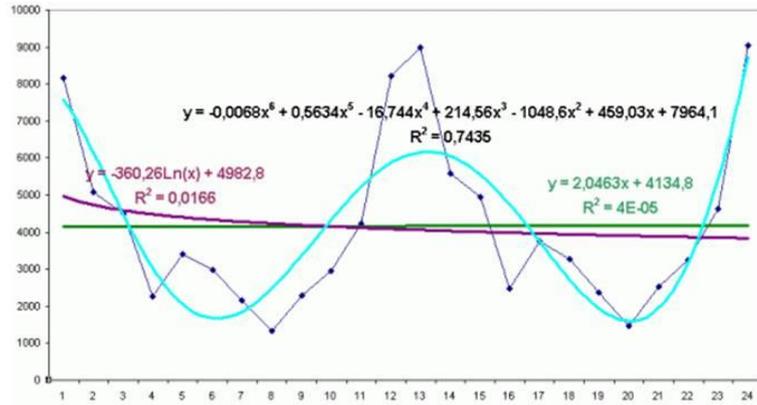


6

Рисунок А.6 — Плакаты презентации №6

# Системы прогнозирования данных:

## Линии тренда в программе Microsoft Office Excel

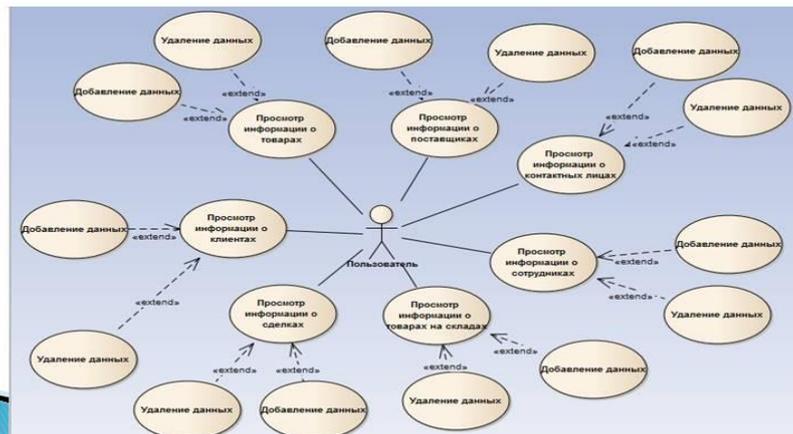


7

Рисунок А.7 — Плакат презентации №7

# Реализация дипломной работы

## ► Разработка информационной системы учета информации

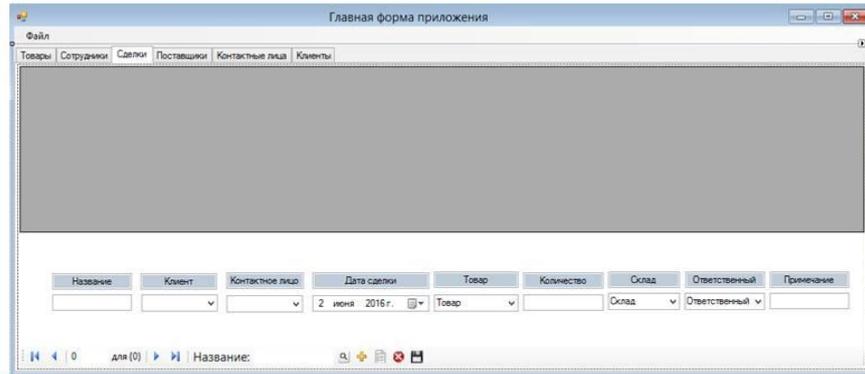


8

Рисунок А.8 — Плакат презентации №8

# Реализация дипломной работы

- Разработка информационной системы учета информации

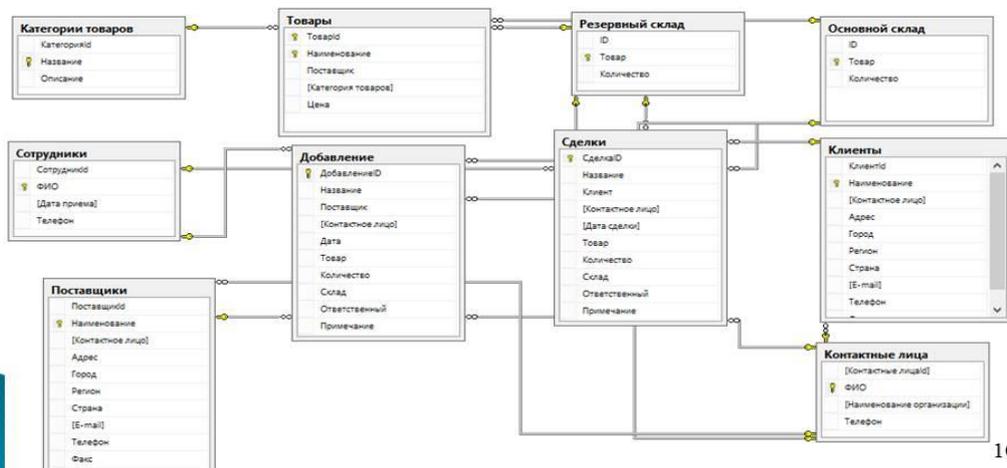


9

Рисунок А.9 — Плакат презентации №9

# Реализация дипломной работы

- Определена структура базы данных

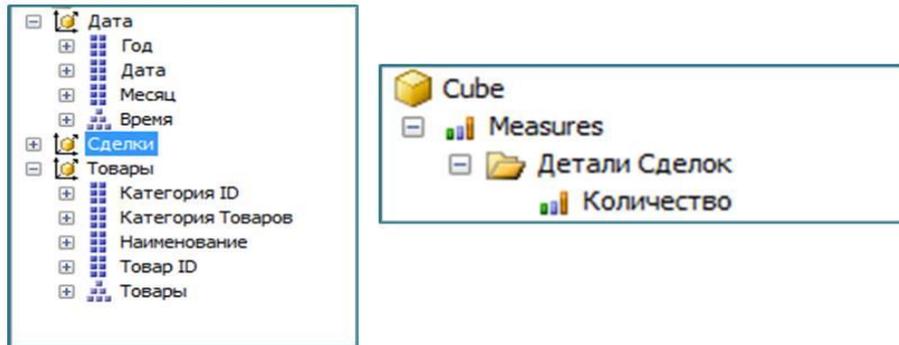


10

Рисунок А.10 — Плакат презентации №10

# Реализация дипломной работы

- Определены измерения и меры OLAP-куба



11

Рисунок А.11 — Плакат презентации №11

# Реализация дипломной работы

- Создан OLAP-куб с возможностью импорта данных в Excel

