

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт ИнЭО

Направление подготовки Информатика и вычислительная техника

Кафедра Автоматики и компьютерных систем

**БАКАЛАВРСКАЯ РАБОТА**

Тема работы
Программный модуль для преобразования данных инженерных служб

УДК 004.415:004.67

Студент

Группа	ФИО	Подпись	Дата
3-8В2В2	Носкова Алена Викторовна		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Баночкин Павел Иванович			

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Рахимов Т.Р.	к.т.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Н.В.	к.т.н.		

**ДОПУСТИТЬ К ЗАЩИТЕ:**

И. о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев М.С.	к.т.н.		

Томск 2017

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЕ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
<i>Профессиональные компетенции</i>	
ПК1	Разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.
ПК2	Осваивать методики использования программных средств для решения практических задач
ПК3	Разрабатывать интерфейсы "человек - электронно-вычислительная машина".
ПК4	Разрабатывать модели компонентов информационных систем, включая модели баз данных.
ПК5	Разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования.
ПК6	Обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.
ПК7	Готовить презентации, научно-технические отчеты по результатам выполненной работы, оформлять результаты исследований в виде статей и докладов на научно-технических конференциях.
ПК8	Готовить конспекты и проводить занятия по обучению сотрудников применению программно-методических комплексов, используемых на предприятии.
ПК9	Участвовать в настройке и наладке программно-аппаратных комплексов.
ПК10	Сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем.
ПК11	Инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем.

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт ИнЭО

Направление подготовки Информатика и вычислительная техника

Кафедра Автоматики и компьютерных систем

УТВЕРЖДАЮ:

Зав. кафедрой

\_\_\_\_\_  
(Подпись)      (Дата)      (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Бакалаврской работы

Студенту:

Группа	ФИО
3-8В2В2	Носкова Алена Викторовна

Тема работы:

Программный модуль для преобразования данных инженерных служб	
Утверждена приказом директора (дата, номер)	от 18.04.2017 №2773/с
Срок сдачи студентом выполненной работы:	30.05.2017

## ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p><b>Исходные данные к работе</b></p> <p>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</p>	<p><b>Объект исследования</b> – программное приложение для инженерных служб DailyReport.</p> <p><b>Предмет исследования</b> – преобразование данных.</p> <p><b>Цель исследования</b> - разработка модуля преобразования данных инженерных служб для приложения DailyReport.</p>
<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</p>	<ol style="list-style-type: none"><li>1 изучить существующую структуру приложения DailyReport и схему базы данных приложения;</li><li>2 изучить программную среду <i>Delphi XE3</i> для разработки программной части модуля;</li><li>3 изучить программную среду <i>PL/SQL Developer</i> для работы со схемой данных приложения, а также с хранимыми пакетами;</li><li>4 разработать и реализовать алгоритм преобразования данных, обеспечивающий выполнение следующих требований:<ul style="list-style-type: none"><li>• для запуска процесса преобразования данных пользователь должен указать дату начала, дату окончания и цех предприятия;</li><li>• результат преобразования данных должен осуществляться в таблицу READIND_TAB базы данных <i>DailyReport_output</i> под управлением <i>Oracle</i> из баз данных <i>ASU-TP</i>, <i>TELEMERADCVS</i> под управлением <i>Microsoft SQL</i>, расположенных двух серверах;</li><li>• импортируемые данные должны быть синхронизированы по идентификаторам с базой данных <i>DailyReport_output</i>;</li><li>• при импорте данных должна осуществляться проверка, если показатель на это время уже записан, то запись не производится для числовых значений;</li><li>• обновление строковых значений показателей должно осуществляться с использованием разделителя с сохранением предыдущего значения;</li><li>• данные должны импортироваться по запросу пользователя.</li></ul></li><li>5 спроектировать и реализовать алгоритм передачи преобразованных данных из приложения <i>DailyReport</i> серверу <i>Oracle</i>;</li><li>6 реализовать графический интерфейс модуля, отвечающий требованиям:<ul style="list-style-type: none"><li>• модуль содержит окно выбора параметров</li></ul></li></ol>

	преобразования данных; <ul style="list-style-type: none"> <li>• результаты преобразования данных отображаются в табличной форме.</li> </ul>
<b>Перечень графического материала</b> <i>(с точным указанием обязательных чертежей)</i>	---
<b>Консультанты по разделам выпускной квалификационной работы</b> <i>(с указанием разделов)</i>	
Раздел	Консультант
<b>Финансовый менеджмент</b>	Рахимов Т.Р.
<b>Социальная ответственность</b>	Маланова Н.В.
<b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b>	
<b>Заключение</b>	

<b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b>	
---	--

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассисент	Банокин Павел Иванович			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
3-8В2В2	Носкова Алена Викторовна		

**Министерство образования и науки Российской Федерации**  
 федеральное государственное автономное образовательное учреждение  
 высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт ИнЭО

Направление подготовки (специальность) Информатика и вычислительная техника

Уровень образования Бакалавр

Кафедра АиКС

Период выполнения весенний семестр 2016/2017 учебного года

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН  
 выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:

30.05.2017

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	<i>Архитектура приложения DailyReport, источники входных данных, описание результатов консолидации</i>	20
	<i>Процесс преобразования данных</i>	25
	<i>Выбор и тестирование алгоритмов передачи преобразованных данных серверу Oracle</i>	25
	<i>Проверка результатов преобразования данных</i>	10
	<i>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</i>	10
	<i>Социальная ответственность</i>	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Баночкин Павел Иванович			

**СОГЛАСОВАНО:**

И. о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев М.С.	к.т.н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСО-  
СБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
3-8В2В2	Носкова Алена Викторовна

<b>Институт</b>	<b>ИнЭО</b>	<b>Кафедра</b>	<b>АиКС</b>
<b>Уровень образования</b>	Бакалавр	<b>Направление/специальность</b>	Информатика и вычислительная техника

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Человеческие ресурсы: 1 чел.
2. <i>Нормы и нормативы расходования ресурсов</i>	
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<p>Проведена оценка коммерческого потенциала: 1. Потенциальные потребители результатов исследования. 2. Анализ конкурентных технических решений 3. SWOT-анализ</p> <p>Произведен расчет бюджета научных исследований. Определена ресурсная, финансовая, бюджетная эффективность исследования посредством расчета интегрального финансового показателя, интегрального показателя ресурсоэффективности и эффективности.</p>
2. <i>Планирование и формирование бюджета научных исследований</i>	
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	

**Перечень графического материала:**

---
-----

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент	Рахимов Т.Р.	к.т.н.		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
3-8В2В2	Носкова Алена Викторовна		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
3-8В2В2	Носкова Алена Викторовна

<b>Институт</b>	<b>ИнЭО</b>	<b>Кафедра</b>	<b>АиКС</b>
<b>Уровень образования</b>	Бакалавр	<b>Направление/специальность</b>	Информатика и вычислительная техника

**Исходные данные к разделу «Социальная ответственность»:**

*Описание рабочего места (рабочей зоны, технологического процесса, механического оборудования)*

Рабочее место программиста/пользователя программного модуля

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

<b>1. Производственная безопасность</b>	<i>Анализ опасных и вредных факторов, возникающих при разработке и эксплуатации программного модуля приложения DailyReport: -повышенный уровень электромагнитных излучений; -повышенный уровень наличия статического электричества; -нагрузка, связанная с длительным сидячим положением тела; -высокая нагрузка на зрительный аппарат; -умственное перенапряжение; -отсутствие или недостаток естественного света; -электрический ток.</i>
<b>2. Экологическая безопасность</b>	<i>Защита окружающей среды: утилизация отработанной техники.</i>
<b>3. Безопасность в чрезвычайных ситуациях</b>	<i>Меры действия при возникновении пожара.</i>
<b>4. Правовые и организационные вопросы обеспечения безопасности</b>	<i>-правовые вопросы обеспечения безопасности; -организационные вопросы обеспечения безопасности.</i>

**Перечень графического материала:**

---

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Инженер	Маланова Н.В.	к.т.н.		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
3-8В2В2	Носкова Алена Викторовна		



## РЕФЕРАТ

Выпускная квалификационная работа 98 с., 25 рис., 25 табл., 21 источник, 2 прил.

Ключевые слова: преобразование данных, *Oracle*, база данных, инженерные службы, вода.

Объектом исследования является программное приложение для инженерных служб *DailyReport*.

Цель работы: разработка программного модуля для преобразования данных инженерных служб для приложения *DailyReport* на языке программирования *Delphi* с использованием СУБД *Oracle* и *Microsoft SQL*.

В процессе исследования проводились: анализ архитектуры подключаемых модулей приложения *DailyReport*. Изучение базы данных исходных и обработанных показаний качества воды. Проектирование алгоритмов передачи преобразованных данных из программного модуля в СУБД *Oracle*. Сравнение быстродействия алгоритмов.

В результате исследования: разработан программный модуль для преобразования данных инженерных служб для приложения *DailyReport* на языке *Delphi*, а также создан набор хранимых процедур и функций на сервере СУБД *Oracle* на языке *PL/SQL*.

Степень внедрения: внедрено на предприятии ООО «Водоканал» г. Новокузнецка.

Область применения: автоматизация учета качества производства на предприятии, осуществляющем добычу, очистку и поставку воды для жизнеобеспечения города.

Экономическая эффективность/значимость работы: разработка является экономически эффективной.

В будущем планируется: расширение функциональности созданного программного модуля возможностью запуска процесса преобразования данных по расписанию.

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	12
1 АРХИТЕКТУРА ПРИЛОЖЕНИЯ <i>DAILYREPORT</i> , ИСТОЧНИКИ ВХОДНЫХ ДАННЫХ, ОПИСАНИЕ РЕЗУЛЬТАТОВ ПРЕОБРАЗОВАНИЯ ДАННЫХ.....	16
1.1 Архитектура подключаемых модулей приложения <i>DailyReport</i> .....	16
1.2 Источники входных данных.....	18
1.3 Описание результатов преобразования данных.....	20
1.4 Требования к графическому интерфейсу.....	22
2 ПРОЦЕСС ПРЕОБРАЗОВАНИЯ ДАННЫХ.....	25
2.1 Получение параметров от пользователя.....	27
2.2 Формирование серии <i>SQL</i> -запросов к СУБД.....	31
2.3 Выполнение серии <i>SQL</i> -запросов.....	34
2.4 Сохранение данных в базу данных <i>DailyReport_output</i> .....	39
3 ВЫБОР И ТЕСТИРОВАНИЕ АЛГОРИТМОВ ПЕРЕДАЧИ ПРЕОБРАЗОВАННЫХ ДАННЫХ СЕРВЕРУ <i>ORACLE</i> .....	46
3.1 Алгоритмы передачи данных серверу <i>Oracle</i> .....	46
3.2 Реализация алгоритмов передачи данных серверу <i>Oracle</i> .....	48
3.3 Время работы алгоритмов передачи данных серверу <i>Oracle</i> .....	51
4 ПРОВЕРКА РЕЗУЛЬТАТОВ ПРЕОБРАЗОВАНИЯ ДАННЫХ.....	53
5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБРЕЖЕНИЕ.....	56
5.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения.....	57
5.2 Определение возможных альтернатив проведения научных исследований.....	61
5.3 Планирование научно-исследовательских работ.....	61
5.4 Бюджет научно-технического исследования (НТИ).....	68

5.5	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования .....	74
	Заключение .....	77
6	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....	79
	Введение.....	79
6.1	Производственная безопасность .....	80
6.2	Экологическая безопасность .....	87
6.3	Безопасность в чрезвычайных ситуациях .....	88
6.4	Правовые и организационные вопросы обеспечения безопасности.....	89
	ЗАКЛЮЧЕНИЕ .....	94
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	96
	Приложение А .....	99
	Приложение Б.....	102

## ВВЕДЕНИЕ

Управление системой водоснабжения является важным элементом жизнеобеспечения городской инфраструктуры. Задачей предприятий, выполняющих эту функцию, является обслуживание водозаборных и очистных сооружений, насосных станций, фильтровальных систем, сетей водоснабжения и водоотведения. Поставка качественной питьевой воды населению, юридическим лицам, государственным органам и органам местного самоуправления, является ключевой конечной целью предприятия, выполняющего данный вид деятельности. Для качественного оказания данной услуги, необходимо постоянно следить за ключевыми параметрами:

- количественными, такими как:
  - расход воды,  $\text{м}^3/\text{ч}$ ;
  - расход электроэнергии,  $\text{кВтч/сут}$ ;
  - расход удельной электроэнергии,  $\text{кВтч/м}^3$  [1];
  - накопленный объем,  $\text{м}^3$ ;
  - перепад давления, м. вод. ст.;
  - скорость фильтрации,  $\text{м/ч}$ .
- качественными, такими как:
  - мутность воды,  $\text{мг/м}^3$ ;
  - водородный показатель, ед. рН;
  - температура воды, град. С;
  - хлор остаточный свободный,  $\text{мг/дм}^3$ ;
  - запах, балл;
  - цветность, град;
  - взвешенные вещества,  $\text{мг/дм}^3$ ;
  - хлоропоглощаемость,  $\text{мг/дм}^3$ .

ООО «Водоканал» г. Новокузнецка относится к группе предприятий жизнеобеспечения города.

Выше упомянуто, что одним из важных моментов в предоставлении услуги поставки питьевой воды, является регулярный контроль за ключевыми параметрами. Для сбора показателей по этим ключевым параметрам в Водоканале установлены различные датчики (например, хлора, мутности, уровня воды и т.д.), счетчики. Ежесуточные, среднечасовые, секундные данные с приборов учета и контроля пишутся в две базы данных *ASU-TP*, *TELEMERA-DCVS* под управлением *Microsoft SQL*. Эти базы развернуты на разных серверах.

На предприятии ООО «Водоканал» производственными цехами и отделом главного инженера используется приложение *DailyReport*. Приложение *DailyReport* обеспечивает хранения коллекции значений параметров. Параметры представляют собой показатели. Показатели могут быть обычные и вычисляемые по формулам. Каждый показатель имеет пороговое значение: не более, не менее или в диапазоне. По каждому показателю пишется набор значений – показаний.

На основе хранимых показаний, главный инженер и его подчиненных осуществляют контроль работы очистных и водозаборных цехов, отслеживают аварийные ситуации на участках сетей [2].

Приложение *DailyReport* позволяет формировать настраиваемые пользователем отчеты.

Ежедневно начальники смен водозаборных и очистных сооружений при непосредственном доступе к приборам учета осуществляют сбор данных с трех цехов в одно хранилище – базу данных *DailyReport\_output*.

Выявлены следующие недостатки неавтоматизированного ввода показаний приборов:

- трудоемкость процесса – на предприятии ежечасно три сотрудника осуществляют ввод показаний. На основании проведенных расчетов, каждый сотрудник вводит 1992 показателя в сутки. При таком количестве повышается вероятность ввода ошибочных данных.

- вычислительная ошибка – отдельные показатели считываются не на прямую с приборов, а содержат математическую формулу. В связи с большим количеством вводимых данных, высока вероятность совершения вычислительной ошибки. Такие ошибки в дальнейшем могут привести к большей погрешности итоговых показаний.
- сложность соблюдения расписания ввода показаний – невозможно точно следовать расписанию ввода показаний.

Автоматизированное преобразование данных позволяет:

- снизить вероятность вычислительной ошибки;
- сократить время получения ключевых параметров;
- осуществлять запись параметров точно по расписанию;
- оперативно доставить обработанные данные руководителю технологического отдела предприятия. Это позволит более оперативно реагировать на изменение параметров.

В результате преобразование данных позволяет объединить данные с двух баз данных *ASU-TP*, *TELEMERA-DCVS* и добавить новые вычисляемые показатели.

Целью данной работы является разработка программного модуля для преобразования данных инженерных служб для приложения *DailyReport* на языке программирования *Delphi* с использованием СУБД *Oracle* и *Microsoft SQL*.

Для достижения данной цели были поставлены следующие задачи:

- 1 изучить существующую структуру приложения *DailyReport* и схему базы данных *DailyReport\_output* под управлением *Oracle*;
- 2 изучить программную среду *Delphi XE3* для разработки программной части модуля;
- 3 изучить программную среду *PL/SQL Developer* для работы со схемой базы данных *DailyReport\_output*, а так же с хранимыми пакетами;
- 4 разработать и реализовать алгоритм преобразование данных, обеспечивающий выполнение следующих требований:

- для запуска процесса преобразования данных пользователь должен указать дату начала, дату окончания и цех предприятия;
  - результат преобразования данных должен храниться в таблице READIND\_TAB базы данных *DailyReport\_output* под управлением *Oracle* из баз данных *ASU-TP*, *TELEMERA-DCVS* под управлением *Microsoft SQL*, расположенных двух серверах;
  - импортируемые данные должны быть синхронизированы по идентификаторам с базой данных *DailyReport\_output*;
  - при импорте данных должна осуществляться проверка, если показатель на это время уже записан, то запись не производится для числовых значений;
  - обновление строковых значений показателей должно осуществляться с использованием разделителя с сохранением предыдущего значения;
  - данные должны импортироваться по запросу пользователя.
- 5 спроектировать и реализовать передачи преобразованных данных из приложения *DailyReport* серверу *Oracle*;
- 6 реализовать графический интерфейс модуля, отвечающий требованиям:
- модуль содержит окно выбора параметров преобразования данных;
  - результаты преобразования данных отображаются в табличной форме.

# 1 АРХИТЕКТУРА ПРИЛОЖЕНИЯ *DAILYREPORT*, ИСТОЧНИКИ ВХОДНЫХ ДАННЫХ, ОПИСАНИЕ РЕЗУЛЬТАТОВ ПРЕОБРАЗОВАНИЯ ДАННЫХ

## 1.1 Архитектура подключаемых модулей приложения *DailyReport*

Приложение *DailyReport* имеет модульную архитектуру. Основные модули приложения приведены в таблице 1.

