



**ФЕДЕРАЛЬНОЕ АГЕНТСТВО МОРСКОГО И РЕЧНОГО ТРАНСПОРТА**

**Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Государственный университет морского и речного флота имени адмирала С.О. Макарова»  
(ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»)  
Воронежский филиал**

Кафедра математики, информационных систем и технологий

Направление подготовки 09.03.02 Информационные системы и технологии  
(код, наименование направления подготовки/специальности)

Форма обучения очная

«К ЗАЩИТЕ ДОПУЩЕН(А)»  
Заведующий кафедрой

(подпись)

Черняева С. Н.

(ФИО)

20

**Выпускная квалификационная работа**

Обучающегося Кравцовой Александры Игоревны  
(фамилия, имя, отчество)

Вид работы выпускная квалификационная работа бакалавра  
(выпускная квалификационная работа бакалавра, специалиста, магистра)

**Пояснительная записка**

Тема Разработка АИС обмена данными между системами оперативного учета в среде 1С:Предприятие (на примере РУСЭЛПРОМ-ЛЭЗ)

(полное название темы квалификационной работы, в соответствии с приказом об утверждении тематики ВКР)

Руководитель работы к.т.н, доцент Матьцина И. А.  
(должность, подпись, фамилия, инициалы, дата)

Консультант \_\_\_\_\_  
(при наличии) (должность, подпись, фамилия, инициалы, дата)

Консультант \_\_\_\_\_  
(должность, подпись, фамилия, инициалы, дата)

Обучающийся Кравцова А. И.  
(подпись, фамилия, инициалы, дата)

Воронеж  
2024

ФЕДЕРАЛЬНОЕ АГЕНТСТВО МОРСКОГО И РЕЧНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Государственный университет морского и речного флота имени адмирала С.О. Макарова»  
(ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»)  
Воронежский филиал

Кафедра математики, информационных систем и технологий

Направление подготовки 09.03.02 Информационные системы и технологии  
(код, наименование направления подготовки/специальности)

Форма обучения очная

УТВЕРЖДАЮ  
Заведующий кафедрой

(подпись)

Черняева С. Н.

(ФИО)

2024

**Задание  
на выпускную квалификационную работу**

Вид работы ВКР бакалавра  
(ВКР бакалавра, ВКР специалиста, ВКР магистра)

Обучающемуся Кравцовой Александре Игоревне  
(фамилия, имя, отчество)

Тема Разработка АИС обмена данными между системами оперативного учета в среде 1С:Предприятие

Утверждена приказом ректора университета от \_\_\_\_\_ 20\_\_\_\_, № \_\_\_\_\_

Срок сдачи законченной работы \_\_\_\_\_ 20\_\_\_\_

Исходные данные (или цель ВКР):

Целью дипломной работы является разработка автоматизированной информационной системы обмена данными между системами оперативного учета в среде 1С:Предприятие

Перечень подлежащих исследованию, разработке, проектированию вопросов (краткое содержание ВКР):

*(актуальность темы, цели и задачи ВКР; аналитический обзор литературных источников; постановка задачи исследования, разработки, проектирования; содержание процедуры исследования, разработки, проектирования; обсуждение результатов; дополнительные вопросы, подлежащие разработке; заключение – выводы по работе в целом, оценка степени решения поставленных задач, практические рекомендации; и др.*

– Введение. Актуальность выбранной темы, цель и задачи ВКР  
(наименование вопроса, раздела и его краткое содержание)

– Исследовательский раздел. Общая характеристика предприятия и выявление проблемы, Элементы информационного и технического обеспечения, функционирующие на

предприятия ООО «РУСЭЛПРОМ-ЛЭЗ», Постановка задачи и выделение основных требований к проектируемой системе. Обзор существующих программ, предназначенных для автоматизации деятельности на предприятии, Обмен данными в 1С:Предприятие.

(наименование вопроса, раздела и его краткое содержание)

– Проектная часть. Решение проблемы затрат времени при обмене данными, Как происходит обмен данными с помощью 1С:УПП и КИСПа

(наименование вопроса, раздела и его краткое содержание)

– Заключение. Выводы по работе в целом. Оценка степени решения поставленных задач

(наименование вопроса, раздела и его краткое содержание)

### Практические рекомендации

Перечень графического материала (или презентационного материала):

1. Титульный лист
2. Цель и задачи ВКР
3. Контекстная диаграмма ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как есть» и ее декомпозиция
4. Контекстная диаграмма ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как стало» и ее декомпозиция
5. Интерфейс пользователя
6. Интерфейс пользователя (продолжение)
7. Интерфейс пользователя (продолжение)
8. Интерфейс пользователя (продолжение)
9. Интерфейс пользователя (продолжение)
10. Интерфейс пользователя (продолжение)
11. Интерфейс пользователя (продолжение)
12. Результаты ВКР

Консультанты по разделам ВКР (при наличии):

1. \_\_\_\_\_  
(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)
2. \_\_\_\_\_  
(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)
3. \_\_\_\_\_  
(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)

Дата выдачи задания: \_\_\_\_\_ 20\_\_

Задание согласовано и принято к исполнению: \_\_\_\_\_ 20\_\_

Руководитель ВКР: к.т.н, Матыцина Ирина Александровна \_\_\_\_\_  
(должность, ученая степень, ученое звание, ФИО) (подпись)

Обучающийся: ИТ-4-1, Кравцова Александра Игоревна \_\_\_\_\_  
(учебная группа, ФИО) (подпись)

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	6
1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ .....	9
1.1 Характеристика предприятия. Организационная структура управления ООО «РУСЭЛПРОМ-ЛЭЗ» и ее характеристика.....	9
1.2 Элементы информационного и технического обеспечения, функционирующие на предприятии ООО «РУСЭЛПРОМ-ЛЭЗ».....	12
1.3 Постановка задачи и выделение основных требований к проектируемой системе .....	14
1.4 Обзор существующих программ, предназначенных для автоматизации деятельности на предприятии .....	17
1.4.1 «БухСофт» как решение для автоматизации предприятия .....	17
1.4.2 «СБИС бухгалтерия» как решение для автоматизации предприятия .....	18
1.4.3 «Мое дело» как решение для автоматизации предприятия .....	19
1.4.4 «Инфо-бухгалтер» как решение для автоматизации предприятия.....	19
1.4.5 Программы компании «Контур» как решение для автоматизации предприятия .....	21
1.4.6 «1С:Предприятие» как решение для автоматизации предприятия .....	22
1.5 Обмен данными в 1С:Предприятие.....	23
1.6 Использование брокера сообщений .....	25
1.6.1 Анализ популярных брокеров сообщений RabbitMQ и Apache Kafka ...	26
1.6.2 Архитектурные различия Kafka и RabbitMQ .....	27
1.6.3 Обработка сообщений в Kafka и RabbitMQ.....	29
1.6.4 Ключевые отличия Kafka от RabbitMQ.....	31
1.6.5 Сходства Kafka и RabbitMQ .....	32
1.6.6 Выбор между брокерами сообщений Kafka и RabbitMQ.....	34
1.7 Комплексная информационная система предприятия .....	34
1.8 Разбор XML документа, чтение XML документа, фабрика XDTO .....	36
2 ПРОЕКТНЫЙ РАЗДЕЛ .....	39

2.1 Решение проблемы затрат времени при обмене данными .....	39
2.2 Как происходит обмен данными с помощью 1С:УПП и КИСПа.....	41
ЗАКЛЮЧЕНИЕ .....	50
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	52
ПРИЛОЖЕНИЕ .....	55

## ВВЕДЕНИЕ

В настоящее время информационные технологии играют важную роль в управлении предприятием, и одной из важных аспектов является обмен данными между системами оперативного учета.

Системы автоматизации современных предприятий обычно состоят из отдельных баз данных и имеют географически разрозненную структуру. Эффективное функционирование таких систем невозможно без правильной реализации обмена данными.

Системы оперативного учета, построенные на платформе 1С, обеспечивают автоматизацию бизнес-процессов, учет основных операций, управление запасами и многие другие функции, их правильная интеграция и согласованность данных имеют решающее значение для эффективного управления предприятием.

В данной работе будет рассмотрена тема обмена данными между системами оперативного учета на платформе 1С. Она включает в себя анализ основных аспектов этого процесса, методы интеграции, технологии и инструменты, используемые для передачи информации между различными учетными системами. Также будет освещена роль обмена данными в оптимизации бизнес-процессов, улучшении качества управленческих решений и обеспечении точности и актуальности информации.

Работа посвящена изучению механизмов обмена данными между системами оперативного учета на платформе 1С, исследованию современных подходов к этому процессу, а также выявлению основных преимуществ и вызовов, стоящих перед организациями при реализации интеграции между информационными системами.

Разработка автоматизированной информационной системы позволит улучшить и ускорить обмен данными между 1С:УПП и КИСП.

Объектом исследования является ООО «РУСЭЛПРОМ-ЛЭЗ».

Предметом исследования – обмен данными между 1С:УПП и КИСП.

Целью дипломной работы является разработка автоматизированной информационной системы обмена данными между системами оперативного учета в среде 1С:Предприятие.

Для достижения поставленной цели, необходимо решить ряд следующих задач:

- проанализировать предприятие и выявить недостатки;
- изучить основные технологии и инструменты для реализации обмена данными;
- изучить готовые решения для поставленной цели;
- автоматизировать процесс обмена данными между системами оперативного учета.

Для реализации указанной выше цели и решения задач данной работы будут применены разнообразные методы исследования. Эти методы включают как теоретические, так и практические подходы, которые помогут получить всестороннее и объективное представление о предмете исследования.

Таким образом, завершение всех этапов проекта позволит создать приложение, соответствующее современным стандартам и требованиям. Разработанное приложение будет значимым инструментом для оптимизации обмена данными между системами оперативного учета в среде 1С:Предприятие.

Исследовательские методы включают в себя анализ научной и технической литературы, моделирование процессов, проектирование и проверку программного обеспечения. Для разработки автоматизированных систем применяются современные технологии и инструменты программирования [6].

Дипломная работа состоит из 3 глав.

В первой главе выпускной квалификационной работы рассматривается основная характеристика предприятия ООО «РУСЭЛПРОМ-ЛЭЗ», описаны характеристики рабочих мест рабочих и проанализирована организационная

структура предприятия. Также в первой главе были выявлены недостатки в существующем обмене данными.

Во второй главе была проанализирована теоретическая информация, которая в дальнейшем была использована на практике.

В третьей главе представлена проектная часть работы, в которой описывается реализованная АИС.



# 1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ

## 1.1 Характеристика предприятия. Организационная структура управления ООО «РУСЭЛПРОМ-ЛЭЗ» и ее характеристика

Предприятие имеет следующие характеристики и реквизиты:

- Полное название фирмы: ООО «РУСЭЛПРОМ-ЛЭЗ»;
- Адрес: 196641, г. Санкт-Петербург, пос. Металлострой, дор. на Металлострой, д. №5, корпус 45, литер АО, каб. 63;
- Контактные данные: тел – 8 (812) 462-88-29, эл. почта – info@lez.ru;
- ИНН: 7817067478;
- КПП: 781701001;
- ОГРН: 1177847078518;
- Дата создания: 28.02.2017.

Ленинградский электромашиностроительный завод начал работать с 1933 года. Нынешний облик предприятия начал формироваться в начале 2000-х годов после реорганизации АО «ЛЭЗ». В декабре 2003 года завод вошел в Концерн Русэлпром.

В настоящее время ООО «Русэлпром-Ленинградский электромашиностроительный завод» (ЛЭЗ) – крупнейший производитель синхронных и асинхронных электрических машин переменного тока мощностью до 20 МВт и более общепромышленного и специального исполнения, турбогенераторов мощностью до 40 МВт и гидрогенераторов мощностью до 250 МВт.

ЛЭЗ представляет собой оптимально выстроенное предприятие с налаженной инфраструктурой. Предприятие состоит из четырех основных производств:

- сварочно-заготовительное;
- механообрабатывающее;
- обмоточно-изоляционное;
- сборочно-выпускающее.

Сегодня в номенклатуру ЛЭЗ входят:

- асинхронные электродвигатели с короткозамкнутым ротором мощностью до 20 МВт;
- асинхронные электродвигатели с фазным ротором мощностью до 7 МВт;
- синхронные электродвигатели мощностью до 12 МВт;
- гидрогенераторов мощностью от 400 кВт до 250 МВт;
- турбогенераторы с воздушным охлаждением мощностью до 40 МВт;
- дизель-генераторы от 1 до 9 МВт;
- гребные электродвигатели до 30 МВт.

Схема организационной структуры ООО «Русэлпром-Ленинградский электромашиностроительный завод» представлена на рисунке 1.1.

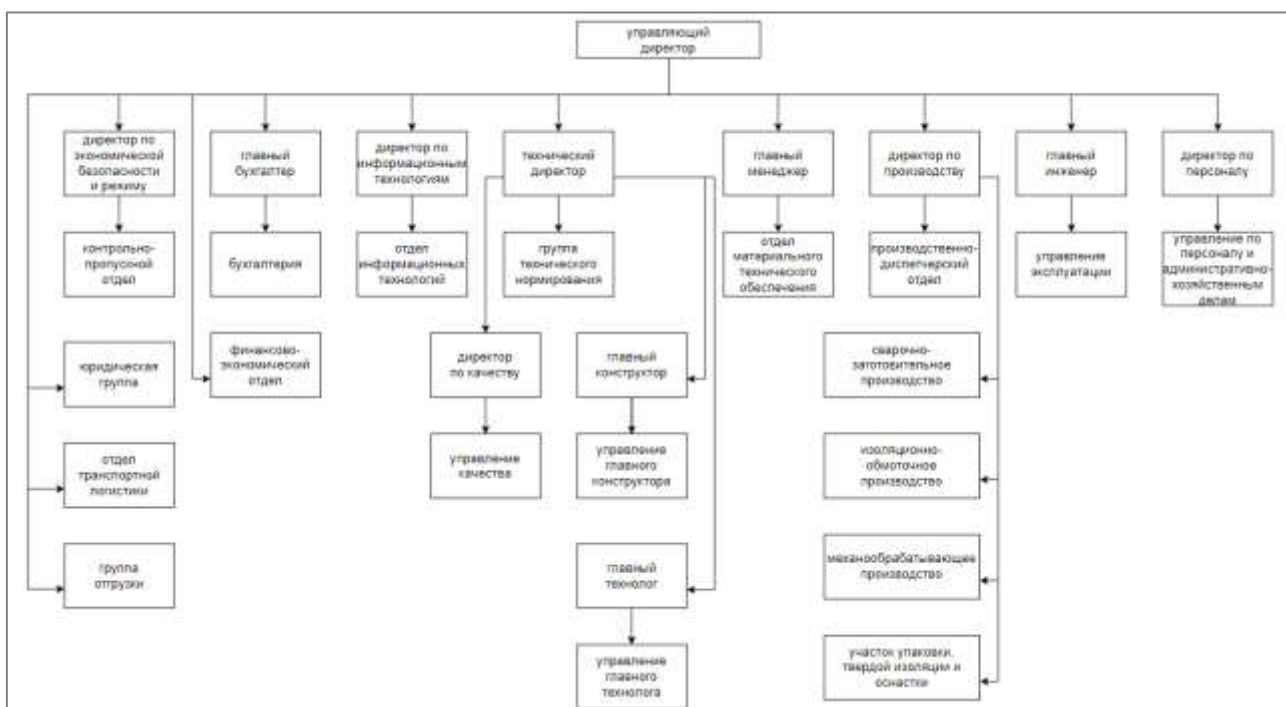


Рисунок 1.1 – Организационная структура управления предприятия ООО «РУСЭЛПРОМ-ЛЭЗ»

Организационная структура управления предприятия ООО «РУСЭЛПРОМ-ЛЭЗ» является линейно-функциональной, так как в ней содержатся как линейные, так и функциональные подразделения [16].

В рамках линейного подразделения осуществляется постановка задач руководством и выбор исполнителей. Функциональное подразделение выполняет поставленные задачи руководства (технического директора и директора по производству).

На рисунке 1.2 представлена схема бизнес-процессов оперативного учета материалов в концерне Русэлпром.