The screenshot shows a Microsoft Excel spreadsheet with an OLAP cube data table. The table has columns for 'Количество' (Quantity) and 'Наименование товаров' (Goods Name). The data is summarized in the following table:

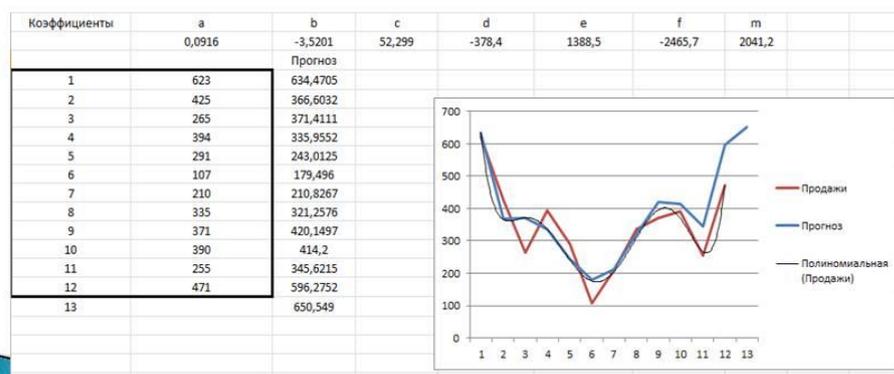
Количество	Наименование товаров	Condiments	Confections	Dairy Products	Grains/Cereals	Meat/Poultry	Produce	Seafood	Общий итог
1396	1842	962	1157	2056	549	900	549	1286	9543
1297	3956	2895	4137	4174	2656	2189	1263	3679	25489
1308	3694	1441	2812	2689	1377	1060	856	2716	16247
Общий итог	9532	5298	7996	9149	4042	4199	2999	7081	51317

12

Рисунок А.12 — Плакат презентации №12

## Реализация дипломной работы

- ▶ Спрогнозированы данные на следующий период



13

Рисунок А.13 — Плакат презентации №13

## Заключение

- ▶ Выполнен анализ существующих программных продуктов;
- ▶ Разработана система учета;
- ▶ Спроектирована база данных;
- ▶ Построен OLAP-куб;
- ▶ Разработана система прогнозирования.

14

Рисунок А.14 — Плакат презентации №14