Таблица 1 – Основные модули приложения *DailyReport*

Название модуля	Описание
<i>UnitLogin</i>	Модуль, реализующий процесс авторизации пользователей приложения.
<i>UnitMain</i>	Управляющий модуль. Описывает графический интерфейс главного окна приложения <i>DailyReport</i> .
<i>UnitDM</i>	Модуль предназначен для хранения основных настроек приложения.
<i>UnitStruct</i>	Модуль, реализующий обзор отчетов в табличной форме.
<i>UnitStructConst</i>	Модуль, реализующий конструктор отчетов в табличной форме.
<i>UnitReadings</i>	Модуль, реализующий отображение хранимых показаний в виде отчетов в табличной форме.
<i>UnitReport</i>	Модуль, реализующий формирование настраиваемых отчетов в форматах <i>xls</i> , <i>xltx</i> , <i>xltm</i> .
<i>Excel_TLB</i>	Внешняя библиотека, позволяющая производить работу с файлами Excel.
<i>UnitNorms</i>	Модуль настроек пороговых значений показателей.
<i>UnitSetPeriod</i>	Модуль настроек параметров.
<i>UnitSync</i>	Реализуемый модуль преобразования данных.



Архитектура приложения *DailyReport* приведена на рисунке 1 в виде диаграммы пакетов.

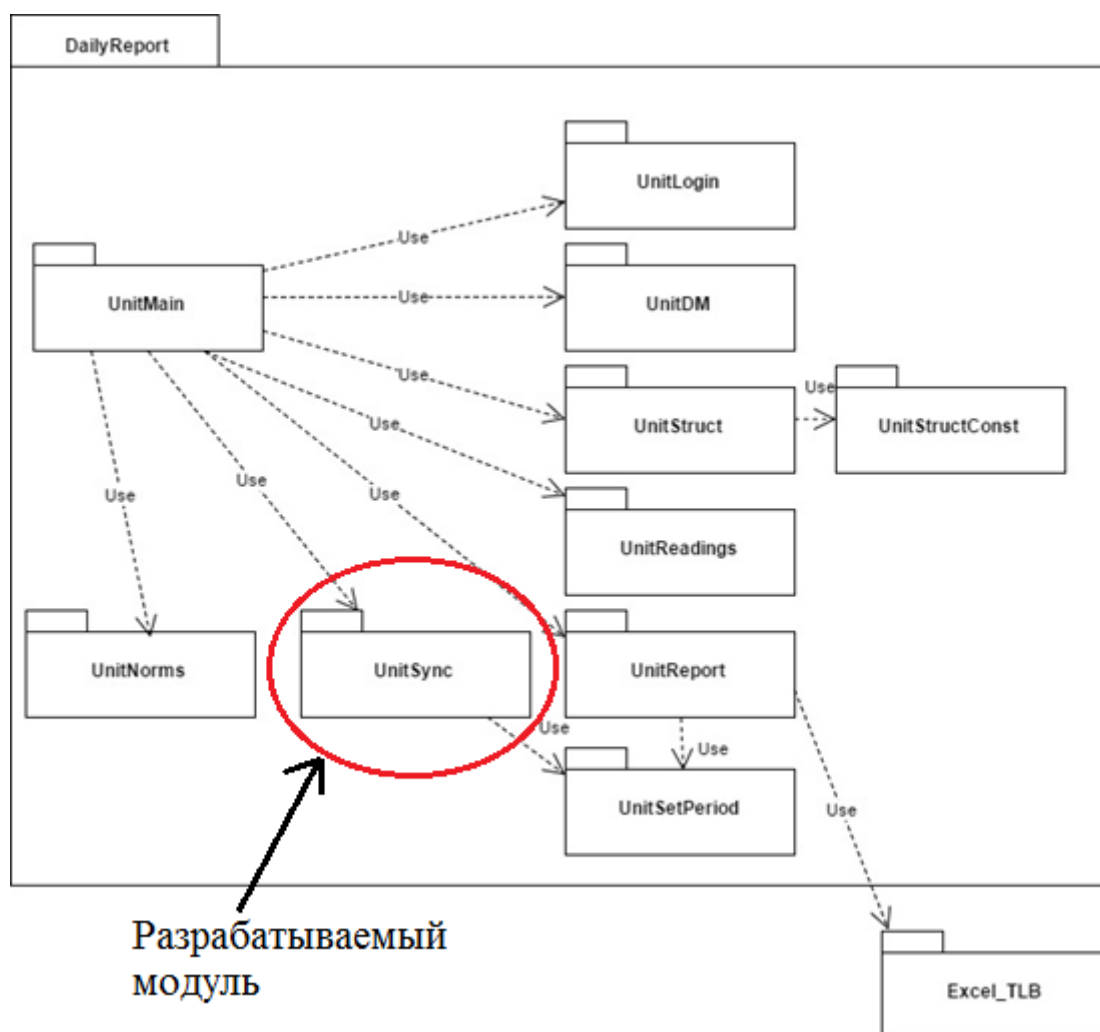


Рисунок 1 – Архитектура приложения *DailyReport*

При реализации приложения *DailyReport* были использованы дополнительно устанавливаемые компоненты *Direct Oracle Access (DOA)*. *DOA* позволяет организовать интеграцию инструмента для разработки Delphi и базы данных *Oracle*, повышая производительность в сравнении со стандартными средствами [3].

Создание нового модуля в приложении осуществляется путем наследования модуля *UnitSetPeriod*, в котором реализованы настройки. Новый модуль создан при помощи *New -> Other -> UnitSetPeriod -> Create*.

## 1.2 Источники входных данных

В качестве источников входных данных выступают базы данных *ASU-TP*, *TELEMERA-DCVS* под управлением *Microsoft SQL*. Обслуживаемые СУБД развернуты на двух серверах. *UML*-диаграмма развертывания приведена на рисунке 2.

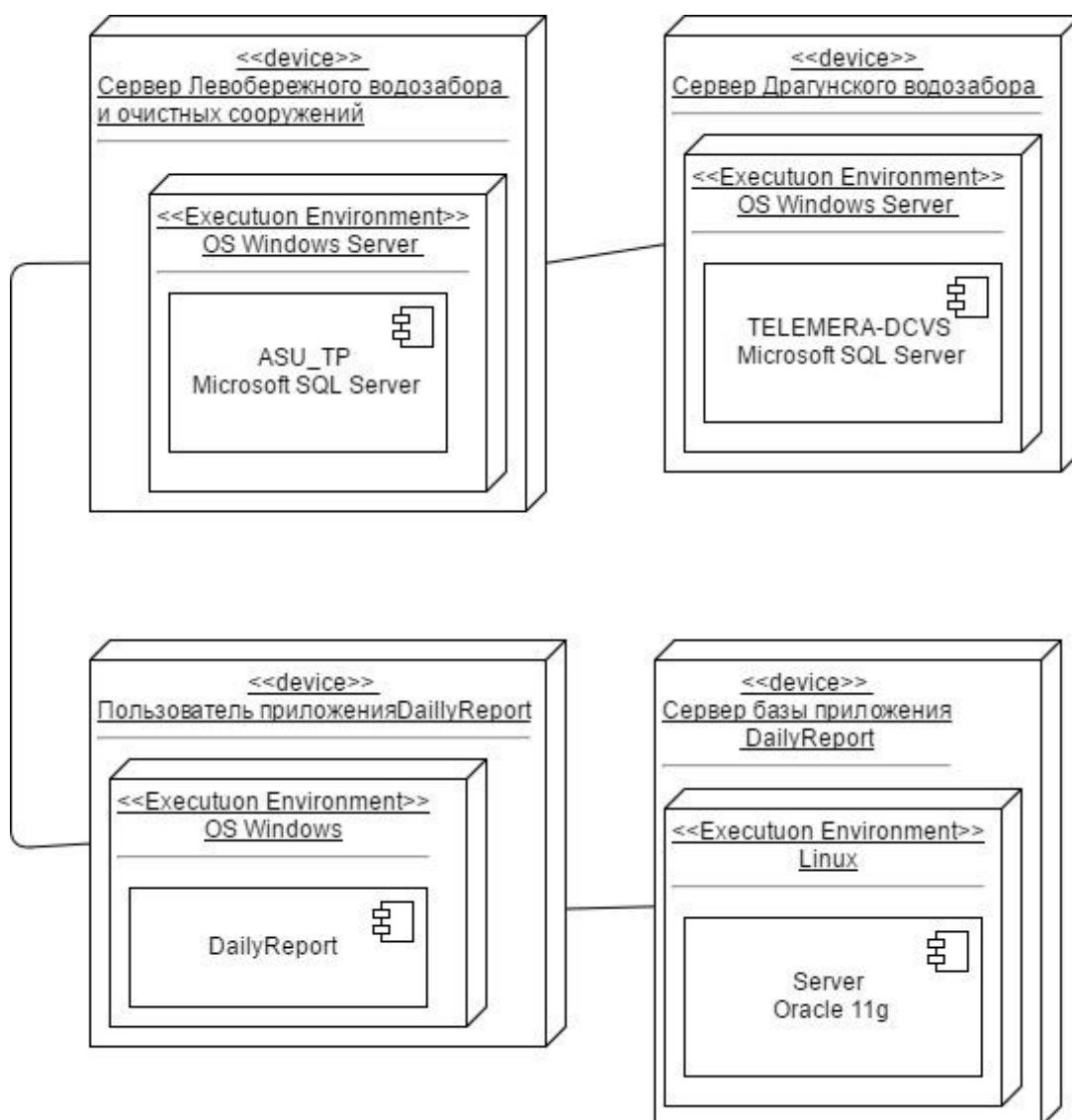


Рисунок 2 – Диаграмма развертывания

Показания с приборов учета и контроля качества записываются в базы данных *ASU-TP* и *TELEMERA-DCVS* в несколько таблиц:

- *Days* - дневные показатели;
- *Mains* - среднечасовые показатели;

- *Shorts* - секундные показатели.

Базы данных *ASU-TP* и *TELEMERA-DCVS* имеют одинаковую схему данных таблиц, задействованных в преобразовании данных за исключением атрибутов *Status* и *State* (рисунок 3). В базе *ASU-TP* используется атрибут *Status*, в *TELEMERA-DCVS* – *State*.

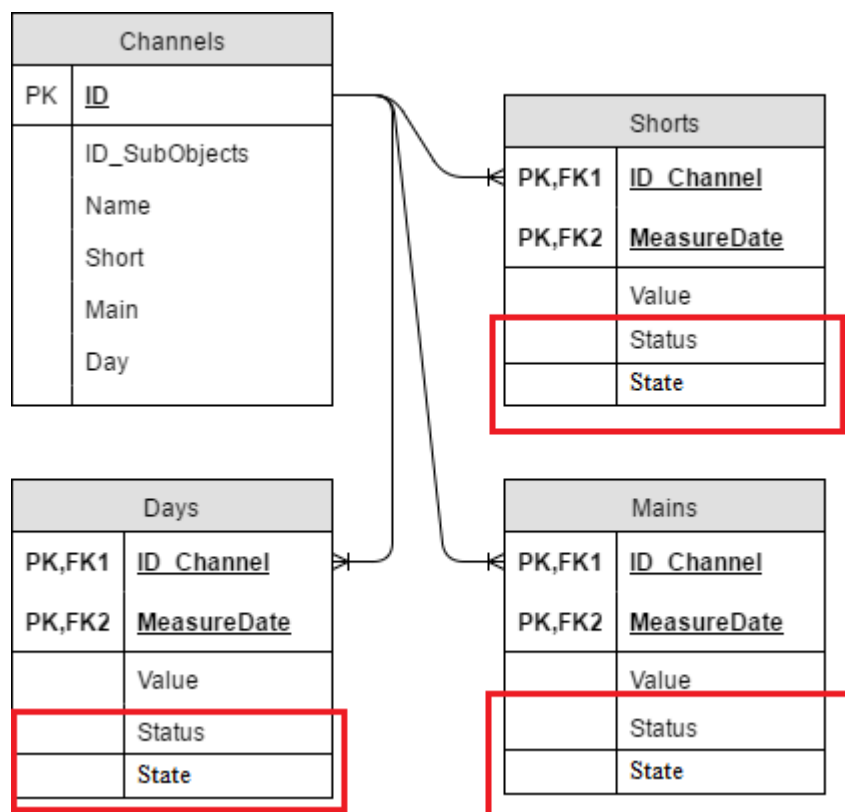


Рисунок 3 – ER-диаграмма таблиц базы данных *ASU-TP* и *TELEMERA-DCVS*, задействованных в преобразовании данных

Помимо упомянутых выше таблиц на схеме представлена таблица *Channels*, которая служит для хранения настроек процесса внесения показаний в таблицы *Days*, *Mains*, *Shorts*. Каждая запись таблицы *Channels* содержит настройки каналов данных.

1) Таблица *Channels* имеет следующие атрибуты:

- *ID* – первичный ключ. Хранит идентификатор канала данных.
- *ID\_SubObjects* - идентификатор объекта канала данных;
- *Name* – название канала данных;
- *Short* – метка о записи в таблицу *Shorts*;

- *Main* – метка о записи в таблицу *Mains*;
  - *Day* - метка о записи в таблицу *Days*.
- 2) *Days* – среднесуточные показатели по определенным каналам данных. Имеет составной первичный ключ. Содержит атрибуты:
- *ID\_Channel* – идентификатор канала данных;
  - *MeasureDate* – дата и время показателя;
  - *Value* – значение показателя;
  - *Status (State)* – статус источника данных (имеет значение 192, если прибор исправлен).
- 3) *Mains* – среднечасовые показатели. Структура таблицы *Mains* идентична таблице *Days*.
- 4) *Shorts* – короткие (ежесекундные) показатели. Структура таблицы *Shorts* идентична таблице *Days*.

### 1.3 Описание результатов преобразования данных

Результат преобразования данных должен храниться в таблице *READING\_TAB* (значения показателей) обслуживаемой СУБД *Oracle*. Структура таблицы приведена на рисунке 4.

READING_TAB	
PK	<u>READING_ID</u>
	RATE_ID
	READING_DATE
	READING_VALUE
	NOTE_TEXT
	CREATE_DATE
	EDIT_USER
	CREATE_USER
	EDIT_DATE
	SERVICE
	VALUE_SIGN
	STRING_VALUE
	ARITHMETIC

Рисунок 4 – Таблица *READING\_TAB*

В таблице *READING\_TAB* поле *READING\_ID* является первичным ключом. Так как СУБД *Oracle* не предоставляет возможности создания автоинкрементных ключей, для автоматического увеличения индекса идентификатора показания, создана последовательность *READING\_SEQ*, а так же триггер *ts\_READING\_TAB\_READING\_SEQ*, обрабатывающий событие после вставки [4].

Таблица *READING\_TAB* имеет в своей структуре следующие поля:

- *READING\_ID* (*number*) – идентификатор показания;
- *RATE\_ID* (*number*) – идентификатор родительского показателя;
- *READING\_DATE* (*date*) – дата/время показания;
- *READING\_VALUE* (*number*) – значение показания (числовое значение);
- *NOTE\_TEXT* (*varchar2(1000)*) – примечание;
- *CREATE\_DATE* (*date*) – дата создания записи;
- *EDIT\_USER* (*varchar2(100)*) – последний вносивший изменения пользователь;
- *CREATE\_USER* (*varchar2(100)*) – пользователь, создавший запись;
- *EDIT\_DATE* (*date*) – дата последнего редактирования записи;
- *SERVICE* (*varchar2(20)*) – информация об обслуживании оборудования, к которому относится показатель;
- *VALUE\_SIGN* (*varchar2(10)*) – знак значения показателя (>,<=);
- *STRING\_VALUE* (*varchar2(4000)*) - значение показания (строковое);
- *ARITHMETIC* (*varchar(10)*) – метка для работы с формулой.

Также таблица *READING\_TAB* имеет дополнительное ограничение целостности на поля *RATE\_ID* и *READING\_DATE*: для каждого показателя не может существовать две записи на одно и то же время.

В результате преобразования данных в таблице *READING\_TAB* должны появиться соответствующие записи, импортированные из баз данных *ASU-TP*, *TELEMERA-DCVS*, а также создаются записи по вычисляемым полям. Например: из базы данных *ASU-TP* хлор остаточный в правом коллекторе ле-

вобережного водозабора (идентификатор 77) на 12.02.2017 00:00 должен быть записан в таблицу *READING\_TAB* базы данных *DailyReport\_output* с идентификатором 711 на 12.02.2017 00:00, соответственно.

Как описано выше, данные из баз данных *ASU-TP*, *TELEMERA-DCVS* выгружаются из разных таблиц и с различной периодичностью. Например, показатели по реке (мутность и водородный показатель), необходимые для водозаборного цеха, необходимо выгружать каждые два часа по нечетным часам. Показатель мутности по водоводу для этого же цеха уже необходимо получать каждые два часа по четным часам. А показатель по остаточному свободному хлору необходимо получать каждый час. Для пользователя все эти процессы должны быть скрыты. В результате преобразования данных пользователь в стандартном средстве отображения отчетов в табличной форме должен увидеть записи по определенным показаниям в нужные часы (четные, не четные, каждый час или прочие условия).

#### **1.4 Требования к графическому интерфейсу**

При использовании модуля преобразования данных пользователю предоставляется следующие операции:

- импортирование данных (опция: настройка параметров);
- формирования отчета, отображающего результат преобразования данных (опция: настройка параметров).

Соответствующий функционал отражен на UML-диаграмме вариантов использования, изображенной на рисунке 5.

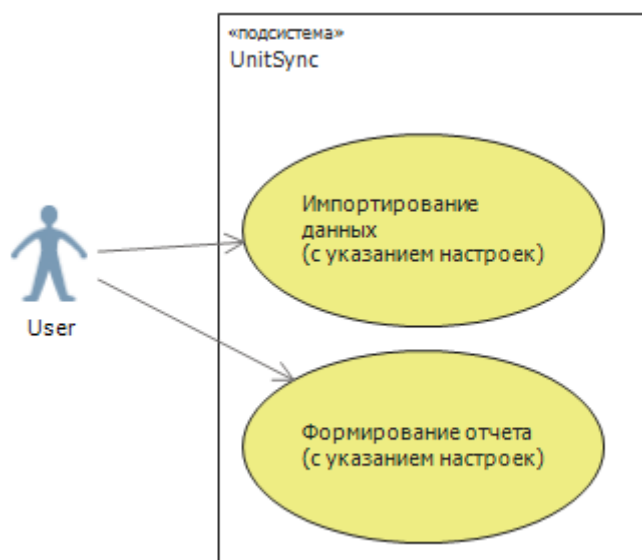


Рисунок 5 – UML-диаграмма вариантов использования

Преобразование данных должно осуществляться по нажатию на кнопку, как приведено в макете на рисунке 6.