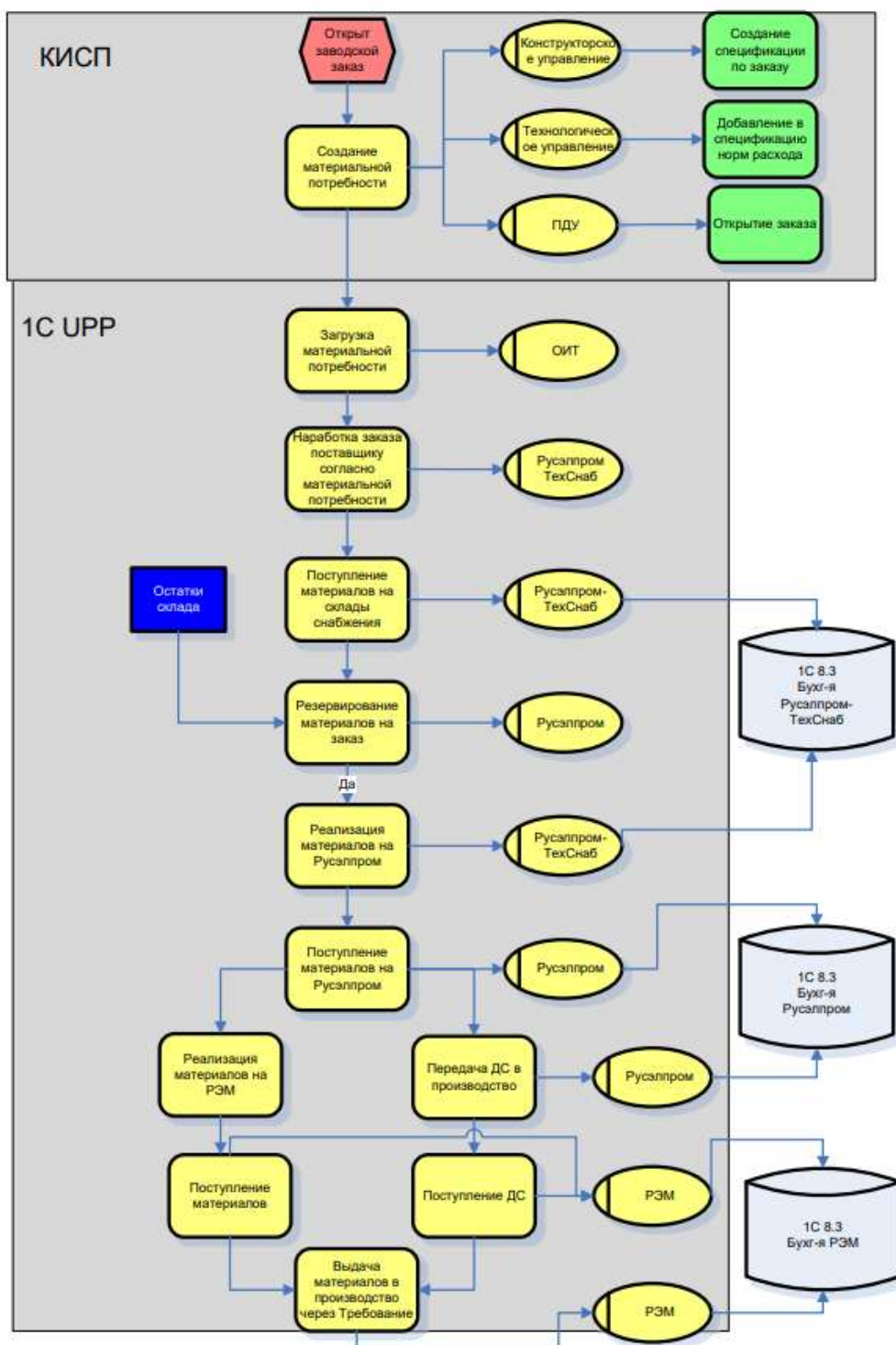


Рисунок 1.2 – Бизнес-процессы оперативного учета материалов

Бизнес-процесс – это повторяющаяся последовательность действий, которые выполняются на предприятии для достижения ценного конечного продукта [19].

Бизнес-процессы имеют 4 характеристики, которые описывают переход ресурсов в продукт:

- стоимость – количество затраченных ресурсов для получения результата;
- длительность – время превращение ресурсов в результат;
- эффективность – соотношение затрат к точности результата;
- качество – степень соответствия результата ожиданиям клиента.

## 1.2 Элементы информационного и технического обеспечения, функционирующие на предприятии ООО «РУСЭЛПРОМ-ЛЭЗ»

Тема диплома заключается в разработке АИС данными между системами оперативного учета в среде 1С:Предприятие. Разработанной автоматизированной системой будут пользоваться две стороны, поэтому программное и аппаратное обеспечение будет рассмотрено на двух примерах:

1. Рабочее место кладовщика;
2. Рабочее место бухгалтера.

Компоненты рабочей системы кладовщика в ООО «РУСЭЛПРОМ-ЛЭЗ»:

- операционная система Windows 10;
- процессор Intel Celeron G4930 CPU @ 3.20GHz;
- ОЗУ (оперативная память) DEXP [DEXP8GD3UD16] 8 Гб, тактовая частота 1600 МГц;
- видеокарта интегрированная;
- МФУ Xerox WorkCentre 3335;
- монитор Samsung SR35 S24R358FZI.

Программные продукты, которые установлены на рабочей системе кладовщика в ООО «РУСЭЛПРОМ-ЛЭЗ»:

- 1С:Предприятие;
- антивирус Kaspersky;
- браузер Firefox/Chrome;
- Microsoft\_Office/LibreOffice;
- почтовый клиент Thunderbird/Outlook;
- pdf просмотр: Adobe\_Acrobat.

Компоненты рабочей системы бухгалтера в ООО «РУСЭЛПРОМ-ЛЭЗ»:

- операционная система Windows 10;
- процессор Intel Celeron G4930 CPU @ 3.20GHz;
- ОЗУ (оперативная память) DEXP [DEXP8GD3UD16] 8 Гб, тактовая частота 1600 МГц;
- видеокарта интегрированная;
- МФУ Kyocera 2235;
- монитор Acer KA240Ybi.

Программные продукты, которые установлены на рабочей системе бухгалтера в ООО «РУСЭЛПРОМ-ЛЭЗ»:

- 1С:Предприятие;
- антивирус Kaspersky;
- браузер Firefox/Chrome;
- Microsoft\_Office/LibreOffice;
- почтовый клиент Thunderbird/Outlook;
- pdf просмотр: Adobe\_Acrobat.

Целью бухгалтера является регистрация документов бухгалтерского учета. Цель кладовщика заключается в выдаче материалов в производство под ответственность материально ответственного лица.

С помощью обозначенных выше лиц производится выдача материала в производство на предприятии ООО «Русэлпром-Ленинградский электромашиностроительный завод».

### 1.3 Постановка задачи и выделение основных требований к проектируемой системе

Согласно индивидуальному заданию, требуется разработать АИС обмена данными между системами оперативного учета в среде 1С:Предприятие.

В настоящий момент на предприятии было произведено внедрение современной информационной платформы КИСП, так как в прошлом, для решения задач поддержки процессов управления производством, на предприятиях Русэлпром функционировало большое количество разнородных информационных систем, которые фактически не поддерживались или поддерживались на минимальном уровне.

Большая часть ИС была построена на сильно устаревших информационных платформах, поддержка этих систем осуществлялась командами из 2-3 человек. Это приводило к распылению ресурсов, создавало зависимость компании от конкретных сотрудников, а также повышало риски полной безвозвратной остановки функционирования систем.

После внедрения в КИСПе был разработан базовый функционал, который позволил поддержать базовые процессы, в которых задействованы головные предприятия Концерна. Помимо этого, внедрение позволило выстраивать единообразные контрольные процессы на всех производственных предприятиях Концерна.

Такой подход позволил значительно сократить объем «вредного информационного обмена», который реализовывался как в виде многочисленных выгрузок, так и в виде двойного ручного ввода.

В настоящее время бухгалтерский и управленческий по учет движению материалов для производства велся в 1С:УПП. Кладовщики вручную вводили документы по ТМЦ (товарно-материальные ценности) в 1С, после происходил обмен данными между 1С и КИСП.

Для выдачи материалов требовалось три документа (реализация, приход и требование-накладная), это занимало длительное количество времени. Чтобы получить требование накладную, нужно было провести два документа (реализация и приход). Физический материал можно было получить только после полного оформления документа.

Полное оформление заключалось в нескольких этапах:

- 1 создание документа реализация;
- 2 создание документа приход;
- 3 создание и распечатка документа требование накладная;
- 4 сбор подписей и печатей материально ответственных лиц (МОЛ).

Первые три этапа производятся в бухгалтерии. После завершения третьего пункта в бухгалтерию приходит МОЦ, которое получает на руки требование накладную, а затем идет к каждому МОЦу лично, чтобы собрать подписи и печати.

Только после завершения всех пунктов, материально ответственное лицо, собирающее подписи, может получить материал со склада.

Так как существующий метод требовал много времени, было принято решение о введении в первичную систему КИСП информацию движения материала, это обусловило уменьшение времени по оприходованию материалов на склад и списыванию материалов в производство.

Контекстная диаграмма ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как есть» показана на рисунке 1.3.

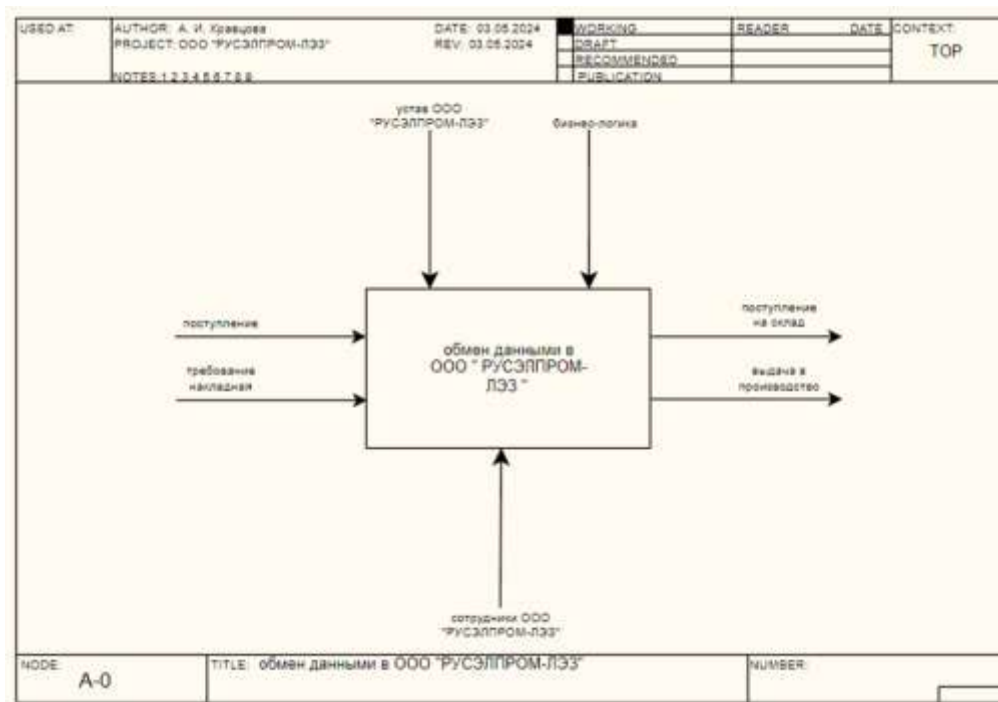


Рисунок 1.3 – Функциональная модель

Входными данными здесь являются документ «Требование-накладная» и документ «Поступление». Выходными данными – документ «Поступление на склад» и документ «Выдача в производство».

Декомпозиция контекстной диаграммы ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как есть» показана на рисунке 1.4.

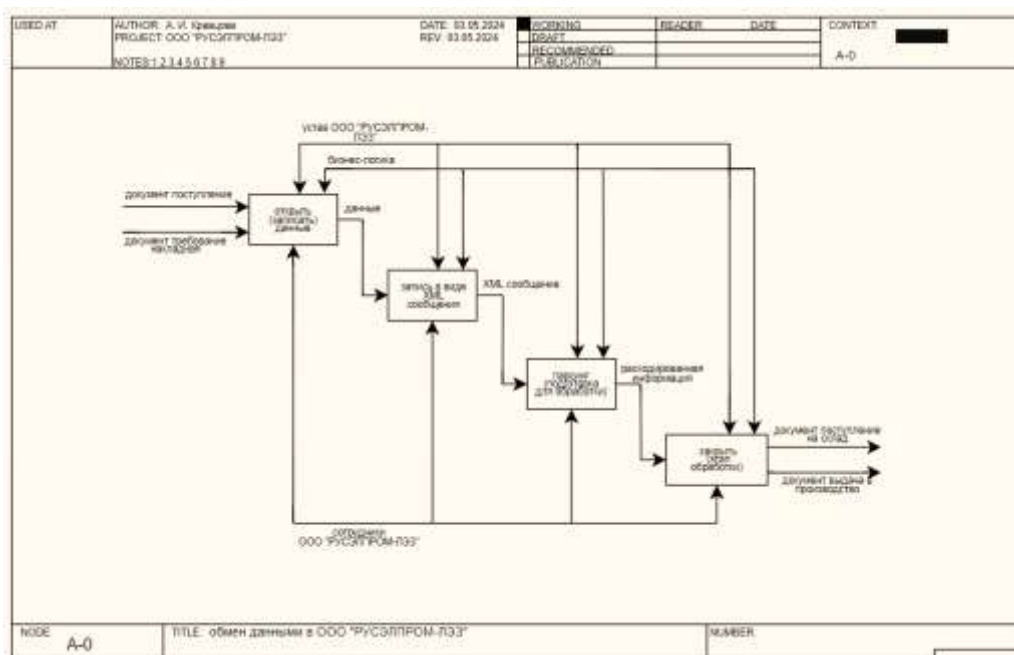


Рисунок 1.4 – Декомпозиция контекстной диаграммы



Раньше требовалось создать сразу несколько документов в 1С, чтобы произвести обмен данными. Приемником данных являлась база данных в среде КИСП (комплексная информационная система), источником данных – 1С:УПП. Бухгалтер создавал все документы вручную, эти действия способствовали частым ошибкам в документации, а также, занимали большую часть времени.

#### 1.4 Обзор существующих программ, предназначенных для автоматизации деятельности на предприятии

На рынке представлено несколько решений для задачи по автоматизации деятельности на предприятии. Данные программы необходимы для того, чтобы существовала возможность интегрировать различные конфигурации, которые способны охватить все направления и сферы деятельности компании, а также обеспечить достижение необходимых результатов в работе.

Автоматизация деятельности предприятия играет важную роль в наше время, так как приносит значительные выгоды:

- улучшение производительности;
- улучшение контроля;
- улучшение качества;
- улучшение адаптивности бизнеса [7].

Ниже в дипломной работе будут рассмотрены существующие решения для автоматизации предприятий.

##### 1.4.1 «БухСофт» как решение для автоматизации предприятия

Продукт предназначен для автоматизированного ведения бухгалтерского, налогового, управленческого и складского учета, а также автоматизации работы кадровых служб. Одним из значительных

преимуществ ПО является возможность удаленного подключения с любого рабочего места, так как это онлайн-сервис [3].

Продукт является «условно бесплатным»: пользователям предоставляется бесплатный тестовый период на три месяца, после чего за использование сервиса необходимо платить.

Программа достаточно универсальна, не требует длительной и тщательной настройки, а обновления происходят автоматически. Среди явных плюсов БухСофта можно отметить обширный функционал программы и круглосуточную поддержку.

Основным недостатком ПО БухСофт является невозможность доработки под специфические нужды предприятия. Программу можно использовать только в компаниях со стандартными учетными процессами, а внесение специфических изменений в функционал может осуществлять только сам разработчик.

#### 1.4.2 «СБИС бухгалтерия» как решение для автоматизации предприятия

Востребованным решением на рынке программ для ведения учета является комплекс программ СБИС. Изначально это был сервис для документооборота, но со временем разработчики расширили функционал до полноценного ПО. СБИС предлагает комплексные решения для ведения учета и составления отчетности на предприятии [15].

Это удобный онлайн-сервис, который подходит как для индивидуальных предпринимателей, так и для крупных предприятий. В программе предусмотрены функции ведения учета, составления отчетности, начисления зарплаты и взаимодействия с контрагентами, включая ЭДО и интеграцию с электронной почтой и банковскими системами.

СБИС является одним из немногих IT-продуктов, адаптированных для обмена информацией с 1С и торговым оборудованием.

Также в программе предусмотрены понятные и наглядные отчеты для руководителя и система контроля правильности начисления налогов.

К недостаткам продукта можно отнести сложный интерфейс и значительные трудозатраты на настройку программы для решения специфических задач. Программа предлагает только стандартный план счетов и типовые проводки, поэтому для отдельных задач (нетиповые операции, корреспонденция счетов, справочники по подразделениям и договорам, порядок расчета оплаты труда с доплатами и надбавками) потребуется ручная настройка.

Среди плюсов можно отметить бесплатное обслуживание и возможность распознавания сканов и фотодокументов.

#### 1.4.3 «Мое дело» как решение для автоматизации предприятия

Ещё одно отличное программное обеспечение для небольших предприятий, ведущих бухгалтерию по стандартной схеме, – это платный онлайн-сервис, который позволяет оформлять первичную документацию, вести учет, и формировать отчетность с последующей отправкой в ИФНС и фонды, аналогично 1С [10].

Благодаря простому интерфейсу систему можно быстро освоить и полноценно использовать, имея доступ в интернет с любого рабочего места.

Однако, как и большинство упрощенных онлайн-сервисов, «Мое дело» не подходит для компаний, занимающихся специфической деятельностью, так как в программе отсутствует возможность настройки под нужды конкретного предприятия.

#### 1.4.4 «Инфо-бухгалтер» как решение для автоматизации предприятия

Линейка программ компании "Инфо-бухгалтер" достаточно широка и охватывает автоматизацию самых распространенных направлений

бухгалтерского учета. Программное обеспечение существует на рынке с 1990 года, что свидетельствует о его востребованности [2].

ПО предназначено для ведения бухгалтерского и налогового учета в компаниях различных сфер бизнеса и систем налогообложения.

