

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «РОССИЙСКИЙ  
ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ  
ИМЕНИ Г.В. ПЛЕХАНОВА»

Институт цифровой экономики и информационных технологий

Базовая кафедра цифровой экономики института развития информационного общества

«Допустить к защите»

Заведующий Базовой кафедрой

Цифровой экономики ИРИО



Уринцов Аркадий Ильич, д.э.н., проф.

«\_\_» \_\_\_\_\_ 2020 г.

**Выпускная квалификационная работа**

Направление 09.04.03 «Прикладная информатика»

Магистерская программа «Консалтинг в сфере информатизации предприятий и организаций»

ТЕМА Внедрение и доработка BI-решения для процесса, автоматизированного на платформе Pega BPM

Выполнил студент Матинян Сурен Гарникович

Группа 291М-05ПИ/18м

Научный руководитель выпускной квалификационной работы

Афанасьев Михаил Анатольевич, к.э.н., доцент  
Базовой кафедры цифровой экономики ИРИО



(подпись)

Автор



(подпись)

Москва 2020

**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Российский экономический университет имени Г.В. Плеханова»**

**АННОТАЦИЯ**

**выпускной квалификационной работы**

Матинян Сурена Гарниковича

на тему: Внедрение и доработка BI-решения для процесса, автоматизированного на платформе Pega BPM

Выпускная квалификационная работа посвящена исследованию возможностей по разработке BI-решения, позволяющего выгружать данные из Pega BPM приложения в формат, наиболее оптимальный для использования профессиональным BI-приложением.

В работе произведен анализ предметной области посредством изучения имеющихся литературных источников и составлены выводы на данной основе. Изучены особенности кредитных организаций, использующих способы автоматизации, интересные для исследования, и приведены детали реализации некоторых процессов. Автором составлен набор уникальных рекомендаций по решению ключевой проблемы исследования и проведена их практическая апробация. Организационный и экономический аспекты реализации также затронуты.

Работа состоит из введения, четырех основных глав, заключения.

**The ministry of science and high education of the Russian Federation  
Federal state-funded educational institution of higher education  
«Plekhanov Russian University of Economics»**

**ANNOTATION**

**Of final qualifying work**

by Matinyan Suren Garnikovich

on the topic of: Development of BI solution for the process implemented on Pega BPM platform

The final qualification work is devoted to the study of possibilities of development of the BI solution for the process implemented on Pega BPM platform.

The author analyzes the subject area by studying the available literature and draws conclusion on this basis. The key features of the objects of the study are described in detail. The author compiled a set of unique recommendations for solving the key research problem and carried out their practical testing. Organizational and economic aspects of implementation are also affected.

The work contains introduction, four main chapters and conclusion.

Автор ВКР

«» Матинян Сурен Гарникович

## СОДЕРЖАНИЕ

СПИСОК ТЕРМИНОВ И СОКРАЩЕНИЙ .....	4
ВВЕДЕНИЕ .....	7
1. ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ ПРИМЕНЕНИЯ ВРМ-РЕШЕНИЙ СОВМЕСТНО С ВІ-РАЗРАБОТКАМИ .....	11
1.1 Описание и обзор класса систем управления бизнес-процессами .....	11
1.2 Описание концепции Business Intelligence, общий обзор архитектуры реализации и практики применения .....	15
1.3 Анализ концепции процессной аналитики в организации, сконцентрированной на управлении бизнес-процессом.....	22
2. АНАЛИЗ КОНТЕКСТА ИССЛЕДОВАНИЯ .....	27
2.1 Обзор деятельности Альфа-Банка и Сбербанка как объектов исследования .....	27
2.2 Разбор типового процесса, являющегося предметом исследования, и проблем-катализаторов проекта автоматизации .....	33
2.3 Анализ технической реализации системы управления бизнес-процессом	35
3. РАЗРАБОТКА РЕШЕНИЯ ВЫГРУЗКИ ДАННЫХ ДЛЯ ВІ-ПРИЛОЖЕНИЯ .....	41
3.1 Разработка альтернативных решений на основе уникального контекста исследования.....	41
3.1.1 Разработка решения для использования данных на основе сохранения текущей структуры их хранения.....	42
3.1.2 Разработка решения для доступа к данным на основе использования индексных таблиц.....	45

3.1.3 Разработка решения для доступа к данным приложения на основе их точечной выгрузки в отдельную базу данных.....	47
3.2 Конкретизация деталей уникального механизма выгрузки данных на основе разработанного метода .....	53
4. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ РЕШЕНИЯ ВЫГРУЗКИ ДАННЫХ ДЛЯ VI-ПРИЛОЖЕНИЯ.....	58
4.1 Детальная разработка архитектуры решения .....	58
4.1.1 Подготовка области выгрузки.....	59
4.1.2 Настройка стандартного дополнения Rega VIX для организации выгрузки .....	61
4.1.3 Автоматизация процесса запуска выгрузки и создание «точек расширения» функционала.....	63
4.1.4 Разработка дополнительного функционала выгрузки .....	69
4.2 Описание особенностей организации проекта и экспертная оценка его эффективности .....	72
ЗАКЛЮЧЕНИЕ.....	79
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ .....	82

## СПИСОК ТЕРМИНОВ И СОКРАЩЕНИЙ

**Кластер** – объединение команд в рамках трайба, каким-либо образом связанных между собой.

**Лидер трайба** – сотрудник, отвечающий за управление продуктом или группой продуктов и достижение бизнес-целей этого трайба. Отвечает за ключевые показатели эффективности трайба в зависимости от его задач (например, P&L, NPS, доля рынка, надежность систем (uptime, количество инцидентов) и так далее).

**Процессная аналитика** – process mining – управление бизнес-процессом на основе получения аналитических данных о его исполнении.

**Спринт** – небольшой промежуток времени (обычно 1-4 недели), в рамках которого команда разрабатывает инкремент, готовый к поставке продукт.

**Трайб** – группа взаимосвязанных команд, сформированная вокруг определенного продукта или бизнес-цели, и отвечающая за бизнес-результат.

**Чантер** – группа специалистов одной области компетенций, объединенных за рамками трайбов.

**ABPD** – automated business process discovery, автоматизированное исследование бизнес-процесса – понятие, которым часто определяют ту процессную аналитику, которая считается автоматизированной в большей степени.

**Ad-hoc analysis** – BI-процесс, спроектированный для получения ответа на единственный и конкретный бизнес-вопрос.

**BLOB** – Binary Large Object – тип данных в базе данных для хранения объекта целиком (термин в рамках платформы Pega BPM).

**BPMS** – Business Process Management System – система управления бизнес-процессами.

**CSV** – comma-separated values – расширение файлов; текстовый файл, использующий запятую в качестве разделителя данных.

**DCO** – Direct Capture of Objectives – процесс описания, организации и распространения идей о бизнес-процессе с использованием возможностей Pega BPM для «проектирования на лету».

**Enterprise Class Structure** – одна из ключевых концепций в рамках платформы Pega BPM, в основе которой – создание множества классов на разных уровнях приложения.

**ETL** (Extract, Transform, Load) – «извлечение, преобразование, загрузка» – один из основных процессов в организации хранилища данных. Включает в себя извлечение данных из внешних систем, их трансформацию и загрузку в хранилище данных.

**Gradle** – современная система автоматизации сборок, в основе которой – использование Java-подобных языков для описания сборки и связанных «задач» как блоков для выполнения сборки.

**Jenkins** – свободно распространяемый сервер автоматизации для сборки, тестирования и развертывания программного обеспечения.

**Maven** – современная система автоматизации сборок, в которой зависимости указываются в формате XML.

**MDM** – Master Data Management – серия технологий и программных инструментов для управления основными данными.

**MVP** – Minimum Viable Product – минимальная первая версия продукта либо новой крупной функциональности в нём, внедряемая для ранней обратной связи от клиентов.

**OLAP** – Online Analytical Processing – технология, служащая основой для многих BI-приложений. Заключается в подготовке агрегированной информации на основе больших массивов данных, структурированных по многомерному принципу.

**RDB** – relational database – реляционная база данных.

**Sbergile** – фреймворк для разработки и поддержки продуктов в Сбербанке, основанный на ценностях гибких подходов к разработке продукта.

*Situational Layer Cake* – концепция в рамках платформы Pega BPM, позволяющая разделить одно приложение на множество слоев на основе бизнес-категоризации.

*XML* – Extensible Markup Language – расширяемый язык разметки.

## ВВЕДЕНИЕ

Работа является заключительным этапом обучения ее автора на магистерской программе в высшем учебном заведении. Согласно различным методическим указаниям, разработанным университетом на основе Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», она должна являться основой для установления уровня сформированности у него, как выпускника, компетенций, предусмотренных учебной программой. Проведение предполагаемого исследования также способствует систематизации и углублению теоретических и практических знаний, полученных в рамках как обучения, так и работы на местах прохождения практик, предшествующих написанию диссертации. Тема диссертации, сформулированная как «Внедрение и доработка BI-решения для процесса, автоматизированного на платформе Pega BPM», является довольно актуальной, что подтверждают факты, описанные в работе далее. Практические и теоретические результаты исследования представляют ценность для организации, на базе которой оно выполнялось. Накопленные в рамках работы знания, умения и навыки оказывают сильное влияние на профессиональное развитие автора работы.

Уникальность и новизна исследования обосновывается описанием и изучением некоторых критически полезных для организации практик и новшеств в рамках затронутой темы, внедрение которых сильно ограничено в процессах, использующих способы автоматизации, подобные описываемым в работе.

Объектом исследования выступает банковская организация, автоматизирующая значительную часть своих бизнес-процессов на базе BPM-решения от компании Pegasystems. Так как автору исследования повезло поработать на двух подобных объектах, являющихся крупнейшими в стране примерами успешного внедрения и использования систем управления бизнес-



процессами, кажется целесообразным описание специфики интересующих проектов как в Сбербанке, так и в Альфа-Банке.

Предметом исследования являются бизнес-процессы, автоматизированные на базе платформы Rega BPM, и возможности по внедрению процессной аналитики для их оптимизации. В качестве интересующих участков ведения деятельности выступают процесс ипотечного кредитования в Альфе-Банке и процесс администрирования кредитных отношений с крупным корпоративным клиентами в Сбербанке (или совокупность процессов).

Целью исследования является разработка BI-решения, представляющего из себя механизм по выгрузке данных процесса, автоматизированного на платформе Rega BPM, в формат, наиболее оптимальный для использования профессиональным BI-приложением.

В рамках достижения цели исследования предполагается выполнение следующих задач:

- Изучение теоретических аспектов исследования: возможностей процессной аналитики и особенностей ее использования в том числе для процессов, автоматизированных с помощью класса систем управления бизнес-процессами, посредством изучения литературных источников и существующих методик и подходов;
- Сбор и анализ эмпирических данных, обзор интересующих аспектов деятельности объекта исследования для составления контекста работы;
- Сравнительный анализ ключевых способов достижения цели исследования и доступных решений в рамках контекста конкретного объекта исследования;
- Разработка уникальных практических рекомендаций и решений на базе знаний, полученных при выполнении предыдущих задач;
- Проведение апробации практических рекомендаций и решений по внедрению процессной аналитики на изучаемые участки процессов с выработкой уникальных деталей реализации;

- Оценка организационной и экономической аспектов реализации практического решения.

Структура работы типичная для подобного рода учебных исследований и состоит из списка терминов и сокращений, введения, четырех основных глав, заключения, списка использованных источников.

В рамках выполнения исследования автором сформулированы следующие положения, выносимые на защиту:

- Структура хранения данных в приложении, автоматизированном на платформе Rega BPM, не представляется оптимальной для целей построения отчетов, и стандартный инструментарий платформы, включающий в себя продукты, приобретаемые по дополнительной лицензии, не является достаточным для подготовки данных BPM-приложения для профессионального BI-решения;

- Разрабатываемый уникальный механизм организации выгрузки данных из BPM-приложения для BI-решения, включающий в себя очистку дубликатов в области выгрузки и некоторые другие преимущества, можно считать MVP-решением, достаточным для подготовки к организации хранилища данных и последующего запуска BI-приложения на основе обработки выгружаемых им данных.

Связь работы с другими похожими исследованиями можно отследить по ссылкам на используемые источники. Основн Автору не удалось найти работы, в значительной степени похожие на текущую. Данный факт является подтверждением некой уникальности проводимого исследования и тем самым его значимости.

С конкретными результатами исследования следует ознакомиться в его заключительных главах. При оценке исследования стоит учитывать специфику работы компаний, на базе которых оно выполнялось. Так как сохранение коммерческой тайны в финансовых организациях особо важно, некоторые

аспекты исследования намеренно упущены или изменены не в пользу совпадения с реальными фактами.

# **1. ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ ПРИМЕНЕНИЯ ВРМ-РЕШЕНИЙ СОВМЕСТНО С BI-РАЗРАБОТКАМИ**

Первая глава работы посвящена теоретическому исследованию ее аспектов на основе изучения литературных источников и практик в современных коммерческих организациях. Предполагается описание популярных ВРМ-решений, особенностей их использования в целом, изучение преимуществ систем BI и их классификация, формирование представления о возможностях и существующих практиках процессной аналитики. Так как в центре внимания стоят данные, отдельная часть главы посвящена разбору некоторых архитектур управлениями данными на предприятиях.

## **1.1 Описание и обзор класса систем управления бизнес-процессами**

Системы управления бизнес-процессами (ВРМ-системы) представляют из себя технологическое программное обеспечение для поддержки концепции ВРМ. Суть самой концепции – управление предприятием на основе управления процессами. ]За основу берется бизнес-процесс, и его совершенствование, оптимизация (управление им) считается основной идеей деятельности. Важной особенностью популярных ВРМ-систем является возможность связать воедино такие понятия, как миссия предприятия, стратегия ее развития, цели, долгосрочные планы, среднесрочные перспективы и конкретные бюджеты на ближайший период. Выстраивается целостная картина большого процесса.

Ориентация на концепцию управления процессами позволяет достичь следующих преимуществ:

- Повышение «гибкости» организации, значительное сокращение временных и прочих издержек, необходимых для поддержания и изменения бизнес-процесса;
- «End-to-end» реализация процесса в рамках единой платформы: четкое видение общей картины всеми участниками, сокращение издержек

взаимодействия, повышение ориентированности сотрудников на достижение бизнес-целей организации и др.

Преимущества выделены исходя из анализа международного опыта внедрения концепции в учреждения банковской, здравоохранительной, телекоммуникационной и транспортной отраслей.

Компании-разработчики BPM-систем руководствуются следующими принципами, которые в общем виде отражают их преимущества перед подходом автоматизации, основанном на разработке приложений с использованием стандартных подходов и языков программирования:

- Low code – возможность автоматизации любых процессов с минимальным программированием;
- Dynamic case management – software-based подход, ориентированный на управление, развитие и в конечном счете автоматизацию работ, помогающий людям лучше понять, как выполнять эти работы. Работы организуются в кейсы (cases) – части работ, приносящие бизнес-результат;
- Процессная архитектура – сквозной взгляд на процесс и сквозное управление всеми действиями в ходе процесса и потоками работ, образующими процесс. К существующей архитектуре предприятия и бизнес-архитектуре управление процессами добавляет ответы на вопросы «как»: как работа должна выполняться и как она меняется.

На рынке представлено много решений для комплексной автоматизации бизнес-процессов на основе концепции управления бизнес-процессами. Самым популярным является решение от компании Pegasystems, о котором и пойдет речь далее (которое и используется на объекте исследования). Известными аналогами в России являются системы IBM BPM и ELMA BPM. В таблице 1 представлен сравнительный анализ платформ [22].

Таблица 1

## Сравнение ведущих BPM-платформ

Критерий сравнения	Peга BPM	Elma BPM	IBM BPM
Симуляция проведения процесса при различных условиях	+	–	–
Графической моделирование	+	+	+
Поддержка каскадных процессов	–	+	+
Поддержка концепции «изменения на лету»	–	+	+
Поддержка матриц эффективности сотрудников	–	+	–
Продукт используется в компании	+	–	–
Поддержка средств отчетности	+	+	+
Развитая поддержка концепции DCO (Directly Capture Objectives)	+	–	–

При сравнении BPM-решений часто обращаются к мнению компании Gartner и ее «магического квадранта». Суть популярного исследования заключается в сравнении решений на основе двух ключевых критериев: Completeness Of Vision (глубина видения) и Ability to Execute (прикладная

ценность). Так, все популярные решения разбиваются на 4 категории в зависимости от уровня соответствия упомянутым критериям (рис. 1).