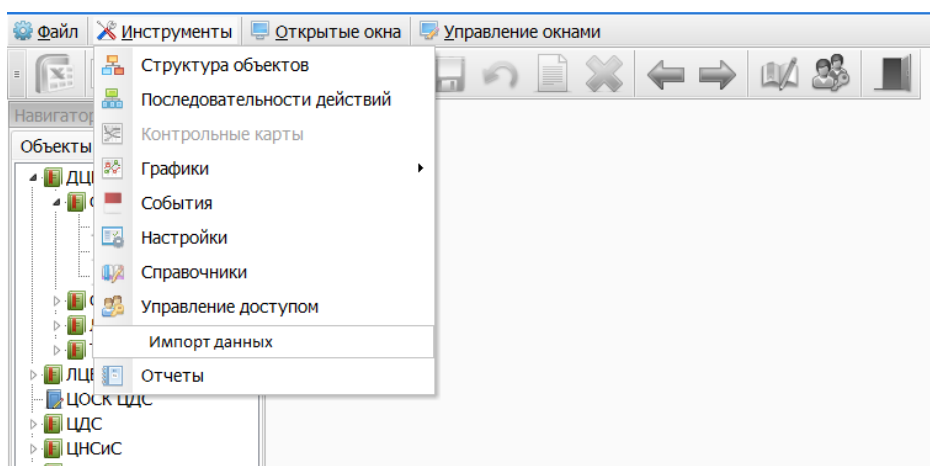


Рисунок 6 – Макет кнопки импорта данных

При нажатии на кнопку открывается форма настроек параметров импорта данных. Макет формы изображен на рисунке 7.

Рисунок 7 – Макет настроек параметров преобразования данных

В результате преобразования, данные должны отображаться при помощи стандартного средства отображения отчетов в табличной форме приложения *DailyReport*. Макет формы отчета изображен на рисунке 8.

Дата/время	Показатели			
		Показания		

Рисунок 8 – Макет окна стандартного средства отображения отчетов в табличной форме приложения *DailyReport*

Заголовки стандартного средства отображения отчетов в табличной форме приложения *DailyReport* формируются посредством показателей. Каждому показателю присваивается свой уникальный код.

Данные для отображения отчета в табличной форме хранятся на сервере *Oracle* в таблице *READING\_TAB*.



## 2 ПРОЦЕСС ПРЕОБРАЗОВАНИЯ ДАННЫХ

Процесс преобразования данных можно осуществить двумя основными способами:

- 1) с использованием гетерогенного сервиса;
- 2) с использованием отдельного программного приложения.

Гетерогенный сервис – компонент, интегрированный в базу данных, который предоставляет осуществлять запросы к сторонним СУБД [5].

Настройка гетерогенного сервиса является оптимальным вариантом для осуществления процесса преобразования данных. Однако, ввиду ограничений, накладываемых политиками безопасности предприятия, создание гетерогенного сервиса является невозможным. Поэтому, процесс преобразования данных реализуется, как модуль программного приложения *DailyReport*.

Процесс преобразования данных изображен на рисунке 9.

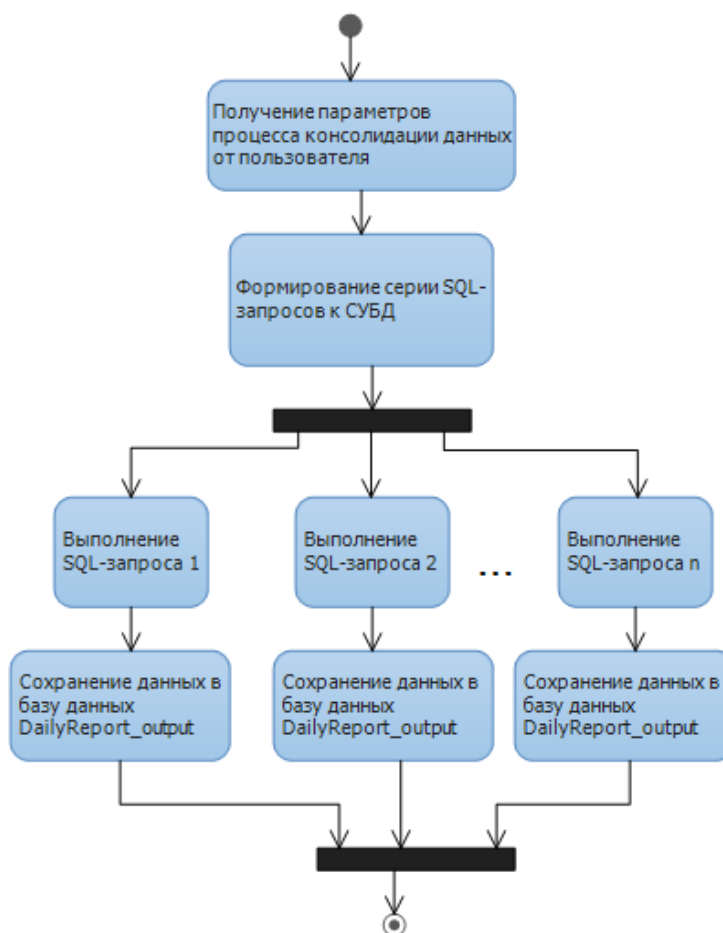


Рисунок 9 – Процесс преобразования данных

Таблица 2 – Входные и выходные данные подпроцессов

Подпроцесс	Входные данные	Выходные данные
Получение параметров процесса преобразования данных от пользователя	— вызов модуля преобразования данных из управляющего; — логин пользователя.	— название цеха; — временной диапазон.
Формирование серии запросов к СУБД	— название цеха.	— коллекция запросов типа <i>TSQLDataSet</i> [6].
Выполнение запроса	— <i>sql</i> -запрос; — временной диапазон.	— коллекция показаний типа <i>TOra-cleDataSet</i> [7].
Сохранение данных в базу данных <i>DailyReport_output</i>	— логин пользователя; — идентификатор показателя; — массив дат; — массив показаний.	

Выполнение процесса преобразования данных обеспечивается серией *SQL*-запросов. Сформированные *SQL*-запросы позволяют получать серию записей в таблице *READING\_TAB*. Значение вычислительных показателей формируется в *SQL*-запросах.

В соответствии с процессом, приведенным на рисунке 9, проектирование и реализация процесса преобразования включает в себя следующие стадии:

- проектирование и реализация пользовательского графического интерфейса (подпроцесс: получение параметров процесса преобразования данных от пользователя);
- формирование *SQL*-запросов на выборку данных из баз данных *ASU-TP* и *TELEMERA-DCVS* и реализация хранения *SQL*-запросов (подпроцесс: формирование серии *SQL*-запросов к СУБД под управлением *Microsoft SQL*);

- проектирование и реализация установки соответствий импортируемых показателей и показателей приложения *Dailyreport* (подпроцесс: выполнение *SQL*-запроса);
- проектирование и реализация алгоритма получения запросов на выборку данных по подразделению предприятия и передачи данных серверу *Oracle* (подпроцесс: выполнение *SQL*-запроса);
- проектирование и реализация алгоритма обработки полученных данных сервером *Oracle*, записи данных (подпроцесс: сохранение данных в базу данных *DailyReport\_output*).

## **2.1 Получение параметров от пользователя**

### **2.1.1 Проектирование и реализация пользовательского графического интерфейса**

Пользовательский интерфейс должен соответствовать макету настроек параметров преобразования данных (рисунок 7).

Для реализации графического интерфейса создан модуль *UnitSyncParam* на основании модуля *UnitSetPeriod*. Модуль *UnitSetPeriod* представлен на рисунке 10. Данный модуль выступает базовым, так как в нем уже реализована настройка временного периода [8].

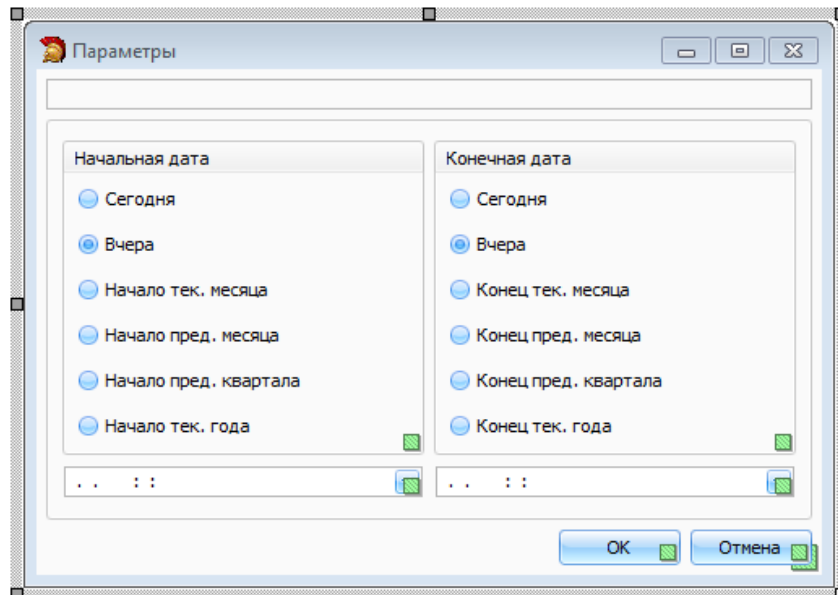


Рисунок 10 – Графический интерфейс модуля настроек параметров  
*UnitSetPeriod*

Модуль *UnitSetPeriod* имеет следующие методы:

- *constructor Create(AName: string; ABeginDate, AEndDate: TDateTime)* – конструктор. Имеет три входных параметра:
  - *AName: string* – заголовок;
  - *ABeginDate: TDateTime* – дата начала по умолчанию;
  - *AEndDate: TDateTime* – дата окончания по умолчанию.
- *procedure rgBDatePresetPropertiesEditValueChanged (Sender: TObject)* – процедура установки даты начала (в зависимости от выбранного параметра). Имеет один входной параметр *Sender: TObject*;
- *procedure rgEDatePresetPropertiesEditValueChanged (Sender: TObject)* – процедура установки даты окончания (в зависимости от выбранного параметра). Имеет один входной параметр *Sender: TObject*;
- *function GetPrevQuarterEndDate(ADate: TDateTime): TDateTime* – функция получения даты конца предыдущего квартала. Имеет один входной параметр *ADate: TDateTime*. Возвращает дату окончания предыдущего квартала;

- *function GetPrevQuarterStartDate(ADate: TDateTime): TDateTime* – функция получения даты начала предыдущего квартала. Имеет один входной параметр *ADate: TDateTime*. Возвращает дату начала предыдущего квартала;
- *procedure FormKeyDown(Sender: TObject; var Key: Word; Shift: TShiftState)* – процедура обработки нажатия на клавишу. При нажатии на клавишу *Enter* вызывает событие нажатия кнопки «OK»;
- *procedure btnOkClick(Sender: TObject)* – процедура, описывающая событие нажатие на кнопку «OK». В процедуре так же осуществляется проверка на корректность введенных дат.

Спроектированный интерфейс модуля-наследника *UnitSyncParam* изображен на рисунке 11.

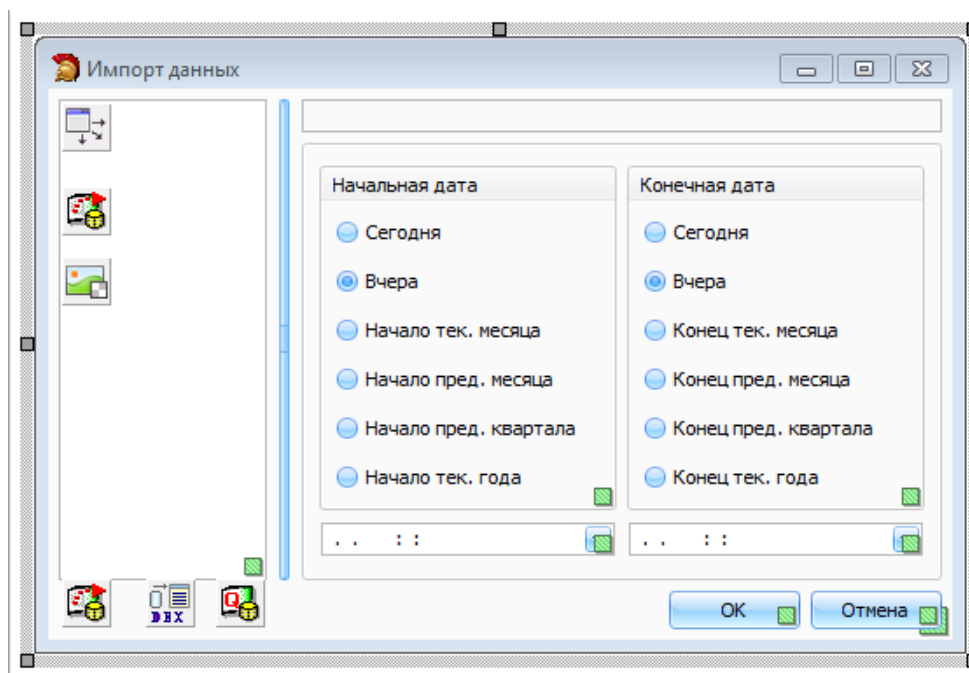


Рисунок 11 – Графический интерфейс модуля настроек параметров преобразования данных *UnitSyncParam*

Форма базового модуля *UnitSetPeriod* расширена компонентом *TreeObjList: TdxDBTreeView*, представляющий дерево объектов для отображения цепочек, по которым осуществляется преобразования данных.

Заполнение компонента *TreeObjList* осуществляется при помощи компонента *odsObjectList: TOracleDataSet*, возвращающего список цехов.

Модуль *UnitSyncParam* реализует следующие методы:

- *constructor Create* – конструктор. Реализует конструктор базового класса. Расширен заполнением дерева объектов.
- *procedure TreeObjListChange(Sender: TObject; Node: TTreeNode)* – метод, описывающий событие выбора цеха в дереве объектов. При выборе цеха в дереве объектов, осуществляется заполнения заголовка для преобразования данных;
- *procedure TreeObjListDbClick(Sender: TObject)* – метод, описывающий событие двойного нажатия по названию цеха в дереве объектов. При двойном нажатии осуществляется проверка корректности выбранного дерева объектов, затем вызывается метод, описывающий событие нажатия на кнопку «ОК».

Получение списка цехов для заполнения дерева объектов, осуществляется посредством запроса к серверу *Oracle*. Для удобства обслуживания и расширения системы, на сервере *Oracle* создан пакет *SYNC\_PKG*, в котором хранятся все процедуры и функции, обрабатываемые на сервере *Oracle* и используемые для процесса преобразования данных.

Реализована процедура *GetQuerySQL(p\_data in out sys\_refcursor, p\_parent varchar2, p\_level number default 1)* для получения списка цехов. Эта функция хранится в пакете *SYNC\_PKG* на сервере *Oracle*. Процедура *GetQuerySQL* имеет три параметра:

- *p\_data in out sys\_refcursor* – курсор. Может быть как входным, так и выходным параметром.
- *p\_parent varchar2* – идентификатор объекта.
- *p\_level number default 1* – уровень. Параметр необходим для определения уровня в иерархических запросах. По умолчанию принимает значение 1.

## 2.2 Формирование серии *SQL*-запросов к СУБД

### 2.2.1 Организация получения необходимых данных из одной базы данных посредством представлений

Базы данных *ASU-TP* и *TELEMERA-DCVS* располагаются на разных серверах. Получение данных из этих баз можно осуществить двумя способами:

- поочередная загрузка данных с каждого сервера баз данных;
- подключение к одной базе данных, в которой созданы специальные представления для получения данных из другой базы данных.

Скорость подключения и надежность соединения сервера базы данных *ASU-TP* выше. В связи с этим, целесообразно в базе данных *ASU-TP* создать представления, получающие необходимые данные из базы данных *TELEMERA-DCVS*. Благодаря такому решению необходимые данные из базы данных *TELEMERA-DCVS* можно получить непосредственно через представления.

В базе данных *ASU-TP* созданы представления:

- *days\_dcvs* – среднесуточные показатели;
- *mains\_dcvs* – среднечасовые показатели;
- *shorts\_dcvs* – мгновенные показатели.

### 2.2.2 Формирование *SQL*-запросов на выборку данных из баз данных *ASU-TP* и *TELEMERA-DCVS*

Процесс преобразования данных должен осуществляться в соответствии в настроенными параметрами:

- по определенному цеху;
- за определенный период.

Для каждого цеха и показателя заданы разные условия. Можно объединить показатели по цехам и предъявляемым к ним условиям с целью форми-

рования запросов для получения данных из баз данных *ASU-TP* и *TELEMERA-DCVS* с последующей выгрузкой полученных данных в базу данных *DailyReport\_output*.

Характеристики исходных данных с детализацией по цехам, представляющие условия для формирования *SQL*-запросов приведены в таблице 3.

Таблица 3 – Характеристики исходных данных с детализацией по цехам

Цех	База данных	Таблица	Описание условия	Количество показателей в запросе
ЦДС	ASU-TP	<i>Shorts</i>	Показания выгружаются раз в два часа по нечетным часам.	36
		<i>Days</i>	Среднесуточные показания на 00:00 часов.	1
		<i>Mains</i>	Расчет суточного расхода: от показаний счетчиков за текущее число на 00:00 часов отнимаются показания счетчиков за предыдущее число на 00:00 часов.	15
ЦОСК	ASU-TP	<i>Mains</i>	Показания выгружаются раз в час.	23
ДЦВС	TELEMERA-DCVS	<i>Shorts</i>	Среднесуточные показания на 9:00 часов.	3
		<i>Mains</i>	Среднесуточные показания на 00:00 часов.	14
		<i>Shorts</i>	Показания выгружаются раз в два часа по нечетным часам.	37
ДЦВС	TELEMERA-DCVS	<i>Shorts</i>	Показания выгружаются раз в час.	24
		<i>Shorts</i>	Показания выгружаются раз в два часа по четным часам.	1
		<i>Shorts</i>	Показания выгружаются по определенным часам: 1:00, 5:00, 9:00, 13:00, 17:00, 21:00.	16
	ASU-TP	<i>Mains</i>	Среднесуточные показания на 00:00 часов.	1
		<i>Shorts</i>	Показания выгружаются раз в два часа по нечетным часам.	4
ЛЦВС	ASU-TP	<i>Mains</i>	Показания выгружаются раз в час.	10
		<i>Shorts</i>	Подсчет накопленных среднечасовых значений (берутся значения за каждую минуту и высчитывается по ним среднее за час).	6
		<i>Mains</i>	Мгновенные значения показаний по определенным часам: 3:00, 9:00, 12:00, 21:00.	6
		<i>Mains</i>	Показания выгружаются раз в два часа по нечетным часам.	12



### 2.2.3 Хранение *SQL*-запросов для выборки данных из баз данных ASU-TP и TELEMERA-DCVS под управлением *Microsoft SQL*

Каждой категории данных, представленных в таблице 2 должен соответствовать *SQL*-запрос, осуществляющий выборку соответствующих данных. Хранение запросов на выборку данных целесообразно производить на сервере *Oracle* в специальной таблице. Это позволит при необходимости быстро расширить импортируемые показатели.