На основе первичных документов программа автоматически формирует регистры бухгалтерского учета (журнал хозяйственных операций, журналы-ордера, главную книгу) и налоговые реестры.

Компания предлагает обширную линейку продуктов: помимо стандартного ПО для ведения бухгалтерского, финансового и управленческого учета предприятий на УСН и ОСН, есть отраслевые решения для ТСЖ, гаражных кооперативов, адвокатов и малых предприятий.

Основная аудитория — предприятия малого и среднего бизнеса.

Программные продукты "Инфо-бухгалтер" интегрируются с торговым оборудованием [1].

Также доступны демоверсии различных отраслевых решений и возможность работы онлайн в облаке.

Существенным недостатком разработок компании является отсутствие встроенного блока ЭДО (электронного документооборота). Для сдачи отчетности в электронном виде или обмена документами с контрагентами через интернет пользователям потребуется приобретать дополнительное ПО, такое как "Астрал", что ведет к дополнительным расходам и трудозатратам.

Среди плюсов следует отметить легкость освоения, доступность поддержки и разветвленную дилерскую сеть. Компания предоставляет круглосуточную поддержку пользователей и поддерживает достаточно лояльные цены.

#### 1.4.5 Программы компании «Контур» как решение для автоматизации предприятия

На сегодняшний день на рынке представлены следующие продукты от компании «Контур»:

1. Контур Эльба: Интернет-сервис для индивидуальных предпринимателей и малого бизнеса.

2. Контур Бухгалтерия: Полноценное программное обеспечение для ведения бухгалтерского учета и составления отчетности, подходящее как для средних, так и для крупных предприятий.

3. Контур Экстерн: Программное обеспечение для предприятий с филиалами.

Одним из преимуществ этого ПО является возможность одновременной работы руководителя и финансовой службы, а также интеграция с большинством распространенных бухгалтерских программ.

Продукты компании "Контур" предлагают полный спектр функций для автоматизации бухгалтерского, налогового и управленческого учета. Программы имеют удобный и интуитивно понятный интерфейс, регулярно обновляются в соответствии с изменениями законодательства и адаптированы для обмена данными с интернет-банками и контролирующими органами [11].

Программное обеспечение "Контур" поддерживает обмен данными с 1С, что позволяет импортировать и экспортировать массивы данных между программами.

Однако стоит отметить, что некоторые бухгалтеры жалуются на невозможность получить полноценную сверку по договорам с контрагентами, а также на то, что не все статьи доходов и расходов учитываются. В "Контуре" может быть сложно вести учет в компаниях с большими оборотами и многочисленным персоналом.

При использовании ПО исключительно для ведения бухгалтерии пользователи обычно не испытывают проблем и нареканий.

#### 1.4.6 «1С:Предприятие» как решение для автоматизации предприятия

Программные решения 1С представляют собой комплекс продуктов, которые позволяют автоматизировать не только бухгалтерский учет, но и практически все бизнес-процессы, от мелких предпринимателей до крупных корпораций. Линейка IT-решений 1С охватывает как бюджетные учреждения, так и коммерческие компании практически любой отрасли экономики [18].

В первую очередь, следует отметить универсальность программ 1С. Они подходят для ведения учета при различных системах налогообложения и формах собственности. Типовыми считаются только те решения, которые не подвергались доработке для специфических нужд предприятия. Продукты, в которых каким-либо образом изменялась конфигурация, считаются нетиповыми.

Основой всех программ является платформа «1С: Предприятие», на которую устанавливаются различные версии продуктов, в зависимости от потребностей пользователя [8].

В линейке программного обеспечения 1С представлены различные конфигурации, каждая из которых предназначена для определенного уровня функциональности и объема задач:

Базовая конфигурация обладает минимальным набором функций и предназначена для работы одного пользователя. Она не поддерживает сетевой режим и не предусматривает доработки.

Профессиональная конфигурация (ПРОФ) расширяет возможности и позволяет вести учет в нескольких организациях, а также вносить изменения в программу.

Корпоративная конфигурация (КОРП) обладает самым широким функционалом и предназначена для автоматизации всех бизнес-процессов крупных предприятий.

В зависимости от масштаба и направления деятельности компании, выбираются соответствующие версии программы. Например, для малого и

среднего бизнеса чаще всего используются 1С Бухгалтерия, 1С Зарплата и управление персоналом, 1С Управление торговлей и другие.

Крупные компании предпочитают использовать комплексные решения, такие как «1С: Комплексная автоматизация», «1С: ERP» и «1С: Управление холдингом».

Разработчики 1С также учитывают специфику различных отраслей экономики, предлагая более 50 специализированных версий программы для учета в здравоохранении, добывающих отраслях, ЖКХ, сельском хозяйстве, строительстве и других сферах. Это позволяет избежать необходимости дополнительной настройки программы и упрощает ее внедрение.

Продукты 1С занимают лидирующее положение на рынке автоматизации бухгалтерского и управленческого учета благодаря своей универсальности и высокой адаптивности к онлайн-банкингу и программному обеспечению контролирующих органов.

В итоге, выбор остается за пользователем, но благодаря оптимальному соотношению «цена-качество», программы 1С удерживают лидирующие позиции на рынке программного обеспечения. Разработчики постоянно улучшают программное обеспечение, стремясь сделать его максимально удобным для клиентов.

Проанализировав рынок, было принято решение об использовании программы «1С:Предприятие» в написании практической части дипломной работы.

### 1.5 Обмен данными в 1С:Предприятие

1С:Предприятие является программным продуктом компании «1С», который предназначен для автоматизации деятельности на предприятии. Он используется для автоматизации управленческого и бухгалтерского учёта, управления кадрами, начисления зарплаты и экономической деятельности предприятия.

Существуют стандартные и индивидуальные конфигурации 1С:Предприятие. К стандартным относятся, например, «1С:Бухгалтерия», «1С:Зарплата и управление персоналом» и «1С:Управление производственным предприятием».

Индивидуальные конфигурации разрабатываются предприятиями самостоятельно для решения своих специфических задач.

Для реализации поставленной задачи будет использоваться конфигурация 1С:Управление производственным предприятием. Это комплексная информационная система, предназначенная для автоматизации управления и производственных процессов в различных компаниях.

1С:УПП обладает широкими функциональными возможностями и простотой использования, что позволяет эффективно управлять производством, складом, финансами, кадрами и другими аспектами деятельности компании [4].

Система дает возможность:

- оптимизировать бизнес-процессы;
- увеличить производственную эффективность;
- управлять запасами на складах;
- контролировать финансовые показатели предприятия;
- контролировать экономические показатели предприятия;
- и многое другое.

Характеристики 1С:УПП:

- гибкость и возможность настройки: платформа позволяет быстро настраивать систему в соответствии с потребностями конкретного предприятия и отрасли;
- интеграция с другими системами: 1С:УПП легко интегрируется с другими информационными системами для обмена данными и совместной работы.



— модульная структура: система состоит из различных функциональных модулей, что позволяет выбирать и активировать только необходимые компоненты.

— безопасность и защита данных: 1С:УПП обеспечивает высокий уровень безопасности и защиту конфиденциальных данных предприятия.

— технологии и архитектура системы: 1С:УПП использует современные технологии, включая клиент-серверную архитектуру, веб-технологии и облачные решения, обеспечивая удобство использования и доступность.

Обмен данными в 1С:Предприятие – это процесс передачи информации между различными информационными системами и базами данных на основе программных продуктов 1С.

Существует множество вариантов обмена данными в 1С:Предприятие:

— внутри единой распределённой базы или между независимыми конфигурациями;

— с помощью различных каналов обмена;

— в автоматическом или ручном режиме по определенному расписанию;

— с ограниченным набором синхронизируемых данных или передачей всей информации;

— с использованием протоколов и форматов обмена данными;

— и многое другое.

В дипломной работе будет рассмотрен обмен данными между 1С:УПП и КИСП. Обмен данными будет осуществляться с помощью брокера сообщений RabbitMQ.

## 1.6 Использование брокера сообщений

Брокер сообщений – это посредник в распределенной системе, который отвечает за прием, хранение и доставку сообщений между различными

компонентами или системами. Основная задача брокера сообщений заключается в управлении отправкой сообщений от отправителя к получателю, обеспечивая надежное и безопасное обмен сообщениями в асинхронном режиме [17].

Брокер сообщений работает по принципу publisher-subscriber (издатель-подписчик): отправители сообщений (издатели) публикуют сообщения в брокере, а получатели сообщений (подписчики) подписываются на определенные каналы или очереди, чтобы получать и обрабатывать сообщения.

Примеры популярных брокеров сообщений включают RabbitMQ, Apache Kafka, ActiveMQ, Amazon SQS и другие. Брокеры сообщений широко используются в распределенных системах и микросервисной архитектуре для обмена данными между различными компонентами и приложениями, обеспечивая масштабируемость, надежность и изоляцию компонентов [14].

#### 1.6.1 Анализ популярных брокеров сообщений RabbitMQ и Apache Kafka

Kafka и RabbitMQ – это системы очередей сообщений, которые можно использовать при потоковой обработке. Поток данных – это непрерывные инкрементные данные большого объема, требующие высокоскоростной обработки.

Например, это могут быть данные датчиков об окружающей среде, которые необходимо постоянно собирать и обрабатывать для наблюдения за изменениями температуры или давления воздуха в реальном времени. RabbitMQ – это распределенный брокер сообщений с открытым исходным кодом, который обеспечивает эффективную доставку сообщений в сложных сценариях маршрутизации. Он называется «распределенным», потому что RabbitMQ обычно работает как кластер узлов, где очереди распределены по узлам – реплицируются для обеспечения высокой доступности и отказоустойчивости.

RabbitMQ использует модель push и предотвращает перегрузку пользователей с помощью ограничения предварительной выборки, настроенного потребителем. Эта модель является идеальным подходом для обмена сообщениями с малой задержкой. Он также хорошо работает с архитектурой на основе очередей RabbitMQ. Думайте о RabbitMQ как о почтовом отделении, которое получает, хранит и доставляет почту, тогда как RabbitMQ принимает, хранит и передает сообщения двоичных данных.

Apache Kafka – это распределенная платформа потоковой передачи событий с открытым исходным кодом, обеспечивающая высокую пропускную способность. Kafka, написанная на Java и Scala, представляет собой шину сообщений pub/sub, ориентированную на потоки и воспроизведение данных с большим объемом входящего трафика. Вместо того, чтобы полагаться на очередь сообщений, Kafka добавляет сообщения в журнал и оставляет их там, где они остаются до тех пор, пока потребитель не прочитает их или не достигнет предела хранения.

Kafka использует подход «на основе извлечения», позволяя пользователям запрашивать пакеты сообщений с определенных смещений. Пользователи могут использовать пакетную обработку сообщений для повышения пропускной способности и эффективной доставки сообщений.

Хотя Kafka поставляется только с клиентом Java, он предлагает адаптер SDK, позволяющий программистам создавать свою уникальную системную интеграцию. Существует также растущий каталог проектов экосистемы сообщества и клиентов с открытым исходным кодом.

Kafka был выпущен в 2011 году, так что это новичок.

### 1.6.2 Архитектурные различия Kafka и RabbitMQ

RabbitMQ и Apache Kafka позволяют производителям отправлять сообщения потребителям. Производители – это приложения, публикующие

информацию, а потребители – приложения, которые подписываются на информацию и обрабатывают ее.

В RabbitMQ и Kafka производители и потребители взаимодействуют по-разному. В RabbitMQ производитель отправляет сообщение и отслеживает, дошло ли оно до предполагаемого потребителя. С другой стороны, производители Kafka публикуют сообщения в очереди независимо от того, получили ли их потребители.

RabbitMQ можно рассматривать как почтовое отделение, которое принимает почту и доставляет ее предполагаемым получателям. Между тем Kafka похож на библиотеку, где на полках раскладываются сообщения разных жанров, публикуемые производителями. Затем потребители читают сообщения с соответствующих полок и запоминают прочитанное.

#### Архитектурный подход RabbitMQ

Брокер RabbitMQ обеспечивает низкую задержку и сложное распределение сообщений с помощью следующих компонентов:

1. Биржа получает сообщения от производителя и определяет, куда их следует направлять.
2. Очередь – это хранилище, которое получает сообщения от биржи и отправляет их потребителям.
3. Привязка – это путь, соединяющий биржу и брокера.

В RabbitMQ ключ маршрутизации – это атрибут сообщения, который используется для маршрутизации сообщений с биржи в определенную очередь. Когда производитель отправляет сообщение на биржу, он включает в него ключ маршрутизации. Затем биржа использует этот ключ маршрутизации, чтобы определить, в какую очередь следует доставить сообщение [13].

#### Архитектурный подход Kafka

Кластер Kafka обеспечивает обработку потоковых событий с более сложной архитектурой с высокой пропускной способностью.

Ниже приведены несколько ключевых компонентов Kafka:

1. Брокер Kafka – это сервер Kafka, который позволяет производителям передавать потоки данных потребителям. Брокер Kafka содержит темы и соответствующие разделы.

2. Тема – носитель данных, в котором сгруппированы похожие данные в брокере Kafka.

3. Раздел – это небольшой носитель данных в теме, на которую подписаны потребители.

4. ZooKeeper – это специальное программное обеспечение, которое управляет кластерами и разделами Kafka для обеспечения отказоустойчивой потоковой передачи. ZooKeeper недавно было заменено протоколом Apache Kafka Raft (KRaft) [20].

Производители в Kafka назначают каждому сообщению ключ сообщения. Затем брокер Kafka сохраняет сообщение в ведущем разделе этой конкретной темы. Протокол KRaft использует алгоритмы консенсуса для определения ведущего раздела.

### 1.6.3 Обработка сообщений в Kafka и RabbitMQ

RabbitMQ и Apache Kafka по-разному передают данные от производителей к потребителям. RabbitMQ – это брокер сообщений общего назначения, который отдает приоритет сквозной доставке сообщений. Kafka – это распределенная платформа для потоковой трансляции событий, поддерживающая непрерывный обмен большими данными в реальном времени.

RabbitMQ и Kafka предназначены для разных примеров использования, поэтому они по-разному обрабатывают обмен сообщениями:

1. Потребление сообщений:

В RabbitMQ брокер гарантирует получение сообщения потребителями. Потребительское приложение играет пассивную роль и ждет, пока брокер RabbitMQ отправит сообщение в очередь. Например, банковское приложение

может ждать SMS-предупреждение от программного обеспечения для централизованной обработки транзакций.

Однако потребители Kafka более активно читают и отслеживают информацию. По мере добавления сообщений в физические файлы журналов пользователи Kafka отслеживают последнее прочитанное сообщение и соответствующим образом обновляют свой трекер смещений. Трекер смещений – это счетчик, показатель которого увеличивается после прочтения сообщения. В случае с Kafka производитель не знает о получении сообщений потребителями.

## 2. Приоритет сообщения:

Брокеры RabbitMQ позволяют программному обеспечению производителя увеличивать количество определенных сообщений, используя очередь приоритетов. Вместо отправки сообщений в порядке «первым пришел — первым ушел» брокер обрабатывает сообщения с более высоким приоритетом, чем обычные сообщения. Например, приложение для розничной торговли может ставить транзакции в очередь каждый час.

Однако если системный администратор отправляет приоритетное сообщение резервного копирования базы данных, то брокер немедленно отправляет его.

В отличие от RabbitMQ, Apache Kafka не поддерживает приоритетные очереди. При распределении по соответствующим разделам все сообщения рассматриваются как равные.

## 3. Порядок сообщений:

RabbitMQ отправляет сообщения и ставит их в очередь в определенном порядке. Если сообщение с более высоким приоритетом не поставлено в очередь в систему, потребители получают сообщения в том порядке, в котором они были отправлены.

Kafka, в свою очередь, использует темы и разделы для постановки сообщений в очередь. Когда производитель отправляет сообщение, оно переходит в определенную тему и раздел. Поскольку Kafka не поддерживает

прямые обмены между производителями и потребителями, потребитель получает сообщения из раздела в другом порядке.

#### 4. Удаление сообщения:

Брокер RabbitMQ маршрутизирует сообщение в очередь назначения. После прочтения потребитель отправляет брокеру ответ с подтверждением (ACK), который затем удаляет сообщение из очереди.

В отличие от RabbitMQ, Apache Kafka добавляет сообщение в файл журнала, который сохраняется до истечения срока хранения. Таким образом, потребители могут повторно обрабатывать потоковые данные в любое время в течение указанного периода.

### 1.6.4 Ключевые отличия Kafka от RabbitMQ

RabbitMQ обеспечивает сложную маршрутизацию сообщений с простой архитектурой, а Kafka предлагает надежную систему брокера сообщений, которая позволяет приложениям обрабатывать данные в истории потоков.

Ниже будут описаны ключевые отличия Kafka от RabbitMQ:

#### 1. Производительность:

И RabbitMQ, и Kafka предлагают высокопроизводительную передачу сообщений для предполагаемых примеров использования.

Kafka может отправлять миллионы сообщений в секунду, поскольку использует последовательный дисковый ввод-вывод для обеспечения высокой пропускной способности обмена сообщениями.