Рисунок 1. «Магический квадрант» Gartner

Как можно заметить, два из трех упомянутых популярных в российских организациях решения являются лидерами в рейтинге Gartner. На данном этапе исследования нам важно понять суть концепции управления бизнес-процессами и описать популярные решения из опыта российского коммерческого

внедрения. В дальнейшем мы более подробно рассмотрим некоторые особенности автоматизации на объекте исследования.

## **1.2 Описание концепции Business Intelligence, общий обзор архитектуры реализации и практики применения**

Важным этапом исследования является обзор систем бизнес-анализа, так как в дальнейшем ключевые преимущества именно подобных решений будут положены в основу формирования практических рекомендаций по внедрению процессной аналитики в организацию с учетом особенностей его развития. Мы в том числе разберем архитектуру BI-решения и определим ту область этой архитектуры, в рамках которой выполняется работа в рамках общей картины конечного решения.

Business Intelligence (BI) – это обозначение компьютерных методов и инструментов, предназначенных для перевода транзакционной деловой информации в человекочитаемую форму для возможности осуществления бизнес-анализа [55]. Говоря простым языком, подобные системы ставят перед собой ключевую цель выделения из больших массивов данных полезной информации в формате, удобном для бизнес-пользователя с целью проведения бизнес-аналитики и совершенствования эффективности работы предприятия в конкретных точках его деятельности. Сама формулировка концепции впервые, как и многое, связанное с компьютерными технологиями, во второй половине XX века, возникла в статье исследователя из IBM Ханса Питера Луна [36]. В своей работе в 1958 году он определил BI как «возможность определения внутренних взаимосвязей имеющихся фактов для принятия правильных решений по отношению к достижению поставленных целей». Так как в работе ранее мы уже обращались к мнению специалистов из Gartner, то для понимания полезно также ознакомиться с описанием термина Business Intelligence, приведенным Говардом Дреснером, аналитиком той самой исследовательской



компании [34]. Он определил BI как некий общий термин для описания концепции и методов для улучшения принятия бизнес-решений с использованием систем на основе бизнес-данных.

Конечно, те определения, которые мы даем ключевым концепциям в данном разделе работы, не являются исчерпывающими, но, по мнению автора, они являются наиболее простыми для формирования того понимания, которое требуется на данном этапе исследования. Более подробно с термином можно ознакомиться в научных трудах, ссылки на которые приведены в работе. Сейчас важно определить, что BI – это не просто набор технологий или методологий, а мощная концепция управления предприятием, которая в случае правильной реализации способна наделять организацию множеством преимуществ.

Применение концепции Business Intelligence оказывает значительное прямое влияние на определение стратегии организации, принятие тактических и операционных бизнес-решений. Данный факт не в последнюю очередь оказал влияние на выбор данного направления в качестве одной из ключевых концепций исследования. Приведем несколько преимуществ и причин необходимости применения BI-решений на различных участках производственного процесса:

- Возможность более точного измерения KPI (Key Performance Indicators, ключевых индикаторов эффективности), основываясь на исторических данных;
- Возможность идентификации и установления стандартов для различных подпроцессов в рамках единого производственного процесса;
- Используя системы BI, организации могут своевременно идентифицировать рыночные тренды и изменение их динамики, благодаря чему лица, принимающие решения, будут способны своевременно выделять ресурсы на решение ключевых вопросов, стоящих перед предприятием;

- BI-решения также помогают в упрощении восприятия информации (в том числе благодаря возможностям по визуализации данных), что сокращает количество ресурсов, необходимых для обработки данных и повышает качество принятия решений на их основе и т.п. [35]

Если говорить о недостатках и ошибках внедрения BI-решений, то в различных трудах выделяются следующие:

- Высокая стоимость лицензий и прочих затрат, связанных с внедрением профессионального решения в организацию [34].
- Связанная с этим недоступность некоторых передовых решений для малых и средних организаций.
- Сложность при конструировании хранилищ данных и связанные ограничения по организации хранения данных.
- Внедрение BI решений происходит обычно после накопления определенного количества данных. Работа с данными из различных источников, каждый из которых реализовывал уникальный подход к их организации, требует больших усилий и высокой степени взаимодействия подразделений на начальном этапе внедрения. По похожим причинам реализация концепции в крупной организации может занимать продолжительное время.

Понятие BI неотделимо связано с некоторыми другими ключевыми определениями, связанными с управлением данными (такими как хранилище данных, управление качеством данных, управление мастер-данными и версиями данных, контекстная аналитика и т.п.). Некоторые из смежных понятий мы затронем далее.

Говоря о компонентах BI-решений, выделим следующие ключевые части, которые можно выстроить в процесс создания аналитического отчета для конечного пользователя (рис. 3):

– Системы-источники данных. Обычно крупные организации хранят данные из различных систем в различных базах данных, часто имеющих реплики.

– Механизмы реализации процесса ETL (Extract, Transform, Load). Это процесс извлечения данных из системы-источника или какой-либо промежуточной системы и трансформации этих данных для дальнейшей загрузки в хранилище данных. Трансформация данных может происходить по следующим причинам: для обеспечения их консистентности (например, устранения различий в способе обозначении пола покупателя в разных системах), перевода данных в формат, легкий для понимания бизнес-пользователем (денормализация данных и приведение к так называемым схемам «звезда» или «снежинка») или обогащения данных (рис. 2) [17].

– Хранилище данных. Сама система, в которой собраны данные из различных источников для использования в аналитических системах.

– Инструменты работы с хранилищем данных (средства анализа и визуализации данных). Некоторые примеры подобных систем будут приведены далее.

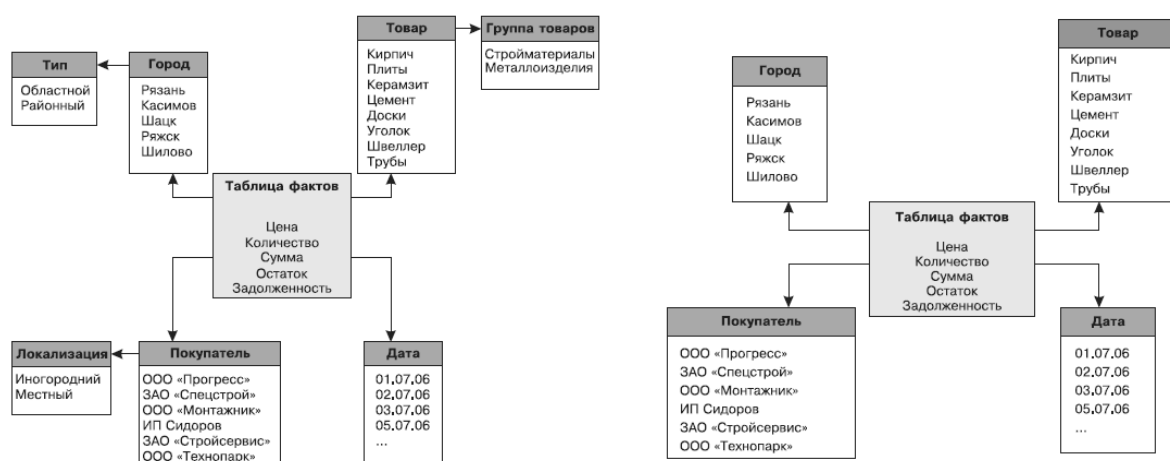


Рисунок 2. Схемы снежинка (слева) и звезда (справа)

Сразу отметим, что исследование в конечном итоге ставит перед собой цель разработки именно механизма извлечения данных из системы, работающей на платформе Pega BPM в промежуточную базу для их дальнейшей загрузки в хранилище данных.

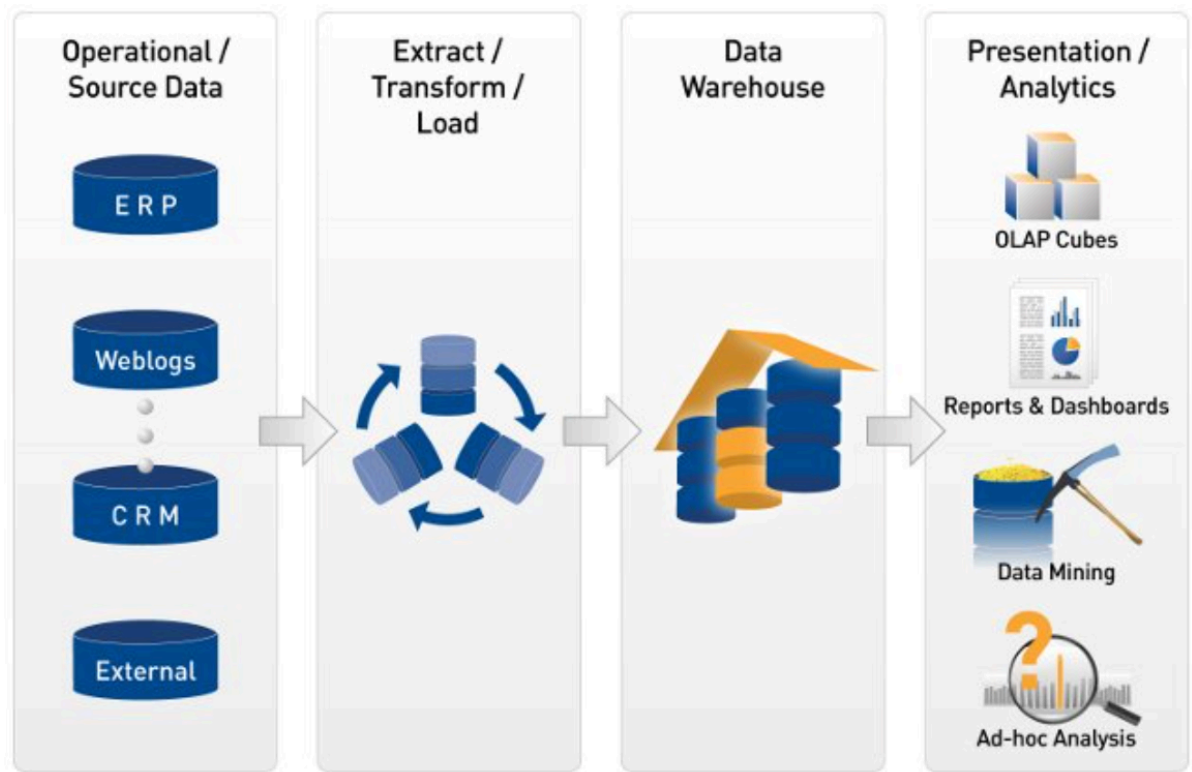


Рисунок 3. Компоненты BI-решения.

Перед внедрением любого решения важно иметь представление о наиболее полезных его функциях для конечного потребителя для выбора правильной реализации и корректной расстановки приоритетов. Согласно исследованию компании SelectHub, далее перечислены несколько ключевых функций BI-решений в порядке их важности для бизнес-потребителя [35]:

- Возможность ранжирования элементов бизнес-процесса по влиянию на бизнес-показатели на основе различных критериев для определения сильных и слабых компонентов предприятия. Возможности для проведения автоматической аналитики.

- Возможность проведения анализа «What-If», когда можно определить влияние потенциального решения будущее бизнеса посредством моделирования на данных из прошлых периодов.

- Возможность построения меняющихся в реальном времени отчетов по операционной деятельности и по стратегическим показателям для менеджмента организации.

- Возможности по визуализации отчетов и интерактивного взаимодействия с ними. Возможность простой генерации новых типов отчетов «на лету».

Наиболее популярными BI-решениями на рынке являются следующие (имеется в виду часть комплексного решения: конечные приложения, которые работают уже с подготовленным хранилищем данных):

- Tableau – одно из самых популярных решений на рынке от компании Salesforce. Включает в себя несколько компонентов. Одним из основных преимуществ пользователи называют простоту использования, когда не технические специалисты способны самостоятельно создавать для себя уникальные отчеты (рис. 4). Именно данное решение будет одним из главных конечных пользователей хранилища, для которого мы опишем механизм по извлечению данных из специфического бизнес-процесса.

- Qlik. Не менее популярное решение, предоставляющее невероятно мощные инструменты для анализа данных. Из других преимуществ: наличие мобильных инструментов и положительные отзывы по поводу производительности и качества визуализации.

- Microsoft Power BI. Одним из ключевых преимуществ решения является наличие бесплатной (хоть и ограниченной) версии. Также программное обеспечение предлагает интерактивную визуализацию, предиктивную аналитики, разные форматы экспорта подготовленной аналитики.

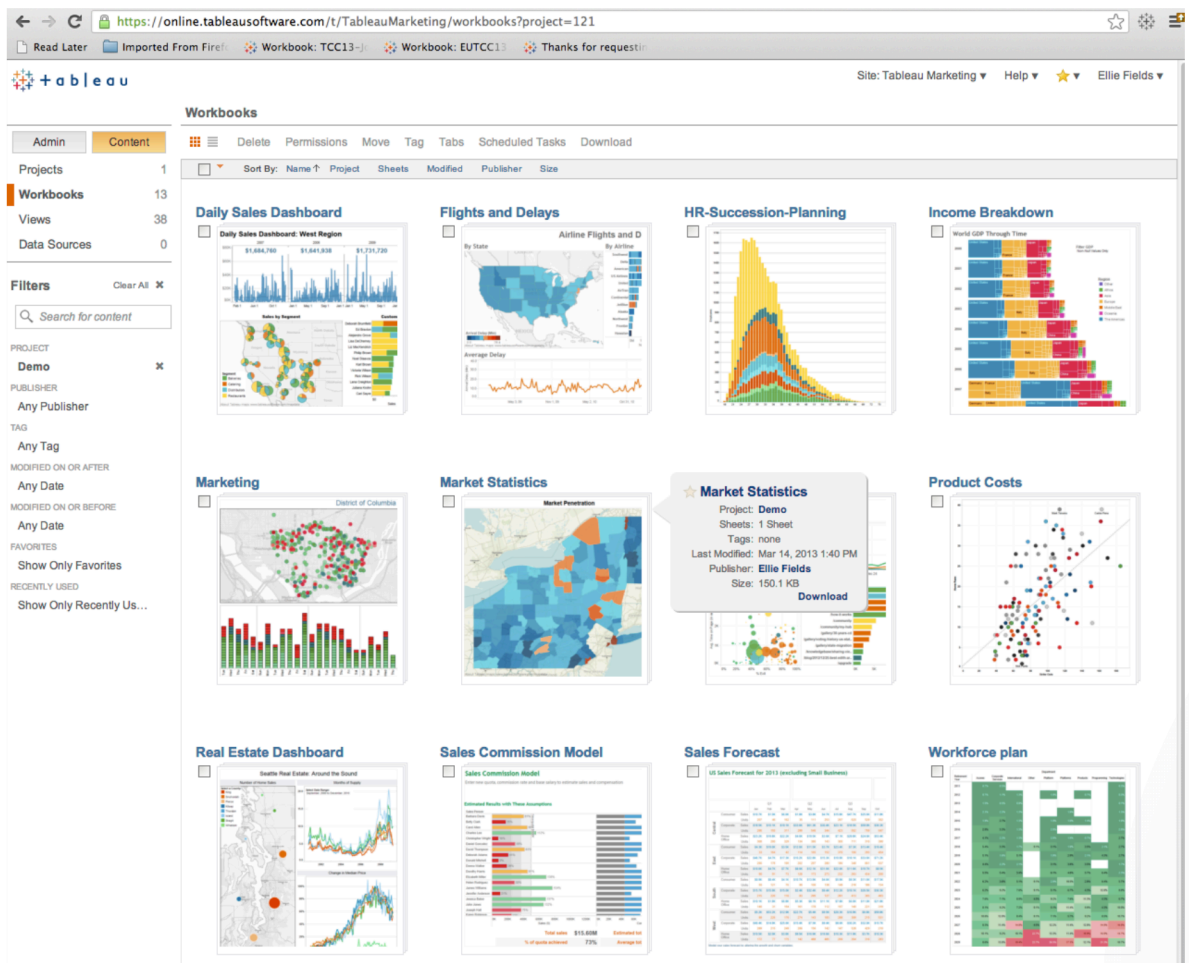


Рисунок 4. Портал пользователя Tableau.