Для хранения запросов на получение данных, в базе данных *DailyReport\_output* создана таблица *SQL\_TAB* (рисунок 12).

SQL_TAB	
PK	<u>ID_QUERY</u>
	ID_PARENT_ NAME_ SQL_ NOTE_ ORDER_ FROM_

Рисунок 12 – Структура таблицы для хранения запросов *SQL\_TAB*

Таблица *SQL\_TAB* имеет следующие атрибуты:

- *ID\_QUERY\_* (*varchar2*(20)) – идентификатор запроса/объекта;
- *ID\_PARENT\_* (*varchar2*(20)) – идентификатор родительского объекта;
- *NAME\_* (*varchar2*(100)) – название цеха;
- *SQL\_* (*varchar2*(4000)) – хранимый запрос;
- *NOTE\_* (*varchar2*(100)) – примечание к запросу;
- *ORDER\_* (*number*(2)) – порядок (для цехов);
- *FROM\_* (*varchar2*(10)) – из какой базы данных производится выгрузка.

Структура таблицы *SQL\_TAB* построена таким образом, что позволяет хранить и заголовки цехов и запросы для каждого цеха. Такая структура таб-

лицы позволяет легко расширять как показатели для импорта, так и запросы, которые относятся к ним [9].

## 2.3 Выполнение серии SQL-запросов

### 2.3.1 Проектирование и реализация сопоставления идентификаторов показателей входных и выходных данных

В процессе преобразования данных одним из важнейших пунктов является установка соответствия между показателями из баз данных, предоставляющих данные (*ASU-TP* и *TELEMERA-DCVS*) и показателями приложения *DailyReport*. Установка соответствий позволит обеспечить, корректность и точность импортируемых данных, исключит вероятность того, что в показателях приложения *DailyReport* появятся посторонние значения.

При установке соответствий важно учесть, что получение данных осуществляется из разных баз данных. Таким образом, появляется вероятность, что в разных базах данных, предоставляющих данные, может быть один номер идентификатора. Однако, в приложении *DailyReport* это два разных показателя.

Для установки соответствий при преобразовании данных оптимальным решением является таблица соответствий. Это решение позволит с легкостью расширять количество показателей для преобразования данных.

В базе данных *DailyReport\_output* для связи данных с другими базами данных, предоставляющими данные, создана таблица *SYNC\_TAB* (рисунок 13).

SYNC_TAB	
PK,FK1	<u>DB_ID_RATE</u>
PK,FK2	<u>DR_ID_RATE</u>
	SYNC_DATE_ DB_NAME_RATE_ UPDATE_ USER_LOGIN_ FROM_ HAR_

Рисунок 13 –Таблица синхронизации показателей *SYNC\_TAB*

Таблица *SYNC\_TAB* имеет следующие атрибуты:

- *DB\_ID\_RATE* (*number*) – атрибут составного первичного ключа. Хранит идентификатор показателя в базе данных, предоставляющей данные;
- *DR\_ID\_RATE* (*number*) – атрибут составного первичного ключа. Хранит идентификатор показателя базы данных *DailyReport\_output*;
- *SYNC\_DATE\_* (*date*) – дата/время последнего импорта данных по текущему показателю;
- *DB\_NAME\_RATE* (*varchar2(200)*) – наименование показателя;
- *UPDATE\_* (*number*) – метка: осуществляется ли перезапись значения по текущему показателю, если оно уже имеется;
- *USER\_LOGIN\_* (*varchar2(20)*) – логин пользователя, осуществившего импорт данных;
- *FROM\_* (*varchar2(10)*) – из какой базы данных производится импорт данных;
- *HAR\_* (*varchar2(10)*) – характеристики значения: *CURDAY* – на текущий день, *PREVDAY* – на предыдущий день; *NEXTDAY* – на следующий день.

Сопоставления во вспомогательной таблице вносятся администратором.

## 2.3.2 Проектирование и реализация алгоритма получения запросов на выборку данных по цеху и передачи данных серверу Oracle

Блок-схема алгоритма получения запросов на выборку данных по цеху и передачи данных серверу Oracle изображена на рисунке 14.

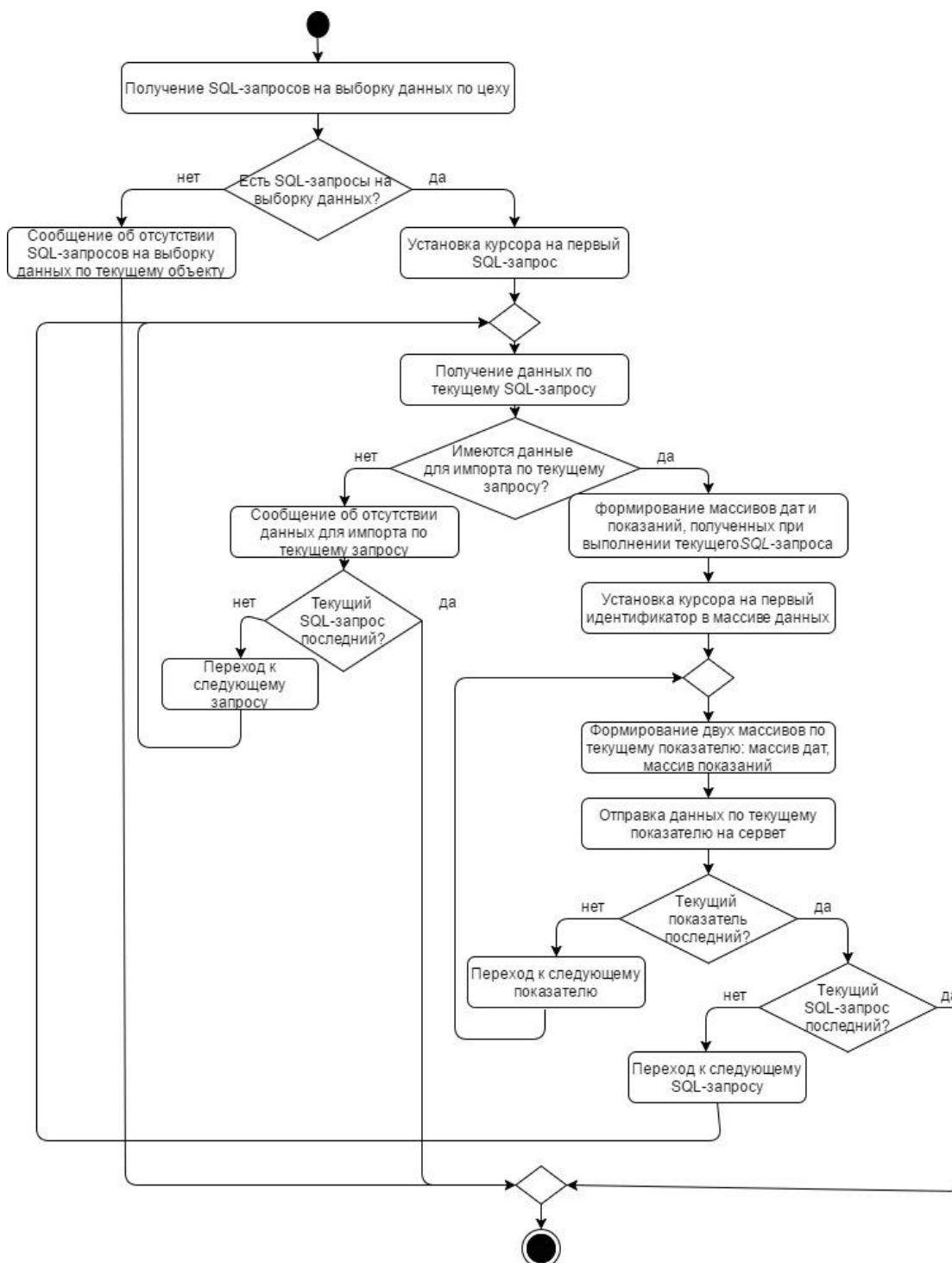


Рисунок 14 – Блок-схема алгоритма получения запросов на выборку данных по цеху и передачи данных серверу Oracle

Алгоритм получения запросов на выборку данных по цеху и передачи данных серверу *Oracle* реализован в модуле *UnitSyncParam* в методах:

- *procedure btnOkClick(Sender: TObject)* – процедура реализует событие нажатия на кнопку «ОК». Имеет один входной параметр. В процедуре происходит вызов метода *GetData(string, string)*. Процедура является входной точкой для алгоритма, представленного на рисунке 14
- *procedure GetData(sql,nameSQL: string)* – в процедуре осуществляется получение *SQL*-запросов на выборку данных по цеху, а так же проверка на наличие *SQL*-запросов по выбранному цеху. В случае отсутствия, информирует пользователя об этом. Процедура имеет два входных параметра:

— *sql: string* - *sql*-запрос;

— *nameSQL: string* - название *sql*-запроса.

Так же в процедуре *GetData* происходит вызов методов:

— *GetValues(var TDateArr; var TValuesArr);*

— *GetValue(integer; TDateArr; var TDateArr; TValuesArr; var TValueArr);*

— *ImportData(Integer; TDateArr; TValueArr; String, String).*

- *procedure GetValues(var ADates: TDateArr; var AValues: TValuesArr)* - в процедуре осуществляется получение данных из баз данных *ASU-TP* и *TELEMERA-DCVS*, формирование массивов дат и показаний, полученных при выполнении текущего *SQL*-запроса. Процедура имеет два входных параметра:

— *var ADates: TDateArr* - массив дат. Тип данных *TDateArr* представляет одномерный массив данных типа *TDateTime*;

— *var AValues: TValuesArr* - массив значений. Тип данных *TValuesArr* представляет двумерный массив данных типа *String*.

- *procedure GetValue(Aix: integer; ADates: TDateArr; var ADate: TDateArr; AValues: TValuesArr; var AValue: TValueArr)* – в процедуре осуществляется формирование двух одномерных массивов по текущему показате-

лю: массив дат и массив показаний. Процедура имеет пять входных параметров:

- *Aix: integer* – порядковый номер показателя;
- *ADates: TDateArr* – массив дат, полученный из исходных данных, полученных при выполнении запроса;
- *var ADate: TDateArr* – массив дат, формируемый непосредственно для набора показаний по текущему показателю;
- *AValues: TValuesArr* - массив показаний, полученный из исходных данных, полученных при выполнении запроса;
- *var AValue: TValueArr* - массив показаний, формируемый непосредственно для набора показаний по текущему показателю. Тип данных *TValueArr* представляет одномерный массив данных типа *String*.

- *procedure ImportData(Aid: Integer; ADates: TDateArr; AValue: TValueArr; ALogin, AFrom: String)* – в процедуре осуществляется отправка готового набора данных (идентификатор показания, одномерный массив дат и одномерный массив значений, логин пользователя, название базы данных) по текущему показателю на сервер *Oracle*. Процедура имеет пять входных параметра:

- *Aid: Integer* – идентификатор показателя;
- *ADates: TDateArr* – одномерный массив по текущему показателю;
- *AValue: TValueArr* – одномерный массив показаний по текущему показателю;
- *ALogin: String* – логин пользователя, осуществляющего импорт;
- *AFrom: String* – название базы данных, из которой получены данные.

Для реализации алгоритма получения запросов на выборку данных и передачи данных серверу *Oracle* были использованы следующий компоненты:

- *odsSQLList: TOracleDataSet* – осуществляет получение всех запросов по текущему показателю из сервера *Oracle*;
- *dsImportData: TSQLDataSet* – осуществляет получение данных по запросу из баз данных на сервере *Microsoft SQL* [6];
- *oqData: TOracleQuery* – осуществляет передачу данных серверу *Oracle*.  
Выполнение запросов и передача данных серверу *Oracle* осуществляется последовательно.

Получение списка запросов по параметру осуществляется вызовом процедуры *GetQuerySQL* (*sys\_refcursor*, *varchar2*, *number*) с передачей идентификатора цеха и уровня, равного больше единицы.

## **2.4 Сохранение данных в базу данных *DailyReport\_output***

### **2.4.1 Проектирование и реализация алгоритма обработки полученных данных сервером *Oracle* и записи данных**

Блок-схема алгоритма обработки полученных данных сервером *Oracle* и записи данных изображена на рисунке 15.

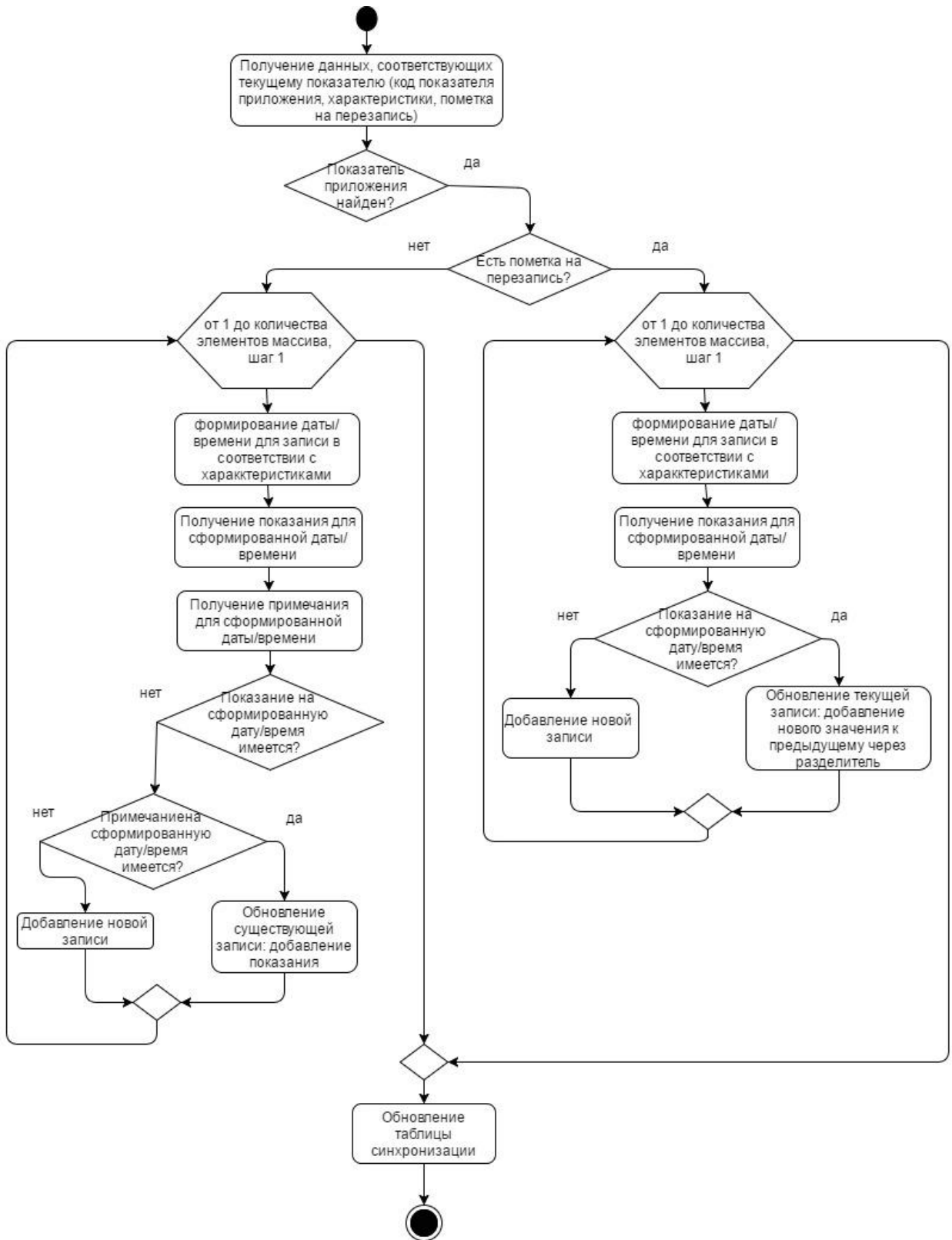


Рисунок 15 – Блок-схема реализации алгоритма обработки полученных данных сервером *Oracle* и записи данных



Для реализации алгоритма обработки полученных данных сервером *Oracle* и последующей записи данных, в пакете *SYNC\_PKG* на сервере *Oracle* используется набор пользовательских типов данных [10]:

- *TDate* – таблица дат, индексируется типом *binary\_integer*;
- *TFloat* – таблица значений вещественного типа *float*, индексируется типом *binary\_integer*;
- *c\_rate* – строка для записи показания числового значения. Содержит атрибуты:
  - *dr\_id\_rate\_sync\_tab.dr\_id\_rate\_%type* – код показания приложения *DailyReport*;
  - *from\_sync\_tab.from\_%type* – название базы данных, из которой осуществляется импорт данных;
  - *har\_sync\_tab.har\_%type* – характеристики показателя;
  - *update\_sync\_tab.update\_%type* – метка, осуществляется ли обновление;
  - *date\_sync\_tab.sync\_date\_%type* – дата синхронизации;
  - *value\_number* – числовое значение;
  - *note\_reading\_tab.note\_text%type* – примечание.
- *c\_rate\_vvarchar* – строка для записи показаний строкового значения. Содержит атрибуты:
  - *dr\_id\_rate\_sync\_tab.dr\_id\_rate\_%type* – код показания приложения *DailyReport*;
  - *from\_sync\_tab.from\_%type* – название базы данных, из которой осуществляется импорт данных;
  - *har\_sync\_tab.har\_%type* – характеристики показателя;
  - *update\_sync\_tab.update\_%type* – метка, осуществляется ли обновление;
  - *date\_sync\_tab.sync\_date\_%type* – дата синхронизации;
  - *value\_vvarchar2(20)* – строковое значение;

— *note\_reading\_tab.note\_text%type* – примечание.