Последовательный дисковый ввод-вывод – это система хранения данных, которая хранит данные из смежного пространства памяти и получает к ним доступ быстрее, чем произвольный доступ к диску.

RabbitMQ также может отправлять миллионы сообщений в секунду, но для этого требуется несколько брокеров. Обычно производительность

RabbitMQ составляет в среднем тысячи сообщений в секунду и может замедлиться, если очереди RabbitMQ перегружены.

## 2. Безопасность:

RabbitMQ и Kafka позволяют приложениям безопасно обмениваться сообщениями, но с использованием разных технологий.

RabbitMQ поставляется с инструментами администрирования для управления разрешениями пользователей и безопасностью брокеров.

Между тем архитектура Apache Kafka обеспечивает безопасные потоки событий с помощью TLS и службы аутентификации и авторизации Java (JAAS). TLS – это технология шифрования, предотвращающая непреднамеренное прослушивание сообщений, а JAAS контролирует, какое приложение имеет доступ к брокерской системе.

## 3. Язык программирования и протоколы:

И Kafka, и RabbitMQ поддерживают различные языки, платформы и протоколы, знакомые разработчикам.

Возможно писать код на Java и Ruby при создании клиентских приложений для Kafka и RabbitMQ. Кроме того, Kafka поддерживает Python и Node.js, а RabbitMQ поддерживает JavaScript, Go, C, Swift, Spring, Elixir, PHP и .NET.

Kafka использует протокол передачи двоичных данных через TCP для потоковой передачи сообщений по конвейерам данных в реальном времени, а RabbitMQ по умолчанию поддерживает протокол расширенной очереди сообщений (AMQP).

### 1.6.5 Сходства Kafka и RabbitMQ

Для обмена данными в облаке приложениям нужны надежные брокеры сообщений. И RabbitMQ, и Kafka предоставляют масштабируемые и отказоустойчивые платформы для удовлетворения растущих требований к трафику и высокой доступности.



Ниже приведены некоторые ключевые сходства между RabbitMQ и Kafka:

### 1. Масштабируемость:

RabbitMQ может расширить возможности обработки сообщений как по горизонтали, так и по вертикали. Серверу RabbitMQ можно выделить больше вычислительных ресурсов, чтобы повысить эффективность обмена сообщениями. В некоторых случаях разработчики используют метод распределения сообщений под названием консистентный хеш-обмен RabbitMQ, чтобы сбалансировать обработку нагрузки несколькими брокерами.

Кроме того, архитектура Kafka позволяет добавлять больше разделов к определенной теме для равномерного распределения нагрузки сообщений.

### 2. Отказоустойчивость:

И Kafka, и RabbitMQ представляют собой надежные архитектуры очередей сообщений, устойчивые к сбоям системы.

Можно сгруппировать несколько брокеров RabbitMQ в кластеры и развернуть их на разных серверах. RabbitMQ также реплицирует сообщения в очереди между распределенными узлами. Это позволяет системе восстанавливаться после сбоя, затрагивающего любой сервер.

Как и RabbitMQ, Apache Kafka обладает аналогичными возможностями восстановления и резервирования благодаря хостингу кластеров Kafka на разных серверах. Каждый кластер состоит из реплик файлов журналов, которые можно восстановить в случае сбоя.

### 3. Простота использования:

Обе системы очередей сообщений имеют мощную поддержку сообщества и библиотеки, упрощающие отправку, чтение и обработку сообщений. Это упрощает разработку клиентских приложений для разработчиков обеих систем.

Например, можно использовать Kafka Streams (клиентскую библиотеку) для создания систем обмена сообщениями на Kafka, а Spring Cloud Data Flow

– для создания управляемых событиями микросервисов с помощью RabbitMQ.

### 1.6.6 Выбор между брокерами сообщений Kafka и RabbitMQ

Важно понимать, что RabbitMQ и Kafka не являются конкурирующими брокерами сообщений. Обе версии были разработаны для поддержки обмена данными в разных примерах использования, где одна из них подходит больше, чем другая.

Ниже будут обозначены аспекты, из-за которых был сделан выбор в сторону RabbitMQ:

1. RabbitMQ обеспечивает гибкость для клиентов с приблизительными требованиями или сложными сценариями маршрутизации. Например, можно настроить RabbitMQ для маршрутизации данных в разные приложения с разными привязками и обменами.

2. RabbitMQ применяет модель push, что означает, что производитель знает, использовало ли клиентское приложение сообщение. Этот вариант подходит для приложений, которые должны соблюдать определенные последовательности и предоставлять гарантии доставки при обмене данными и их анализе.

3. RabbitMQ также поддерживает более широкий спектр языков программирования по сравнению с Kafka.

### 1.7 Комплексная информационная система предприятия

КИСП (комплексная информационная система предприятия) – платформа, с помощью которой реализована информационная система производственного планирования и учета, которая позволяет:

- управлять закупками;
- своевременно обеспечивать производственные заказы материалами;

- оптимизировать использование ресурсов;
- повысить эффективность работы.

Комплексная информационная система предприятия (КИСП) представляет собой интегрированный набор программных продуктов и модулей, предназначенных для автоматизации различных аспектов деятельности предприятия. КИСП позволяет объединить в единую систему все ключевые бизнес-процессы, обеспечивая централизованное управление информацией и ресурсами организации.

На рисунке 1.5 представлена схема подпроцесса поступления, оприходования и оперативного учета ТМЦ в среде КИСП.

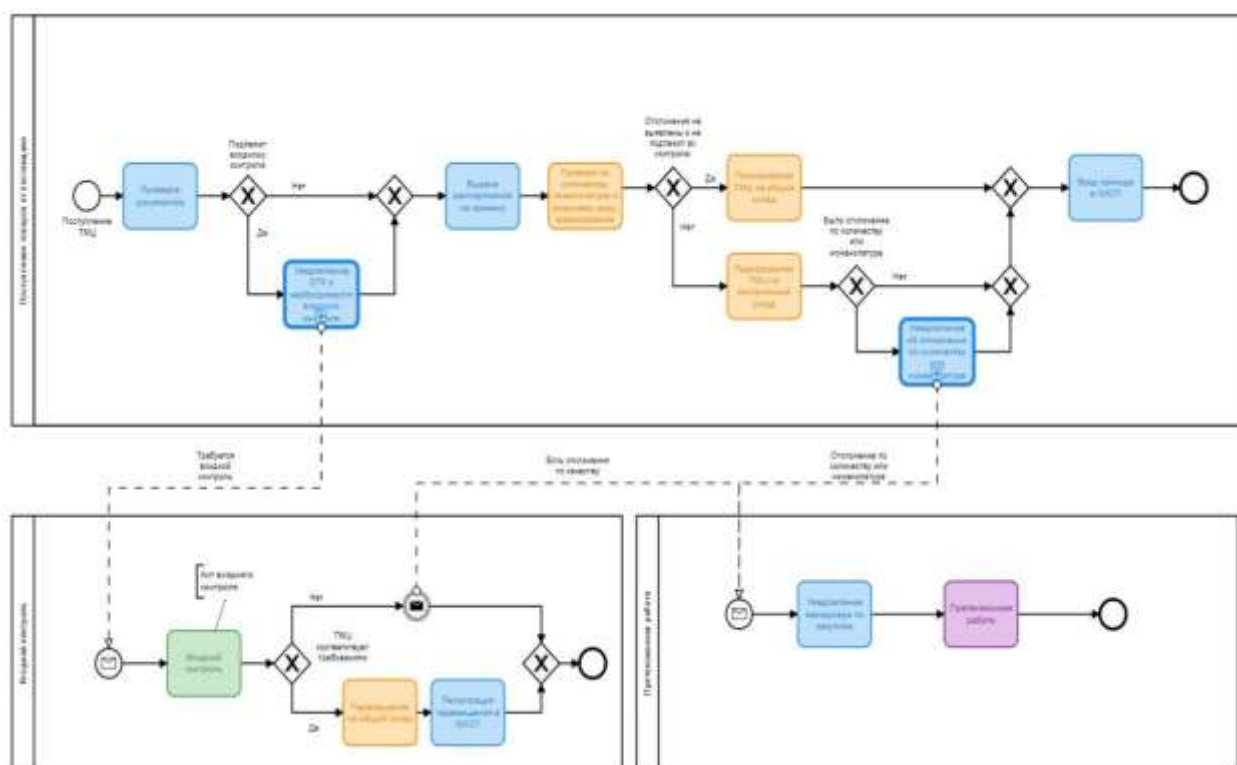


Рисунок 1.5 – Схема подпроцесса поступления, оприходования и оперативного учета ТМЦ

Основные компоненты КИСП обычно включают в себя следующие модули:

1. ERP (Enterprise Resource Planning) – модуль планирования ресурсов предприятия, который обеспечивает автоматизацию бухгалтерского учета, управление финансами, закупками, складским учетом, управлением продажами и другими оперативными процессами.

2. CRM (Customer Relationship Management) – модуль управления отношениями с клиентами, предназначенный для автоматизации работы с клиентами, учета контактов, управления продажами и маркетинговой деятельностью.

3. BI (Business Intelligence) – модуль бизнес-аналитики, который обеспечивает сбор, обработку и анализ данных, предоставляя руководителям и менеджерам предприятия информацию для принятия обоснованных управленческих решений.

4. Модуль управления производственными процессами, который позволяет автоматизировать производственные операции, планировать производственные потребности, контролировать качество выпускаемой продукции.

5. Модуль управления персоналом, который обеспечивает автоматизацию кадровых процессов, учет рабочего времени, расчет заработной платы, поддержку процессов обучения и развития сотрудников.

КИСП представляет собой целостную информационную систему, совокупность взаимосвязанных модулей, которые позволяют повысить эффективность управления предприятием, улучшить качество принимаемых решений, оптимизировать бизнес-процессы и повысить конкурентоспособность организации на рынке. Внедрение КИСП позволяет существенно сократить временные и финансовые затраты на управление бизнесом, снизить риски ошибок и упростить взаимодействие между подразделениями организации.

## 1.8 Разбор XML документа, чтение XML документа, фабрика XDTO

XML-файлы являются одним из основных видов электронных документов. В данном формате отправляется большинство отчетов в государственные органы, осуществляется обмен складскими документами

между контрагентами. Помимо этого, внутренний документооборот в крупных компаниях часто ведется в этом формате [9].

В 1С:Предприятие существует несколько способов обработки XML документов:

- простое последовательное чтение XML;
- получение из XML файла документа DOM и последовательный перебор всех узлов полученного документа;
- получение из XML файла документа DOM и отбор требуемых узлов в DOM документе с помощью выражений XPath;
- разбор XML документа с помощью фабрики XDTO.

В выпускной квалификационной работе используется работа с XML записями с помощью фабрики XDTO.

Аббревиатура XDTO (XML Data Transfer Objects) переводится как Объекты переноса данных XML. Этот механизм, разработанный компанией 1С и используемый исключительно в её экосистеме, предназначен для интеграции и обмена данными между 1С и другими системами в формате XML. XDTO позволяет работать с XML в объектно-ориентированном стиле, не требуя от разработчика заботиться об атрибутах, тегах и узлах [12].

Механизм XDTO включает несколько объектов встроенного языка и объект конфигурации «XDTO-пакет».

Помимо использования XDTO, в 1С существует множество способов парсинга XML файлов – от простого последовательного чтения текста до применения внешних DLL библиотек. Однако, использование XDTO имеет несколько преимуществ:

1. Быстродействие: Конвертация файла в Объект XDTO происходит относительно быстро, и время выполнения линейно зависит от объема данных.

2. Удобство программирования: всего три строки кода позволяют получить готовый Объект XDTO из любого XML, с которым удобно работать, используя синтаксис 1С "через точку".

3. Проверка соответствия XSD-схеме: при наличии XSD-схемы можно сразу проверять соответствие файла этой схеме [5].

## 2 ПРОЕКТНЫЙ РАЗДЕЛ

### 2.1 Решение проблемы затрат времени при обмене данными

КИСП (комплексная информационная система предприятия) – платформа, с помощью которой реализована информационная система производственного планирования и учета, которая позволяет:

- управлять закупками;
- своевременно обеспечивать производственные заказы материалами;
- и многое другое.

КИСП является обширной базой данных, которая хранит в себе множество данных о заказчиках, заказах, складах и т.д.

Так как уже существующий метод обмена данными требовал много времени, было принято решение о введении в первичную систему КИСП информацию движения материала, это обусловило уменьшение времени по оприходованию материалов на склад и списыванию материалов в производство.

Контекстная диаграмма ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как стало» показана на рисунке 2.1.

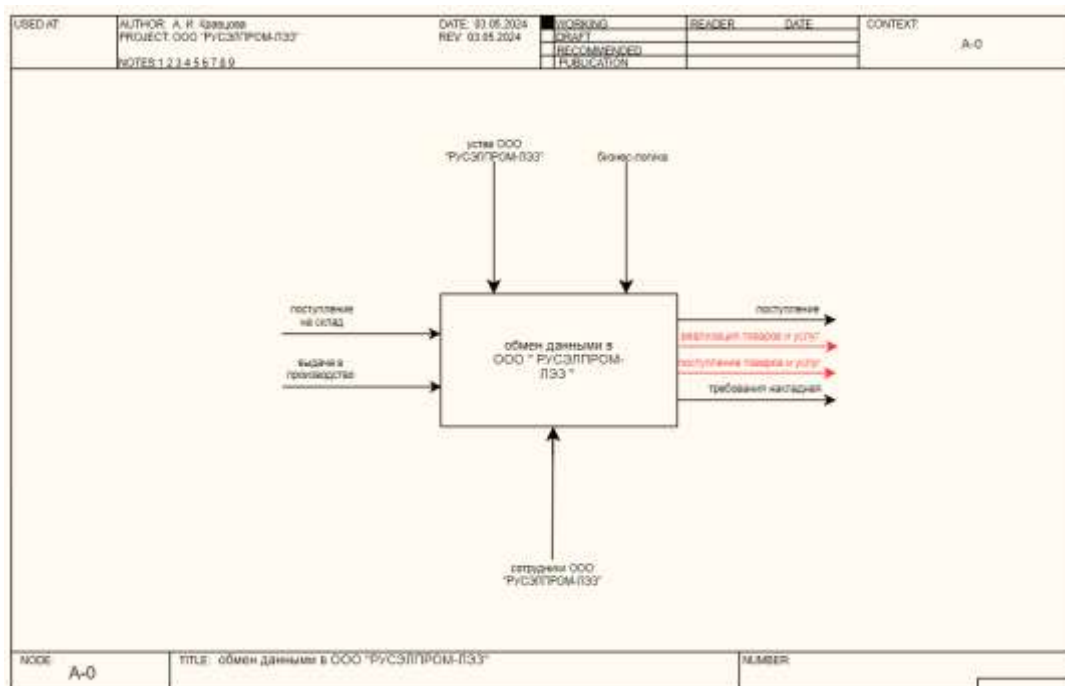


Рисунок 2.1 – Функциональная модель

Входными данными здесь являются документ «Поступление на склад» и документ «Выдача в производство». Выходными данными – документ «Поступление», документ «Требование-накладная», документ «Реализация товаров и услуг» или документ «Поступление товаров и услуг».

Декомпозиция контекстной диаграммы ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как стало» показана на рисунке 2.2.

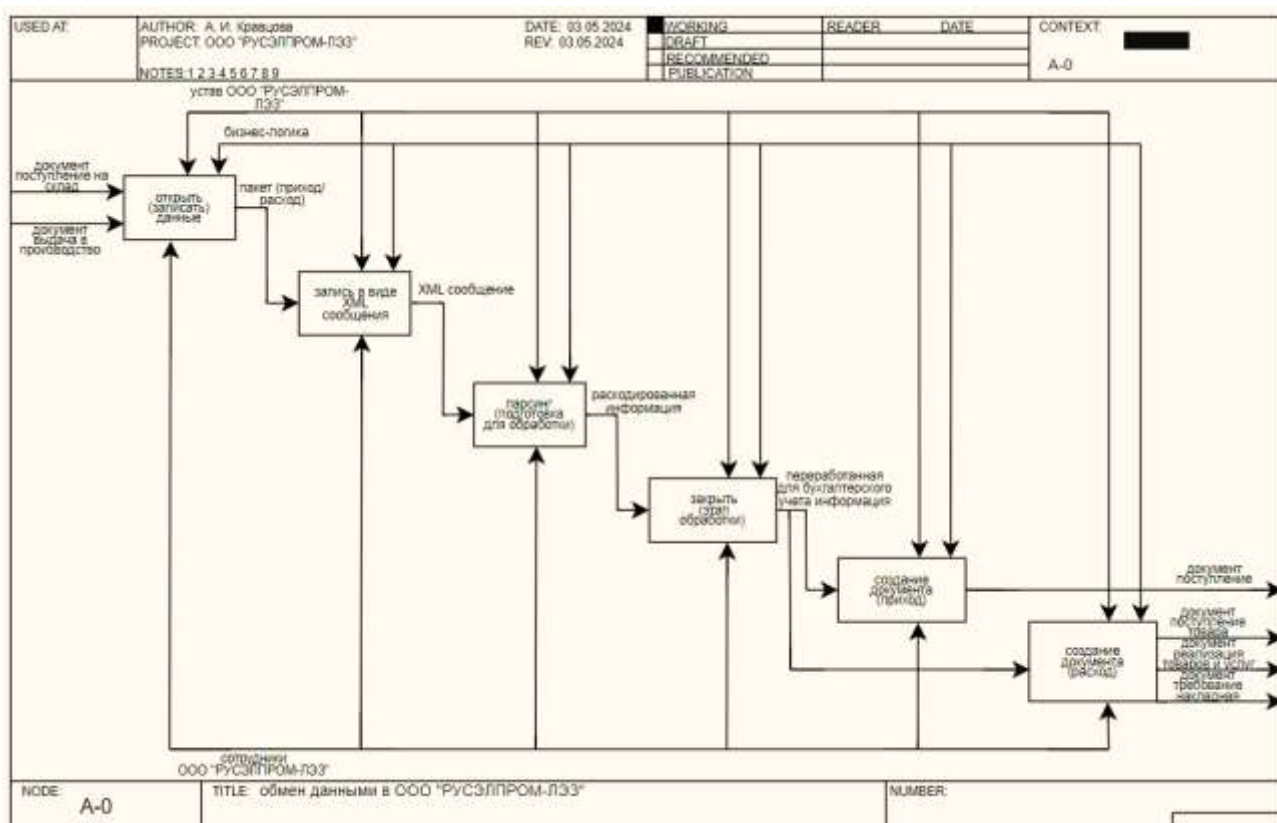


Рисунок 2.2 – Декомпозиция контекстной диаграммы

Глядя на рисунок 3.2, на котором изображена декомпозиция контекстной диаграммы ООО «РУСЭЛПРОМ-ЛЭЗ» обмена данными «Как стало», можно увидеть заметную разницу:

- этапов стало больше;
- на выходе можно выбрать информацию, которая необходима бухгалтеру;
- затраты по времени для обмена данными заметно уменьшились.