Выделим некоторые тренды в области систем Business Intelligence [19]:

- Согласно исследованию компании Gartner, наблюдается тенденция на использование решений с искусственным интеллектом для замещения некоторых функций ЛПР и автоматического принятия решений в реальном времени.
- Внедрение инструментов совместной работы в коробочный функционал BI-решений.
- Развитие концепции модульности, когда одно решение может встраиваться в другое для расширения своих возможностей на основе потребностей конкретного потребителя.

– «Облачная аналитика», когда бизнесу предлагаются уже развернутые на серверах производителя и готовые к работе технологии. Согласно последним исследованиям, динамика роста затрат организаций на внедрение облачных решений увеличится в 4.5 раза за следующие 2 года.

Таким образом, имея более глубокое представление о концепции Business Intelligence и типовой архитектуре реализации BI-решения (и того, с какой частью данной архитектуры мы работаем в рамках данного исследования), можем перейти к более детальному рассмотрению специфики предполагаемого проекта.

### **1.3 Анализ концепции процессной аналитики в организации, сконцентрированной на управлении бизнес-процессом**

Изучение концепции процессной аналитики является последним блоком исследования, сконцентрированным на изучении теоретического материала. Данная концепция важна для дальнейшей работы и является по сути симбиозом содержания двух предыдущих подглав.

Процессной аналитикой (process mining) называют методы и подходы, предназначенные для анализа и усовершенствования процессов в информационных системах или бизнес-процессов на основании изучения системных данных о выполненных операциях в системе. В некоторых ситуациях process mining называют data mining в рамках концепции управления бизнес-процессами (business process management, рис. 5) [51]. Банальное построение отчетов на основе внутренних данных BPM-приложения на предприятии необходимо как для реализации стандартных бизнес-функций наподобие отслеживания статистики продаж, так и для воплощения в жизнь процессной аналитики в описываемом направлении [57].

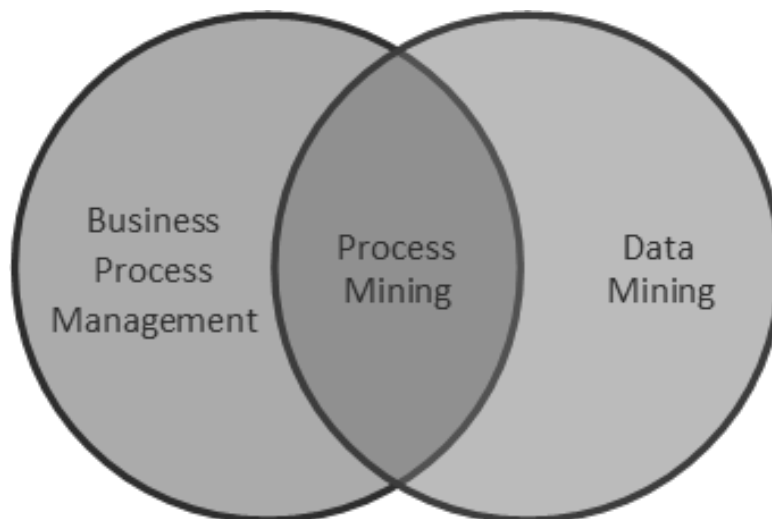


Рисунок 5. Process mining – data mining в рамках концепции business process management.

Основная причина реализации процессной аналитики – выявление причин различий между задуманным, «идеальным» процессом и имеющимся в действительности. Существует довольно большое количество примеров успешного использования концепции на реальных предприятиях совершенно различных областей деятельности. Помимо того, грамотное сочетание преимуществ процессной аналитики с другими инструментами совершенствования бизнес-процесса может дать компании еще более заметное преимущество. К примеру, именно процессная аналитика позволяет выявить наиболее удобные участки процесса для внедрения «роботов» (robotic process automation, RPA) [50].

Выделим некоторые преимущества проведения процессной аналитики на проекте:

- Сокращение издержек. Процессная аналитика позволяет выявить части процесса, на которых происходит необоснованное потребление ресурсов.
- Повышение прозрачности. Сложные бизнес-процессы обычно создают в процессе своего выполнения множество данных. Подход



позволяет в том числе выявить наиболее ценные для бизнеса данные с целью дальнейшего их использования.

- Сокращение задержек. Устранение потерь, связанных с падением производительности на отдельных участках процесса (выявление данных участках и причин подобного поведения).

На данном этапе стоит также отметить различие между понятиями процессной аналитики (Process Mining) и автоматического изучения процесса (automated business process discovery, ABPD). Хотя во многих источниках данные понятия не различаются и считаются синонимами для обозначения единой концепции, в некоторых работах можно столкнуться со мнением, заключающемся в том, что ABPD – это следующий этап развития process mining. Авторы таких работ представляют стандартную процессную аналитику в виде в большей части ручного процесса изучения бизнес-процесса на основе его системных логов. ABPD же рассматривается как дальнейшее развитие концепции, опирающееся на современные инструменты автоматизации деятельности по анализу данных процесса. Если придерживаться подобной точки зрения, то мы говорим скорей об автоматическом изучении процесса, чем о процессной аналитике, хотя далее в работе разница между данными понятиями не проводится.

Выделим следующие альтернативы процессной аналитике или вообще управлению бизнес-процессами на предприятии и их преимущества и недостатки:

- Управление поручениями. Акцент на контроле за тем, когда и кем будут выполняться рабочие задачи. Являлся популярным и эффективным подходом в двадцатом веке, но изжил себя с развитием новых качеств и ценностей сотрудником.

- Управление проектами. Акцент на уникальности деятельности, а не его шаблонности. Не подходит для исследования, так как описанная в

работе бизнес-операция является типичным примером повторяющегося бизнес-процесса.

– Управление на основе отчетности и аналитики. Как и два предыдущих подхода частично является частью управления бизнес-процессами, но уже давно не представляет интереса для самостоятельного рассмотрения.

Процессная аналитика представляет из себя перспективную концепцию, сформированную на стеке двух других, рассмотренных выше: Business Intelligence, извлечения полезного материала из данных в общем, и Business Process Management, управления бизнес-процессом как ключевой сущностью деятельности предприятия. При правильной реализации подобного рода аналитики перед предприятием открываются перспективы получения такого рода выводов об узких местах бизнес-процесса (при этом практически любой степени его декомпозиции), которые невозможно сформулировать при ручном его изучении. Ведь, как доказала практика, нет большого смысла во внедрении информационных технологий на предприятие, когда связанные процессы очевидно не скомпилированы до той степени оптимальности, которой можно достичь без использования ИТ. Далее в работе мы также коротко описываем организации, являющиеся объектами исследования, и делаем выводы о том, что менеджмент в разные периоды старался оптимизировать процессы в компаниях с использованием различных подходов (от производственной системы Toyota до принципов Agile) и добился определенных успехов. Важно понимать, что применение процессной аналитики с использованием современных информационных систем можно считать обоснованным на определенном этапе развития процесса для поиска тех выводов о его состоянии, которые уже невозможно определить без дорогостоящего внедрения BI-решения.

Таким образом, в данной части исследования мы изучили три ключевые для текущей работы концепции: Business Process Management (управление бизнес-процессом), Business Intelligence (бизнес-анализ) и Automated Business Process

Discovery (автоматизированная процессная аналитика). Эти подходы, которые мы свяжем между собой в рамках дальнейшего более подробного изучения концепции процессного управления в современной организации (объекте исследования) являются самыми важными для внедрения инициативы по реализации процессной аналитики в конкретной компании и процессе, основанной на описанных принципах, но уникальной в деталях реализации, и являющейся основным результатом данной работы.

## **2. АНАЛИЗ КОНТЕКСТА ИССЛЕДОВАНИЯ**

Вторая глава исследования посвящена изучению его контекста – реальной организации – и бизнес-процессов, выступающих в качестве предмета исследования. Стоит учитывать, что вся предоставленная информация взята с открытых источников и может быть неактуальной на данный момент или неполной. Тем не менее раскрытие какой-либо информации, не обнародованной уполномоченными представителями организаций, ограничено трудовым соглашением с автором исследования.

### **2.1 Обзор деятельности Альфа-Банка и Сбербанка как объектов исследования**

В качестве объекта исследования выступает акционерное общество «Альфа-Банк». Также в данной главе будет предоставлено короткое описание деятельности публичного акционерного общества «Сбербанк», так как описываемый проект автоматизации может быть применен в данной организации, а опыт его исследования по результатам работы окажется полезным.

АО «Альфа-Банк» – крупнейший по размеру совокупного капитала и кредитному портфелю частный банк страны, основанный в 1990 году и ведущий свою деятельность с 1991 года. Компания входит в список системно значимых кредитных организаций страны. Кредитный рейтинг компании – «BB+» (Fitch Ratings), это один из самых высоких показателей среди российских банков.

В банке работает около 25 тысяч сотрудников. Клиентская база – более 16 миллионов физических лиц и более 500 тысяч корпоративных клиентов. Высокое качество предоставляемых банком услуг неоднократно подтверждалось многими престижными наградами. Так, в 2016 году международная платежная система VISA отметила организацию в рамках глобальной программы Visa Global Service Quality. Головной офис организации

расположен в Москве, но деятельность компании ведет и за пределами России. Дочерние компании банка ведут свою деятельность в Великобритании, Нидерландах, на Кипре и в других странах.

Банк динамично развивается. За последние несколько месяцев он обновил подход к брендингу, стратегию и формулировку миссии. На текущий момент миссия организации звучит следующим образом: «Мы – партнёр для активных людей и компаний. Мы создаём уверенность в успехе и каждый день делаем их жизнь лучше.» Интересным может показаться сравнение текущей формулировки миссии со старой: «Мы верим, что свобода – ключевая ценность современного человека. Объединяя неравнодушных людей, их опыт и энергию, мы помогаем вам быть свободнее в поступках и мечтах.»

Банк признает свою репутацию в качестве наиболее ценного актива, которым владеет. Для ее поддержания с самого момента своего основания он ведет активную социальную деятельность:

- Собственная стипендиальная программа для поддержки талантливой молодежи «Альфа-Шанс»;
- Ежегодный международный фестиваль электронной музыки Alfa Future People, один из крупнейших в мире;
- Финансовая поддержка благотворительных фондов, членство в корпоративном клубе WWF России;
- Организация является официальным европейским банком Чемпионата мира по футболу FIFA 2018 и Кубка Конфедераций FIFA 2017.

Среди целей ведения деятельности компании можно выделить следующие, закрепленные в уставе:

- Содействие росту инвестиционной и коммерческой активности в экономике Российской Федерации;
- Содействие становлению и развитию частного предпринимательства;

- Получение оптимального размера прибыли от использования собственных и привлеченных средств [25].

Для достижения указанных целей банк осуществляет следующие основные виды деятельности [4]:

- Привлекает денежные средства юридических и физических лиц в рублях и иностранной валюте, мобилизует кредитные ресурсы на внутреннем и международном финансовых рынках;
- Организует и осуществляет расчеты своих клиентов, связанные с их производственной, торговой и иной деятельностью;
- Осуществляет кредитование операций, связанных с производственной, торговой и другими видами деятельности клиентов банка;
- Предоставляет все виды банковских услуг в соответствии с лицензией, выданной Банком России;
- Оказывает консультационные услуги в области банковской и финансовой деятельности;
- Осуществляет иные виды деятельности в соответствии с законодательством РФ.

Основной владелец банка – акционерное общество «АБ Холдинг» (99%), 1 % акций принадлежит материнской компании AVH Financial Ltd., зарегистрированной на Кипре. Упомянутые организации практически полностью контролируются одним из крупнейших в России частным финансово-инвестиционным консорциумом «Альфа-Групп» [30].

Вот некоторые интересные для работы выдержки из обновленной стратегии развития на 2019-2021 годы, ключевые инициативы, на которые будут направлены основные ресурсы в следующие 2 года:

- Фокус на создании лучшего мобильного банка в России как центра взаимодействия с клиентами (тем не менее банк не собирается

отказываться от других каналов взаимодействия с клиентами и не позиционирует себя как ИТ-компанию) [29];

- Реализация концепции первого в России безбумажного универсального банка (успехи в данном направлении уже есть, с некоторыми достижениями можно ознакомиться на ресурсе, ссылка на который размещена в конце работы);

- Трансформация сети отделений в так называемую «сеть нового поколения» (реализация обслуживания без паспорта и бумажных документов и т.п.);

Одна из ключевых целей стратегии банка – удвоение количества клиентов в 2022 по сравнению с 2017.

ИТ-стратегия банка на ближайшие несколько лет включает в себя инициативы по сокращению количества используемых информационных систем и унификации применяемых технологий.

Публичное акционерное общество «Сбербанк» – крупнейший банк в России, Центральной и Восточной Европе, один из ведущих международных финансовых институтов. Миссия организации звучит следующим образом: «Мы даем людям уверенность и надежность, делаем их жизнь лучше, помогая реализовывать устремления и мечты». Сегодня Сбербанк является самым дорогим брендом в России и самым дорогим банковским брендом в мире. На долю Сбербанка приходится треть совокупных банковских активов России, а его активными клиентами являются 60% населения. Количество стран, в которых присутствует организация, – 22, количество сотрудников – свыше 300 000 [31].

Ключевой целью организации, согласно последнему годовому отчету, является превращение банка в одну из лучших финансовых и технологических компаний в мире. Главная цель актуальной стратегии – выйти на новый уровень конкурентоспособности, который даст возможность конкурировать с глобальными технологическими компаниями, оставаясь лучшим банком для населения и бизнеса. Как утверждает руководство организации, компания не

считает себя просто банком, а позиционируется как крупная технологическая компания. Именно крупнейшие в мире транснациональные технологические компании воспринимаются организацией в качестве самых серьезных стратегических конкурентов на фоне цифровизации банковских услуг и взрывного развития финтех-продуктов, в большей части предлагаемых не традиционными финансовыми организациями. Суть – стать ИТ-компанией с банковской лицензией. Гибкость и преимущества, что дают современные информационные технологии и развитая корпоративная культура, признаются ключевым конкурентным преимуществом в стратегической перспективе, без которых деятельность организации в долгосрочной перспективе считается провальной.

Ключевыми стратегическими приоритетами на 2018-2020 года являются:

- Достижение лучшего клиентского опыта и построение экосистемы – не только в финансовой, но и в других сферах жизни клиента.
- Технологическое лидерство: надежность и эффективность; новая платформа; безопасность; компания, управляемая с помощью данных и алгоритмов; инновации.
- Люди нового качества в эффективных командах. Новые компетенции, команды вместо иерархии, современная культура и HR [32].

Ключевым акционером банка является Правительство РФ (до недавних пор – Центральный Банк РФ).

Важно отметить, что кроме классической организационной структуры в банке принята и другая, вдохновленная философией Agile (так называемый Sbergile, рис. 6) [38]. Компания поделена на трайбы (дословно переводится как «племя»), каждый трайб сосредоточен на достижении какой-то бизнес-цели в рамках стратегии и обладает своей структурой, включающей лидеров и кураторов трайба. Трайб состоит из множества команд, которые работают по философии Agile обычно на основе адаптированного фреймворка SCRUM. Стандартный цикл работы команды – двухнедельный спринт [21].





Рисунок 6. Базовые процедуры, используемые при реализации Agile в Сбербанке

Трайбы в свою же очередь могут быть разбиты на более мелкие объединения команд, кластеры. Обычно трайб насчитывает в себе не более 1500 сотрудников, кластер – не более 150. Тем не менее, данные числа не являются строгой рекомендацией. Также в новой организационной структуре существует понятие чаптера, объединения специалистов в рамках какой-либо профессиональной области (в рамках разных трайбов).

В организации налажена тесная работа технических специалистов и «представителей бизнеса». Пространство одного из ключевых офисов – «дома Agile» – устроено по примерам крупнейших технологических компаний, когда предусмотрено много физического места для совместных встреч и отдыха или работы за пределами стандартного рабочего стола. У лидеров трайба так же, как

и у других сотрудников, не предусмотрено наличие отдельного кабинета, за основу взята концепция «открытого пространства» («open space»). Считается, что случайные встречи и спонтанные обсуждения рабочих вопросов могут привести к творческим идеям, поэтому это поощряется, в том числе на уровне архитектуры офиса. Кроме того, в компании уважаются не только ценности философии Agile, но другие доказавшие свою эффективность мировые практики менеджмента. Например, в деталях работы современного Сбербанка можно найти много отсылок к легендарной производственной системе Toyota (Lean, или бережливое производство).