Также для реализации алгоритма обработки полученных данных сервером *Oracle* и последующей записи данных, в пакете *SYNC\_PKG* на сервере *Oracle* добавлены следующие методы:

- *procedure ImportData(p\_db\_id\_rate in sync\_tab.dr\_id\_rate\_%type, p\_dates in TDate, p\_values in TFloat, p\_login in user\_tab.user\_login%type, p\_from in sql\_tab.from\_%type)* – процедура, в которую осуществляется передача данных для преобразования из приложения. В процедуре осуществляется получение данных по полученному показателю путем вызова функции *GetRate*. Далее проверяет наличие показателя в базе *DailyReport\_output*, соответствующего входному параметру *p\_db\_id\_rate*. Затем в полученных данных анализируется значение пометки на перезапись. В зависимости от этого значения вызывается метод, реализующий запись показания числового (*ImportData*) или строкового (*ImportDataVarchar*) типа. Процедура имеет пять входных параметров:
  - *p\_db\_id\_rate in sync\_tab.dr\_id\_rate\_%type* – идентификатор показателя базы данных, предоставляющей данные;
  - *p\_dates in TDate* – массив дат;
  - *p\_values in TFloat* – массив показаний по текущему показателю;
  - *p\_login in user\_tab.user\_login%type* – логин пользователя, запустившего процесс преобразования данных;
  - *p\_from in sql\_tab.from\_%type* – название базы данных, предоставляющей данные для преобразования.
- *function GetRate(p\_db\_rate\_id in sync\_tab.db\_id\_rate\_%type, p\_from in sync\_tab.from\_%type)* - функция получения данных по полученному показателю (код показателя приложения *DailyReport*, характеристики, пометка на перезапись). Функция возвращает результат типа *s\_rate*. Имеет два входных параметра:
  - *p\_db\_rate\_id in sync\_tab.db\_id\_rate\_%type* – идентификатор показателя из базы данных, предоставляющей данные;

— *p\_from in sync\_tab.from\_%type* – название базы данных, предоставляющей данные.

- *procedure ImportData(p\_meas\_date in reading\_tab.reading\_date%type, p\_meas\_value in reading\_tab.reading\_value%type, p\_login in user\_tab.user\_login%type, p\_rate in out c\_rate)* – процедура осуществляет создание/обновление записей в таблице *READING\_TAB* с показани-ями числового типа. В процедуре осуществляется формирование да-ты/времени в соответствии с характеристиками для записи (вызов ме-тода *GetDate*). Затем осуществляется попытка получения показания и примечаний к нему по заданному времени и показателю (вызов метода *GetRateValue*). При отсутствии такого показателя в базе *DailyRe- port\_output* и примечания по нему, осуществляется добавление новой строки в таблицу *READING\_TAB*. При отсутствии показателя, но нали-чии комментария по нему, осуществляется обновление существующей строки в таблице *READING\_TAB*. Процедура *ImportData* имеет четыре параметра:

— *p\_meas\_date in reading\_tab.reading\_date%type* – дата;

— *p\_meas\_value in reading\_tab.reading\_value%type* – показание;

— *p\_login in user\_tab.user\_login%type* – логин пользователя;

— *p\_rate in out c\_rate* – строка данных для записи числового значе-ния.

- *function GetDate(p\_date in date, p\_har in sync\_tab.har\_%type)* – функция формирования даты для записи. Возвращает дату. Имеет два входных параметра:

— *p\_date in date* – дата;

— *p\_har in sync\_tab.har\_%type* – характеристика: *CURDAY* – текущая дата, *PREVDAY* – предыдущий день, *NEXTDAY* – следующий день.

- *function GetRateValue (p\_id\_rate in reading\_tab.reading\_value%type, p\_date in reading\_tab.reading\_date%type)* – функция получения показа-

ния числового типа по дате/времени и показателю. Возвращает значение числового типа данных. Имеет два входных параметра:

- *p\_id\_rate in reading\_tab.reading\_value%type* – идентификатор показателя приложения *DailyReport*;
- *p\_date in reading\_tab.reading\_date%type* – дата/время.

- *procedure ImportDataVarchar (p\_meas\_date in reading\_tab.reading\_date%type, p\_meas\_value in reading\_tab.reading\_value%type, p\_login in user\_tab.user\_login%type, p\_rate\_varchar in out c\_rate\_varchar)* - процедура осуществляет создание/обновление записей в таблице *READING\_TAB* с показаниями строкового типа. В процедуре осуществляется формирование даты/времени в соответствии с характеристиками для записи (вызов метода *GetDate*). Затем осуществляется попытка получения показания по заданному времени и показателю (вызов метода *GetRateValueVarchar*). При отсутствии такого показателя в базе *DailyReport\_output* осуществляется добавление новой строки в таблицу *READING\_TAB*. Если показатель имеется, осуществляется обновление существующей строки в таблице *READING\_TAB* с записью нового значения через разделитель. Имеет четыре входных параметра:

- *p\_meas\_date in reading\_tab.reading\_date%type* – дата;
- *p\_meas\_value in reading\_tab.reading\_value%type* – показание;
- *p\_login in user\_tab.user\_login%type* – логин пользователя;
- *p\_rate\_varchar in out c\_rate\_varchar* – строка данных для записи строкового значения.

- *function GetRateValueVarchar (p\_id\_rate in reading\_tab.reading\_value%type, p\_date in reading\_tab.reading\_date%type)* - функция получения строкового значения по дате/времени и показателю. Возвращает значение строкового типа данных. Имеет два входных параметра:

- *p\_id\_rate in reading\_tab.reading\_value%type* - идентификатор показателя приложения *DailyReport*;
- *p\_date in reading\_tab.reading\_date%type* – дата/время.

### **3 ВЫБОР И ТЕСТИРОВАНИЕ АЛГОРИТМОВ ПЕРЕДАЧИ ПРЕОБРАЗОВАННЫХ ДАННЫХ СЕРВЕРУ ORACLE**

Одним из важных критериев в реализации процесса преобразования данных является время ожидания пользователем. При выполнении процесса преобразования осуществляется импорт большого количества данных, что также значительно влияет на время ожидания. Поэтому, при реализации важно учесть все возможные факторы и получить быстродействующий алгоритм.

Одним из важных этапов является передача данных, полученных при выполнении запроса, в базу данных *DailyReport\_output*.

#### **3.1 Алгоритмы передачи данных серверу Oracle**

После проведения анализа, определено три основных алгоритма передачи данных серверу *Oracle*:

- алгоритм на основе вставок кортежей – в базу данных передается отдельно каждый набор данных со значениями показателей. Блок-схема алгоритма приведена на рисунке 16;
- алгоритм на основе передачи одного массива - передача данных в базу данных осуществляется одномерным массивом типа *String*. Набор данных для каждого показания записан в одну ячейку через разделитель. Блок-схема алгоритма приведена на рисунке 17. Алгоритм требует дополнительного преобразования данных СУБД, поэтому обработка алгоритма осуществляется двумя программными компонентами: клиентским приложением и сервером Oracle;
- алгоритм на основе передачи отдельных массивов данных - передача данных в базу данных осуществляется тремя одномерными массивами: массив целочисленных значений, массив дат, массив вещественных чисел. Каждый *i*-ый элемент массивов соответствует набору данных. Блок-схема алгоритма приведена на рисунке 18. Алгоритм требует до-

полнительного преобразования данных СУБД, поэтому обработка алгоритма осуществляется двумя программными компонентами: клиентским приложением и сервером Oracle.

Для получения максимально приближенных к реальной задаче результатов, в качестве входных данных служит набор данных: целое число, дата, вещественное число. Такой набор данных характеризует одно показание в формате: идентификатор показателя, дата, показание.



Рисунок 16 – Блок схема реализации алгоритма на основе вставок кортежей

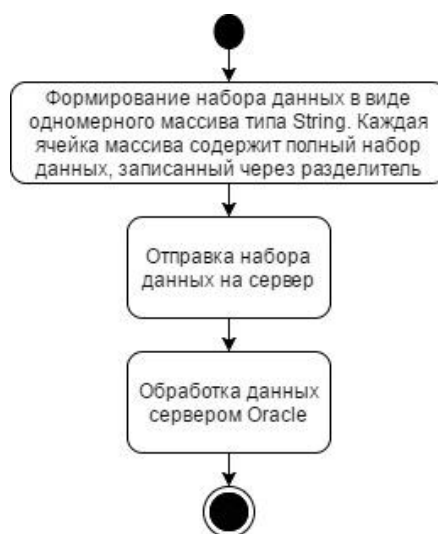


Рисунок 17 – Блок схема реализации алгоритма на основе передачи одного массива



Рисунок 18 – Блок схема реализации алгоритма на основе передачи отдельных массивов данных

### 3.2 Реализация алгоритмов передачи данных серверу Oracle

Для определения наиболее эффективного алгоритма реализовано приложение *ArrToOra*. Работа приложения заключается в том, чтобы отследить время выполнения каждого алгоритма и таким образом определить наиболее эффективный алгоритм (с минимальным временем передачи данных).

Приложение *ArrToOra* содержит один модуль *ArrayToOracle*, который описывает пользовательский графический интерфейс (рисунок 19) и алгоритмы передачи данных серверу.

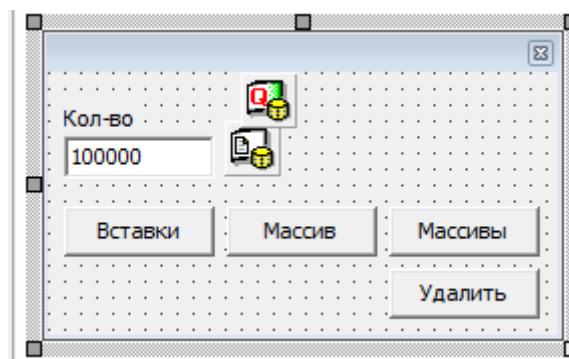


Рисунок 19 – Пользовательский графический интерфейс модуля *ArrayToOracle*



Модуль *ArrayToOracle* реализует следующие методы:

- *procedure FormCreate(Sender: TObject)* – процедура, описывающее событие создания формы. В процедуре осуществляется подключение к серверу;
- *function GetInsArr* – функция создания двумерного массива, содержащего набор значений. Возвращает параметр типа *Variant*;
- *procedure GetInsArrays(var AInt: TIntArr; var ADate: TDateArr; var ANum: TNumArr)* – процедура формирования трех одномерных массивов. Каждый *i*-ый элемент массивов соответствует набору данных. Процедура имеет три входных параметра:
  - *var AInt: TIntArr* – массив целочисленных значений;
  - *var ADate: TDateArr* – массив дат;
  - *ANum: TNumArr* – массив вещественных чисел.
- *function GetInsArrStr* – функция формирования одномерного массива типа *String*. Каждая ячейка массива содержит полный набор данных, записанный через разделитель. Функция возвращает параметр типа *Variant*;
- *procedure SingleInsClick(Sender: TObject)* – процедура описывает событие нажатия на кнопку «Вставки». В процедуре осуществляется формирование массива данных и отправка поочередно на сервер каждого показания. Время выполнения алгоритма фиксируется и после окончания работы отображается в заголовке формы;
- *procedure ArrayInsClick(Sender: TObject)* – процедура описывает событие нажатия на кнопку «Массив». В процедуре осуществляется формирование одномерного массива данных типа *String* (вызовом функции *GetInsArrStr*) и отправки массива данных на сервер *Oracle*. Время выполнения алгоритма фиксируется и после окончания работы отображается в заголовке формы;
- *procedure ArraysInsClick(Sender: TObject)* – процедура описывает событие нажатия на кнопку «Массивы». В процедуре осуществляется фор-

мирование трех одномерных массивов (вызовом процедуры *GetInsArrays*) и отправки массивов на сервер *Oracle*. Время выполнения алгоритма фиксируется и после окончания работы отображается в заголовке формы;

- *procedure DeleteAllClick(Sender: TObject)* – процедура описывает событие нажатия на кнопку «Удалить». В процедуре осуществляется выполнение запроса на удаление всех данных из тестовой таблицы в базу данных *Oracle*;
- *procedure StartTime* – процедура запуска таймера;
- *procedure StopTime* – процедура остановки таймера;
- *procedure ShowTime* – процедура отображения времени выполнения алгоритма;
- *procedure ClearTime* – процедура сброса времени отображения;
- *procedure FormClose(Sender: TObject; var Action: TCloseAction)* – процедура, описывающая событие закрытия формы. В процедуре осуществляется завершение сессии подключения к серверу *Oracle*.

Для записи тестовых данных на сервере *Oracle* создана таблица *TEST\_TAB* (рисунок 20).

TEST_TAB
ID_
DATE_
VALUE_

Рисунок 20 – Таблица *TEST\_TAB*

Для обработки полученных данных, на сервере *Oracle* создан пакет *TEST\_PKG*. В пакете *TEST\_PKG* реализованы пользовательские типы данных:

- *TStr* – таблица, содержащая набор данных типа *varchar2(100)*;
- *TInt* – таблица, содержащая набор данных типа *integer*;
- *TDate* – таблица, содержащая набор данных типа *date*;

- *TNum* – таблица, содержащая набор данных типа *number*;  
Так же в пакете *TEST\_PKG* реализованы методы:
- *function ParseRec(p\_rec in varchar2)* – функция реализует разбор строки параметров набора данных, записанных через разделитель. Возвращает параметр типа *TStr*. В качестве входного параметра *p\_rec* принимает строку параметров с разделителями;
- *procedure Import(p\_data in TStr)* - процедура разбора одномерного массива данных типа *string* и записи параметров в таблицу *TEST\_TAB*;
- *procedure Import(p\_ids in TInt, p\_dates in TDate, p\_values in TNum)* – процедура имеет три входных параметра:
  - *p\_ids in TInt* – массив целочисленных значений;
  - *p\_dates in TDate* – массив дат;
  - *p\_values in TNum* – массив вещественных чисел.

В процедуре осуществляется запись каждого *i*-то элемента массивов, как набора данных в таблицу *TEST\_TAB*.

### 3.3 Время работы алгоритмов передачи данных серверу *Oracle*

В результате выполнения программы *ArrToOra* полученные временные результаты по алгоритмам (данные фиксировались на 1000 элементов). Результат по каждому алгоритму приведен в 4 таблице, а также на рисунках 21-23).

Таблица 4 – Результат тестирования алгоритмов передачи данных

Алгоритм	на основе вставок кортежей	на основе передачи одного массива	на основе передачи отдельных массивов данных
Время выполнения	0,871 сек	0,118 сек	0,045

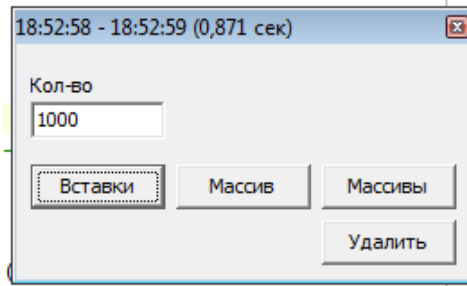


Рисунок 21 – Результат работы программы *ArrToOra* алгоритмом на основе вставок кортежей на 1000 элементов

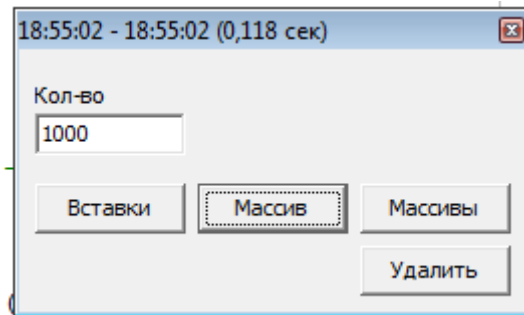


Рисунок 22 – Результат работы программы *ArrToOra* алгоритмом на основе передачи одного массива на 1000 элементов

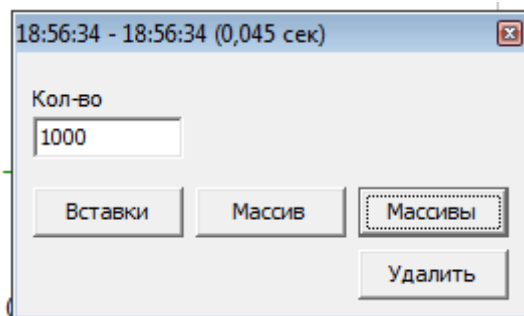


Рисунок 23 – Результат работы программы *ArrToOra* алгоритмом на основе передачи отдельных массивов данных на 1000 элементов

На основании полученных результатов, сделан вывод, что алгоритм на основе передачи отдельных массивов данных является наиболее эффективным алгоритмом передачи данных серверу *Oracle*. Этот алгоритм был использован при реализации модуля преобразования данных приложения *DailyReport*.

#### 4 ПРОВЕРКА РЕЗУЛЬТАТОВ ПРЕОБРАЗОВАНИЯ ДАННЫХ

Для проверки результатов преобразования данных использовался набор показаний по определенным показателям за период с 13.05.2017 00:00:00 по 14.05.2017 00:00:00.

В 5 таблице приведены коды показателей качества поставляемой воды и требования к преобразованию данных по ним.

Таблица 5 – Пример исходных данных для проверки результатов преобразования данных

Название показателя (единица измерения)	Код показателя в приложении <i>DailyReport</i>	Код соответствующего показателя в базе, предоставляющей данные / формула	Периодичность импорта преобразованных данных	Количество знаков после запятой при округлении
Расход воды на технологических пунктах (м <sup>3</sup> /час)	'86'	'553'	По нечетным часам.	0
	'87'	'554'		0
	'2129'	'231'		0
Накопленный объем воды на технологических пунктах (м <sup>3</sup> )	'8040'	'555'/10	Раз в сутки на 00:00 часов	1
	'8041'	'556'/10		1
Итоговая сумма подачи воды по технологическим пунктам (м <sup>3</sup> /час)	'8060'	('555'(от показания на текущее число отнять показание за предыдущее число) + '556'(от показания на текущее число отнять показание за предыдущее число))/10		1

В таблице 6 приведены значения показаний на определенную дату/время в том виде, в котором они хранятся в базах данных *ASU-TP*, *TELEMERA-DCVS*.