## 2.2 Как происходит обмен данными с помощью 1С:УПП и КИСПа

Для начала работы необходимо создать документ «Поступление товаров на склад» на основании бумажного универсального передаточного документа (УПД) от поставщика в систему КИСП. Данный документ создает/вносит кладовщик (рис. 2.3).

Склад входит в отдельную юридическую компанию РУСЭЛПРОМ-ТЕХСНАБ.

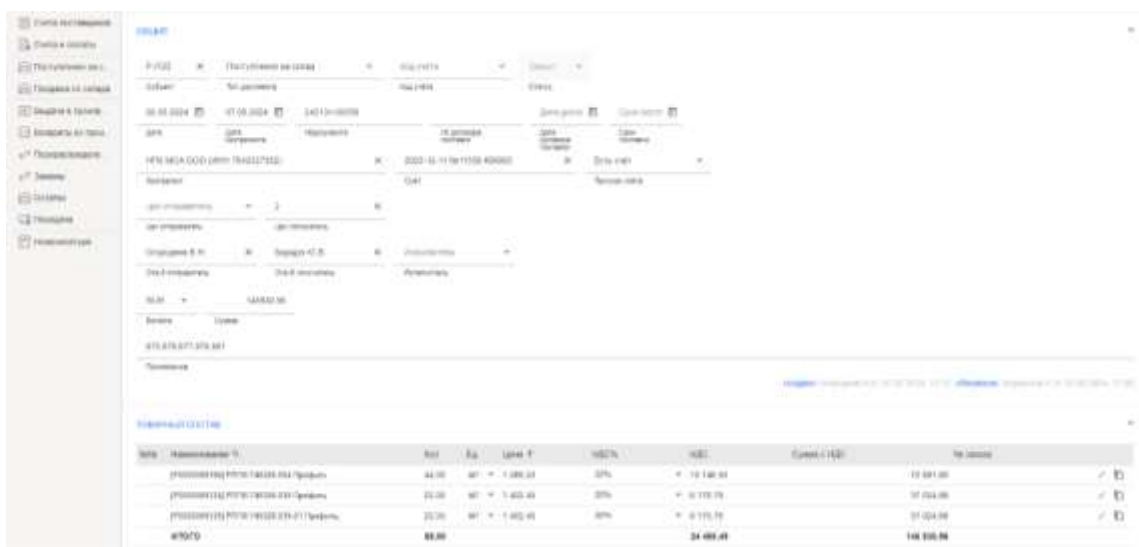


Рисунок 2.3 – Внесение данных в новый созданный документ

Если в базе КИСП не найден поставщик, то кладовщик создает новую карточку справочника контрагента. Карточка справочника представлена на рисунке 2.4.



Рисунок 2.4 – Карточка справочника контрагента

То же самое происходит в том случае, если не существует карточки номенклатуры (рис. 2.5.).

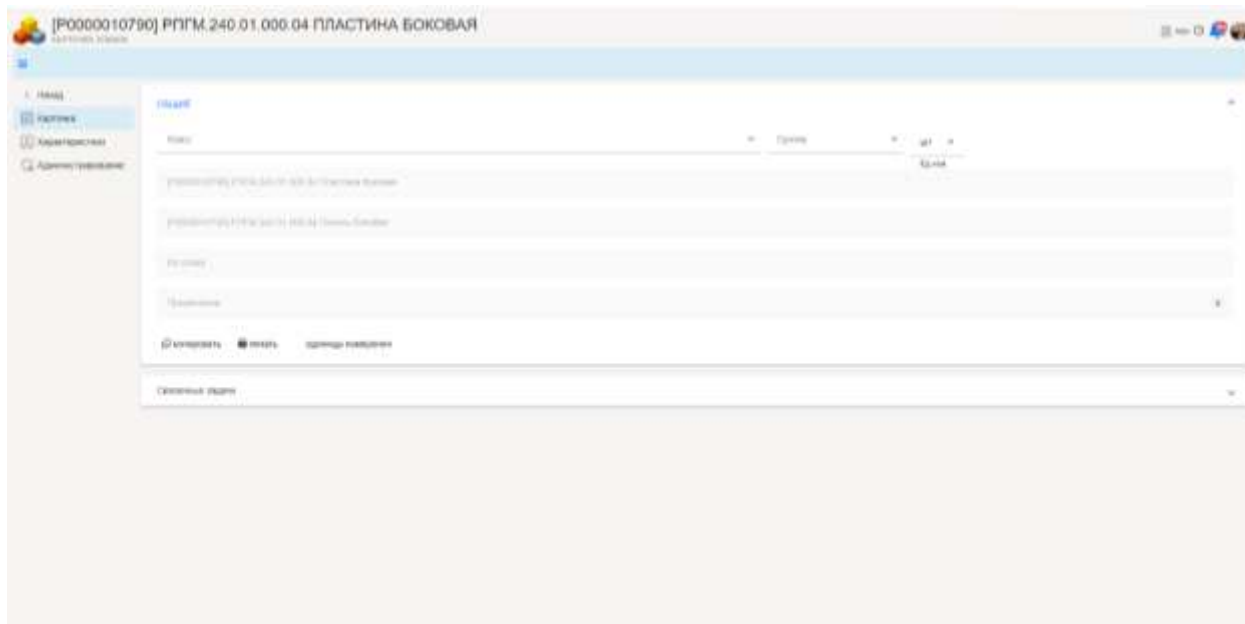


Рисунок 2.5 – Карточка номенклатуры

На этом работа кладовщика заканчивается. Итогом работы кладовщика является формирование папки для обмена в 1С:УПП (рис 2.6.).

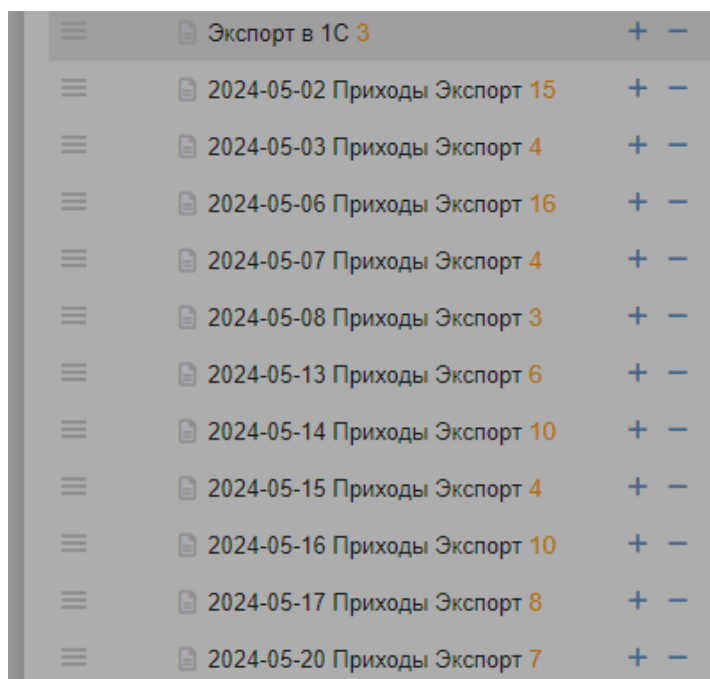


Рисунок 2.6 – Созданные папки

После этого нажимается кнопка «экспортировать», чтобы система сформировала два протокола (рис. 2.7-2.8). Экспорт объектов настраивается администратором системы. В настоящий момент система поддерживает два протокола: экспорт в файл или отправка сообщения с помощью брокера сообщений RabbitMQ.

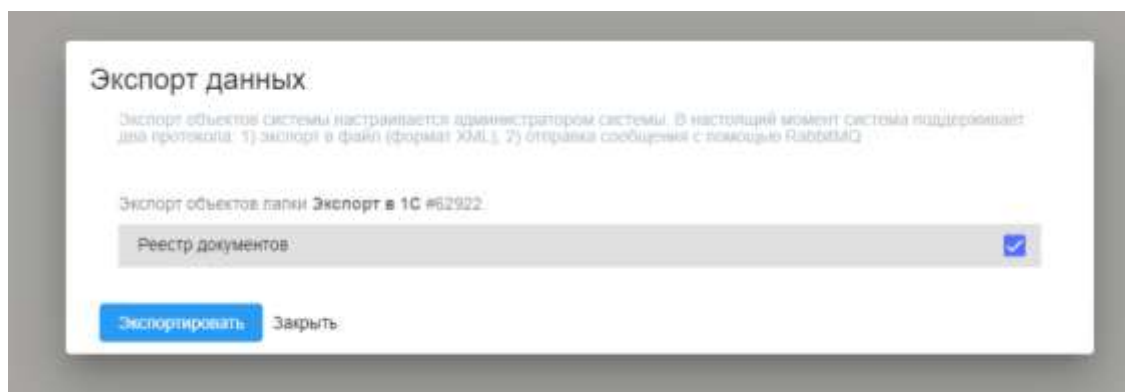


Рисунок 2.7 – Экспорт данных

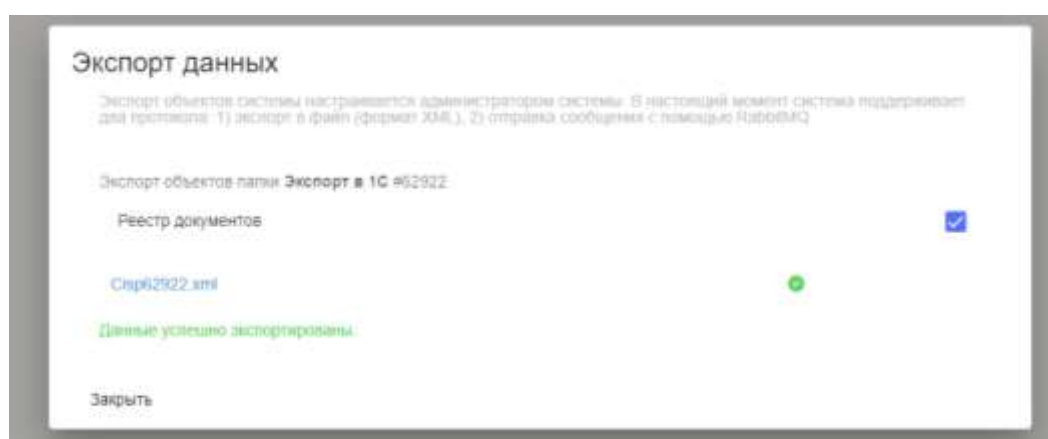


Рисунок 2.8 – Успешный экспорт данных

На выходе экспорта получается сообщение в виде строки XML-записи (рис. 3.9).

```
<Документ ДокументКод="26915" ДокументДата="08.05.2024" ДокументНомер="240131-00058" ДокументТип="Поступление материала на склад"
ДокументПримечание="875,876,877,878,881" Клиент="НПК МСА 000" КлиентИН="7842327352" КлиентДата="31.01.2024" Кладовщик="Бордук Юрий
Вячеславович"><Строки><Строка ProductId="9829" Код="P0000009125" Наименование="РПГМ.745326.039-01 Профиль" ЕдИзм="шт"
Количество="22.000000" СуммаБНДС="30853.900000" СуммаСНДС="37024.680000" /><Строка ProductId="25361" Код="P0000009106"
Наименование="РПГМ.745326.054 Профиль" ЕдИзм="шт" Количество="44.000000" СуммаБНДС="60734.670000" СуммаСНДС="72881.600000" /><Строка
ProductId="30924" Код="P0000009124" Наименование="РПГМ.745326.039 Профиль" ЕдИзм="шт" Количество="22.000000" СуммаБНДС="30853.900000"
СуммаСНДС="37024.680000" /></Строки></Документ><Документ ДокументКод="26955" ДокументДата="16.05.2024" ДокументНомер="ЦБ-395"
ДокументТип="Поступление материала на склад" ДокументПримечание="КОРы" Клиент="ОВЕН СПб 000" КлиентИН="7806447654"
КлиентДата="01.02.2024" Кладовщик=""><Строки><Строка ProductId="14530" Код="P0000003464" Наименование="Фиксатор торцевой МТУ-51"
ЕдИзм="шт" Количество="36.000000" СуммаБНДС="1710.000000" СуммаСНДС="2052.000000" /><Строка ProductId="33992" Код="P0000010819"
Наименование="Нарисовка пружинных клемм 2.5мм2, пустая, МТС-2.5НС" ЕдИзм="шт" Количество="800.000000" СуммаБНДС="1740.400000"
СуммаСНДС="2088.480000" /><Строка ProductId="34011" Код="P0000010837" Наименование="Клемма пружинная проходная МТС-2.5" ЕдИзм="шт"
Количество="600.000000" СуммаБНДС="23940.000000" СуммаСНДС="28728.000000" /><Строка ProductId="34012" Код="P0000010838"
Наименование="Заглушка торцевая МТС-Р2.5" ЕдИзм="шт" Количество="40.000000" СуммаБНДС="600.400000" СуммаСНДС="720.480000" /><Строка
ProductId="34013" Код="P0000010839" Наименование="Клемма пружинная заземления МТС-2.5РЕ" ЕдИзм="шт" Количество="150.000000"
```

Рисунок 2.9 – Строка XML-записи

После происходит парсинг, в ходе которого происходит раскодирование информации (рис. 2.10). По итогу парсинга в 1С:УПП приходит сообщение об успешной записи данных (рис. 2.11).

```
<Сообщение Источник="CISP_REM" Организация="РЭМ" Агент="CISP" ВерсияАгента="1" Запрос="1445" Папка="62922">
<Документы>
<Документ ДокументКод="26189" ДокументДата="27.04.2024" ДокументНомер="2" ДокументТип="Поступление материала на склад"
ДокументПримечание="819,820,821 Б240С" Клиент="ЗАВОД ПСКОВЭЛЕКТРОИТ ООО" КлиентИНН="6027153968" КлиентДата="10.01.2024"
Кладовщик="Лебедев Антон Владимирович"><Строки>
<Строка ProductId="33954" Код="P0000010790" Наименование="РПГМ.240.01.000.04 Пластина боковая" ЕДИЗМ="шт" Количество="6.000000"
СуммаЕНДС="14850.000000" СуммаСНДС="17820.000000" />
</Строки>
```

Рисунок 2.10 – Раскодированная информация



Рисунок 2.11 – Сообщение об успешной записи данных

На этом этапе начинается работа бухгалтера, который уже может сформировать два документа на выбор (рис. 2.12):

- 1 приход;
- 2 расход.