Информация, предоставленная о компаниях, минимальная, но, как мне кажется, достаточная для формирования понимания соответствия предлагаемых далее инициатив интересам стратегий организаций.

## **2.2 Разбор типового процесса, являющегося предметом исследования, и проблем-катализаторов проекта автоматизации**

Конечной целью исследования является воплощение в жизнь проекта, который позволит повысить эффективность изучаемых процессов на объекте исследования. Предполагается создание инструментов, удобных для использования владельцами процесса, и пригодных для управления процессом с целью его постоянного совершенствования. В данной части коротко рассмотрим сам процесс и некоторые «узкие места» для понимания контекста исследования.

Подробное описание бизнес-процессов не требуется в работе, так как оно построено на принципе внедрения решения в определенных условиях автоматизации вне значительной зависимости от бизнес-контекста. Тем не менее исследование в большей части выполнялось на базе 2 процессов, которые очень коротко описаны далее.

Первый процесс – ипотечное кредитование в Альфа-Банке. Процесс выдачи кредита на приобретение жилья (или рефинансирование другого похожего кредита) стандартный и состоит из следующих этапов [6]:

- Менеджер продаж оформляет заявку, собирает пакет документов и передает заявку и документы в кредитно-аналитический отдел;
- Кредитно-аналитический отдел проверяет заявку и документы, формирует решение и передает заявку и документы в отдел заключения сделок;
- Отдел заключения сделок собирает документы по залогу, формирует кредитно-обеспечительную документацию, резервирует договора в банковской системе, передает все в отдел выдачи кредита;
- Сотрудник отдела выдачи кредита производит выдачу кредита.

В Сбербанке же рассматривались типичные процессы по администрированию кредитных отношений с корпоративными клиентами: постановке кредитных клиентов на внутреннее администрирование, их снятие с администрирования, работа с данными по расходованию выданных кредитных средств, анализ сомнительных транзакций и т.п. Детализация процессов с точки зрения бизнес-операций не представляет интереса для работы, важна их техническая реализация (способ автоматизации), которая описана в следующей части.

Выделим некоторые конкретные проблемы на рассматриваемом бизнес-процессе, которые послужили катализатором запуска проекта по внедрению процессной аналитики (проблемы на примере Альфа-Банка, но их можно отнести и к деятельности Сбербанка). Они позволят на примере разобраться в преимуществах реализации концепции:

- Неэффективность периодически реализуемых технологических доработок процесса. Владельцы процесса не могут понять, какая задача по автоматизации способна принести наибольший бизнес-эффект.

- Отсутствие подробной отчетности по бизнес-показателям каждого отдельного участка процесса.
- Возрастающая сложность процесса и невозможность ручной обработки некоторых данных.
- Стремительный рост направления кредитования и нагрузки на менеджеров. Возрастающая необходимость идентификации «проблемных» участков процесса.
- Невозможность или излишняя сложность реализации некоторых требований регулирующих органов без описываемых в работе доработок процессов.
- Необходимость пересмотра подхода к управлению процессом вследствие рыночного давления.

Таким образом, мы максимально коротко описали бизнес-процессы, являющиеся основным контекстом исследования, и выделили некоторые проблемы, послужившие катализатором возникновения идеи подготовки к внедрению процессной аналитики в организации. Можно переходить к рассмотрению технических особенностей реализации упомянутых процессов, так как именно они являются ключевыми для исследования.

### **2.3 Анализ технической реализации системы управления бизнес-процессом**

Данная подглава посвящена описанию особенностей работы приложений на базе платформы Rega BPMS для понимания сути предлагаемых в следующей подглаве альтернатив реализации способов выгрузки данных из BPM-приложения для BI-решения. Акцент сделан на рассмотрении способов хранения данных процесса, так как именно этот аспект нас интересует в большей степени.

В проектах Pega BPMS используется концепция Situational Layer Cake, позволяющая разделить одно приложение на множество слоев на основе бизнес-категоризации (различных регионов, категорий клиентов и т.д.) Для реализации данной концепции применяется подход Enterprise Class Structure, в основе которого – создание множества классов на разных уровнях приложения, которые в свою очередь служат локацией для расположения так называемых rules («рулов», правил) – базовых единиц при разработке приложения на данной платформе. Для хранения же данных приложения предусмотрены две стандартные схемы в реляционной базе данных в текущих актуальных версиях платформы:

- PegaRULES – для хранения системной информации;
- PegaDATA – для хранения более близких пользователю данных о, например, созданных заявках, справочнике доступных продуктовых предложений и т.п.

Особенностью платформы является то, что практически каждая таблица в БД привязана к упомянутым выше классам в приложении через специальные «рулы» связи. Переменные в классе и в классах-предках представляют из себя доступный набор полей для соответствующей таблицы. Когда мы создаем объект какого-либо конкретного класса, он сохраняется в базе данных в виде записи. Некоторые объекты могут содержать в себе сотни полей совершенно разного формата, поэтому хранение всех этих данных в виде отдельных колонок в БД представляется неэффективным. Pega хранит всю информацию об объекте в формате XML в колонке под стандартным названием rzPvStream в таблице класса объекта (BLOB datatype – Binary Large Object, рис. 7). Преимуществами данного подхода являются:

- Данные сжимаются перед сохранением и занимают меньше дискового пространства;

- На информацию не накладываются никакие ограничения в виде максимального размера, количества символов, типа данных и тому подобного;
- Структура хранимого объекта может быть любой сложности (например, содержать множество вложенных объектов);
- Правила для структуры объекта могут меняться без вмешательства администратора базы данных, так как это не потребует изменения структуры БД;
- Вся информация об объекте считывается сразу, что быстрее в стандартных сценариях работы.

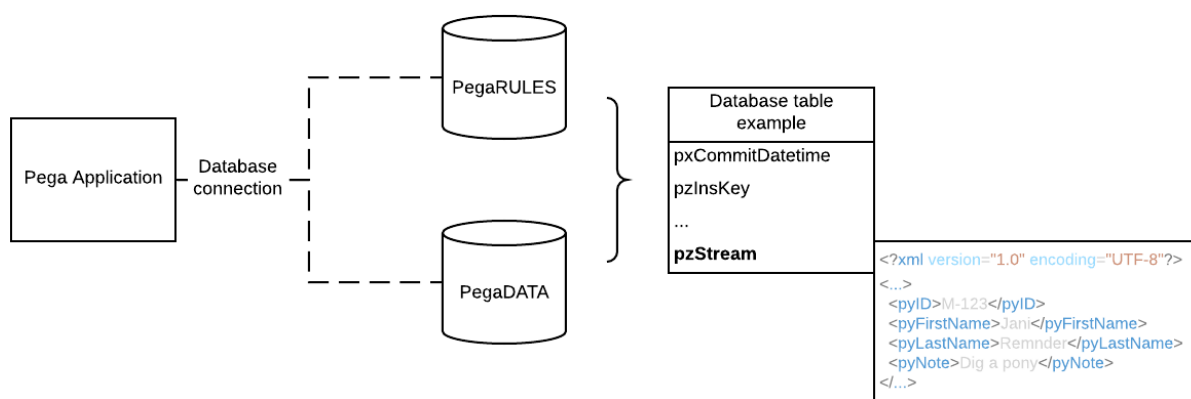


Рисунок 7. Пример хранения данных в приложении

Само собой, существуют и недостатки у данного подхода. К примеру, в случае необходимости построения каких-либо отчетов, которые обычно содержат определенное небольшое количество полей, нам требуется получать эти поля из XML-файла, что задействует довольно много ресурсов по сравнению с обычным считыванием значения из колонки в БД (мы считываем и разбираем весь объект). Данный недостаток является причиной реализации описываемого решения для выгрузки данных, так как продвинутое BI-приложение потребляют

данные из специально организованного хранилища данных, а не из специфичной XML-структуры, оптимальной для работы BPM-решения.

Можно сделать вывод о том, что способ хранения данных в BPM-решении ориентирован на те стандартные способы их использования, что предлагаются платформой и достаточны для реализации серьезных приложений и автоматизации, возможно, любых бизнес-процессов. Производительность подобного подхода находится на высоком уровне в стандартных сценариях использования приложения и тем самым оправдана. Тем не менее на определенных этапах развития проектов автоматизации бизнес-процессов команды сталкиваются с новыми сильно отличающимися направлениями оптимизации вроде необходимости реализации BI-решений для выявления тех особенностей процесса, что невозможно отличить «ручной аналитикой». В подобных ситуациях специфичная структура хранения данных и автоматизация на платформе-конструкторе оказывается скорее не преимуществами, а серьезными недостатками, и порождают громадные проблемы, связанные с тем, что приходится разбирать весь накопленный объем данных по сути по отдельным сущностям и преобразовывать его в формат традиционных хранилищ данных. Подобный процесс «разбора» очень болезненный и чуть ли не точечный в рамках отдельных сущностей по следующим причинам:

- Проекты автоматизации на системах-конструкторах, освоить которые в базовой степени возможно очень быстро, обычно воспринимаются менеджерами организации как возможность по привлечению к работе максимального количества специалистов младшего уровня. И хотя в краткосрочной перспективе данный подход экономит деньги на оплате труда и время на поиск и подготовку специалистов, в долгосрочной он приводит к большим проблемам.

- Ключевым преимуществом BPM-решений считается скорость автоматизации бизнес-задач, в разы превышающая разработку на стандартных языках программирования. И хотя поставщики платформы

предоставляют огромное количество обучающих материалов и делают акцент на необходимости разработки с использованием «лучших практик», недалёковидные менеджеры проектов соглашаются на увеличение скорости разработки в ущерб ее качеству, не задумываясь о долгосрочных последствиях. Например, мой опыт работы на проекте по автоматизации ипотечного процесса привел к одному из следующих выводов: накопленный объем работ по устранению «технического долга» на проекте превышает затраты на реализацию всех запланированных будущих бизнес-задач. И чем дольше он игнорируется, тем выше скорость его прироста. Напомним, что Pegasystems неоднократно подчеркивает в своих материалах: нормальной ситуацией считается отсутствие «технического долга» в принципе.

– Свободная структура хранения данных, располагающихся в реляционной базе данных, но не привязанных к нормам нормализации, приводит к тому, что трудно выделить какие-то общие правила организации структуры данных, на которые можно опираться при разработке. О требованиях BI-решений к структуре данных задумываются лишь в тот момент, когда наступает необходимость их реализации, и когда накоплены уже значительные массивы неструктурированных данных.

В итоге мы владеем ситуацией, когда огромная часть уже накопленных данных хранится в приложении в структурах, сильно отличающихся друг от друга и не имеющих малейшего намека на наличие какого-либо архитектурного видения при их проектировании. Игнорировавшееся в прошлом низкое качество разработки ради получения мимолетной выгоды привело к ситуации, когда для решения с первого вида небольших проблем с данными приходится затрачивать огромное количество ресурсов.

Таким образом, в данной главе мы выстроили понимание контекста исследования посредством изучения объекта и предмета исследования. Нами были выделены не только общие особенности бизнес-процесса и некоторые его



проблемы, но и детали технической реализации. Как оказалось, подготовке данных процесса для реализации процессной аналитики препятствуют некоторые специфичные особенности хранения данных в платформе Pega BPM, использованной для автоматизации.

### **3. РАЗРАБОТКА РЕШЕНИЯ ВЫГРУЗКИ ДАННЫХ ДЛЯ ВІ-ПРИЛОЖЕНИЯ**

#### **3.1 Разработка альтернативных решений на основе уникального контекста исследования**

Как было отмечено в предыдущей главе, исследование сконцентрировано на реализации уникального решения, связанного с выгрузкой данных из системы, которая представляет из себя автоматизацию процесса в рамках концепции управления бизнес-процессом на платформе Pega BPM. Напомним, что, согласно схеме на рисунке 3, работа посвящена части реализации ВІ-решения, связанной с процессом ETL (Extract, Transform, Load). В большей части – Extract (выгрузка данных в промежуточную БД для организации хранилища данных).

Ключевая проблема исследования связана с обоснованием выбора наиболее подходящего решения для достижения максимально возможного положительного эффекта от реализации процессной аналитики на описанном ранее процессе (в области подготовки данных). Для решения данной проблемы произведен глубокий анализ разработанных альтернативных вариантов, произведено сравнение и описаны преимущества и недостатки каждого выбора. Стоит отметить, что в следующей главе работы также более подробно описаны уникальные детали реализации решения.

На основе текущих обстоятельств был проведен анализ и выделены следующие уникальные для ситуации подходы к организации выгрузки данных для отчетности в рамках реализации процессной аналитики для процесса, автоматизированного на платформе Pega BPM:

- Сохранение текущей структуры хранения данных и использование механизмов для построения отчетов или организации хранилища данных на основе данных из XML-файла (вариант 1);
- Дублирование данных из XML-схемы в отдельных колонках базы данных, использование индексных таблиц (вариант 2);

- Использование отдельного специального продукта (дополнения) для платформы Rega BPM для создания отдельной промежуточной базы данных для целей ВІ с необходимостью его доработки (вариант 3).

Важно понимать, что под источником данных для ВІ-решения мы имеем в виду источник не для самого ВІ-приложения, а для хранилища данных. В следующих подглавах опишем каждое из разработанных направлений более подробно и придем к формированию окончательного уникального варианта реализации.

### **3.1.1 Разработка решения для использования данных на основе сохранения текущей структуры их хранения**

Разберем первый предложенный командой автоматизации вариант, заключающийся в сохранении текущей структуры данных и использовании механизмов построения отчетов на основе данных из XML (скорей всего, с предварительной загрузкой в хранилище данных).

Нами были выделены следующие преимуществу использования данной методики:

- Простота в реализации и минимальное количество работы по преобразованию данных. Автором имеется в виду работа по преобразованию данных со стороны их поставщика. То есть, по сути, предполагается, что BPM-приложение даст доступ к копии своей базы и простой инструмент для их извлечения, а необходимую работу по применению данного инструмента для настройки источника данных возьмет на себя разработчик конечного ВІ-решения. Данный подход, с одной стороны, может показаться безответственным, но он имеет свои преимущества (например, в виде сокращения затрат ресурсов на определенного рода коммуникации между подразделениями посредством своеобразного разграничения их работы).

- Возможность постоянного доступа к «свежим копиям». Так как в данном варианте реализации данные для BI-решения предполагается брать напрямую из реплики промышленной базы данных, то актуальность данных может быть обеспечена периодичностью создания резервных копий. Учитывая важность данных по кредитному процессу, предполагается в целевом варианте создание резервных копий с периодичностью в 1 час.

Для воплощения данной идеи в жизнь мы выделили следующие основные работы, которые требуется провести (если тут же возвращаться к простоте его реализации):

- Организовать периодическое создание отдельной резервной копии основной БД для целей BI;
- Доработать стандартную функцию по извлечению данных из XML-структуры с целью максимальной оптимизации ее производительности под наши нужды.

Данный метод рассматривался в качестве довольно серьезной альтернативы на начальных этапах проекта вследствие перечисленных преимуществ и сжатых сроков, но он был отвергнут на определенной стадии реализации вследствие некоторых критичных недостатков, среди которых:

- Не соответствующая ожиданиям поставщиков и стандартам современных систем скорость построения отчетов. После первой же демонстрации прототипа стало очевидным невозможность комфортной работы с интерактивными отчетами при использовании данного подхода к хранению и доступа к данным. Предположение о низкой производительности решения было заложено в его описание изначально, но прототип доказал наличие слишком существенного падения производительности.
- В рамках проработки решения нами были выявлены некоторые критичные недостатки стандартного инструмента по извлечению данных

объекта приложения из стандартной XML-образной структуры. В результате объем доработок оказался значительно больше ожидаемого, а уровень производительности мог бы упасть еще ниже после их осуществления. В частности:

- Придется разрабатывать дополнительный механизм извлечения данных из XML структуры по объектам, являющимися массивами данных. Стандартный инструмент работает с ними некорректно.
- Придется дополнительно конвертировать значения некоторых типов полей (например, дат), так как в XML-структуре они хранятся в своеобразной форме.