Таблица 6 – Исходные показания для преобразования данных

Дата	Код показателя в базах данных <i>ASU-TP, TELEMERA-DCVS</i> .				
	553	554	231	555	556
13.05.2017 00:00:00	26,69	37,240001	515,5	13534957	24450294
13.05.2017 01:00:00	26,329999	36,86	460,8999	13535188	24450618
13.05.2017 02:00:00	22,76	31,51	181,6	13535394	24450916
13.05.2017 03:00:00	23,049999	27,75	179,5	13535601	24451208
13.05.2017 04:00:00	22,18	30,729999	171,89999	13535804	24451486
13.05.2017 05:00:00	22,34	29,299999	213,19999	13536006	24451772
13.05.2017 06:00:00	25,29	35,470001	332,20001	13536206	24452056
13.05.2017 07:00:00	26,229999	36,25	478,79998	13536435	24452388
13.05.2017 08:00:00	25,799999	34,720001	522,59997	13536517	24452572
13.05.2017 09:00:00	24,85	36,220001	552,20001	13536617	24452772
13.05.2017 10:00:00	24,819999	37,830001	568,90002	13536759	24453014
13.05.2017 11:00:00	24,829999	36,130001	550,29998	13536924	24453290
13.05.2017 12:00:00	24,469999	34,330001	554,90002	13537151	24453622
13.05.2017 13:00:00	25,6	36,580001	550,09997	13537385	24453966
13.05.2017 14:00:00	24,42	34,169998	543,29998	13537624	24454308
13.05.2017 15:00:00	24,579999	36,150001	537,59997	13537863	24454652
13.05.2017 16:00:00	24,889999	34,810001	538,70001	13538100	24455000
13.05.2017 17:00:00	22,319999	33,459999	538,09997	13538336	24455344
13.05.2017 18:00:00	22,52	31,090000	521,09997	13538573	24455690

Продолжение таблицы 6

Дата	Код показателя в базах данных <i>ASU-TP, TELEMERA-DCVS</i> .				
	553	554	231	555	556
13.05.2017 19:00:00	23,129999	39,159999	517	13538810	24456028
13.05.2017 20:00:00	22,639999	31,209999	533,29998	13539052	24456374
13.05.2017 21:00:00	21,799999	32,880001	523,5	13539295	24456728
13.05.2017 22:00:00	23,09	32,11	525,09997	13539552	24457096
13.05.2017 23:00:00	22,92	40,169998	517,20001	13539811	24457470
14.05.2017 00:00:00	23,94	33,33	462,6	13540078	24457844

На рисунке 24 приведен результат преобразования данных, приведенных в 6 таблице.

Дата/время	Начальник смены	12 ГУ				Сумма	Перемычка d500	Примечание	
		Левый		Правый					Подача
		Расход м3/час	Накопленный объём м3	Расход м3/час	Накопленный объём м3				
* Добавить новую запись									
13.05 01:00		26	37			461			
13.05 03:00		23	28			180			
13.05 05:00		22	29			213			
13.05 07:00		26	36			479			
13.05 09:00	Михеева Г.Ф.	25	36			552			
13.05 11:00		25	36			550			
13.05 13:00		26	37			550			
13.05 15:00		25	36			538			
13.05 17:00		22	33			538			
13.05 19:00		23	39			517			
13.05 21:00	Кружилина Е.В.	22	33			524			
13.05 23:00		23	40			517			
14.05 00:00			1354007,8		2445784,4	1267,1			
Границы нор		15   75		10   75		300   750			
Кол-во знач	4	24	3	24	3	3	24	0	
Нестандарт		1	3	0	3	3	4		
Минимально		13	1353495,7	21	2445029,4	1267,1	177		
Максимально		26	1354559	40	2446580,8	1392,7	557		
Среднее		23,04167	1354020,83333	33,54167	2445798,2	1335,8	467,75		
Сумма		553	4062062,5	805	7337394,6	4007,4	11226		

- Ниже нормы   
 - В норме   
 - Выше нормы   
 - Недоступно

Рисунок 24 – Результат преобразования данных

## 5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБРЕЖЕНИЕ

Приложение *DailyReport* используется на предприятии ООО «Водоканал» г. Новокузнецка, осуществлявшего поставку питьевой воды жителям и организациям города.

Приложение *DailyReport* позволяет формировать различные отчеты, содержащие анализ ключевых параметров. Ввод ключевых параметров осуществляется вручную начальниками смены водозаборных и очистных цехов при непосредственном доступе к приборам учета, что влечет за собой следующие негативные факторы:

- трудоемкость процесса;
- вычислительные ошибки;

Автоматизированное преобразование данных позволяет:

- снизить вероятность вычислительной ошибки;
- значительно сократить время получения ключевых параметров;
- осуществлять запись параметров в режиме реального времени;
- повысить скорость доставки обработанных данных руководителю технологического отдела предприятия, что позволит более оперативно реагировать на изменение параметров.

Целью данной работы является разработка программного модуля преобразования данных инженерных служб для приложения *DailyReport* на языке *Delphi*.

Для достижения данной цели были поставлены следующие задачи:

- 1 изучить существующую структуру приложения *DailyReport* и схему базы данных приложения;
- 2 изучить среду для разработки программной части модуля;
- 3 изучить среду для работы со схемой данных приложения, а так же с хранимыми пакетами;



- 4 разработать и реализовать алгоритм преобразования данных, отвечающий требованиям заказчика;
- 5 разработать и протестировать оптимальный алгоритм обмена массива данных из приложения *DailyReport* с базой данных *Oracle*;
- 6 реализовать графический интерфейс модуля, отвечающий требованиям заказчика.

К реализации программного модуля является технически актуальной. Таким образом, к разработке можно предъявить следующие критерии эффективности:

- надежность;
- скорость;
- время обработки;
- качество обслуживания.

## **5.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

### **5.1.1 Потребители результатов разработки**

Модуль преобразования данных разрабатывался для существующего приложения *DailyReport*, используемого на предприятии ООО «Водоканал» г. Новокузнецка.

Целевой рынок разработки – служба главного инженера ООО «Водоканал» г. Новокузнецка.

Качество новой разработки можно оценить при помощи *QuaD* технологии. Технология *QuaD* представляет собой гибкий инструмент изменения характеристик, описывающих качество новой разработки и её перспективность на рынке и позволяющие принимать решение целесообразности вложения

денежных средств в научно-исследовательский проект. Оценка качества и перспективы проекта приведена в таблице 7 [11].

Таблица 7 – Оценка качества и перспективности проекта

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значения
<b>Показатели оценки качества разработки</b>					
Надежность	0,3	85	100	0,85	0,255
Скорость передачи данных	0,2	90	100	0,9	0,18
Простота эксплуатации	0,05	95	100	0,95	0,0475
Стабильность	0,2	85	100	0,85	0,17
Качество интерфейса	0,05	95	100	0,95	0,0475
<b>Показатели оценки коммерческого потенциала разработки</b>					
Сопровождение	0,08	87	100	0,87	0,0696
Цена	0,04	70	100	0,7	0,028
Наличие аналогов	0,08	95	100	0,95	0,076
Итого:					0,8736

Средневзвешенное значение показателя качества и перспективности научной разработки равно 0,8736. Это означает, что разработка является перспективной.

*SWOT*-анализ отражает сильные и слабые стороны, возможности и угрозы. *SWOT*-анализ по данной разработке приведен в таблице 8.

Таблица 8 – SWOT-анализ

	<p>С1. Разработка представлена в виде встроенного модуля, а не отдельной программы;</p> <p>С2. Высокая скорость преобразования большого объема данных;</p> <p>С3. Надежность преобразования данных;</p> <p>С4. Возможность доработки под дополнительные условия заказчиков в рамках сопровождения.</p>	<p>Сл1. Счетчики фиксируют некоторое значение даже тогда, когда насосы отключены. Эти значения так же импортируются.</p> <p>Сл2. Работа модуля преобразования напрямую зависит от работоспособности баз данных, предоставляющих данные;</p> <p>Сл3. Настройка параметров преобразования данных производится непосредственно при подключении к базе данных.</p>
<p>В1. Автоматизация всех показателей;</p> <p>В2. Отсутствие встраиваемых аналогов под существующее на предприятии программное приложение;</p> <p>В3. Настройка импорта данных по расписанию.</p>	<p>С2-В1. Реализовать автоматизацию всех показателей .</p> <p>С3-В1В2. Реализовать преобразование данных под настраиваемое расписание.</p>	<p>Сл1-В3. Реализовать преобразование данных под настраиваемое расписание. Благодаря этому «лишние» показания не будут импортироваться.</p>
<p>У1. Поломка баз данных, предоставляющих данные;</p> <p>У2. Неактуальность приложения, в которое встроен разрабатываемый модуль, в связи с появлением лучших аналогов конкурентов.</p>	<p>С1С4-У2. Систематизировать релиз обновлений с постоянным усовершенствованием текущего алгоритма и добавлением нового функционала.</p>	<p>Сл2-У1. Организовать мониторинг работоспособности баз данных, предоставляющих данные. В случае непредвиденных поломок, максимально быстро восстанавливать их работоспособность.</p> <p>Сл3-У2. Реализовать графический интерфейс для администратора. Это повысит удобство пользования и актуальность приложения в целом (запросы сможет добавлять даже обычный пользователь).</p>

При помощи интерактивной матрицы были рассмотрены всевозможные комбинации взаимосвязей областей *SWOT*-матрицы.

В таблице 9 показана интерактивная матрица проекта полей «Сильные стороны и возможности».

Таблица 9 – Интерактивная матрица проекта (С-В)

Сильные стороны проекта					
Возможности проекта		С1	С2	С3	С4
	В1	-	+	+	+
	В2	+	-	-	-
	В3	-	-	+	+

В таблице 10 показана интерактивная матрица проекта полей «Слабые стороны и возможности».

Таблица 10 – Интерактивная матрица проекта (Сл-В)

Слабые стороны проекта				
Возможности проекта		Сл1	Сл2	Сл3
	В1	-	-	-
	В2	-	-	-
	В3	+	-	-

В таблице 11 показана интерактивная матрица проекта полей «Сильные стороны и угрозы».

Таблица 11 – Интерактивная матрица проекта (С-У)

Сильные стороны проекта					
Угрозы проекта		С1	С2	С3	С4
	У1	-	-	-	-
	У2	+	-	-	+

В таблице 12 показана интерактивная матрица проекта полей «Слабые стороны и угрозы».

Таблица 12 – Интерактивная матрица проекта (Сл-У)

Слабые стороны проекта				
Угрозы проекта		Сл1	Сл2	Сл3
	У1	-	+	-
	У2	-	-	+

## 5.2 Определение возможных альтернатив проведения научных исследований

Для выделения возможных альтернатив проведения исследования был применен морфологический подход (результат приведен в таблице 13).

В рамках данного исследования были выделены три альтернативных варианта проведения исследования.

Таблица 13 – Морфологический анализ исследования

	1	2	3
А. Разработчик	Инженер	Руководитель	Передать проект на аутсорсинг
Б. Количество человек, участвующих в разработке	1	2	3
В. Встраиваемость разрабатываемого решения	Встраиваемый в приложение <i>DailyReport</i>	Отдельное программное приложение	
Г. Среда разработки программного модуля	<i>Microsoft Visual Studio</i>	<i>Delphi XE3</i>	
Д. Алгоритм передачи данных серверу <i>Oracle</i>	алгоритм на основе вставок кортежей	алгоритм на основе передачи одного массива	алгоритм на основе передачи отдельных массивов данных

Варианты решений:

- 1) А1Б2В1Г2Д3
- 2) А2Б1В1Г2Д1
- 3) А3Б3В2Г1Д2

## 5.3 Планирование научно-исследовательских работ

Перечень этапов работ и распределение исполнителей приведено в таблице 14.

Таблица 14 – Перечень этапов работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Разработка технического задания	1	Разработка задания.	Руководитель
Изучение исходных данных	2	Изучение архитектуры подключаемых модулей приложения <i>DailyReport</i> .	Исполнитель
	3	Изучение источников входных данных.	Исполнитель
	4	Рассмотрение результатов преобразования данных.	Исполнитель
	5	Изучение требований к графическому интерфейсу.	Исполнитель
Разработка процесса преобразования данных	6	Разбиение процесса преобразования данных на подпроцессы.	Исполнитель Руководитель
	7	Проектирование первого подпроцесса: получение параметров от пользователя.	Исполнитель Руководитель
	8	Реализация первого подпроцесса: получение параметров от пользователя.	Исполнитель
	9	Проектирование второго подпроцесса: формирование серии запросов к СУБД.	Исполнитель Руководитель
	10	Реализация второго подпроцесса: формирование серии запросов к СУБД.	Исполнитель
	11	Проектирование третьего подпроцесса: хранение <i>SQL</i> -запросов для выборки данных	Исполнитель Руководитель
	12	Реализация третьего подпроцесса: хранение <i>SQL</i> -запросов для выборки данных	Исполнитель
	13	Проектирование четвертого подпроцесса: сохранение данных в базу данных <i>DailyReport_output</i> .	Исполнитель Руководитель

Продолжение таблицы 14

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Разработка процесса преобразования данных	14	Реализация четвертого под-процесса: сохранение данных в базу данных <i>DailyReport_output</i> .	
Выбор и тестирование алгоритмов передачи преобразованных данных серверу <i>Oracle</i>	15	Проектирование алгоритмов передачи преобразованных данных серверу <i>Oracle</i>	Исполнитель Руководитель
	16	Реализация и тестирование алгоритмов передачи преобразованных данных серверу <i>Oracle</i>	Исполнитель
Проверка результатов преобразования данных	17	Тестирование результатов исследования	Исполнитель

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования [11].

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов. Для определения ожидаемого (среднего) значения трудоемкости  $t_{ожі}$  используется следующая формула:

$$t_{ожі} = \frac{3t_{mini} + 2t_{maxi}}{5},$$

где  $t_{ожі}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы чел.-дн.;

$t_{mini}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.;

$t_{\max i}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн [11].

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях  $T_p$ , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как удельный вес зарплаты в общей сметной стоимости научных исследований составляет около 65 %.

$$T_{p_i} = \frac{t_{\text{ож}i}}{Ч_i},$$

где  $T_{p_i}$  – продолжительность одной работы, раб. дн.;

$t_{\text{ож}i}$  – ожидаемая трудоемкость выполнения одной работы, чел.-дн.

$Ч_i$  – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел [11].

В таблице 15 представлен план работ с рассчитанной длительностью каждого этапа работы в календарных часах

Таблица 15 – перечень длительности этапов и работ разработки

№	Содержание работ	Должность исполнителя	Кол-во челоло-век	$t_{\min i}$	$t_{\max i}$	$t_{\text{ож}i}$	$T_{p_i}$	$T_{k_i}$	№ вар
1	Разработка задания.	Руководитель ВКР	1	0,25	0,75	0,45	0,45	0,665	В1
		Руководитель	1	0,5	1	0,7	0,7	1,034	В2
		Руководитель	1	0,5	1	0,7	0,7	1,034	В3
2	Изучение архитектуры подключаемых модулей приложения <i>DailyReport</i> .	Исполнитель	1	1,2	1,5	1,32	1,32	1,951	В1
		Руководитель	1	1	1,3	1,12	1,12	1,655	В2
		Инженер	3	1,1	1,2	1,14	0,38	0,562	В3



Продолжение таблицы 15

№	Содержание работ	Должность исполнителя	Кол-во человек	$t_{\min i}$	$t_{\max i}$	$t_{\text{ож } i}$	$T_{pi}$	$T_{ki}$	№ вар
3	Изучение источников входных данных.	Исполнитель	1	1,2	2	1,52	1,52	2,246	B1
		Руководитель	1	0,7	1,5	1,02	1,02	1,507	B2
		Инженер	3	0,8	1,9	1,24	0,41	0,611	B3
4	Рассмотрение результатов преобразования данных.	Исполнитель	1	1,5	3	2,1	2,1	3,103	B1
		Руководитель	1	1	2	1,4	1,4	2,069	B2
		Инженер	3	0,9	2,1	1,38	0,46	0,68	B3
5	Изучение требований к графическому интерфейсу.	Исполнитель	1	0,8	1	0,88	0,88	1,3	B1
		Руководитель	1	0,5	1	0,7	0,7	1,034	B2
		Инженер	3	0,5	1	0,7	0,23	0,345	B3
6	Разбиение процесса преобразования данных на подпроцессы.	Исполнитель, Руководитель ВКР	2	2,5	3,8	3,02	1,51	2,231	B1
		Руководитель	1	2,2	3	2,52	2,52	3,724	B2
		Инженер	3	2,8	3,5	3,08	1,03	1,517	B3
7	Проектирование первого подпроцесса: получение параметров от пользователя.	Исполнитель, Руководитель ВКР	2	3,5	4,9	4,06	2,03	3	B1
		Руководитель	1	2,9	3,5	3,14	3,14	4,64	B2
		Инженер	3	3,1	4,5	3,66	1,22	1,803	B3
8	Реализация первого подпроцесса: получение параметров от пользователя.	Исполнитель	1	10	12,6	11,04	11	16,31	B1
		Руководитель	1	9,7	13	11,02	11	16,29	B2
		Инженер	3	11,5	13	12,1	4,03	5,96	B3
9	Проектирование второго подпроцесса: формирование серии запросов к СУБД.	Исполнитель, Руководитель ВКР	2	2	3,5	2,6	1,3	1,921	B1
		Руководитель	1	2,3	4	2,98	2,98	4,404	B2
		Инженер	3	3	3,8	3,32	1,11	1,635	B3
10	Реализация второго подпроцесса: формирование серии запросов к СУБД.	Исполнитель	1	6	8,6	7,04	7,04	10,4	B1
		Руководитель	1	5,6	8,5	6,76	6,76	9,99	B2
		Инженер	3	5,7	8,9	6,98	2,33	3,438	B3
11	Проектирование третьего подпроцесса: хранение SQL-запросов для выборки данных.	Исполнитель, Руководитель ВКР	2	3	4	3,4	1,7	2,512	B1
		Руководитель	1	3,5	5	4,1	4,1	6,059	B2
		Инженер	3	1,5	2,1	1,74	0,58	0,857	B3

Продолжение таблицы 15

№	Содержание работ	Должность исполнителя	Кол-во человек	$t_{\min i}$	$t_{\max i}$	$t_{\text{ож } i}$	$T_{pi}$	$T_{ki}$	№ вар
12	Реализация третьего подпроцесса: хранение <i>SQL</i> -запросов для выборки данных	Исполнитель	1	7	8,5	7,6	7,6	11,23	B1
		Руководитель	1	7	8,3	7,52	7,52	11,11	B2
		Инженер	3	3,9	4,7	4,22	1,41	2,079	B3
13	Проектирование четвертого подпроцесса: сохранение данных в базу данных <i>DailyReport_output</i> .	Исполнитель, Руководитель ВКР	2	10	13	11,2	5,6	8,276	B1
		Руководитель	1	14	15,7	14,68	14,7	21,69	B2
		Инженер	3	6	7,3	6,52	2,17	3,212	B3
14	Реализация четвертого под-процесса: сохранение данных в базу данных <i>DailyReport_output</i>	Исполнитель	1	15	17,5	16	16	23,64	B1
		Руководитель	1	14	16,4	14,96	15	22,11	B2
		Инженер	3	9	10,1	9,44	3,15	4,65	B3
15	Проектирование алгоритмов передачи преобразованных данных серверу <i>Oracle</i> .	Исполнитель, Руководитель ВКР	2	5	6,1	5,44	2,72	4,02	B1
		Руководитель	1	6,5	8	7,1	7,1	10,49	B2
		Инженер	3	4	5,5	4,6	1,53	2,266	B3
16	Реализация и тестирование алгоритмов передачи преобразованных данных серверу <i>Oracle</i> .	Исполнитель	1	10	12,6	11,04	11	16,31	B1
		Руководитель	1	9,8	10,5	10,08	10,1	14,9	B2
		Инженер	3	8	9,3	8,52	2,84	4,197	B3
17	Тестирование результатов исследования	Исполнитель	2	0,5	0,9	0,66	0,33	0,488	B1
		Руководитель	1	0,4	0,7	0,52	0,52	0,768	B2
		Инженер	3	0,5	0,9	0,66	0,22	0,325	B3
Итого календарных дней:	Исполнитель, Руководитель ВКР (Вариант 1)							109,6	
	Руководитель (Вариант 2)							133,5	
	Инженер (Вариант 3)							35,1	

На рисунке 25 изображен календарный план-график для максимального по длительности исполнения работ (вариант 2).