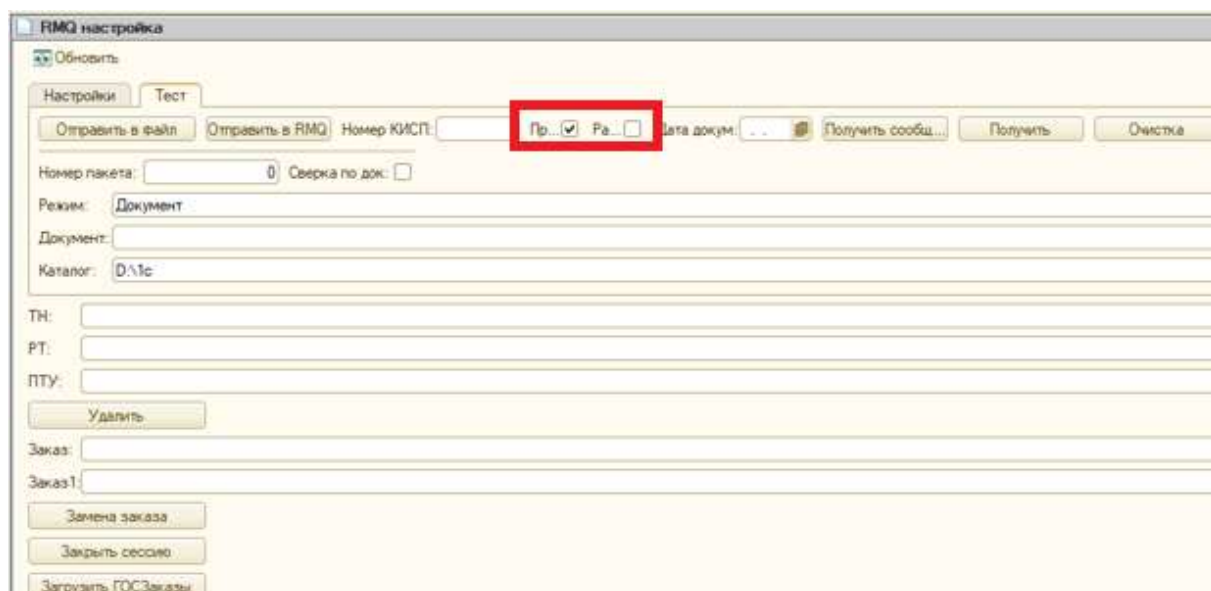


Рисунок 2.12 – Приход или расход

Если бухгалтер решит создать документ «Приход», то получится документ «Поступление товаров и услуг» (рис. 2.13).

Поступление товаров и услуг: покупка, комиссия. Проведен \*

Операция ▾ Цены и валюта... Действия ▾ Заполнить и провести

Номер: STСБ0000176 от: 08.04.2024 9:00:00 Отобразить в:  упр. учете  бух. учете  налог. уче...

Организация: ООО "Русэлпром-Техснаб" На склад: Менеджмент (37)

Контрагент: МСА НПК ООО Договор: основной

Заказ поставщику:  По сделке с поставщиком нет долга

Заказы Товары (3 поз.) Услуги (0 поз.) Дополнительно

№	Номенклатура	Хар...	Серия	Количест...	Ед.	К.	Цена	Сумма	% Н...	Сумма...	Всего	Заказ
1	РПГМ.745326.03...			22,000	шт	1,000	1 402,45	30 853...	20%	6 170,78	37 024,68	
2	РПГМ.745326.05...			44,000	шт	1,000	1 380,33	60 734...	20%	12 146...	72 881,60	
3	РПГМ.745326.03...			22,000	шт	1,000	1 402,45	30 853...	20%	6 170,78	37 024,68	

Тип цен: цены закупки Всего (руб.): 146 930,96

Счет-фактура: Ввести счет-фактуру НДС (сверху): 24 488,49

Номер в КИСП: 26915 Комментарий:

ТОРГ-12 (Товарная накладная за поставщика с услугами) Резервирование Печать ОК Записать Закрыть

Рисунок 2.13 – Сформированный документ «Поступление товаров и услуг»

Если бухгалтеру необходимо создать документ «Расход», то итогом создания будут три документа:

- 1 требование-накладная (рис. 2.14-2.16);
- 2 реализация товаров и услуг (рис. 2.17-2.18);
- 3 поступление товаров и услуг: в переработку (рис. 2.19-2.20).

Требования-накладные: 2024 г.

Действия

У	Б	Н	Дата	Номер	По	Склад	Организация	Ответственный	Комментарий
✓	✓	✓	16.05.2024 16:00:00	S000000..	Уч.	Склад давальческ..	ООО "Русэлпрон-П..	Администратор	КИСП №26368 от 16...
✓	✓	✓	16.05.2024 16:00:00	S000000..	Уч.	Склад давальческ..	ООО "Русэлпрон-П..	Администратор	КИСП №26368 от 16...
✓	✓	✓	16.05.2024 16:00:00	S000000..	Уч.	Склад давальческ..	ООО "Русэлпрон-П..	Администратор	КИСП №26368 от 16...
✓	✓	✓	16.05.2024 16:00:00	S000000..	Уч.	Склад давальческ..	ООО "Русэлпрон-П..	Администратор	КИСП №26368 от 16...

Рисунок 2.14 – Требования-накладные выбранных заказов

Требование-накладная: Проведен \*

Действия

Номер: S0000000098 от: 16.05.2024 16:00:00

Организация: ООО "Русэлпрон-ПЗЗ"

Подразделение: Участок сборки и электромонтажа

Отразить в:  упр. уче.  бух. уче.  налог. уче.

Склад: Склад давальческого сырья (9)

Подразделение организации: Участок сборки и электромонтажа

Материалы | НДС | Дополнительно

№	Номенкл.	Характерис. номенклату...	Серия номенклатуры	Количество	Ед.	К.	Статья затрат	Хар-р затрат
1	Винт М4-6x16..			16,000	шт	1,000	Материалы принятые в переработку	Производстве расходы
2	ВАКИ.30.. Крышка			1,000	шт	1,000	Материалы принятые в переработку	Производстве расходы
3	Заклеп. резьбов...			6,000	шт	1,000	Материалы принятые в переработку	Производстве расходы
4	Наконеч. НШВИ...			24,000	шт	1,000	Материалы принятые в переработку	Производстве расходы
5	Стойка РСНСN-10			1,000	шт	1,000	Материалы принятые в переработку	Производстве расходы
6	Коробка			8,000	шт	1,000	Материалы принятые	Производстве

Номер в КИСП: 26368

Комментарий: КИСП №26368 от 16.05.2024

Требование-накладная | Печать | ОК | Записать | Закрыть

Рисунок 2.15 – Требование-накладная одного заказа

Печать документов

Печать копий 1

### Требование-накладная № 98 от 16 мая 2024

Организация: ООО "Русэлпром-ЛЗЗ"

Склад: Склад давальческого сырья (9)

Подразделение: Участок сборки и электромонтажа блоков и кассет

№	Материал	Количество	
1	Винт М4-6х16.58.016 ГОСТ 17475-80 DIN 965 DIN 963	16	шт
2	ВАКИ.301255.003 Крышка	1	шт
3	Защелка-гайка резьбовая рифленая с потайным фланцем М4 L=11	6	шт
4	Наконечник НШВИ 0,5-8	24	шт
5	Стойка РСНСН-10	1	шт
6	Коробка распаячная 120x80x50 IP55 67051	8	шт
7	Контакт вилочный CDSM-0,75	12	шт
8	РПГМ БЦСВ 1.00.21.00 Корпус	1	шт
9	Лак Gramolin Plastik(1л)	0,028	л

Отпустил \_\_\_\_\_ Получил \_\_\_\_\_

Рисунок 2.16 – Готовая требование-накладная

Реализация товаров и услуг: 2024 г.

Действия: [Иконки] Перейти: [Иконки] Оформить документы [Иконки] ЭДО

У	Б	Н	Дата	Номер	Вид операции	Вид перед.	Сумма	Вал.	Контрагент	Сделка	Склад
✓	✓	✓	16.05.2024 12:00:00	STC5000...	продажа, ко...	Со склада	20 330,61	руб.	РУСЭЛПРОМ М...		Менеджмент (
✓	✓	✓	16.05.2024 12:00:00	STC5000...	продажа, ко...	Со склада	728,33	руб.	РУСЭЛПРОМ М...		Менеджмент (
✓	✓	✓	16.05.2024 12:00:00	STC5000...	продажа, ко...	Со склада	1 505,33	руб.	РУСЭЛПРОМ М...		Менеджмент (
✓	✓	✓	16.05.2024 12:00:00	STC5000...	продажа, ко...	Со склада	88,20	руб.	РУСЭЛПРОМ М...		Склад Техсна

Рисунок 2.17 – Документы «Реализация товаров и услуг»

Реализация товаров и услуг: продажа, комиссия. Проведен \*

Операция ▾ Цены и валюта... Действия ▾ Перейти ▾ Заполнить и провести >>

Номер: STСB0000105 от: 16.05.2024 12:00:00 Отобразить в:  упр. учете  бух. учете  налог. уче...

Организация: ООО "Русэлпром-Техснаб" Со склада: Менеджмент (37)

Контрагент: РУСЭЛПРОМ МЕНЕДЖМЕНТ ООО Договор: Договор продажи материалов

Заказ покупат...

Товары (9 поз.) Тара (0 поз.) Услуги (0 поз.) Дополнительно Печать

№	Номенклатура	Характеристика номен...	Серия номенклатуры	Кол...	Ед.	К.	Сумм
1	Винт М4-6gx16.58.016 ГОСТ 17475-80...			16.0...	шт	1,000	
2	ВАКИ.301255.003 Крышка			1,000	шт	1,000	1
3	Заклепка-гайка резьбовая рифле...			6,000	шт	1,000	
4	Наконечник НШВИ 0,5-8			24,0...	шт	1,000	
5	Стойка РСНСN-10			1,000	шт	1,000	
6	Коробка распаячная 120x80x50 IP55...			8,000	шт	1,000	
7	Контакт вилочный CDSM-0,75			12,0...	шт	1,000	
8	РПГМ.БЦСВ 1.00.21.00 Корпус			1,000	шт	1,000	
9	Лак Gramolin Plastik(1л)			0,028	л	1,000	

Тип цен: Отпускные Всего (руб.): 20 330,61

Счет-фактура: Вести счет-фактуру НДС (сверху): 3 388,43

Номер в КИСП: 26368 Комментарий: КИСП №26368 от 16.05.2024

Расходная накладная Печать Чек ОК Записать Закрыть

Рисунок 2.18 – Документ «Реализация товаров и услуг»

Поступления товаров и услуг: 2024 г.

Действия ▾ Перейти ▾ ЗДО ▾

У	Б	Н	Дата	Вид операции	Номер	Вид по...	Сумма	Валюта	Контрагент	Склад
✓	✓	✓	16.05.2024 14:0...	в переработ...	SO00000...	На skl...	16 942,18	руб.	РУСЭЛПРОМ МЕНЕДЖМЕ...	Склад давальческогог
✓	✓	✓	16.05.2024 14:0...	в переработ...	SO00000...	На skl...	606,94	руб.	РУСЭЛПРОМ МЕНЕДЖМЕ...	Склад давальческогог
✓	✓	✓	16.05.2024 14:0...	в переработ...	SO00000...	На skl...	1 254,44	руб.	РУСЭЛПРОМ МЕНЕДЖМЕ...	Склад давальческогог
✓	✓	✓	16.05.2024 14:0...	в переработ...	SO00000...	На skl...	73,50	руб.	РУСЭЛПРОМ МЕНЕДЖМЕ...	Склад давальческогог

Рисунок 2.19 – Документы «Поступления товаров и услуг»



Поступление товаров и услуг: в переработку. Проведен \*

Операция ▾ Цены и валюта... Действия ▾ Перейти ▾ Заполнить и провести

Номер: SO000000103 от: 16.05.2024 14:00:00 Отобразить в:  упр. учете  бух. учете  налог. уче...

Организация: ООО "Русэлпром-ПЭЗ" На склад: Склад давальческого сырья (9)

Контрагент: РУСЭЛПРОМ МЕНЕДЖМЕНТ ООО Договор: Договор подряда № М-2021/018 от 05.05.202...

Заказ покупателя: Заказ покупателя 820ДМ1 от 25.01.2...

Материалы (1 поз.) Дополнительно

Заполнить ▾ Проверить ▾ Изменить Подбор Серийные номера Переоценка

№	Номенклатура	Хара...	Серия	Количество	Ед.	К.	Цена	Сумма	Склад
1	Разъем TJS-6P6C (К			3,000	шт	1,000	24,50	73,50	Склад даваль...

Тип цен: Не заполнено! Всего (руб.): 73,50

Счет-фактура: Ввести счет-фактуру НДС (сверху):

Номер в КИСП: 26368 Комментарий:

ТОРГ-12 (Товарная накладная за поставщика с услугами) | Резервирование | Печать | **OK** | Записать | Закрыть

Рисунок 2.20 – Документ «Поступление товаров и услуг: в переработку»

На этом работа бухгалтера окончена. Итогом работы бухгалтера являются созданные документы: или документ «Приход», или документ «Расход».

## ЗАКЛЮЧЕНИЕ

В заключении дипломной работы на тему «Разработка АИС обмена данными между системами оперативного учёта в среде 1С:Предприятие» можно отметить, что цель исследования – автоматизация процесса обмена данными между информационными системами была достигнута.

В ходе выполнения выпускной квалификационной работы были выполнены все поставленные задачи, а именно:

- проанализировать предприятие и выявить недостатки;
- изучить основные технологии и инструменты для реализации обмена данными;
- автоматизировать процесс обмена данными между системами оперативного учета.

Разработанная АИС позволяет производить оперативный обмен данными между 1С:УПП и КИСП.

Разработанное программное решение успешно внедрено и теперь функционирует в компании ООО «РУСЭЛПРОМ-ЛЭЗ». В дальнейшем планируется его техническое сопровождение.

В заключение, стоит отметить, что разработанная АИС для обмена данными между системами оперативного учета обладает значительным потенциалом для будущего развития и совершенствования.

Путем внедрения новых технологий, расширения функциональности и адаптации к изменяющимся требованиям рынка приложение будет сохранять свою актуальность и оставаться важным инструментом в области логистики.

Успешная реализация перспективных направлений развития обеспечит приложению конкурентные преимущества и способствует повышению эффективности при обмене данными.

Таким образом, завершение всех этапов проекта позволило создать приложение, соответствующее современным стандартам и требованиям. Разработанное приложение будет значимым инструментом для оптимизации

обмена данными между системами оперативного учета в среде 1С:Предприятие.

Исследовательские методы включают в себя анализ научной и технической литературы, моделирование процессов, проектирование и проверку программного обеспечения. Для разработки автоматизированных систем применяются современные технологии и инструменты программирования.

## СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Абдуллаев А. Сравнительный анализ использования отечественных и зарубежных бухгалтерских информационных систем //Общество и инновации. – 2021. – Т. 2. – №. 3/S. – С. 250-263.
2. Аблякимова А. С. Преимущества и формы автоматизации бухгалтерского учета //Финансово-экономическое и информационное обеспечение инновационного развития региона. – 2020. – С. 211-214.
3. Белова О. А. КАДРОВЫЙ И БУХГАЛТЕРСКИЙ УЧЕТ КОМАНДИРОВОК В ПРОГРАММЕ" БУХСОФТ: ПРЕДПРИЯТИЕ" //Состояние, направления и перспективы развития среднего профессионального образования. – 2019. – С. 12-18.
4. Вятчин А. А., Глотина И. М. АДМИНИСТРИРОВАНИЕ ПОЛЬЗОВАТЕЛЕЙ В 1С: УПП //Интернаука. – 2019. – №. 29. – С. 13-14.
5. Городнов Е. В. ПОКАЗАТЕЛИ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ //Современные информационные технологии: интеграция науки и практики. – 2019. – С. 32-37.
6. Громак В. И. и др. Программа научно-исследовательской практики для специальности: 1-31 80 03 Математика и компьютерные науки, 1-31 80 04 Механика и математическое моделирование. № УД-9024/уч. – 2020.
7. Доильницын Д. В., Шевченко О. И. СФЕРА ВЫСОКИХ ТЕХНОЛОГИЙ КАК ДВИГАТЕЛЬ ЭКОНОМИЧЕСКОГО РОСТА //Символ науки. – 2023. – №. 12-2. – С. 99-103.
8. Ефремова Л. И., Колекина А. О. Выбор системы электронного документооборота для предприятия //Вестник Волжского университета им. ВН Татищева. – 2019. – Т. 2. – №. 1. – С. 23-31.
9. Крыжановская Ю. А., Науменко П. С. Дистанционное обучение: решение на платформе 1С //Современные информационные технологии и ИТ-образование. – 2021. – Т. 17. – №. 3. – С. 757-770.

10. Купченко М. В., Инзарцев А. В. Обзор программных решений для управления продажами малого предприятия розничной торговли //Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований. – 2022. – С. 413-415.

11. Латышева З. И., Николенко Д. В. Цифровизация управленческого учета в России: современные тенденции и перспективы //Вестник Курской государственной сельскохозяйственной академии. – 2021. – №. 2. – С. 74-80.

12. Литасов А. В. Способы интеграции и обмена данными сторонних приложений с системой 1С //Прогрессивные технологии и экономика в машиностроении: сборник трудов X Всероссийской научно-практической конференции для студентов и учащейся молодежи, 4-6 апреля 2019 г., Юрга.—Томск, 2019. – 2019. – С. 200-202.

13. МУЛИНА А. В., ШОРИНА И. С. ОБЗОР ПРОТОКОЛА AMQR ДЛЯ ИНТЕРНЕТА ВЕЩЕЙ И ЕГО УЯЗВИМОСТЕЙ //Мавлютовские чтения. – 2022. – С. 644-651.

14. Новиков М. О., Якунин А. Г. АНАЛИЗ МЕТОДОВ ОДНОСТОРОННЕЙ СИНХРОНИЗАЦИИ БАЗ ДАННЫХ //Измерение, контроль, информатизация. – 2023. – С. 37-42.

15. Осадчая К. И., Иванова Е. В., Халина Я. Ю. Компьютерные программы для автоматизации бухгалтерского учета //ЭКОНОМИКА И ПРАВО-2023. – 2023. – С. 58-64.