Предполагается, что одним из главных недостатков данного решения, как и следующего, является также невозможность легкого преобразования данных. Как мы отметили в предыдущей главе, накопленные данные в приложении хранятся неструктурировано, элементы одной и той же сущности могут быть разбросаны по разным объектам. Обычно при проектировании BI-приложения исходные данные сначала приводятся к определенной степени нормальности и затем только денормализуются при загрузке в само хранилище данных. Полноценного доступа к нормализованным данным, по сути, не будет. Придется производить две операции при загрузке: сбор нужных данных из разных объектов, своеобразная нормализация, и денормализация уже при загрузке в хранилище данных.

Нами был сделан максимально простой протип подобного решения, и его эксплуатация на тестовых стендах вызвала крайне негативную реакцию со стороны членов кроссфункциональной команды, ответственных за тестирование. Позже была проведена и внутренняя демонстрация (одно из мероприятия в рамках SCRUM-фреймворка). В данной ситуации гибкий подход к разработке продемонстрировал свои несомненные преимущества. Вызывающее негативные эмоции решение было отвергнуто сразу после первой

итерации реализации, и команда изменила свои планы в сторону работы над более сложными в воплощении, но более правильными с архитектурной точки зрения выработанными методами, описанными далее. Стоит упомянуть, что опыт, полученный от работы над данной методикой, пригодился. Некоторые преимущества подхода все же получилось перенести в другое конечное решение, он также несет ценность в повседневной работе с платформой.

### **3.1.2 Разработка решения для доступа к данным на основе использования индексных таблиц**

В первую очередь поясним, что мы имеем в виду под индексными таблицами в данном случае. Это не имеет ничего общего с индексами в базе данных, как может показаться, а является термином в рамках ВРМ-платформы от Pegasystems. Как я упоминал ранее, объект в приложении обычно хранится в виде записи в базе данных и имеет специальную колонку, содержащую в себе все его данные в виде XML-структуры (как раз той, с которой мы пытались работать в предыдущем методе). Но в этой же таблице могут быть и другие колонки, представляющие различные свойства объекта и дублирующие содержимое этой XML-структуры. Состав этих «отдельно вынесенных в базе данных» колонок мы можем регулировать самостоятельно. Почему их требуется создавать отдельно: как раз по причине оптимизации для отчетности. Кроме того, если объект имеет в себя вложенную структуру в виде массива, то мы можем создать отдельную табличку для описания свойств этого массива в виде отдельных колонок в БД с соответствующими связями с текущей таблицей. Именно такого рода таблицы и называются индексными. Обновление данных в колонках предусмотрено в рамках коробочного решения и происходит автоматически при сохранении объекта.

Суть второго предложенного нами варианта по использованию данных ВРМ-приложения для целей ВІ заключается в следующем: в качестве первого

источника данных использовать также реплику промышленной базы данных, но опираться не на XML-структуру, а на отдельные колонки в базе данных. Для этого потребуется предварительно проанализировать набор требуемых для отчетности объектов и свойств и предусмотреть их сохранение в виде отдельных колонок в БД приложения.

По аналогии с предыдущим предложением опишем выделенные нами преимущества разработанного подхода, приведшие к его выдвижению:

- Сравнительно небольшая сложность реализации. Подход с расширением колонок базы данных для целей отчетности является распространенным среди разработчиков на ВРМ-платформе. Для внутренних простых отчетов он уже используется долгое время и доказал свою эффективность и простоту.

- Отсутствие необходимости в покупке дополнительных лицензий. Далее мы рассмотрим еще одну похожую разработанную альтернативу, но данная отличается от нее в лучшую сторону отсутствием трат на приобретение лицензий на программное обеспечение.

Автором сделаны предположения по требуемому выполнению следующих действий для применения подхода:

- Нужно проанализировать используемые для построения отчетов объекты и их свойства, описать их типы и требования к ячейке хранения в специфике конкретной СУБД;

- Провести работу по созданию отдельных колонок в базе данных для выделенных полей и первичную миграцию данных в них из xml-структуры (в дальнейшем она будет производиться автоматически при сохранении объекта).

Подход относительно легкорезализуем и соответствует лучшим практикам разработки на платформе, но имеет свои недостатки. Так, в процессе анализа требуемых для построения базовых отчетов свойств выяснилось, что количество создаваемых в базе данных новых колонок очень велико, значительно выше

предполагаемого. И это с учетом того, что в будущем их потребуется еще больше. В краткосрочной перспективе вырисовывалась картина вынесения практически всех свойств объектов системы в отдельные колонки в базе данных. Данная ситуация практически уничтожает все преимущества использования XML-структуры для хранения данных объекта и в конечном итоге может привести нас к окончательному отходу от базовых принципов и преимуществ использования платформы.

В предыдущей главе мы также описывали недостаток текущих данных в приложении, заключающийся в неструктурированности способа их хранения (из-за часто низкого качества проработки технических решений при разработке). Очевидным является необходимость трансформации данных и в некоторой степени их нормализации перед подготовкой к загрузке в хранилище данных. Как и в случае с предыдущим вариантом реализации, это будет трудно сделать, используя реплику той же БД, что является первоисточником данных. Поэтому отсутствие отдельной области выгрузки также служит недостатком данного подхода.

Таким образом, мы пришли к выводу, что использование индексных таблиц является хорошим решением для построения немногочисленных простых отчетов внутри самого приложения, но не подходит для целей разработки профессионального BI-решения. С данным выводом команда автоматизации пришла к третьей, описанной в следующей части, методике выгрузки данных.

### **3.1.3 Разработка решения для доступа к данным приложения на основе их точечной выгрузки в отдельную базу данных**

Третьим разработанным методом по организации доступа к данным приложения является их периодическая выгрузка в отдельную промежуточную реляционную базу данных, сконструированную именно для целей поддержки



процесса построения хранилища. Данный метод является наиболее перспективным с точки зрения реализации, так что рассмотрим его подробнее.

Стоит отметить, что идея реализации взята из практики, рекомендуемой поставщиком платформы Pega BPM, но в значительной степени доработана. Так, суть предложения заключается в следующем:

- Организовать отдельную реляционную БД для периодической выгрузки в нее требующихся для отчетности данных. Также можно ограничиться созданием отдельной схемы в текущей базе данных (такой вариант хорошо подходит для тестовых сред). Чтобы не дублировать основную базу, предполагается выгрузка только необходимых свойств необходимых объектов.

- Разработать инструмент (или доработать стандартные решения), позволяющий с уровнем производительности, соответствующим заявленным требованиям, извлекать данные из XML-образной структуры объекта из основной базы данных и помещать их в другую реляционную базу данных.

- Разработать механизм, позволяющий автоматизировать запуск упомянутого инструмента по настраиваемому расписанию.

Учитывая то, что объем данных в основном кредитном приложении велик, от разрабатываемого решения по данной схеме требуется соответствие следующим критериям:

- Возможность осуществления выборочной выгрузки данных (как по сущностям, так и по их свойствам). То есть мы можем выгружать не все объекты в системе и не все свойства выгружаемого объекта, а только требуемые для BI-решения.

- Возможность осуществления выгрузки только тех данных, которые изменились с момента последней миграции (отслеживание состояния объектов), так как выгрузка всей базы даже в области только необходимых сущностей очевидно будет занимать много ресурсов, не является

необходимой при неизменности данных и может приводить к появлению лишних дубликатов в области выгрузки.

- Одновременно с этим необходима возможность повторного запуска выгрузки по всем сущностям вне зависимости от того, изменились они или нет. Это потребуется в том случае, если по мере развития отчетности или кредитного ВРМ-приложения появится требование выгрузки дополнительных свойств объектов.

- Обеспечение уникальности записей в области выгрузки. То есть при выгрузке нового состояния объекта должна быть уничтожена запись о его предыдущем состоянии, так как она является ненужным дубликатом.

Для достижения описанных в рамках подхода критериев к инструменту мы можем пойти двумя путями:

- Разработать собственное решение с нуля;
- Использовать готовые решения.

Ни один из этих вариантов на самом деле не подходит в реальной ситуации, так как разработка собственного подобного решения слишком ресурсозатратна и может уступать в производительности, ключевом показателе, другим проверенным механизмам. Готовых же решений, которые удовлетворяют всем критериям, перечисленным выше (и некоторым менее значимым, поэтому не обозначенным здесь) на рынке просто не существует. В итоге командой автоматизации было принято решение взять за основу готовое программное обеспечение, в какой-либо степени удовлетворяющее перечисленным критериям, и разработать целостный механизм, в рамках которого это ПО будет работать, и который будет реализовывать его отсутствующие функции.

Можно утверждать, что последний описанный подход является правильным с архитектурной точки зрения (как в плане возможностей по расширению, так и в плане рекомендаций со стороны разработчика ВРМ-платформы). Кроме того, он позволит разграничить область выгрузки и основную базу данных приложения, без чего невозможно сохранение

преимуществ двух разных подходов: хранения информации об объекте целиком в XML-образной структуре и хранения различных его свойств в разных колонках базы данных. Стоит признать, что каждый из указанных методов хранения имеет свои преимущества в различных ситуациях и логичным и единственным решением представляется создание отдельных баз данных для ВРМ-приложения и для ВІ-решения (области выгрузки) с одинаковыми данными, но разной структурой их хранения. Помимо прочего, отдельная область выгрузки значительно расширяет возможности по трансформации данных в ней и приведения их в то состояние (ту степень нормальности), которая наиболее соответствует требованиям заказчика (подразделения, занимающегося последующей трансформацией и загрузкой данных в хранилище данных).

Очевидными недостатками разработанного третьего подхода является дублирование части данных и возможные дополнительные затраты на приобретение оборудования и лицензий для организации отдельной базы данных и программного решения по ее заполнению на основе принципов, кратко изложенных выше. Тем не менее можно признать, что данные издержки являются необходимыми и избавляют команду от множества возможных проблем в будущем.

Резюмируем наш короткий сравнительный анализ по описанным альтернативным разработкам в таблице 2. Заметим, что каждый из подходов имеет как свои преимущества, так и недостатки. В случае выбора любого из них нам придется столкнуться со сложностями и ограничениями, но опыт реализации ИТ-проектов подсказывает, что обойтись без каких-либо ограничений обычно не получается вне зависимости от степени предварительной проработки разрабатываемого решения.

Сравнение доступных вариантов реализации

№	Описание	Преимущества	Недостатки
1	Сохранение текущей структуры хранения данных и использование механизмов для построения отчетов на основе данных из XML-файла	<ul style="list-style-type: none"> <li>- Отсутствие необходимости в изменении схемы данных и способа работы с ними;</li> <li>- Самый простой способ реализации.</li> </ul>	<ul style="list-style-type: none"> <li>- Очень низкая скорость построения отчетов;</li> <li>- Возможно, потребуются дополнительные трудозатраты для разработки собственного решения или приспособления стороннего решения для наиболее эффективного извлечения данных из XML.</li> </ul>
2	Дублирование данных из XML-схемы в вынесенных колонках базы данных, использование индексных таблиц	<ul style="list-style-type: none"> <li>- Низкая трудоемкость первичной реализации;</li> <li>- Отсутствие необходимости в приобретении дополнительных продуктов и услуг.</li> </ul>	<ul style="list-style-type: none"> <li>- Чрезмерное усложнение схемы данных;</li> <li>- Нивелирование преимуществ текущей реализации;</li> <li>- Задвоение данных.</li> </ul>

№	Описание	Преимущества	Недостатки
3	Использование отдельного специального продукта (дополнения) для платформы Rega для создания отдельной базы данных для отчетности	<ul style="list-style-type: none"> <li>- Рекомендуемый поставщиком подход для решения задачи;</li> <li>- Сохранение преимуществ текущей реализации с устранением недостатков;</li> <li>- Наиболее правильное решение с позиции архитектурного подхода.</li> </ul>	<ul style="list-style-type: none"> <li>- Возможно, потребуется приобретение дополнительных лицензий или разработка собственного решения;</li> <li>- Часть данных будет дублироваться;</li> <li>- Потребуется закупка дополнительного оборудования и выделение ресурсов поддержки.</li> </ul>

Из описания разработанных механизмов для достижения цели уже становится очевидным то решение, которое является наиболее предпочтительным. Это третий подход, заключающийся в организации отдельной области выгрузки данных для BI-решения и разработке описанного уникального механизма по выгрузке данных в эту область. Конечно, он обладает своими недостатками. В дальнейшем будут произведены попытки их минимизации и достижения некоторых преимуществ альтернативных подходов в конечном решении.

### **3.2 Конкретизация деталей уникального механизма выгрузки данных на основе разработанного метода**

В предыдущей части был разработан целевой подход по выгрузке данных и требования к решению, в данной попытаемся немного детализировать его особенности на основе описанных выше критериев его реализации.

Так, нами было принято решение об использовании готового инструмента по выгрузке данных из основной базы ВРМ-приложения из XML-образной структуры в другую реляционную базу данных. Таких инструментов не так уж и много (конкретно известен один). Business Intelligence Exchange (BIX) – дополнительный продукт (дополнение) к ВРМ-платформе Rega для извлечения данных во внешнюю реляционную базу данных или в файл определенного формата (XML, CSV и т.п.), его и будем использовать в качестве каркаса конечного решения (рис. 8). Программное обеспечение требует покупки дополнительной лицензии, но его стоимость незначительна по сравнению с ценой лицензии самой платформы и не является значительной суммой для организации в данной категории трат. Разработчиком решения является компания Regasystems, оно органично вписывается в саму платформу, хотя и обладает рядом недоработок, которые в рамках исследования потребуются устранить.

В официальном руководстве выделяется три формата, в которые может экспортировать данные BIX:

- XML. Можно выгружать либо все переменные класса в XML-формат, либо лишь обозначенные в extract-правиле.
- CSV (comma-separated values). Можно производить экспорт в один или множество текстовых файлов (множественные файлы генерируются в случае выгрузки объектов, имеющих вложенные элементы).
- Relational database. Для записи данных в БД используется стандартная команда INSERT. BIX нормализует экспортируемые данные, помещая их во множество таблиц реляционной базы данных с

соответствующими внешними ключами. При настройке существует возможность обозначения желаемых имен таблиц и колонок (на самом деле нам требуется обозначить их в специальных «рулах», extract-правилах, а потом создать в базе данных самостоятельно с возможностью использованию сгенерированных платформой SQL-скриптов). Важно следить за правильным обозначением типов данных и их длины в экспортируемой базе данных.

Интересна выгрузка именно в отдельную реляционную базу данных. В следующей главе будут описаны некоторые детали организации данной БД и процесса настройки самой выгрузки. Далее лишь отметим некоторые особенности решения, послужившие основанием для его выбора в качестве целевого (с дополнительными доработками в рамках конечного уникального механизма).

Таким образом, с помощью VIX мы сможем реализовать следующие требования разработанного подхода:

- Осуществлять выборочную выгрузку данных (на основе указания определенных сущностей для выгрузки и их свойств) в реляционную базу данных (решение также позволяет выгружать данные в CSV и XML форматы, но нам это не требуется).
- Выгружать только те данные, которые изменились с момента последней выгрузки. Достигается это посредством создания в основной базе данных отдельной таблицы, содержащей в себе дату последней выгрузки по сущности. При следующей выгрузке стандартный механизм сверяет дату обновления записи и дату последней выгрузки по сущности и на основе этого принимает решение о необходимости ее обновления.