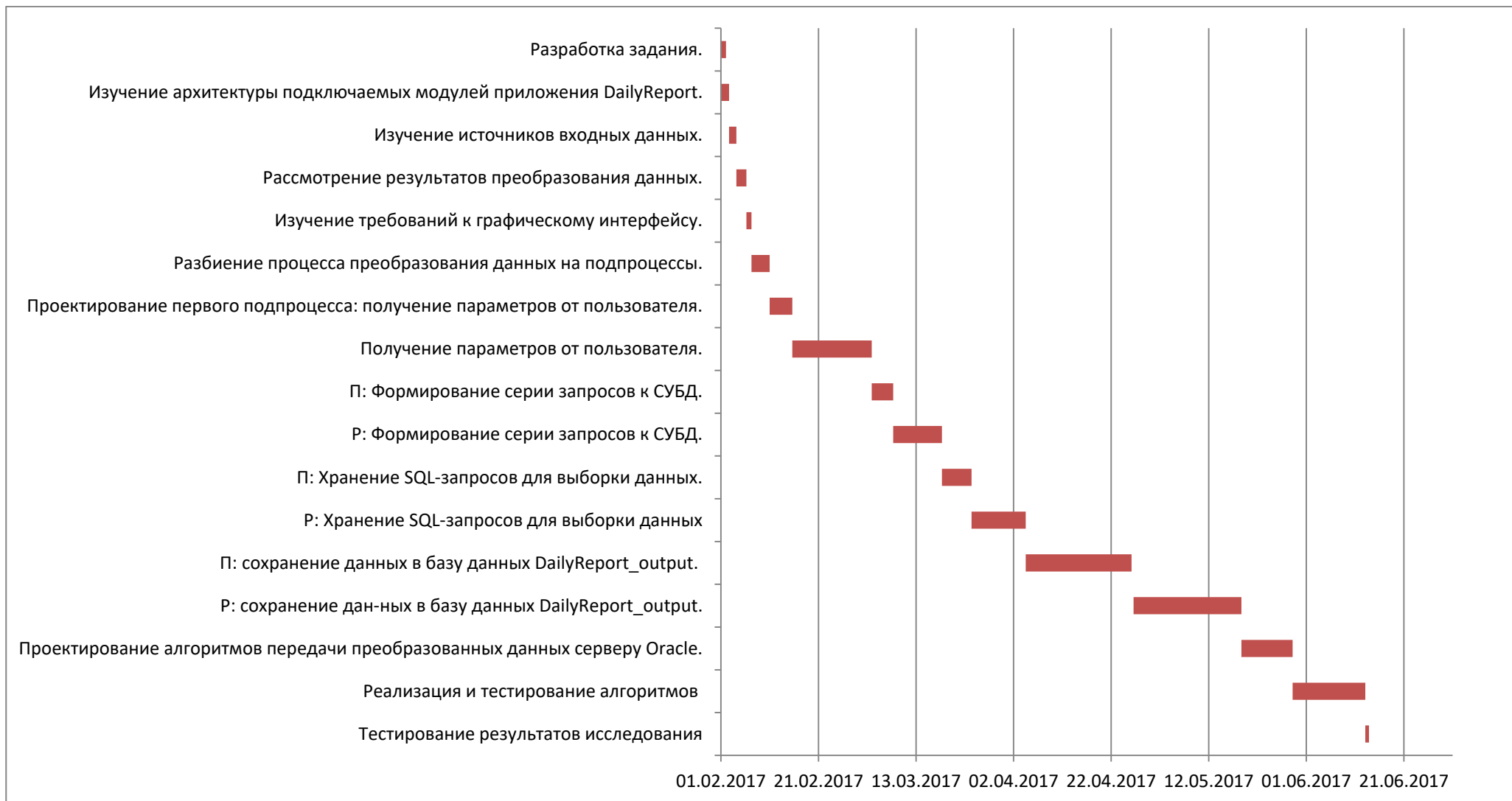


Рисунок 25 - Календарный план-график для максимального по длительности варианта исполнения работ

## 5.4 Бюджет научно-технического исследования (НТИ)

В состав бюджета выполнения работ по научно-технической работе включает вся себя стоимость всех расходов, необходимых для их выполнения. При формировании бюджета используется группировка затрат по следующим статьям:

- основная заработная плата исполнителей темы;
- дополнительная заработная плата исполнителей темы;
- отчисления во внебюджетные фонды (страховые отчисления);
- накладные расходы [11].

### 5.4.1 Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_E) \cdot \sum_{i=1}^m C_i \cdot N_{\text{расх } i}$$

где  $m$  - количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{\text{расх } i}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м<sup>2</sup> и т.д.);

$C_i$  – цена приобретения единицы  $i$ -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м<sup>2</sup> и т.д.);

$k_T$  – коэффициент, учитывающий транспортно-заготовительные расходы.

Величина коэффициента ( $k_T$ ), отражающего соотношение затрат по доставке материальных ресурсов и цен на их приобретение, зависит от условий договоров поставки, видов материальных ресурсов, территориальной удаленности поставщиков и т.д [11].

Материальные затраты, необходимые для данной разработки, заносятся в таблицу 16.

Таблица 16 – Материальные затраты

Наименование	Ед. изме- мере- ре- ния	Количество			Цена за ед., руб.			Затраты на матери- алы, ( $Z_m$ ), тыс.руб.		
		Ис- п.1	Исп. 2	Исп. 3	Исп.1	Исп.2	Исп.3	Исп. 1	Исп.2	Исп. 3
Амортизация оборудования	шт	1	1	0	3300	3300	3300	3300	3300	0
<i>Delphi XE3</i>	шт	1	1	0	1200	1200	1200	1200	1200	0
<i>PL/SQL Developer</i>	шт	1	1	0	3900	3900	3900	3900	3900	0
Итого								8400	8400	0

В ходе разработки использовалось имеющееся оборудование, поэтому в материальные расходы внесены затраты на его амортизацию за 4 месяца. Также в материальные расходы занесены затраты на приобретение лицензий для используемого программного обеспечения.

Третий вариант предполагает передачу разработки на аутсорсинг, поэтому по нему нет материальных затрат.

#### 5.4.2 Расчет основной заработной платы исполнителей системы

В данную статью расходов включается заработная плата научного руководителя и студентов, а также премии и доплаты. Расчет выполняется на основе трудоемкости выполнения каждого этапа и величины почасовой оплаты работы исполнителей [11].

Основной расчет фонда заработной платы выполняется по формуле:

$$Z_{зп} = Z_{осн} + Z_{доп}$$

где  $Z_{осн}$  – основная заработная плата;

$Z_{доп}$  – дополнительная заработная плата (12-20 % от  $Z_{осн}$ ).

Основная заработная плата ( $Z_{\text{осн}}$ ) руководителя (лаборанта, инженера) от предприятия (при наличии руководителя от предприятия) рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{дн}} \cdot T_p$$

где  $Z_{\text{осн}}$  – основная заработная плата одного работника.

$T_p$  – продолжительность работ, выполняемых научно-техническим работником, раб. часов.

$Z_{\text{дн}}$  – среднедневная заработная плата работника, руб [11].

Расчет основной заработной платы приведен в таблице 17.

Таблица 17 – Расчет основной заработной платы

№	Содержание работ	Должность исполнителя	Кол-во человек	Трудоемкость, чел.-дн.	Зарплата, приходящаяся на один чел.-дн., тыс.руб.	Всего заработная плата по тарифу (окладам), тыс.руб.	№ вар
1	Разработка задания.	Руководитель ВКР	1	0,5			B1
		Руководитель	1	0,7	3500	2450	B2
		Руководитель	1	0,7	3500	2450	B3
2	Изучение архитектуры подключаемых модулей приложения DailyReport.	Исполнитель	1	1,3	1200	1584	B1
		Руководитель	1	1,1	3500	3920	B2
		Инженер	3	1,1		0	B3
3	Изучение источников входных данных.	Исполнитель	1	1,5	1200	1824	B1
		Руководитель	1	1,0	3500	3570	B2
		Инженер	3	1,2		0	B3
4	Рассмотрение результатов преобразования данных.	Исполнитель	1	2,1	1200	2520	B1
		Руководитель	1	1,4	3500	4900	B2
		Инженер	3	1,4		0	B3
5	Изучение требований к графическому интерфейсу.	Исполнитель	1	0,9	1200	1056	B1
		Руководитель	1	0,7	3500	2450	B2
		Инженер	3	0,7		0	B3

Продолжение таблицы 17

№	Содержание работ	Должность исполнителя	Кол-во человек	Трудоемкость, чел.-дн.	Заработная плата, приходящаяся на один чел.-дн., тыс.руб.	Всего заработная плата по тарифу (окладам), тыс.руб.	№ вар
6	Разбиение процесса преобразования данных на подпроцессы.	Исполнитель, Руководитель ВКР	1	3,0	1200	3624	B1
		Руководитель	1	2,5	3500	8820	B2
		Инженер	3	3,1		0	B3
7	Проектирование первого подпроцесса: получение параметров от пользователя.	Исполнитель, Руководитель ВКР	1	4,1	1200	4872	B1
		Руководитель	1	3,1	3500	10990	B2
		Инженер	3	3,7		0	B3
8	Получение параметров от пользователя.	Исполнитель	1	11,0	1200	13248	B1
		Руководитель	1	11,0	3500	38570	B2
		Инженер	3	12,1		0	B3
9	П: Формирование серии запросов к СУБД.	Исполнитель, Руководитель ВКР	1	2,6	1200	3120	B1
		Руководитель	1	3,0	3500	10430	B2
		Инженер	3	3,3		0	B3
10	Р: Формирование серии запросов к СУБД.	Исполнитель	1	7,0	1200	8448	B1
		Руководитель	1	6,8	3500	23660	B2
		Инженер	3	7,0		0	B3
11	П: Хранение SQL-запросов для выборки данных.	Исполнитель, Руководитель ВКР	1	3,4	1200	4080	B1
		Руководитель	1	4,1	3500	14350	B2
		Инженер	3	1,7		0	B3
12	Р: Хранение SQL-запросов для выборки данных	Исполнитель	1	7,6	1200	9120	B1
		Руководитель	1	7,5	3500	26320	B2
		Инженер	3	4,2		0	B3

Продолжение таблицы 17

№	Содержание работ	Должность исполнителя	Кол-во человек	Трудоемкость, чел.-дн.	Заработная плата, приходящаяся на один чел.-дн., тыс.руб.	Всего заработная плата по тарифу (окладам), тыс.руб.	№ вар
13	П: сохранение данных в базу данных DailyReport_output.	Исполнитель, Руководитель ВКР	1	11,2	1200	13440	B1
		Руководитель	1	14,7	3500	51380	B2
		Инженер	3	6,5		0	B3
14	Р: сохранение данных в базу данных DailyReport_output.	Исполнитель	1	16,0	1200	19200	B1
		Руководитель	1	15,0	3500	52360	B2
		Инженер	3	9,4		0	B3
15	Проектирование алгоритмов передачи преобразованных данных серверу Oracle.	Исполнитель, Руководитель ВКР	1	5,4	1200	6528	B1
		Руководитель	1	7,1	3500	24850	B2
		Инженер	3	4,6		0	B3
16	Реализация и тестирование алгоритмов	Исполнитель	1	11,0	1200	13248	B1
		Руководитель	1	10,1	3500	35280	B2
		Инженер	3	8,5		0	B3
17	Тестирование результатов исследования	Исполнитель	1	0,7	1200	792	B1
		Руководитель	1	0,5	3500	1820	B2
		Инженер	3	0,7		0	B3
Итого	Вариант 1 (включая ЗП Руководителя ВКР = 20 час * 220 = 4400)					111104	
	Вариант 2					316120	
	Вариант 3					2450	

Для третьего варианта рассчитывается заработная плата только на разработку технического задания, которое в последующем передается на выполнение сторонней организации.



### 5.4.3 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников. Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$З_{внеб} = k_{внеб} \cdot (З_{осн} + З_{доп})$$

где  $k_{внеб}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2017 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30% [11].

Отчисления во внебюджетные фонды представлены в таблице 18.

Таблица 18 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.		
	Вар. 1	Вар.2	Вар. 3
Руководитель ВКР	4400	-	-
Исполнитель	106704	-	-
Руководитель	-	316120	2450
Итого:	111104	316120	2450
Коэффициент отчислений во внебюджетные фонды	0,3		
Коэффициент:			
Вариант 1:	33331,2		
Вариант 2:	94836		
Вариант 3:	735		

### 5.4.4 Контрагентные расходы

Третий вариант исполнения исследования предусматривает передачу реализации на аутсорсинг.

Третий вариант при исполнении группой разработчиков из трех предусматривает 35,1 календарных дня исполнения.

Расходы на оплату услуг составят  $35,1 \cdot 11428 = 401\,122,8$  р.

#### 5.4.5 Формирование бюджета затрат научно-исследовательского проекта

Сумма затрат по всем статьям расходов рассчитывается заносится на данном этапе в таблицу 19.

Таблица 19 – Бюджет затрат научно-исследовательского проекта

Наименование статьи	Сумма, руб.		
	Вариант 1	Вариант 2	Вариант 3
1. Материальные затраты НТИ	8400	8400	0
2. Затраты по основной заработной плате исполнителей темы	111104	316120	2450
3. Отчисление во внебюджетные фонды	33331,2	94836	735
4. Контрагентные расходы	0	0	401122,8
5. Накладные расходы	24453,63	67096,96	64689,25
Бюджет затрат НТИ	177288,83	486452,96	468997,05

#### 5.5 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}$$

где  $I_{\text{финр}}^{\text{исп.}i}$  – интегральный финансовый показатель разработки;

$\Phi_{pi}$  – стоимость  $i$ -го варианта исполнения;

$\Phi_{\max}$  – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги) [11].

$$\text{Вариант 1: } I_{\text{финр}} = \frac{177288,83}{486452,96} = 0,36$$

$$\text{Вариант 2: } I_{\text{финр}} = \frac{486452,96}{486452,96} = 1$$

$$\text{Вариант 3: } I_{\text{финр}} = \frac{468997,05}{486452,96} = 0,96$$

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное удешевление стоимости разработки в размах.

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i$$

где  $I_{pi}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент  $i$ -го варианта исполнения разработки;

$b_i^a, b_i^p$  – бальная оценка  $i$ -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

$n$  – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности рекомендуется проводить в форме таблицы (таблица 20).

Таблица 20 – Сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования Критерии	Весовой коэффициент параметра	Вариант 1	Вариант 2	Вариант 3
1) Встраиваемость реализуемого модуля	0,15	5	5	0
2) Скорость преобразования данных	0,25	5	2	4
3) Насколько язык разрабатываемого модуля соответствует стандартам предприятия ООО «Водоканал»	0,1	5	5	0

Продолжение таблицы 20

Объект исследования Критерии	Весовой ко- эффициент параметра	Вариант 1	Вариант 2	Вариант 3
4) Сопровождение модуля преобразования данных	0,25	5	3	2
5) Цена	0,2	5	1	2
6) Скорость реализации	0,05	2	1	5
ИТОГО	1			

$$I_{p-вар1} = 0,15*5+0,25*5+0,1*5+0,25*5+0,2*5+0,05*2 = 4,85$$

$$I_{p-вар2} = 0,15*5+0,25*2+0,1*5+0,25*3+0,2*1+0,05*1 = 2,75$$

$$I_{p-вар3} = 0,15*0+0,25*4+0,1*0+0,25*2+0,2*2+0,05*5 = 2,15$$

*Интегральный показатель эффективности вариантов исполнения разработки* ( $I_{исп.1}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр}}$$

$$I_{вар1} = 4,85 / 0,36 = 13,47$$

$$I_{вар2} = 2,75 / 1 = 2,75$$

$$I_{вар3} = 2,15 / 0,96 = 2,23$$

Полученное значение интегрального показателя эффективности исполнения разработки превысило максимальный балл оценки. Таким образом, результат работы можно считать положительным, так как оценка интегрального показателя ресурсоэффективности близка к максимальной, при этом стоимость разработки ниже, чем у ряда аналогов, рассмотренных при анализе конкурентных решений.

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта ( $\mathcal{E}_{cp}$ ):

$$\mathcal{E}_{cp} = \frac{I_{ucn.1\_i}}{I_{ucn.2\_min}}$$

Сравнительная эффективность разработки представлена в таблице 21.