16. Попов Д. В., Кутикова С. П. Формирование системы управления организации //ТЕХНИЧЕСКОЕ ТВОРЧЕСТВО МОЛОДЕЖИ Учредители: Московский государственный технологический университет" Станкин". – 2022. – №. 3. – С. 38-43.

17. Сааков В. В. и др. БРОКЕРЫ СООБЩЕНИЙ: ДЛЯ ЧЕГО ОНИ НУЖНЫ //СОВРЕМЕННАЯ НАУКА И МОЛОДЫЕ УЧЁНЫЕ. – 2022. – С. 64-67.

18. Савина А. Г., Малявкина Л. И. Концепция построения архитектуры системы «1С: Предприятие» и средства разработки прикладных решений //Экономическая среда. – 2021. – №. 1. – С. 63-69.

19. Шевченко И. В., Похлебаев А. В., Уксекова М. А. СЦЕНАРИИ И ИНСТРУМЕНТЫ УПРАВЛЕНИЯ АРХИТЕКТУРОЙ ПРЕДПРИЯТИЯ В КОНТЕКСТЕ ОПТИМИЗАЦИИ БИЗНЕС-ПРОЦЕССОВ //Экономика устойчивого развития. – 2021. – №. 2. – С. 177-181.

20. Шуляк А. В. АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ ХРАНИЛИЩА ETCD //Международный научный. – 2021. – С. 132.

## ПРИЛОЖЕНИЕ

```
Ниже представлен код «чтение из шлюза»:
//ВидОчереди - Приход,Расход,ОчисткаПриход,ОчисткаРасход
//ДатаДокументов - дата документов в КИСПе
//Сообщение - вывести строку сообщения RabbitMQ в окно
сообщений
//НомерКИСП - номер документа в КИСПе
//НомерПакета - номер пакета, если очередь состоит из нескольких
пакетов
&НаСервере
Функция
ПолучитьXMLнаСервере (ВидОчереди,ДатаДокументов,Сообщение,НомерКИ
СП,НомерПакета) Экспорт
// Получение настроек
Сервер = ПолучитьНастройку ("Сервер");
Логин = ПолучитьНастройку ("Логин");
Пароль = ПолучитьНастройку ("Пароль");
ВиртХост = ПолучитьНастройку ("Вирт. хост");
ТочкаОбмена = ПолучитьНастройку ("Точка обмена");
КлючТО = ПолучитьНастройку ("Ключ ТО для реестра");
ПарXML = Новый Структура
("Источник,Статус,ТипСобытия,ОписаниеОшибки,ВидДокумента");
ПарXML.Статус = "ОК";
Если ПустаяСтрока (Сервер) ИЛИ ПустаяСтрока (Логин) ИЛИ
ПустаяСтрока (Пароль) Тогда
Сообщение1 = Новый СообщениеПользователю;
Сообщение1.Текст = "Проверьте настройки";
Сообщение1.Сообщить ();
Возврат Ложь;
КонецЕсли;
Макет = ПолучитьОбщийМакет ("RMQ_Компонента");
Адрес = ПоместитьВоВременноеХранилище (Макет);
Попытка
// Первая попытка
ПодключитьВнешнююКомпоненту (Адрес, "BITERP",
ТипВнешнейКомпоненты.Native);
Исключение
Попытка
// Вторая попытка
ПодключитьВнешнююКомпоненту (Адрес, "BITERP",
ТипВнешнейКомпоненты.Native);
Исключение
ПарXML.Статус = "ERR";
ПарXML.ОписаниеОшибки = "Ошибка подключения
внешней компоненты";
Сообщение1 = Новый СообщениеПользователю;
Сообщение1.Текст = "Ошибка подключения внешней
компоненты";
Сообщение1.Сообщить ();
Возврат Ложь;
КонецПопытки;
КонецПопытки;
```

```

Кролик = Новый("AddIn.BITERP.PinkRabbitMQ");

Попытка
    Кролик.Connect(Сервер, 5672, Логин, Пароль, ВиртХост);
    сообщить("подключился");
Исключение
    Кролик = Неопределено;
    ПарXML.Статус = "ERR";
    ПарXML.ОписаниеОшибки = "Не удалось соединиться с
сервером RMQ " + Сервер;
    Сообщение1 = Новый СообщениеПользователю;
    Сообщение1.Текст = "Не удалось соединиться с сервером
RMQ ";
    Сообщение1.Сообщить();
    Возврат Ложь;
КонецПопытки;

Попытка

    Если ВидОчереди = "Приход" или ВидОчереди = "ОчисткаПриход"
Тогда
        ИмяОчереди = "rem.cisp-upp.supply";
        ИначеЕсли ВидОчереди = "Расход" или ВидОчереди = "Сверка"
или ВидОчереди = "ОчисткаРасход" Тогда
            ИмяОчереди = "rem.cisp-upp.lzk";
            КонецЕсли;
            Потребитель = Кролик.BasicConsume(ИмяОчереди, "", Ложь,
Ложь, 0);
            ОтветноеСообщение = ""; // Обнуляем, чтобы избежать утечку
памяти
            ТегСообщения = 0; // Обнуляем, чтобы избежать утечку памяти
            КоличествоДокументов = 0;

            РезультатРазборки = Новый Массив;

            НомПакета=1;
            Пока Кролик.BasicConsumeMessage("", ОтветноеСообщение,
ТегСообщения, 5000) Цикл
                //Пока Кролик.BasicConsumeMessage("",
ОтветноеСообщение, 5000) Цикл
                    Если ВидОчереди = "ОчисткаРасход" или ВидОчереди =
"ОчисткаПриход" Тогда
                        Если НомерПакета = 0 ИЛИ НомПакета = НомерПакета
Тогда
                            Кролик.BasicAck(ТегСообщения);
                            КонецЕсли;
                            НомПакета = НомПакета +1;
                            Продолжить;
                            КонецЕсли;
                            Сообщить("Успешно! Из очереди прочитано сообщение " +
ОтветноеСообщение);
                            КоличествоДокументовП = 0;

```



```

        КоличествоДокументовД = 0;
        КоличествоДокументов = КоличествоДокументов + 1;
        //СоздатьДок(ОтветноеСообщение, Таб);
        ОтветноеСообщение1 = "<?xml version="+""+"1.0"+""+"
encoding="+""+"UTF-
8"+""+"?>"+Символы.ПС+СокрЛП(ОтветноеСообщение);
        ОтветноеСообщение1
        СтрЗаменить(ОтветноеСообщение1, "<Документы", "<Документы
xmlns=""http://localhost/testXML"" ");
        Ниже представлен код «Получение настроек»:
        //Возвращает настройки сервера из справочника RMQ_Сервер Функция
        ПолучитьНастройку(ИмяНастройки)
            Значение
        Справочники.RMQ_Сервер.НайтиПоНаименованию(ИмяНастройки).Значени
        е;
        Возврат Значение;
        КонецФункции // ПолучитьНастройку

        нутри
        //ПолучитьXMLнаСервере(ВидОчереди, ДатаДокументов, Сообщение, Номер
        К//ИСП, НомерПакета) Экспорт
        ОтветноеСообщение1 = "<?xml version="+""+"1.0"+""+"
encoding="+""+"UTF-
8"+""+"?>"+Символы.ПС+СокрЛП(ОтветноеСообщение);
        ОтветноеСообщение1
        СтрЗаменить(ОтветноеСообщение1, "<Документы", "<Документы
xmlns=""http://localhost/testXML"" ");

        ЧтениеXML = Новый ЧтениеXML;
        // Устанавливаем строку брокера сообщения RabbitMQ,
        содержащую текст XML, для чтения объектом ЧтениеXML
        ЧтениеXML.УстановитьСтроку(ОтветноеСообщение1);
        // Считывается узел XML сообщение
        Пока ЧтениеXML.Прочитать() = Истина Цикл
            //Проверка свойств ЧтениеXML.ЛокальноеИмя и
        ЧтениеXML.ТипУзла значению начало элемента XML
            Если ЧтениеXML.ЛокальноеИмя = "Сообщение" И
        ЧтениеXML.ТипУзла = ТипУзлаXML.НачалоЭлемента Тогда
                ТипXDTO
            ФабрикаXDTO.Тип("http://localhost/testXML", "Документы");
                Парам
            ФабрикаXDTO.ПрочитатьXML(ЧтениеXML, ТипXDTO);
                сообщить("Ок");
                // указываем Фабрике XDTO в пакете список
        объектов Документы или одиночный объект Документы
                Если ТипЗнч(Парам.Документы) <>
        Тип("СписокXDTO") Тогда
                    МассивXDTO = Новый Массив;
                    МассивXDTO.Добавить(Парам.Документы);
                Иначе
                    МассивXDTO = Парам.Документы;
                КонецЕсли;

```

```

//Считывается узел Документы
Для каждого Доки из МассивХДТО Цикл

// указываем Фабрике ХДТО в пакете
список объектов Документ или одиночный объект Документ
Если
Доки.Последовательность().Количество()=1 Тогда
    МассивДоки = Новый Массив;
    МассивДоки.Добавить(Доки.Документ);
Иначе
    МассивДоки = Доки.Документ;
КонецЕсли;

//Считывается узел Документ
// обход массива документов, содержащих
реквизиты
Для каждого Док из МассивДоки Цикл
    КоличествоДокументовП =
КоличествоДокументовП + 1;
    // чтение и преобразование атрибута
ДатыДокумента в тип Дата
    ДатаД1 =
строковыефункцииклиентсервер.СтрокаВДату(Строка(Док.ДокументДата
));
    Год = Год(ДатаД1);
    Месяц = Месяц(ДатаД1);
    День = День(ДатаД1);
    Час = Час(ТекущаяДата());
    Минута = Минута(ТекущаяДата());
    Секунда = Секунда(ТекущаяДата());
    ДатаД =
Дата(Год, Месяц, День, Час, Минута, Секунда);

    Если НомерПакета <> 0 И
КоличествоДокументов <> НомерПакета Тогда
        Продолжить;
        КонецЕсли;

        ПустаяДата =
'00010101000000'; // '00010101'
        // проверка команды импортировать
документы по указанному реквизиты формы ДатаДокумента
        Если ДатаДокументов <> ПустаяДата
Тогда
            Если НачалоДня(ДатаД) <>
НачалоДня(ДатаДокументов) Тогда
                Продолжить;
            Иначе
                сообщить(Док.ДокументКод);
                КоличествоДокументовД =
КоличествоДокументовД + 1;
            КонецЕсли;

```

```

КонецЕсли;

// проверка команды импортировать
один документ по указанному реквизиту формы НомерКИСП
Если СтрДлина (Строка (НомерКИСП) ) >
0 Тогда
Если СокрЛП (НомерКИСП) <>
СокрЛП (Док.ДокументКод) Тогда
Продолжить;
Иначе
сообщить (Док.ДокументКод);
КоличествоДокументовД =
КоличествоДокументовД + 1;
КонецЕсли;
КонецЕсли;

// проверка команды вывести строку
сообщения RabbitMQ в окно сообщений
Если Сообщение = "Да" Тогда
Продолжить;
КонецЕсли;

ДатаП = Дата (Год, Месяц, День, 9, 0, 0);
ДатаР =
Дата (Год, Месяц, День, 12, 0, 0);
ДатаПТ =
Дата (Год, Месяц, День, 14, 0, 0);
ДатаТ =
Дата (Год, Месяц, День, 16, 0, 0);

ПарXML.ВидДокумента = Док.ДокументТип;
// Загрузка документов Поступление
на склад
Если ВидОчереди = "Приход" Тогда
//поиск по уникальному коду
ДокументКод в 1С был ли загружен документ
ПоступлениеТоваровУслуг ранее
ДокументПоступление =
Документы.ПоступлениеТоваровУслуг.НайтиПоРеквизиту ("НомерВходяще
гоДокументаЭлектронногоОбмена", Док.ДокументКод);
Если ДокументПоступление <>
Документы.ПоступлениеТоваровУслуг.ПустаяСсылка () Тогда
ДокументПоступление1 =
ДокументПоступление.ПолучитьОбъект ();
НомерДок =
ДокументПоступление1.Номер;
КонецЕсли;

//структура, описывающая
реквизиты и табличную часть документа ПоступлениеТоваровУслуг
ДокРеквизиты = Новый
Структура ("Товары, Услуги, ТоварыИЗаказы");

```

```

ДокРеквизиты.Вставить ("НомерВходящегоДокументаЭлектронногоОб
мена", Док.ДокументКод);
ПарXML.ТипСобытия = "Новый
КИСП";
// при перезаписи документа
номер документа сохраняется
Если
СтрДлина (СокрЛП (НомерДок) ) > 0 Тогда
ДокРеквизиты.Вставить ("Номер", НомерДок);
ПарXML.ТипСобытия =
"Перезапись КИСП";
КонецЕсли;
НомерДок = "";
Заказ =
Справочники. RMO_ГОСЗаказы.ПустаяСсылка ();
Попытка
Разделитель =
Строка (Док.РазделительКод);
// поиск необходимого
Склада в справочнике RMO_ГОСЗаказы по атрибуту РазделительКод
Заказ =
Справочники. RMO_ГОСЗаказы.НайтиПоРеквизиту ("РазделительКод", Сокр
ЛП (Разделитель) );
Если Заказ <>
Справочники. RMO_ГОСЗаказы.ПустаяСсылка () Тогда
ЗаказГОС =
Заказ.ПолучитьОбъект ();
ЗаказПокупателя =
ЗаказГОС.ЗаказПокупателя;
ДокРеквизиты.Вставить ("СкладОрдер", ЗаказГОС.Склад);
Иначе
сообщить ("не нашел
склад"+ Разделитель);
Заказ =
Справочники. RMO_ГОСЗаказы.ПустаяСсылка ();
// по умолчанию
Склад Менеджмент, если РазделительКод пустой
СкладМен =
Справочники.Склады.НайтиПоКоду ("000000037");
ДокРеквизиты.Вставить ("СкладОрдер", СкладМен);
КонецЕсли;
Исключение
Заказ =
Справочники. RMO_ГОСЗаказы.ПустаяСсылка ();
СкладМен =
Справочники.Склады.НайтиПоКоду ("000000037");

```

```

// по умолчанию Склад
Менеджмент, если нет атрибута РазделительКод

ДокРеквизиты.Вставить ("СкладОрдер", СкладМен);
КонецПопытки;
ДокРеквизиты.Вставить ("Сделка", Документы.ЗаказПоставщику.ПустаяСсылка ());

Попытка
КлиентДата1 =
строковыефункции.клиентсервер.СтрокаВДату (Строка (Док.КлиентДата))
;
Год = Год (КлиентДата1);
Месяц =
Месяц (КлиентДата1);
День = День (КлиентДата1);
Час = Час (ТекущаяДата ());
Минута =
Минута (ТекущаяДата ());
Секунда =
Секунда (ТекущаяДата ());
КлиентДата =
Дата (Год, Месяц, День, Час, Минута, Секунда);
// заполняем дату
поставщика

ДокРеквизиты.Вставить ("ДатаВходящегоДокумента", КлиентДата);
Исключение
сообщить ("Не заполнена
дата поставщика");
КонецПопытки;

ТехСнаб =
Справочники.Организации.НайтиПоКоду ("000000007");

ДокРеквизиты.Вставить ("Дата", ДатаП);

ДокРеквизиты.Вставить ("ДатаВходящегоДокументаЭлектронногоОбм
ена", ДатаП);

// поиск Контрагента в
справочнике Контрагенты по атрибуту КлиентИНН
Попытка
Контрагент =
Справочники.Контрагенты.НайтиПоРеквизиту ("ИНН", СокрЛП (Док.Клиент
ИНН));
Если Контрагент =
Справочники.Контрагенты.ПустаяСсылка () Тогда
сообщить ("не
правильный ИНН "+СокрЛП (Док.КлиентИНН));
Иначе

ДокРеквизиты.Вставить ("Контрагент", Контрагент.Ссылка);

```

```

        Выборка =
Справочники.ТипыЦенНоменклатурыКонтрагентов.Выбрать(, Контрагент)
;
        Пока
Выборка.Следующий() Цикл
        //установка
типа цен по владельцу Контрагент

        ДокРеквизиты.Вставить("ТипЦен", Выборка.Ссылка);
        КонецЦикла;
        Выборка =
Справочники.ДоговорыКонтрагентов.Выбрать(, Контрагент);
        Пока
Выборка.Следующий() Цикл
        Если
Выборка.Организация = ТехСнаб И Выборка.ВидДоговора =
Перечисления.ВидыДоговоровКонтрагентов.СПоставщиком и
СокрЛП(Выборка.ВалютаВзаиморасчетов.Наименование) = "руб." Тогда
        //установка договора по владельцу Контрагент

        ДокРеквизиты.Вставить("ДоговорКонтрагента", Выборка.Ссылка);
        КонецЕсли;
        КонецЦикла;
        КонецЕсли;
        Исключение
        КонецПопытки;

        // заполнение необходимых
реквизитов документа

        ДокРеквизиты.Вставить("НомерВходящегоДокумента", Док.Документ
Номер);

        ДокРеквизиты.Вставить("Организация", ТехСнаб);

        ДокРеквизиты.Вставить("ОтражатьВУправленческомУчете", Истина)
;

        ДокРеквизиты.Вставить("ОтражатьВБухгалтерскомУчете", Истина);

        ДокРеквизиты.Вставить("ОтражатьВНалоговомУчете", Истина);

        ДокРеквизиты.Вставить("Ответственный", ПараметрыСеанса.Текущи
йПользователь);

        ДокРеквизиты.Вставить("ВидОперации", Перечисления.ВидыОпераци
йПоступлениеТоваровУслуг.ПокупкаКомиссия);

        ДокРеквизиты.Вставить("ВидПоступления", Перечисления.ВидыПост
упленияТоваров.НаСклад);