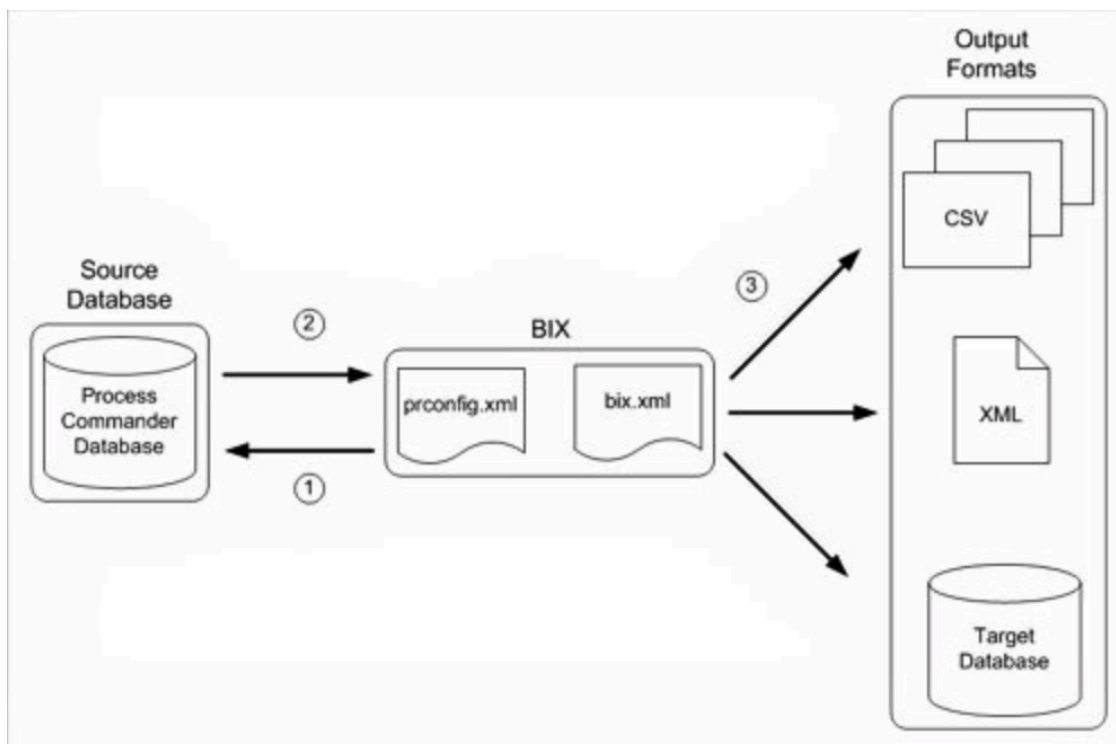


Рисунок 8. Схема выгрузки данных для отчетности с использованием инструмента Business Intelligence Exchange (BIX)

Тем не менее, как упоминалось, для реализации разработанного подхода применение одного стандартного программного обеспечения недостаточно, так как:

- Оно не обеспечивает уникальность записей в области выгрузки. Каждая новая выгрузка измененных записей создает дубликаты, а не обновляет предыдущие.
- Стандартный подход развертывания приложения устарел и не позволяет с должным удобством и производительностью настроить график автоматического осуществления выгрузок в часы наименьшей нагрузки на основную БД.

Для преодоления перечисленных недостатков мы разработаем упомянутый выше уникальный механизм, в рамках которого будет работать стандартное решение. Механизм будет включать в себя:



– Использование современной системы сборок проектов, которая позволит запустить ВІХ с одновременным запуском прочих скриптов, дополняющих его функционал. В частности, будут сделаны скрипты, обеспечивающие возможность выгрузки всех данных с нуля и те, что будут гарантировать отсутствие дубликатов в базе выгрузки. Важно, чтобы разработанные скрипты и проект в системе сборки были оптимизированы и обеспечивали высокую производительность (имеются в виду не только SQL-скрипты, но и прочие, доступные для запуска в системе автоматизации сборок).

– Систему автоматизированного запуска выгрузки посредством использования специальных фоновых агентов или в целевом варианте проекта на сервере автоматизации.

Более подробно детали реализации рассмотрим в следующей главе. Сделаем вывод о том, что стандартное дополнение к платформе Pega ВІХ рекомендуется к использованию поставщиком самой платформы и реализует значительную часть функционала, необходимого для организации ВІ-решения, но не весь. Современные системы автоматизации сборок являются привычным и желательным явлением в проектах разработки программного обеспечения. Данное утверждение можно отнести и к использованию серверов автоматизации, наиболее популярные (и одновременно продвинутые) из которых предлагаются по лицензии open source, предполагающей свободное распространение.

Таким образом, большая часть работы заключалась в анализе и выделении альтернативных подходов использования данных приложения с учетом конкретной специфики контекста исследования для воплощения в жизнь процесса ETL и подготовки хранилища данных – источника для описанных в первой главе исследования комплексных ВІ-решений. Сравнительный анализ данных альтернатив позволил выделить наиболее оптимальную в конкретном

случае. Основные детали разработки выбранного решения были описаны более подробно, что позволяет нам перейти к его практической реализации.

## **4. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ РЕШЕНИЯ ВЫГРУЗКИ ДАННЫХ ДЛЯ VI-ПРИЛОЖЕНИЯ**

В данной главе работы мы поговорим о самой практической реализации описанного ранее метода выгрузки данных. Стоит учитывать, что хотя далее и приведена некоторая детализация архитектуры, все же она значительно размыта с целью сокрытия коммерческой тайны.

### **4.1 Детальная разработка архитектуры решения**

Основываясь на предположениях, сделанных в предыдущих главах, выясним, что нам в итоге потребуется 3 главных компонента в конечном решении:

- База данных ВРМ-приложения в качестве исходного источника данных. Стандартом является СУБД Oracle (к примеру, последняя версия 19с – в актуальной версии платформы Rega ВРМ заявлена ее поддержка). В ней большая часть исходных данных будет храниться в полях специального типа в формате XML, как мы и описывали ранее (компонент 1).
- Механизм выборочного извлечения и переноса данных из первой сущности во вторую. Требования к нему мы описывали в предыдущей главе, в следующей части более подробно опишем реализацию (компонент 2).
- Реляционная база данных, куда мы будем выгружать данные. Будет правильным использовать ту же самую СУБД той же самой версии (компонент 3).

### 4.1.1 Подготовка области выгрузки

Подготовка области выгрузки заключается в развертывании отдельной реляционной БД, в которую в итоге будут попадать в формате, более удобном для дальнейшей организации хранилища, данные из исходной базы данных.

В ситуации выбора системы управления базами данных нет особых альтернатив. Решение от Oracle обладает множеством преимуществ и проверенной надежностью и уже используется как база данных для ВРМ-приложения (откуда выгружать). Желательно, чтобы исходная и целевая базы данных были одинаковыми. На текущий момент в обеих организациях в качестве БД для ВРМ-приложений используется версия Oracle 11.2, срок поддержки которой заканчивается уже в течение года. Можно предположить, что в ближайшее время планируется обновление до новой версии. Единственной возможной версией обновления является Oracle Database 19c, гарантирующая поддержку в течение нескольких лет. Сейчас остановимся на версии целевой БД 11.2 с планами на ближайшее обновление параллельно с исходной.

Как только будет организована область выгрузки в среде разработки, потребуется сделать то же самое и для других сред. В организациях по-разному осуществляется деление на среды, возьмем в качестве примерного следующее:

- Среда разработки, на которой осуществляется непосредственно разработка программного обеспечения;
- Среда тестирования (так называемый Q&A, quality and assurance) для первичной проверки разработанного функционала и выявления дефектов;
- Предпромышленная среда, максимально приближенная к промышленной, на которой осуществляются финальные действия для подтверждения качества поставляемого инкремента;
- Промышленная среда, на которой работают с реальными бизнес-данными уже сотрудники организации.

Для того, чтобы не создавать четыре базы данных, воспользуемся следующим подходом: на разработческой и тестовой средах создадим не отдельные базы данных для области выгрузки, а отдельные схемы данных в рамках исходной БД. Заметной разницы быть не должно. Для промышленной и предпромышленной сред уже стоит организовать отдельные базы данных для области выгрузки (для промышленной – для исключения влияния на производительность основного приложения, для предпромышленной – для максимального сходства с «боевой» средой).

Сами технические детали процесса создания отдельных схем в базе данных или отдельной базы данных в контейнере стандартны и не представляют большого интереса для описания в текущей работе. Ознакомиться с ним подробнее при желании можно, перейдя по ссылкам в списке использованных источников.

На данном этапе исследования стоит также отметить особенность организации выгрузки из исходной БД в целевую. Так, мы говорили о том, что в базе данных ВРМ-приложения одни и те же свойства объекта могут храниться в нескольких местах: в специальном поле типа BLOB в xml-образной структуре и в виде отдельных колонок в таблице (как мы помним, некоторые свойства объектов можно отдельно сохранять и таким образом, если они часто отбираются для построения внутренних несложных отчетов в приложении). Часто возникает путаница, откуда в итоге выгружаются данные о сущности, если они могут дублироваться и в различных ситуациях даже отличаться. В большинстве сценариев использования Rega ВРМ и в дополнении Rega ВІХ единственной правдивой версией объекта является та, что хранится в колонке BLOB. Таким образом, даже если свойство уже вынесено в отдельную колонку, откуда ее явно легче выгрузить, инструментарий ВІХ все равно выполнит более сложную с точки зрения производительности операцию и «вытащит» значение свойства из XML-образного представления объекта. По следующим причинам

данное поведение является выгодным и не требует изменения при дальнейших доработках:

- Реляционная база данных требует наложения ограничений на формат и объем информации, хранящейся в колонке. В XML же у свойства есть собственный тег, внутри которого может располагаться любое значение любой длины (отдельные валидации можно предусмотреть в процессной логике). Таким образом, может быть ситуация, когда, например, текстовое значение свойства не уместится в выделенную колонку определенного размера и обрезается с конца (тем временем в XML оно сохраняется полностью) [41].

- Существует множество различных способов «вынесения» свойства в отдельную колонку в базе данных. Нет никаких гарантий, что все дополнительные колонки в исходной базе данных для «вынесенных» свойств были созданы корректно. При неправильной реализации значение в XML по существующим записям может не соответствовать значению в созданной колонке.

Таким образом, после настройки пустой отдельной областью выгрузки в виде реляционной базы данных, можно переходить к дальнейшим шагам.

#### **4.1.2 Настройка стандартного дополнения Rega VIX для организации выгрузки**

Как было сказано, дополнение Rega VIX поставляется различными способами для различных версий платформы и состоит из 2 компонентов [40]:

- «Рулсета» (набора «рулов», см. выше), импортируемого в ВРМ-приложение. В последних версиях платформы они уже стандартно включены в платформу, но рассматриваемые проекты пока не обновились.

- Отдельного консольного приложения, опционального для использования (требуется для запуска выгрузки с отдельного сервера, как в нашем случае).

Для настройки дополнения Pega VIX нам потребовалось осуществить описанный далее набор действий. Стоит отметить, что далее представлен не пересказ инструкции, а практический пример, который в определенной степени отличается от рекомендуемого поставщиком набора действий.

- Импортировать в приложение рулсеты дополнения Business Intelligence Exchange стандартными средствами.

– Было решено создать отдельное приложение в платформе именно для целей выгрузки, построив его на базе существующего бизнес-решения (то есть данное приложение будет наследовать все свойства текущего – особенность организации «слоев» построенных друг на друге приложений в Pega). В созданное приложение включим стандартные рулсеты дополнения Pega VIX и все настройки для осуществления выгрузки (extract-рулы, см. ниже). Использование отдельного приложения выгодно с точки зрения минимизации влияния на основное приложение и возможностей по организации отдельных поставок доработок «куска» VI-решения на платформе и основного BPM-приложения.

– Создать специальные extract-правила в среде разработки Pega для каждого из необходимых классов (связь внутренних классов в платформе со структурой данных описывалась в предыдущих главах). То есть с помощью блоков специального типа точно описали тот набор данных (свойств экземпляров), который нам требуется выгружать из каждой требуемой сущности (класса). С развитием VI-решения и будущими поставками данный набор выгрузки будет расширяться.

– Создать отдельные правила, описывающие в платформе подключение к реляционной базе данных, являющейся целевой для выгрузки. В случае необходимости (которая часто возникает на

разработческих средах для предварительного тестирования) мы можем обойтись без консольного приложения и запустить точечную выгрузку сразу из среды разработки платформы Pega BPM.

Таким образом, у нас уже есть настроенное решение для копирования данных в область выгрузки. В следующих частях мы поговорим о том, как организовать данную выгрузку автоматически и расширить ее функционал для достижения требований, описанных ранее.

#### **4.1.3 Автоматизация процесса запуска выгрузки и создание «точек расширения» функционала**

По результатам требуется осуществлять автоматически по расписанию следующие действия:

- Запуск специально разработанной нами (см. далее) логики для подготовки к выгрузке.
- Запуск выгрузки стандартным механизмом Pega BIX с различными параметрами.
- Запуск специально разработанной логики (см. далее) после завершения выгрузки для очистки дубликатов и выполнения других действий, предусмотренных требованиями к проекту и не предусмотренных стандартным функционалом расширения.

Говоря иначе, на данном этапе требуется автоматически запускать выгрузку и предусмотреть наличие «точек расширения» в определенных местах процесса. Специальных блоков в механизме, в которые можно размещать собственную логику обработки, ориентируясь на требования, изложенные ранее.

К параметрам запуска выгрузки отнесем следующие:



- Тип выгрузки: выгрузка только «дельты» (изменившихся с момента последней выгрузки данных) или полная с предварительной очисткой целевой БД и пересозданием ее структуры;
- Тип выгрузки: выгрузка сначала или с момента, на котором остановилась последняя неуспешная выгрузка;
- Уровень логирования процесса и др.

Автоматизировать процесс запуска выгрузки и выполнения дополнительной логики можно различными способами, выделены два наиболее подходящих:

- Без использования отдельного консольного приложения, поставляемого с дополнением Pega VIX. То есть посредством создания специальной сущности внутри платформы Pega BPM (элемента автоматизации, так называемого «activity»), запускающей выгрузку. Для его работы по расписанию также нужно предусмотреть разработку еще одного элемента, фонового агента.
- Посредством разработки отдельного проекта на современной системе автоматизации сборок и настройка возможности его запуска через сервер автоматизации с поддержкой концепций непрерывной интеграции (continuous integration) и непрерывной поставки (continuous delivery).

Агентом называется специально правило в платформе, настроенное на выполнение определенной логики по заданному расписанию или для обработки предварительно размещенных в специальной очереди объектов.

Не вдаваясь в детали, заметим, что первый вариант автоматизации был отклонен, за основу взят второй. Основными причинами являлись:

- Первый вариант реализации предполагает ограничение возможностей выгрузки стандартными инструментами платформы. И хотя этих инструментов достаточно для реализации механизма любой сложности, все же данная ситуация не похожа на ту, для которой предназначена платформа Pega BPM.

- При реализации первого варианта вся работа по осуществлению выгрузки будет происходить на основном сервере приложений. Заметим, что извлечение данных из XML-структуры и перемещение их в отдельную реляционную БД является довольно ресурсозатратной операцией. Хотя агента, выполняющего ее, можно запускать по ночам, все же логичным является ее максимальное вынесение за рамки выделенного пула ресурсов на поддержку основного ВРМ-приложения. Второй вариант реализации предполагает взаимодействие только с базой данных, работа по осуществлению выгрузки может быть запущена на любом отдельном сервере и может поддерживаться сервером автоматизации.

- Использование сервера автоматизации и современной системы автоматизации сборки приносит дополнительные преимущества в проект, некоторые из которых будут описаны далее.

Таким образом, для реализации целевого решения потребуется осуществить следующий набор действий:

- Выбрать современное средство автоматизации сборки и реализовать на нем проект, который позволяет запустить стандартный инструмент осуществления выгрузки с применением дополнительной логики.

- Реализовать проект на современном сервере автоматизации, который позволит запускать упомянутую сборку автоматически или вручную с минимизацией затрат человеческих ресурсов на рутинные операции.

В официальной инструкции дополнения Pega VIX есть короткое упоминание возможности его использования со средствами автоматизации сборок, но оно устаревшее. Так, предлагается использовать решение Apache Ant, которое очевидно уже не соответствует современным тенденциям разработки ПО.

Рассматривалось использование одной из следующих современных систем автоматизации сборок: Apache Maven или Gradle. Не будем в данной работе

вдаваться в детали их сравнения, в списке использованных источников есть ссылки на множество интересных ресурсов на эту тему. В таблице 3 представлена очень короткая экспертная оценка различных аспектов систем автоматизации сборки (по пятибалльной шкале) [39]. Отметим, что помимо трех упомянутых альтернатив есть еще и другие решения (например, Meson). Тем не менее, именно две основные альтернативы, Maven и Gradle, рассматривались вследствие распространенности их применения на объектах исследования и наличия у них большого сообщества разработчиков и значительных перспектив развития.