```

```

ДокРеквизиты.Вставить ("УчитыватьНДС", Истина);

ДокРеквизиты.Вставить ("ВалютаДокумента", Справочники.Валюты.Наименование);

ДокРеквизиты.Вставить ("КурсВзаиморасчетов", 1);

ДокРеквизиты.Вставить ("КратностьВзаиморасчетов", 1);

ДокРеквизиты.Вставить ("Комментарий", "КИСП" + "№" +
СокрЛП (Док.ДокументКод) + " от " + СокрЛП (Док.ДокументДата));

ДокРеквизиты.Вставить ("ФлагСчета", Истина);
// присвоение типа массив ТЧ
Товары, Услуги, ТоварыИЗаказы
Массив;
ДокРеквизиты.Товары = Новый
Массив;
ДокРеквизиты.Услуги = Новый
Новый Массив;
ДокРеквизиты.ТоварыИЗаказы =

// Загрузка документов Выдача
в производство
ИначеЕсли ВидОчереди = "Расход" или
ВидОчереди = "Сверка" Тогда

    НомерДок = "";
    ТехСнаб =
Справочники.Организации.НайтиПоКоду ("000000007");
//поиск по уникальному коду
ДокументКод в 1С был ли загружен документ ТребованиеНакладная
ранее

    ДокументТребование =
Документы.ТребованиеНакладная.НайтиПоРеквизиту ("НомерВходящегоДокументаЭлектронногоОбмена", Док.ДокументКод);
    Если ДокументТребование <>
Документы.ТребованиеНакладная.ПустаяСсылка () Тогда
        ДокументТребование1 =
ДокументТребование.ПолучитьОбъект ();
        НомерДок =
ДокументТребование1.Номер;
    КонецЕсли;

//структура, описывающая
реквизиты и табличную часть документа ТребованиеНакладная
ДокРеквизиты = Новый
Структура ("Материалы, ТЧ");

ДокРеквизиты.Вставить ("Дата", ДатаТ);

```

```

ДокРеквизиты.Вставить ("НомерВходящегоДокументаЭлектронногоОб
мена", Док.ДокументКод);
ПарXML.ТипСобытия = "Новый
КИСП";
// при перезаписи документа
номер документа сохраняется
Если
СтрДлина (СокрЛП (НомерДок) ) > 0 Тогда
ДокРеквизиты.Вставить ("Номер", НомерДок);
ПарXML.ТипСобытия =
"Перезапись КИСП";
КонецЕсли;
// Организация завод
НПП =
Справочники.Организации.НайтиПоКоду ("000000001");
ДокРеквизиты.Вставить ("Организация", НПП);
// поиск Подразделение в
справочнике Подразделения по атрибуту ПодразделениеВ
ПодразделениеВ =
Строка (Док.ПодразделениеВ);
Если ПодразделениеВ = "1"
Тогда
ПодразделениеНаименование =
"Отдел управления качеством";
ИначеЕсли ПодразделениеВ = "2"
Тогда
ПодразделениеНаименование =
"Участок сборки, монтажа и упаковки систем возбуждения";
ИначеЕсли ПодразделениеВ = "3"
Тогда
ПодразделениеНаименование =
"Участок сборки и электромонтажа блоков и кассет";
ИначеЕсли ПодразделениеВ = "4"
Тогда
ПодразделениеНаименование =
"Участок сборки и монтажа силовой электроники";
ИначеЕсли ПодразделениеВ = "5"
Тогда
ПодразделениеНаименование =
"Испытательная лаборатория";
КонецЕсли;
Подразделение =
Справочники.Подразделения.НайтиПоНаименованию (СокрЛП (Подразделен
иеНаименование) );
Если
Подразделение<>Справочники.Подразделения.ПустаяСсылка () Тогда
ДокРеквизиты.Вставить ("Подразделение", Подразделение);
КонецЕсли;

```



```

ПодразделениеО =
Справочники.ПодразделенияОрганизаций.НайтиПоНаименованию(СокрЛП(
ПодразделениеНаименование));

Если
ПодразделениеО<>Справочники.ПодразделенияОрганизаций.ПустаяСсылк
а() Тогда

    ДокРеквизиты.Вставить("ПодразделениеОрганизации",Подразделен
иеО);

    КонецЕсли;

    // заполнение необходимых
реквизитов документа

    СкладДС =
Справочники.Склады.НайтиПоКоду("000000009"); //выбор склада
Давальческого сырья

    ДокРеквизиты.Вставить("Склад",СкладДС);

    ДокРеквизиты.Вставить("ОтражатьВУправленческомУчете",Истина)
;

    ДокРеквизиты.Вставить("ОтражатьВБухгалтерскомУчете",Истина);

    ДокРеквизиты.Вставить("ОтражатьВНалоговомУчете",Истина);

    ДокРеквизиты.Вставить("Ответственный",ПараметрыСеанса.Текущи
йПользователь);

    ДокРеквизиты.Вставить("Комментарий", "КИСП №"+
СокрЛП(Док.ДокументКод) + " от "+СокрЛП(Док.ДокументДата));

    ДокРеквизиты.Материалы = Новый
Массив;

    ДокРеквизиты.ТЧ = Новый
Массив;

    КонецЕсли;

    //ЗаполнитьЗначенияСвойств(ДокОбъект,ДокРеквизиты);

    сообщить(Док.ДокументКод);

    //Считывается узел Строки
Если
Док.Строки.Последовательность().Количество()>0 Тогда
    //указываем фабрике XDTO в
пакете список объектов Строки или одиночный объект Строки
    Если ТипЗнч(Док.Строки.Строка)
<> Тип("СписокXDTO") Тогда

```

```

Массив;
МассивХДТО1 = Новый
МассивХДТО1.Добавить (Док.Строки.Строка);
Иначе
МассивХДТО1 =
Док.Строки.Строка;
КонецЕсли;
Массив = Новый Массив;
Массив.Добавить (Тип ("Число"));
КЧ = Новый
КвалификаторыЧисла (15, 3);
ОписаниеТиповЧ = Новый
ОписаниеТипов (Массив, , , КЧ);

//Считывается узел Строка
Для каждого Стр из
МассивХДТО1 Цикл
услуга = 0;
// ТЧ документа
ПоступлениеТоваровУслуг
ДокТовары = Новый
Структура ("Склад, Заказ, Количество, СтавкаНДС, Номенклатура, Единица
Измерения, Коэффициент, Цена, Сумма, СуммаНДС");
ДокУслуги = Новый
Структура ("Заказ, Номенклатура, Количество, СтавкаНДС, Цена, Сумма, Су
ммаНДС, СтатьяЗатрат, Подразделение, Содержание, ПодразделениеОргани
зации");
ДокЗаказы = Новый
Структура ("ЗаводскойЗаказ, Заказ, Номенклатура, Материал, Количество
, Цена, Сумма1");
// ТЧ документа
ТребованиеНакладная
ДокМатериалы = Новый
Структура ("Заказ, Количество, Номенклатура, ЕдиницаИзмерения, Коэффи
циент, Качество, СтатьяЗатрат");
Если ВидОчереди =
"Приход" Тогда
//синхронизация
справочника Номенклатура 1с и КИСП по атрибуту Код
НоменклатураКод =
ВернутьНоменклатуру (Стр);
Если НоменклатураКод
<> Справочники.Номенклатура.ПустаяСсылка () Тогда
флНом = 1;

```

```

                                                    Если
НоменклатураКод.ВидНоменклатуры =
Справочники.ВидыНоменклатуры.НайтиПоНаименованию("Услуга") Тогда
                                                    услуга =
1;

    ДокУслуги.Номенклатура = НоменклатураКод.Ссылка;

    ДокУслуги.Содержание = НоменклатураКод.Наименование;
                                                    Иначе

    ДокТовары.Номенклатура = НоменклатураКод.Ссылка;

    ДокТовары.ЕдиницаИзмерения =
НоменклатураКод.ЕдиницаХраненияОстатков;
                                                    КонецЕсли;
                                                    Иначе
наименование у строки "+стр.ProductId);
                                                    сообщить ("нет
кода у строки "+стр.ProductId);
                                                    сообщить ("нет
                                                    КонецЕсли;

                                                    //ТЧ товары
Если услуга = 0
Тогда
                                                    Если Заказ <>
Справочники.RMQ_ГОСЗаказы.ПустаяСсылка() Тогда
                                                    ЗаказГОС =
Заказ.ПолучитьОбъект();

    ЗаказПокупателя = ЗаказГОС.ЗаказПокупателя;

    ДокТовары.Склад = ЗаказГОС.Склад;
                                                    КонецЕсли;

                                                    //поиска Заказа
в документе Поступление
                                                    Попытка

    СтрокаЗаказ = Стр.Заказ;
                                                    Заказ1 =
Справочники.RMQ_ГОСЗаказы.НайтиПоНаименованию(СокрЛП(СтрокаЗаказ
), Истина);
                                                    Если
Заказ1 <> Справочники.RMQ_ГОСЗаказы.ПустаяСсылка() Тогда

    ЗаказГОС1 = Заказ1.ПолучитьОбъект();

    ЗаказПокупателя = ЗаказГОС1.ЗаказПокупателя;

```

```

ДокТовары.Склад = ЗаказГОС1.Склад;

Иначе

СтрокаЗаказ = "";

КонецЕсли;

Исключение

СтрокаЗаказ = "";

КонецПопытки;

Если Заказ =
СкладМен =
Справочники.РМЦ_ГОСЗаказы.ПустаяСсылка() Тогда
Справочники.Склады.НайтиПоКоду("000000037");

ДокТовары.Склад = СкладМен;

Если
СокрЛП(СтрокаЗаказ) = "" Тогда
"Резерв";
'20220101000000';

ЗаказПокупателя
Документы.ЗаказПокупателя.НайтиПоНомеру(Ном, Дат);

КонецЕсли;
КонецЕсли;

ДокТовары.Количество = Число(Стр.Количество);
Если
Число(Стр.Количество) = 0 Тогда

Продолжить;

КонецЕсли;

= Число(Стр.СуммаБНДС);

ДокТовары.Сумма
СуммаНДС =
Число(Стр.СуммаСНДС) - Число(Стр.СуммаБНДС);

ДокТовары.СуммаНДС = СуммаНДС;

```

```

= Число (Стр.СуммаБНДС) /Число (Стр.Количество) ;
<> 0 Тогда
    ДокТовары.Цена
    Если СуммаНДС
        ДокТовары.СтавкаНДС = Перечисления.СтавкиНДС.НДС20;
        Иначе
            ДокТовары.СтавкаНДС = Перечисления.СтавкиНДС.БезНДС;
            КонецЕсли;
        Иначе //ТЧ услуги
            ДокУслуги.Количество = Число (Стр.Количество) ;
            Если
                Число (Стр.Количество) = 0 Тогда
                    Продолжить ;
                    КонецЕсли;
                ДокУслуги.Сумма
                = Число (Стр.СуммаБНДС) ;
                СуммаНДС =
                Число (Стр.СуммаСНДС) -Число (Стр.СуммаБНДС) ;
                ДокУслуги.СуммаНДС = СуммаНДС;
                ДокУслуги.Цена
                = Число (Стр.СуммаБНДС) /Число (Стр.Количество) ;
                Если СуммаНДС
                    <> 0 Тогда
                        ДокУслуги.СтавкаНДС = Перечисления.СтавкиНДС.НДС20;
                        Иначе
                            ДокУслуги.СтавкаНДС = Перечисления.СтавкиНДС.БезНДС;
                            КонецЕсли;
                        КонецЕсли;
                        //ТЧ ТоварыИЗаказы
                        ДокЗаказы.ЗаводскойЗаказ = ЗаводскойЗаказ (ЗаказПокупателя) ;
                        ДокЗаказы.Заказ =
                        ЗаказПокупателя;
                        ДокЗаказы.Номенклатура = ДокТовары.Номенклатура;

```

```

ДокТовары.Номенклатура;
ДокТовары.Цена;
= ДокТовары.Количество;
ДокТовары.Сумма;

ДокЗаказы.Материал =
ДокЗаказы.Цена =
ДокЗаказы.Количество
ДокЗаказы.Сумма1 =

КонецЕсли;

//СтрокаТабличнойЧасти.Заказ =
СтрокаРазмещения.Размещение;
Если ВидОчереди =
"Расход" или ВидОчереди = "Сверка" Тогда
ДокТовары1.Добавить ();
//ДокТоварыСтр =

Если
Число (Стр.КоличествоФакт) = 0 Тогда
Продолжить;
КонецЕсли;

ДокТовары.Количество= Число (Стр.КоличествоФакт) ;
ДокТовары.СтавкаНДС
= Перечисления.СтавкиНДС.НДС20;

ДокТовары.Коэффициент=1;

НоменклатураКод =
ВернутьНоменклатуру (Стр) ;

// проверка
номенклатуры
Если НоменклатураКод
<> Справочники.Номенклатура.ПустаяСсылка () Тогда

ДокТовары.Номенклатура = НоменклатураКод.Ссылка;

ДокМатериалы.Номенклатура = НоменклатураКод.Ссылка;

ДокТовары.ЕдиницаИзмерения =
НоменклатураКод.ЕдиницаХраненияОстатков;

ДокМатериалы.ЕдиницаИзмерения =
НоменклатураКод.ЕдиницаХраненияОстатков;

```

```

ДокМатериалы.Коэффициент                                     =
ДокМатериалы.ЕдиницаИзмерения.Коэффициент;
Иначе
сообщить ("нет
наименование у строки "+стр.ProductId);
сообщить ("нет
кода у строки "+стр.ProductId);
КонецЕсли;

ДокМатериалы.Количество = Число (Стр.КоличествоФакт) ;
Качество                                     =
Справочники.Качество.НайтиПоНаименованию ("Новый" ) ;

ДокМатериалы.Качество =Качество;

Попытка
ном                                     =
СокрЛП (Стр.Заказ) ;
Дат                                     =
'20180101000000';
ЗаказПокупателя
= Документы.ЗаказПокупателя.НайтиПоНомеру (Ном, Дат) ;
Пока
ЗаказПокупателя = Документы.ЗаказПокупателя.ПустаяСсылка () Цикл
Дат                                     =
ДобавитьМесяц (Дат, 12) ;
Если
Дат>ТекущаяДата () Тогда
Прервать;
КонецЕсли;

ЗаказПокупателя                                     =
Документы.ЗаказПокупателя.НайтиПоНомеру (Ном, Дат) ;
КонецЦикла;

ДокМатериалы.Заказ = ЗаказПокупателя;

ДокМатериалы.ЗаказРезерв = ЗаказПокупателя;
ДокТовары.Заказ
= ЗаказПокупателя;
Исключение
КонецПопытки;
СтатьяЗатрат                                     =
Справочники.СтатьиЗатрат.НайтиПоНаименованию ("Материалы принятые
в переработку" ) ; ДокМатериалы.СтатьяЗатрат= СтатьяЗатрат;
КонецЕсли;
Если услуга = 0 тогда
ДокТовары.Коэффициент = 1;

```

```

КонецЕсли;
//ТЧ Услуги
Если услуга = 1 тогда
    СтатьяЗатрат =
Справочники.СтатьиЗатрат.НайтиПоНаименованию ("Транспортные
расходы");
        ДокУслуги.СтатьяЗатрат= СтатьяЗатрат;

        Подразделение =
Справочники.Подразделения.НайтиПоНаименованию ("Центральный
материальный склад");
        ПодразделениеОрганизации =
Справочники.ПодразделенияОрганизаций.НайтиПоКоду ("S00000001");
        ДокУслуги.Подразделение =
Подразделение;
        ДокУслуги.ПодразделениеОрганизации =
ПодразделениеОрганизации;

        КонецЕсли;
        Если ВидОчереди =
"Приход" Тогда
            Если услуга = 1
тогда
ДокРеквизиты.Услуги.Добавить (ДокУслуги);
                Иначе
ДокРеквизиты.Товары.Добавить (ДокТовары);
ДокРеквизиты.ТоварыИЗаказы.Добавить (ДокЗаказы);
                КонецЕсли;
            ИначеЕсли ВидОчереди =
"Расход" или ВидОчереди = "Сверка" Тогда
                ДокРеквизиты.Материалы.Добавить (ДокМатериалы);
                КонецЕсли
                КонецЦикла;
            КонецЕсли;
            //ПарXML.Источник = ДокОбъект;
            //ЗаписьВРеестрСобытий (ПарXML);
РезультатРазборки.Добавить (ДокРеквизиты);
        КонецЦикла;
        КонецЦикла;
        КонецЕсли;
        КонецЦикла;

```