Таблица 3

Экспертное сравнение популярных систем автоматизации сборки по пятибалльной шкале

Критерий	Ant	Maven	Gradle
Сложность изучения	3	4	3
Скорость сборки	4.5	4.5	3.5
Сложность проектов	1.5	4.5	3
Доступность плагинов	4	3	3
Развитость сообщества и документации	3	5	2
Интеграция с инструментами разработки	5	3	4
Итог	21	24	18.5

Каждое решение обладает своими преимуществами, но в проекте было решено взять за основу более современный Gradle. Данная система автоматизации построена на основе концепции запуска зависящих друг от друга задач. Синтаксис проекта автоматизации может быть основан на одном из двух возможных Java-подобных языков: Groovy или Kotlin. Мы пользуемся первым,

так как он кажется команде более удобным. Я не могу приводить в работе структуру самого конечного решения, поэтому на рисунке 9 представлен типичный каркас подобных проектов, конечное решение лишь должно включать в себя несколько дополнительных файлов, среди которых:

- Файлы самого стандартного решения по организации выгрузки;
- SQL-скрипты для создания структуры конечной базы данных;
- Дополнительные скрипты для удаления дубликатов после выгрузки или выполнения любой другой логики.



Рисунок 9. Возможная структура проекта в системе автоматизации сборки Gradle

Сама логика автоматизации сборки (описание «задач» – базовых блоков в Gradle, как мы уже упоминали) в большей степени прописывается в файле build.gradle, конфигурационный файл (в котором могут содержаться в том числе настройки подключения к БД) и который отличается на каждой среде, – settings.gradle.

Упомянутая выше последовательность шагов работы механизма была оформлена в виде взаимозависимых «задач» Gradle. Для управления состоянием базы данных существует множество полезных дополнительных плагинов, которые мы также будем использовать. То есть, по сути, в проекте сборки запуск самой выгрузки представляет из себя одну из «задач» (термин из Gradle), мы

имеем возможность добавить любое количество дополнительных задач для расширения функционала разрабатываемого механизма, о чем будет более подробно сказано далее.

Если говорить о сервере автоматизации, то использование бесплатного и целевого на проекте решения Jenkins является очевидным. На сервере автоматизации предполагается настройка «работы» по автоматическому (или ручному по желанию) запуску разработанного механизма. Для этого сам проект по автоматизации сборки предварительно выкладывается в git-репозиторий, ссылка на который указывается в «работе» Jenkins. В качестве системы управления версиями на проекте используется решение Bitbucket от компании Atlassian.

Предварительная «грубая» схема решения представлена на рисунке 10.

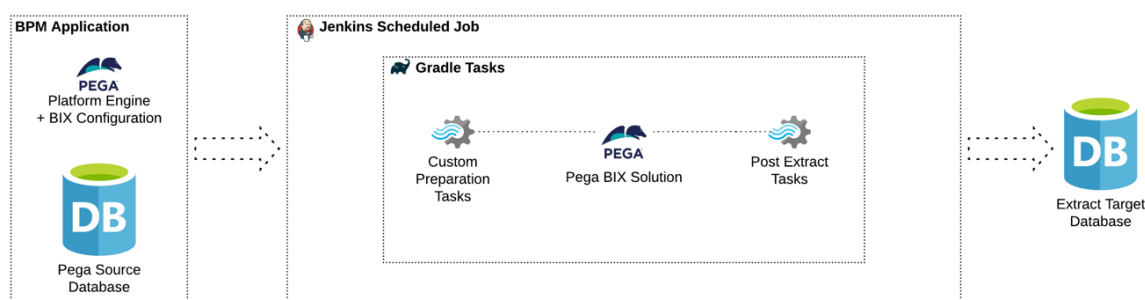


Рисунок 10. Приблизительная схема работы разрабатываемого решения

Таким образом, нами был создан простой механизм запуска выгрузки с возможностью включения дополнительной логики в проект сборки. Упомянутая «дополнительная логика» представляет из себя собственное представление функций, отсутствующих в стандартном решении, и являющихся частью выдвинутых требований к разрабатываемому механизму. Далее опишем ее реализацию более подробно.

#### 4.1.4 Разработка дополнительного функционала выгрузки

Теперь, когда на практике реализован максимально стандартный механизм выгрузки (хотя способ ее запуска относительно уникален), мы можем расширить его дополнительными функциями, которые описывались в предыдущих главах.

Одной из отличительных черт механизма является возможность предварительной проверки структуры базы данных, в которую осуществляется выгрузка, и ее автоматического изменения в случае наличия доработок (приложенных новых версий SQL-скриптов, формирующих ее структуру). Для реализации данного функционала мы использовали популярный плагин к системе автоматизации сборок, flyway. Настроим работу данного расширения следующим образом:

- В директории сборки создается отдельная папка, в которую складываются SQL-скрипты, формирующие структуру области выгрузки. Сами скрипты версионизируются по определенным правилам именования.
- В области выгрузки создается таблица, содержащая в себе список запущенных скриптов с обозначением их версии, наименования и статуса запуска (успешно или неуспешно).
- В проект автоматизации добавляется зависимости от упомянутого дополнения и новая задача из него, совершающая активность по запуску скриптов из указанной выше папки (при этом только тех, что еще не были запущены – на основе данных из описанной таблицы). Основная задача по запуску выгрузки ставится в зависимость от данной, что гарантирует актуализацию структуры БД до запуска самой выгрузки.

Таким образом, мы получаем систему контроля доработок в структуре БД. При выпуске новых версий продукта (добавлении новых полей в выгрузку) потребуется модифицировать структуру области выгрузки, что теперь будет происходить автоматически. Также решение предоставляет некоторые дополнительные преимущества:

- При ручном изменении структуры целевой базы данных следующая выгрузка завершится с ошибкой, так как происходит сравнение контрольных сумм. То есть гарантируется прозрачное применение изменений единым способом.

- В перспективе структура целевой БД станет значительно сложнее, чем в момент запуска. Тем не менее применение решение гарантирует, что мы можем ее полностью воссоздать последовательным запуском версионированных скриптов из единой папки их хранения.

- Плагин добавляет доступ также к некоторым другим задачам Gradle, которые можно запускать отдельно: очистке базы данных, воссоздания ее структуры с нуля и т.д.

В рамках данной работы опишем еще один последний большой «кусочек» доработок, реализованных в рамках выгрузки средствами системы автоматизации сборки по требованиям к конечному решению, предоставленным в предыдущей главе. Речь идет о части механизма, ответственного за удаление дублирующихся записей в области выгрузки.

Как мы упоминали, стандартное решение выгружает обновленные объекты в реляционную базу данных. Так, если выгрузка происходила несколько раз и объект между выгрузками обновлялся (или просто было принято решение по каким-либо причинам выгрузить его снова несмотря на отсутствие обновлений), то в области выгрузки создадутся несколько записей одного и того же объекта: одна актуальная, соответствующая последней его успешной выгрузке, и несколько устаревших. От нас требуется реализовать задачу в системе автоматизации сборок Gradle, которая будет запускаться после осуществления сборки и по определенному алгоритму удалять неактуальные записи. Кажущаяся банальной задача по факту таковой не является, так как:

- Область выгрузки организована специфично с учетом того, что в ней могут содержаться как таблицы с основным выгруженным объектом,

так и связанные с ней дополнительные таблицы, содержащие данные массивов, хранящихся в основном объекте.

– Необходимо предусмотреть множество различных сценариев: например, объект может измениться только таким образом, что будет удален один из элементов вложенного в него массива (который хранится в области выгрузки в отдельной связанной внешним ключом с таблицей основного объекта таблицей).

– Подобный функционал удаления дубликатов критически необходим, но важно сделать его максимально простым на случай будущих доработок и максимально производительным. Так как речь идет о громадных массивах данных, ключевую роль играет не столько работающая реализация, сколько наиболее оптимальная.

Нетривиальная задача с реализацией части механизма по очистке дубликатов доставила много проблем, так как первые варианты ее воплощения не решали поставленную задачу в полной степени, что удавалось обнаружить только при промышленной эксплуатации. Автор исследования участвовал в различных этапах реализации целостного механизма по выгрузке данных, разработка окончательного варианта наиболее оптимального механизма удаления дубликатов стала одной из его основных задач на проекте.

Проблема была решена разработкой специальных SQL-скриптов, автоматически запускающихся в области выгрузки после ее успешного осуществления. Автоматический запуск этих скриптов, конечно, тоже реализован через соответствующую задачу в системе автоматизации сборок Gradle.

Чтобы суммировать вышесказанное, полезно описать наиболее часто используемый и стандартный сценарий осуществления выгрузки:

1. Автоматическая актуализация структуры целевой БД на основе последних еще не запущенных на среде версий SQL-скриптов формирования структуры.



2. Опционально: очистка информации о последней дате выгрузки (эта дата стандартно используется для выборки записей, с того момента измененных) и базы данных выгрузки в случае необходимости ее осуществления «с нуля».

3. Запуск стандартного механизма выгрузки с применением предварительно настроенной/указанной конфигурации.

4. Запуск дополнительных скриптов после осуществления выгрузки для удаления дубликатов, старой информации об обновленных записях.

Таким образом, разработка уникального решения основана на применении современных практик автоматизации и, как предполагается, может служить примером для подобных автоматизаций на смежных ВРМ-проектах в организации. Более детальное рассмотрение некоторых отдельных деталей решения может являться темой дальнейших работ автора. Стоит подчеркнуть, что решение было действительно реализовано в компании и в настоящий момент успешно эксплуатируется.

#### **4.2 Описание особенностей организации проекта и экспертная оценка его эффективности**

В данной части коротко опишем организационную и экономическую проекции проекта. Стоит, однако, иметь в виду, что невозможно точно определить ту явную выгоду, которую может принести реализация концепции процессной аналитики.

Проект планировался и реализовывался с использованием рекомендаций методологии SCRUM. В частности:

- Окончательный функционал был описан в информационной системе управление проектами Atlassian Jira в виде еpic-а и разбит на пользовательские истории. Для организации работы активно

использовались возможности внутренней agile-ориентированной вики-системы Atlassian Confluence.

- Все работы были декомпозированы и описаны в виде задач в бэклоге команды под соответствующим еріс-ом.
- Во время нескольких сеансов командного grooming-a, оценки и планирования задачам были выставлены соответствующие оценки трудозатрат и приоритеты.
- Задачи были спланированы и выполнены в виде MVP в 2 спринта (каждый спринт длится на проекте 2 недели).

Применение стандартного подхода к планированию работ в виде четкого разбиения последовательности действий и построения диаграммы Ганта трудноосуществимо, так как работа в кроссфункциональной команде с использованием итерационного подхода предполагает выполнение различного рода функций параллельно в рамках небольшого промежутка времени и при тесном взаимодействии специалистов разной области. В предыдущей главе отмечалось, что в процессе реализации часто приходилось отказываться от намеченного плана из-за возникающей необходимости в пересмотре выбранного в начале подхода.

Из-за особенностей современных гибких подходов к разработке информационных систем и постоянном развитии модели данных BPM-приложения, отдельные доработки BI-решения осуществляются до сих пор. Далее предоставлены затраты на разработку MVP-решения и экспертная оценка стоимости поддержки будущих доработок.

Для реализации MVP-решения были привлечены следующие специалисты (все являются либо сотрудниками банка, либо сотрудниками компании-подрядчика, постоянно работающими на кредитном проекте):

- Архитектор направления;
- 2 разработчика;
- 3 аналитика (с навыками как бизнес-анализа, так и системного);

- 2 тестировщика.

Автор данного исследования выступал в проекте в качестве разработчика. Он участвовал на всех этапах реализации решения, а его роль заключалась в выполнении части критических работ, акцент на чем сделан в предыдущих частях исследования.

Приблизительная стоимость трудовых ресурсов для реализации проекта представлена в таблице 4.

Таблица 4

Стоимость трудовых ресурсов для реализации проекта

Роль	Средняя ставка, Р/час	Количество отработанных часов	Общая стоимость, Р
Архитектор	1300.00	35	45,500.00
Аналитик	650.00	135	87,750.00
Разработчик	820.00	80	65,600.00
Тестировщик	550.00	75	41,250.00
Всего			240,100.00

Источником данных о средней стоимости сотрудника для организации является Федеральная служба государственной статистики.

Значимыми прочими затратами помимо затрат на трудовые ресурсы в описанных проектах выступают следующие:

- Лицензирование применяемых информационных систем. Данная статья расходов не учитывается по перечисленным далее причинам:
  - Стоимость лицензий на большинство программных комплексов согласовывается в результате переговоров компании-производителя ИС и банка и не подлежит раскрытию;

- Оценка количества всех проектов, которыми совместно используются лицензии, является трудноосуществимой вследствие множества причин;

- Согласно организационно-технологическому архитектурному решению проекта по автоматизации процессов выдачи ипотечных кредитов, приобретение дополнительных лицензий помимо тех, что уже есть у банка, не требуется (так же, как и пересмотр условий существующих).

- Стоимость оборудования. Речь даже идет не столько об оборудовании, а об аренде пула ресурсов. Закупка дополнительного оборудования не требовалась на этапе реализации MVP и может быть востребована только при развитии проекта. Приблизительная ежемесячная стоимость аренды ресурсов указана далее.

- Стоимость аренды рабочих мест для сотрудников. Не является существенной для столь небольшого по продолжительности проекта и включена в стоимость сотрудника, указанную в таблице выше.

- Стоимость электроэнергии и прочих материалов. Минимальна и может быть проигнорирована.

Оценить большое количество расходов не представляется возможным, так как они в большей части представляют из себя результат договоренностей между различными организациями, не подлежащий разглашению, и могут сильно варьироваться в зависимости от множества факторов. Также часть понесенных однажды расходов (например, на единоразовую покупку неограниченной лицензии) распространяется на огромное количество проектов. Тем не менее, указанные затраты не являются сколь либо существенными, основная статья расходов – оплата труда сотрудников.

Для поддержания работоспособности разработанного решения (например, актуализации правил выгрузки данных при изменении их структуры в исходной базе данных или поддержки выполнения механизма выгрузки на промышленной

среде), требуется выделение в среднем 40 часов работы разработчика среднего уровня и 8 часов работы сотрудника поддержки в месяц. Исходя из оценки стоимости часа работы сотрудника, представленной выше, затраты на разработчика составляют 32 800 руб./мес., на специалиста поддержки – 5 200 руб./мес. (на основе стоимости для компании часа работы подобного сотрудника в 650 рублей). Аренда пула задействованных ресурсов, количество которых зарезервировано уже с учетом возможности значительного увеличения объема выгружаемых данных, оценивается в сумму около 90 000 рублей в месяц (на основе данных калькулятора аренды ресурсов Microsoft Azure) [28]. Таким образом, разработка MVP-решения обойдется в 240 000 рублей, поддержка решения без серьезных доработок – в 128 000 рублей в месяц. Для реализации конечного профессионального BI-решения потребуется, конечно, провести еще большой объем работ (вроде организации хранилища данных), но уже на данном этапе, обладая описанной областью выгрузки, можно реализовывать наиболее простую процессную аналитику.

Альтернативой внедрения предложенной системы выступает использование ручного труда для генерации полезных выводов на основе преимущественно ручной обработки разрозненных данных кредитного процесса. Очевидно, что ручная обработка и анализ данных процесса (так называемая ручная процессная аналитика, о которой рассказывалось в первой главе исследования), не может осуществляться в тех же объемах, что и автоматическая. Тем не менее в случае отсутствия предлагаемой разработки она в любом случае требуется для некоторых участков процесса [14]. К таким участкам относятся, к примеру, процессы проведения комплаенс-проверок заемщиков с целью исключения неумышленного финансирования терроризма или других незаконных действий. Уже возникали инциденты, когда из-за программной ошибки или невнимательности сотрудников клиентами различных банков становились лица, обслуживание которых финансовыми организациями противоречит нормам действующего законодательства [7]. Обзор данных

процесса уже после его выполнения критически важен для наиболее быстрого выявления подобных инцидентов, оперативной работы с последствиями и совершенствования системы для их предотвращения в будущем. Ручное выполнение анализа только этого небольшого участка потребует, если исходить из объемов выдаваемых в рассматриваемой организации кредитов (ипотечных в Альфа-Банке), работы минимум 17 аналитиков (при занятости в 40 часов в неделю), что равняется 1 768 000 рублей дополнительных расходов в месяц. Стоит не забывать о том, что речь идет только об одном небольшом участке процесса, анализ данных с которого на текущий момент считается обязательным (что не исключает возможности выделения в ближайшем будущем еще других критических для анализа участков) [5].

На основе приведенного выше примера можно сделать вывод о том, что предлагаемая разработка позволяет экономить миллионы рублей ежемесячно только на некоторых критичных участках процесса. Тем не менее оценить явную экономическую выгоду от реализации невозможно, так как реализация концепции процессной аналитики в долгосрочной перспективе может послужить источником для приобретения огромного количества конкурентных преимуществ. Более подробно о неявных выгодах излагалось в первой главе исследования. Согласно предположениям владельца продукта, основного инициатора доработок, в перспективе применение процессной аналитики на различных участках процесса может привести к инициации доработок, позволяющих сократить все совокупные издержек на реализацию процесса в 1,5 раза. Воплощение проекта в жизнь является выгодной с экономической точки зрения, что также подтверждено инициаторами из числа руководителей организации.

Таким образом, нами были не только описаны альтернативы и уникальные решения по организации выгрузки данных из ВРМ-приложения для реализации VI-решения, но и была проведена успешная апробация рекомендаций. В реальной компании (Альфа-Банк в конкретном случае, хотя опыт похожих

проектов в Сбербанке тоже учитывался) был разработан и успешно внедрен в промышленную эксплуатацию MVP продукт. Сотрудники организации уже могут пользоваться преимуществами конечного BI-решения и оставляют положительную обратную связь по поводу его полезности. Затраты на реализацию MVP оказались небольшими для организации, а стоимость его поддержки минимальна по сравнению с альтернативными вариантами организации анализа данных процесса. Предполагаемые выгоды от разработки значительно превышают объем ресурсов, затраченный на ее осуществление.

## ЗАКЛЮЧЕНИЕ

Подводя итог проделанной работе, можно сделать вывод о достижении поставленной цели. Нами было разработано BI-решение, представляющее из себя механизм по выгрузке данных процесса, автоматизированного на платформе Rega BPM, в формат, наиболее оптимальный для использования профессиональным BI-приложением.

В рамках достижения цели исследования были решены следующие задачи:

- Изучены теоретические аспекты исследования: возможности процессной аналитики и особенности ее использования в том числе для процессов, автоматизированных с помощью класса систем управления бизнес-процессами, посредством изучения литературных источников и существующих методик и подходов;
- Произведены сбор и анализ эмпирических данных, обзор интересующих аспектов деятельности объекта исследования для составления контекста работы;
- Произведен сравнительный анализ ключевых способов достижения цели исследования и доступных решений в рамках контекста конкретного объекта исследования;
- Разработаны уникальные практические рекомендации и решения на базе знаний, полученных при выполнении предыдущих задач;
- Проведена апробация практических рекомендаций и решений по внедрению процессной аналитики на изучаемые участки процессов с выработкой уникальных деталей реализации;
- Произведена оценка организационной и экономической аспектов реализации практического решения.

Внедрение BI-решения на участки организации, сконцентрированные на управлении бизнес-процессами как ключевыми единицами деятельности, может привести к реализации концепции процессной аналитики, несомненно выгодной в описываемых условиях. Для реализации упомянутой концепции и разработки



профессионального BI-решения существует множество преград, заключающихся в специфических способах хранения данных Pega BPM приложениями. И хотя стандартный инструментарий платформы, включающий в себя дополнения, приобретаемые по отдельным лицензиям, не способен в полной степени решить проблему организации данных BPM-приложения в форму, наиболее удобную для дальнейшего использования профессиональным BI-решением, нами был успешно разработан уникальный механизм для достижения данной цели. Разработанный механизм использует стандартные инструменты платформы вкупе с современными инструментами разработки программного обеспечения, что позволяет достичь сформулированных требований. Практическая реализация в виде разработки минимально жизнеспособного продукта доказала успешность проекта и его экономическую целесообразность.

Следующие положения вынесены на защиту по итогам написания выпускной квалификационной работы:

- Структура хранения данных в приложении, автоматизированном на платформе Pega BPM, не является оптимальной для целей построения отчетов, и стандартный инструментарий платформы, включающий в себя продукты, приобретаемые по дополнительной лицензии, не является достаточным для подготовки данных BPM-приложения для профессионального BI-решения;
- Разработанный уникальный механизм организации выгрузки данных из BPM-приложения для BI-решения, включающий в себя очистку дубликатов в области выгрузки и некоторые другие преимущества, можно считать MVP-решением, достаточным для подготовки к организации хранилища данных и последующего запуска BI-приложения на основе использования выгружаемых им данных.

В качестве дальнейшего развития темы выпускной квалификационной работы предлагается описание и разработка универсального механизма трансформации данных для Pega BPM приложения. Исследования в данном направлении представляются актуальными, так как проблема разрозненности и неструктурированности хранения больших уже накопленных массивов данных в BPM-приложении являлась наиболее острой из тех, с которыми автор столкнулся при апробации разработанных рекомендаций. Механизм, который позволил бы наиболее оптимальным способом осуществить трансформацию имеющихся данных для их более правильной с архитектурной точки зрения организации, упростил бы не только доработки BI-решения, но и BPM-приложения в целом.

## СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Конституция РФ. М.: Издательство «Эксмо», 2019
2. Гражданский кодекс РФ на 1 мая 2020 года. М.: Изд-во АСТ. 2020.- 384 с.
3. Налоговый кодекс РФ. Части первая и вторая. – М.: Издательство «Проспект», 2020.-516с.
4. Федеральный закон от 2 декабря 1990 г. № 395-1 «О банках и банковской деятельности».
5. Федеральный закон от 7 августа 2001 г. №115-ФЗ «О противодействии легализации (отмыванию) доходов, полученных преступным путем».
6. Федеральный закон от 16 июля 1998 г. №102-ФЗ «Об ипотеке (залоге недвижимости)».
7. Инструкция Банка России от 25.08.2003 № 105-И «О порядке проведения проверок кредитных организаций (их филиалов) уполномоченными представителями Центрального банка Российской Федерации»
8. Положение Банка России от 16.12.2003 г. № 242-П «Об организации внутреннего контроля в кредитных организациях и банковских группах».
9. Об утверждении порядка проведения государственной итоговой аттестации по образовательным программам высшего образования - программам бакалавриата, программам специалитета и программам магистратуры. Министерство образования и науки Российской Федерации. Приказ от 29 июня 2015 г. N 636.
10. ГОСТ 34.601-90 Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
11. ГОСТ 7.1-2003. Библиографическая запись. Библиографическое описание. Общие требования и правила составления.

12. ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе. Структура и правила оформления».
13. ГОСТ Р ИСО 9000-2008. Системы менеджмента качества. Основные положения и словарь.
14. Авдошин, С. М. Информатизация бизнеса. Управление рисками / С.М. Авдошин, Е.Ю. Песоцкая. - М.: ДМК Пресс, 2016. - 176 с.
15. Бабенко В. В. Практический анализ бизнес-процессов. – Сыктывкар, 2010, 290 с.
16. Балдин, К.В. Информационные системы в экономике: учебник /К.В. Балдин, В.Б. Уткин. — Дашков и К°, 2017. — 395с.
17. Введение в реляционные базы данных /С. Кузнецов. — Национальный Открытый Университет «ИНТУИТ», 2016. — 248с.
18. Ляндау Ю.В., Стасевич Д.И. "Автоматизация бизнес-процессов" М.: Издательство "Издательский Центр РИОР", - 2017.
19. Афанасьев М. А., Староверова О.В., Уринцов А.И. Адаптация как процесс управления хозяйствующим субъектом // Вестник Московского университета МВД России. 2016. № 2. С. 201-206.
20. Афанасьев М.А., Староверова О.В., Уринцов А.И. Компьютерный инструментарий управления эффективностью бизнеса // Вестник Московского университета МВД России. 2016, № 8, С.208-211
21. Долженко Р.А. выпуск возможности каскадирования целей организации среди участников agile-проектов // Инновации №1. 2018. № 231. С. 101-109.
22. Альтернативы ВРМ [Электронный ресурс] / Global CIO. Режим доступа: <http://www.globalcio.ru/workshops/321/> (дата обращения: 12.06.2020).
23. Альфа-банк в 2018 году возобновит собственную программу ипотеки в большинстве регионов [Электронный ресурс] / РИА-Новости.

Режим доступа: [https://realty.ria.ru/mortgage\\_news/20170907/408907024.html](https://realty.ria.ru/mortgage_news/20170907/408907024.html)  
(дата обращения: 27.05.2020).

24. В Альфа-Банке завершен первый этап работ по внедрению системы управления бизнес-процессами Rega PRPC [Электронный ресурс] / Официальный сайт Альфа-банка. Режим доступа: <https://alfabank.ru/press/news/2014/8/26/2.html> (дата обращения: 27.05.2020).

25. Годовой отчет АО «Альфа-Банк» [Электронный ресурс] / Центр раскрытия корпоративной информации. Режим доступа: <https://www.e-disclosure.ru/portal/files.aspx?id=1389&type=2> (дата обращения: 27.05.2020).

26. Информационные технологии в Альфа-Банке [Электронный ресурс] / TAdviser. Режим доступа: <http://www.tadviser.ru/index.php/> (дата обращения: 27.06.2020).

27. «Клиент без смартфона – потенциально не наш клиент» [Электронный ресурс] / Коммерсантъ. Режим доступа: <https://www.kommersant.ru/doc/3932096/> (дата обращения: 27.06.2020).

28. Калькулятор цен Azure [Электронный ресурс] / Microsoft. Режим доступа: <https://azure.microsoft.com/ru-ru/pricing/calculator/> (дата обращения: 29.06.2020).

29. Компания Альфа-банк Россия [Электронный ресурс] / TAdviser. Режим доступа: <http://www.tadviser.ru/index.php/> (дата обращения: 27.05.2020).

30. Организационная структура и менеджмент [Электронный ресурс] / Альфа-Банк. Режим доступа: [https://alfabank.ru/about/corporate\\_governance/orgstructure/](https://alfabank.ru/about/corporate_governance/orgstructure/) (дата обращения: 27.06.2020).

31. Сбербанк вновь самый дорогой российский бренд в мире [Электронный ресурс] / Сбербанк. Режим доступа: <https://www.sberbank.com/ru/news-and-media/press-releases/article?newsID=6ee870ea-fc58-42fb-b946->

7dec5663a481&blockID=7&regionID=77&lang=ru&type=NEWS&\_ga=2.24523706.1694277559.1591639935-696480102.1587295706 (дата обращения: 27.06.2020).

32. Стратегия развития Сбербанка 2020 [Электронный ресурс] / Сбербанк. Режим доступа: <https://www.sberbank.com/common/img/uploaded/files/sberbankdevelopmentstrategyfor2018-2020.pdf> (дата обращения: 27.06.2020).

33. Устав АО «Альфа-Банк» [Электронный ресурс] / Центр раскрытия корпоративной информации. Режим доступа: <https://www.e-disclosure.ru/portal/files.aspx?id=1389&type=1> (дата обращения: 27.06.2020).

34. Cebotarean Elena. Business Intelligence // Journal of Knowledge Management, Economics and Information Technology. 2012, P.7-28

35. 10 Critically Important Business Intelligence Software Features [Электронный ресурс] / SelectHub. Режим доступа: <https://www.selecthub.com/business-intelligence/critical-business-intelligence-features/#survey> (дата обращения: 07.06.2020).

36. A Business Intelligence System [Электронный ресурс] / Altaplana. Режим доступа: <http://altaplana.com/ibm-luhn58-BusinessIntelligence.pdf> (дата обращения: 07.06.2020).

37. About Business Intelligence Exchange (BIX) [Электронный ресурс] / Pega Community. Режим доступа: <https://community.pega.com/knowledgebase/articles/about-business-intelligence-exchange-bix/> (дата обращения: 10.06.2020).

38. Agile на 11 000 сотрудников [Электронный ресурс] / VC. Режим доступа: <https://vc.ru/sberbank/38179-agile-na-11-000-sotrudnikov> (дата обращения: 10.06.2020).

39. Ant vs Maven vs Gradle: Java Build Tools Comparison [Электронный ресурс] / JRebel. Режим доступа: <https://www.jrebel.com/blog/java-build-tools-comparison> (дата обращения: 10.06.2020).

40. BIX 8.1 User Guide [Электронный ресурс] / Pega Community. Режим доступа: <https://community.pega.com/knowledgebase/documents/bix-81-user-guide> (дата обращения: 10.06.2020).
41. BIX extraction process and use of BIX [Электронный ресурс] / Pega Community. Режим доступа: <https://collaborate.pega.com/question/bix-extraction-process-and-use-bix> (дата обращения: 10.06.2020).
42. BIX design models [Электронный ресурс] / Pega Community. Режим доступа: <https://community.pega.com/knowledgebase/articles/business-intelligence-exchange-bix/bix-design-models> (дата обращения: 10.06.2020).
43. Business Intelligence: A Primer [Электронный ресурс] / TIM. Режим доступа: <https://timreview.ca/article/284> (дата обращения: 07.06.2020).
44. Customer Success [Электронный ресурс] / Pegasystems Inc. Режим доступа: <https://www.pega.com/customers/> (дата обращения: 10.06.2020).
45. Example of using Business Intelligence Exchange (BIX) [Электронный ресурс] / iexrertify. Режим доступа: <http://pega.iexpertify.com/2014/09/example-of-using-business-intelligence-exchange-bix.html> (дата обращения: 10.06.2020).
46. Introduction to the SAS 9 Business Intelligence Platform: A Tutorial [Электронный ресурс] / SAS Global Forum. Режим доступа: <https://support.sas.com/resources/papers/proceedings/proceedings/forum2007/2007-2007.pdf> (дата обращения: 07.06.2020).
47. How to access/install Pega BIX [Электронный ресурс] / Pegasystems Inc. Режим доступа: <https://community1.pega.com/community/pega-support/question/how-accessinstall-pega-bix> (дата обращения: 10.06.2020).
48. Pega BIX extract to database [Электронный ресурс] / Pega Collaboration Center. Режим доступа: <https://collaborate.pega.com/question/pega-bix-extract-database> (дата обращения: 10.06.2020).

49. Personalize Your View of Products in the Analytics and Business Intelligence Platforms Market [Электронный ресурс] / Gartner. Режим доступа: <https://www.gartner.com/reviews/market/analytics-business-intelligence-platforms> (дата обращения: 07.06.2020).

50. Process Mining [Электронный ресурс] / PwC. Режим доступа: <https://www.pwc.ru/ru/services/technology/process-mining.html> (дата обращения: 07.06.2020).

51. Process Mining: Data Science in action [Электронный ресурс] / Coursera. Режим доступа: <https://www.coursera.org/learn/process-mining> (дата обращения: 07.06.2020).

52. Reporting VS BIX [Электронный ресурс] / Pega Collaboration Center. Режим доступа: <https://collaborate.pega.com/discussion/reporting-vs-bix> (дата обращения: 07.06.2020).

53. Review study: Business intelligence concepts and approaches [Электронный ресурс] / ResearchGate. Режим доступа: [https://www.researchgate.net/publication/303851229\\_Review\\_study\\_Business\\_intelligence\\_concepts\\_and\\_approaches](https://www.researchgate.net/publication/303851229_Review_study_Business_intelligence_concepts_and_approaches) (дата обращения: 07.06.2020).

54. Support Guide: Configuring BIX to Run from Command Line [Электронный ресурс] / Pega Community. Режим доступа: <https://community.pega.com/video-library/support-guide-configuring-bix-run-command-line> (дата обращения: 07.06.2020).

55. What is Business Intelligence? Definition & Example [Электронный ресурс] / Guru99. Режим доступа: <https://www.guru99.com/business-intelligence-definition-example.html> (дата обращения: 07.06.2020).

56. What is Process Engineering? [Электронный ресурс] / Quora. Режим доступа: <https://www.quora.com/What-is-process-engineering> (дата обращения: 08.06.2020).



57. What is Process Mining? Everything you need to know [Электронный ресурс] / ProcessGold. Режим доступа: <https://processgold.com/process-mining-everything-you-need-to-know/> (дата обращения: 07.06.2020).