Таблица 21 – Сравнительная эффективность разработки

№ п/п	Показатели	Вариант 1	Вариант 2	Вариант 3
1	Интегральный финансовый показатель разработки	0,36	1	0,96
2	Интегральный показатель ресурсоэффективности разработки	4,85	2,75	2,15
3	Интегральный показатель эффективности	13,47	2,75	2,23
4	Сравнительная эффективность вариантов исполнения	6	1,2	1

Сравнение значений интегральных показателей эффективности позволяет понять и выбрать более эффективный вариант решения поставленной в бакалаврской работе технической задачи с позиции финансовой и ресурсной эффективности [11].

Наиболее эффективным для разработки является первый вариант.

### Заключение

В процессе оценки перспективности и альтернатив проведения научного исследования с позиции ресурсоэффективности и ресурсосбережения коммерческого потенциала для работы «Разработка модуля преобразования данных» было отмечено, что целевой потребителем разработки – предприятие ООО «Водоканал», г. Новокузнецка.

Также была произведена оценка качества и перспективности проекта благодаря инструменту *QuaD* технология.

Были проанализированы сильные и слабые стороны решения, его возможности и угрозы, а также корреляция этих показателей в ходе *SWOT*-анализа. Полученные показатели позволили определить направление разви-

тия разработки для достижения наибольшей востребованности среди целевой аудитории. Результаты *SWOT*-анализа показали, что необходимо направить развитие приложения в сторону кроссплатформенности и адаптивности под различные мобильные устройства.

Были рассмотрены три различных варианта исполнения задачи выпускной квалификационной работы. После проведения анализа выявлено, что первый вариант, описывающий реализацию задачи в выпускной квалификационной работе, является наиболее эффективным.

## 6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

### Введение

ООО «Водоканал» г. Новокузнецка является одним из ключевых предприятий жизнеобеспечения города.

Для качественного контроля всех сфер работы, предприятие оснащено большим количеством различных программных приложений, одним из которых является приложение *DailyReport* (Суточный рапорт).

Приложение *DailyReport* предназначено для работы инженерно-технологических служб. Приложение хранит в себе различные технические показатели, в том числе показатели поставляемой питьевой воды, а так же сточных вод:

- количественные показатели (расход воды, м<sup>3</sup>/ч, расход электроэнергии, кВтч/сут, расход удельной электроэнергии, кВтч/м<sup>3</sup>, накопленный объем, м<sup>3</sup>, перепад давления, м. вод. ст., скорость фильтрации, м/ч и так далее);
- качественные показатели (мутность воды, мг/м<sup>3</sup>, водородный показатель, ед. рН, температура воды, град. С, хлор остаточный свободный, мг/дм<sup>3</sup>, запах, балл, цветность, град, взвешенные вещества, мг/дм<sup>3</sup>, хлоропоглощаемость, мг/дм<sup>3</sup> и так далее).

Долгое время эти показатели вводились в приложение вручную контроллерами на цехах при непосредственном контакте с приборами учета.

Целью данной работы является разработка программного модуля преобразования данных инженерных служб для приложения *DailyReport* на языке *Delphi*. Программный модуль преобразования данных позволяет осуществлять импорт показаний приборов контроля качества и количества поставляемой питьевой воды, а так же сточных вод в базу данных приложения *DailyReport* из базы данных под управлением *Microsoft SQL*, в которую регулярно фиксируются данные с приборов.

## 6.1 Производственная безопасность

Разработка и использование программного модуля преобразования данных приложения *DailyReport* предполагает наличие компьютера и специально оборудованного рабочего места. В связи с этим, возникает влияние на человека вредных и опасных факторов, приведенных в таблице 22.

Таблица 22 - Опасные и вредные факторы при использовании, проектировании и реализации алгоритма процесса преобразования данных

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
<p>Разработка и эксплуатация программного модуля:</p> <p>1) длительная работа с персональным компьютером;</p> <p>2) работа в помещениях не (или не достаточно) пропускающих естественный свет.</p>	<p>1) повышенный уровень электромагнитных излучений [12];</p> <p>2) повышенный уровень наличия статического электричества [12];</p> <p>3) нагрузка, связанная с длительным сидячим положением тела при работе;</p> <p>4) высокая нагрузка на зрительный аппарат;</p> <p>5) умственное перенапряжение [12];</p> <p>6) отсутствие или недостаток естественного света [12].</p>	<p>1) Электрический ток</p>	<p>Гигиенические требования к персональным электронно-вычислительным машинам и организации работы устанавливаются СанПиН 2.2.2/2.4.1340-03I [13].</p> <p>Требования к освещению рабочих мест устанавливаются ГОСТ Р 55710-2013 [14].</p> <p>Требования к электромагнитным полям в производственных условиях устанавливаются СанПиН 2.2.4.1191-03 [15].</p> <p>Средства защиты от статического электричества определяются ГОСТ 12.4.124-83 [16].</p>



### 6.1.1 Повышенный уровень электромагнитных излучения

ПЭВМ являются источниками вредных излучений, негативно сказывающихся на здоровье человека.

ПЭВМ является источником таких излучений, как:

- ультрафиолетового 200-400 нм;
- видимого 400-700 нм;
- ближнего инфракрасного 700-1050 нм;
- радиочастотного 3 кГц – 30 МГц;
- электростатических полей.

Превышение норм этих излучений приводит к таким негативным последствиям, как головная боль, дерматит кожи, перегрев тканей, повышение температуры тела.

Предельно допустимые значения ЭМП характеристик устанавливаются СанПиН 2.2.4.1191-03 «Электромагнитные поля в производственных условиях» [15]. Предельно-допустимые значения приведены в таблице 23.

Таблица 23 - Предельно допустимые значения характеристик ЭМП

Наименование параметров	Допустимое значение
Напряженность электромагнитного поля по электрической составляющей на расстоянии 50 см от поверхности видеомонитора	10 В/м
Напряженность электромагнитного поля по магнитной составляющей на расстоянии 50 см от поверхности видеомонитора	0,3 А/м
Напряженность электростатического поля не должна превышать для взрослых пользователей	20 кВ/м
Напряженность электромагнитного поля на расстоянии 50 см вокруг ВДТ по электрической составляющей должна быть не более: - в диапазоне частот 5 Гц – 2 кГц; - в диапазоне частот 2 – 400 кГц.	25 В/м 2,5 В/м
Плотность магнитного потока должна быть не более: - в диапазоне частот 5 Гц – 2кГц; - в диапазоне 2 – 400 кГц.	250 нТл 25 нТл
Поверхностный электростатический потенциал не должен превышать	500 В

ПЭВМ на которых ведется разработка проектируемого модуля или его эксплуатация должны соответствовать требованиям санитарных правил [13].

Согласно СанПиН 2.2.2/2.4.1340-03I временные уровни электромагнитных полей (ЭМП), создаваемых ПЭВМ, не должны превышать значений, указанных в таблице 24 [13].

Таблица 24 - Временные допустимые уровни ЭМП, создаваемых ПЭВМ

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	в диапазоне частот 5 Гц – 2 кГц	25 В/м
	в диапазоне частот 2 кГц – 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц – 2 кГц	250 нТл
	в диапазоне частот 2 кГц – 400 кГц	25 нТл
Электрический потенциал экрана видеомонитора		500 В

### 6.1.2 Повышенный уровень наличия статического электричества

В кабинетах, оборудованных ПЭВМ токи статического электричества часто возникают при прикосновении к любому из элементов ПЭВМ. Небольшие токи опасности для человека не представляют, только могут вызывать неприятные ощущения. Однако, они могут привести к выходу оборудования из строя, в худшем случае к возникновению пожара.

Для нейтрализации статического электричества необходимо использовать заземляющие устройства, нейтрализаторы, увлажняющие устройства, антиэлектростатические вещества, экранирующие вещества [17].

Так же поверхность сиденья, спинки и других элементов стула (кресла) должна быть полумягкой, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием [13].

### **6.1.3 Нагрузка, связанная с длительным сидячим положением тела при работе**

При длительном сидячем положении тела, велика нагрузка на позвоночник. В сидячем положении страдают все отделы позвоночника. Суставы, связывающие отдельные позвонки между собой, при отсутствии нагрузки теряют эластичность, высыхают. Результатом это становится возникновение таких болезней, как искривление позвоночника, остеохондроз, радикулит. Всё это сопровождается болевыми симптомами в пояснице и позвоночнике. Из-за малого количества движений может возникнуть венозная недостаточность.

Для снятия нагрузки, связанной с длительным сидячим положением тела при работе, конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ПЭВМ, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип рабочего стула (кресла) следует выбирать с учетом роста пользователя, характера и продолжительности работы с ПЭВМ [13].

Рабочий стул (кресло) должен быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию [13].

### **6.1.4 Высокая нагрузка на зрительный аппарат**

Зрительный анализатор - это совокупность органов и тканей, обеспечивающая восприятие, кодирование и декодирование зрительной информации.

Во время работы, пользователь ПЭВМ долгое время может быть сосредоточен на мониторе, выполняя работу. Чем больше времени затрачено на сосредоточенное наблюдение, тем выше нагрузка на зрительный аппарат. Все

структуры зрительного анализатора, участвующие непосредственно в зрении, имеют свои резервные возможности. После длительной зрительной нагрузки необходим определённый период отдыха и восстановления. В зрелом возрасте наступает истощение резервных возможностей глаза, которое проявляется в повышенной зрительной утомляемости; если такое состояние сохраняется в течение длительного времени, то создаются предпосылки развития хронических глазных заболеваний.

Для снятия нагрузки на зрительный аппарат экран видеомонитора должен находиться от глаз пользователя на расстоянии 600-700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов [13].

Рабочие места с ПЭВМ при выполнении творческой работы, требующей значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать друг от друга перегородками высотой 1,5-2,0 м [13].

### **6.1.5 Умственное перенапряжение**

Длительная сконцентрированная работа без отдыха может приводить к умственному перенапряжению. Развитие умственного перенапряжения протекает в три стадии:

- первая стадия - появляются субъективные признаки – нарушается сон (человеку трудно заснуть, он быстро просыпается, а утром сложно проснуться), пропадает аппетит. Уставший не выдерживает физические и психические нагрузки.
- вторая стадия - усталость ярко выражена, приводит к неприятному дискомфорту, снижению качества жизни. Кроме того, что человек быстро устает, его беспокоят неприятные ощущения в сердца, после физических нагрузок дрожат конечности, появляется спазм. Если своевременно не принять меры, у человека начнут изменяться все органы, при этом нарушается сон, постоянно беспокоят кошмары. Иногда на

этой стадии появляются всплески работоспособности – рано утром или поздно вечером. Кроме того, на этой стадии возникают сбои в обмене веществ, человек резко теряет вес, появляются проблемы с давлением, ухудшается внешний вид.

- третья стадия – на этом этапе человек страдает от бессонницы, постоянно устает, нарушается работа внутренних органов, человек полностью выбивается из нормальной жизни.

### **6.1.6 Отсутствие или недостаток естественного света**

Выполнение зрительной работы при недостаточной освещенности рабочего места может привести к развитию некоторых дефектов глаз: близорукость ложная и истинная; дальнозоркость истинная и старческая.

К производственному освещению независимо от источника света предъявляются следующие требования:

- достаточная освещенность, т. е. освещенность объекта должна обеспечить комфортные условия для общей работоспособности;
- равномерность освещения, т. е. освещенность должна быть равномерной во времени и пространстве;
- отсутствие блескости в поле зрения работающих.

Рассмотрим некоторые характеристики освещения.

Освещенность – поверхностная плотность светового потока; единица освещенности – люкс.

Яркость – поверхностная плотность силы света в данном направлении, которая определяется из отношения силы света излучаемой поверхности в этом направлении к проекции светящейся поверхности на плоскость, перпендикулярную данному направлению. Единица яркости – кандела на квадратный метр.

Коэффициент отражения  $p$  характеризует способность поверхности отражать падающий на нее световой поток. Если  $p < 0,2$  – фон считается темным; если  $0,2 < p < 0,4$  – средним; при  $p > 0,4$  – светлым.

Контраст объекта с фоном  $K$  определяется из соотношения яркостей рассматриваемого объекта и фона.

Контраст объекта с фоном считается малым, если  $K < 0,2$ , средним – при  $0,2 < K < 0,5$  и большим при  $K > 0,5$ .

Освещение в производственных помещениях в светлое время суток осуществляется естественным источником света – небосводом. Естественное освещение может быть боковым (через окна), верхним (через зенитные фонари) и комбинированным. Применение той или иной системы естественного освещения зависит от назначения и размеров помещения, расположения его в плане здания, а также от светового климата местности.

Интенсивность естественного освещения оценивается коэффициентом естественного освещения (КЕО), показывающего, во сколько раз освещенность в помещении меньше освещенности наружной.

При недостатке или отсутствии естественного света, его дополняют искусственным освещением.

Для рабочих мест, оборудованных персональными компьютерами или мониторами, допустимые значения габаритной яркости осветительных приборов (ОП), отражающихся в экранах мониторов, в зависимости от яркости экранов/мониторов приведены в таблице 25 [15].

Таблица 25 - Таблица зависимости осветительных приборов в зависимости от яркости экранов/мониторов

Класс представления информации	Габаритная яркость ОП, кд/м <sup>2</sup>	Яркость экранов/мониторов, кд/м <sup>2</sup>
А – темные знаки на светлом фоне	До 1500 включ. “3000”	До 200 включ. Св. 200
В – светлые знаки на темном фоне	“1000” “1500”	До 200 включ. Св. 200
Примечание - Значения габаритной яркости ОП определяют под углом не менее 65° от вертикали для рабочих мест, в которых находятся дисплейные экраны с углом наклона 150°. Для рабочих мест, для которых необходимы более чувствительные экраны или экраны с регулируемым наклоном, значения габаритной яркости ОП определяют под углом не более 55°.		

### 6.1.7 Электрический ток

Особую опасность для пользователей и разработчиков системы может представлять электрический ток. Влияние электрического тока может быть незначительным, а может привести к трагическим последствиям.

При обычном пользовании персональным компьютером, электрический ток не оказывает влияние на пользователя. Опасность удара электрическим током может возникнуть только в случае разбора устройств и непосредственного контакта с питанием.

Защитить сотрудника от влияния электрического тока можно обеспечив должностной инструкцией, в которой необходимо запретить сотруднику предприятия вмешиваться в работу техники, нарушая её целостность или разбирая детали.

### 6.2 Экологическая безопасность

Защита окружающей среды является важным вопросом современного общества.

Реализуемый программный модуль преобразования данных не оказывает прямого воздействия на окружающую среду. Однако при разработке и эксплуатации программного модуля используются персональные компьютеры (ПК). Персональные компьютеры имеют ограниченный срок эксплуатации, после которого устройство приходит в негодность и дальнейшая его эксплуатация становится невозможной. На этом этапе и возникает вопрос о влиянии на окружающую среду. Оргтехника содержит в себе различные металлы (свинец, ртуть, сурьма, мышьяк, кадмий), которые под воздействием внешних условий, переходят в органические и растворимые соединения, которые представляют сильнейшие яды.

Согласно Федеральному закону от 10 января 2002 г. №7-ФЗ «Об охране окружающей среды», отходы производства и потребления подлежат сбору, использованию, обезвреживанию, транспортировке, хранению и захоронению, условия и способы которых должны быть безопасными для окружающей среды и регулироваться законодательством Российской Федерации [16].

Следовательно, для того, чтобы исключить вредное воздействие на окружающую среду, после прекращения эксплуатации ПК необходимо подвергнуть утилизации в соответствии с методикой проведения работ по комплексной утилизации драгоценных металлов из отработанных средств вычислительной техники, утвержденной Государственным комитетом РФ по телекоммуникациям (от 19 октября 1999 г.) [18].

### **6.3 Безопасность в чрезвычайных ситуациях**

В процессе реализации или пользования реализуемым программным решением может возникнуть такая чрезвычайная ситуация, как пожар.

Алгоритм действий при пожаре:

- 1 Необходимо немедленно вызвать пожарную охрану по телефону "01", сообщив свой точный адрес, объект пожара и встретить пожарную охрану;



- 2 Если горение только началось, необходимо его затушить следующими способами: водой, толстым одеялом, покрывалом, песком, землей;
- 3 Ни в коем случае не тушить водой горящие электропроводку и электроприборы, находящиеся под напряжением;
- 4 При усилении пожара, срочно покинуть помещение;
- 5 Запрещено прятаться в задымленном помещении в укромные места [19].

При возникновении прочих чрезвычайных ситуация (землетрясение, ураган и т.д.) необходимо совершать действия в соответствии с ГОСТ Р 22.0.01-94 и ГОСТ Р 22.3.03-94 [20, 21].

## **6.4 Правовые и организационные вопросы обеспечения безопасности**

### **6.4.1 Правовые вопросы обеспечения безопасности**

В системе обеспечения безопасности жизни и здоровья работников в процессе их трудовой деятельности основная роль принадлежит нормативным правовым актам по охране труда.

Государственный надзор и контроль в организациях независимо от организационно-правовых форм и форм собственности осуществляют специально уполномоченные на то государственные органы и инспекции в соответствии с федеральными законами. Согласно трудовому кодексу РФ:

- продолжительность рабочего дня не должна превышать 40 часов в неделю;
- во время регламентированных перерывов целесообразно выполнять комплексы упражнений и осуществлять проветривание помещения [19].

Существуют также специализированные органы, осуществляющие государственный контроль и надзор в организациях на предмет соблюдения существующих правил и норм.

К таким органам относятся:

- Федеральная инспекция труда;
- Государственная экспертиза условий труда Федеральной службы по труду и занятости населения;
- Федеральная служба по надзору в сфере защиты прав потребителей и благополучия человека и др.

## **6.4.2 Организационные вопросы обеспечения безопасности**

### **6.4.2.1 Требования к помещениям для работы с ПЭВМ**

Эксплуатация ПЭВМ в помещениях без естественного освещения допускается только при наличии расчетов, обосновывающих соответствие нормам естественного освещения и безопасность их деятельности для здоровья работающих [18].

Естественное и искусственное освещение должно соответствовать требованиям действующей нормативной документации [18].

Оконные проемы должны быть оборудованы регулируемыми устройствами типа: жалюзи, занавесей, внешних козырьков и др [13].

Площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) - 4,5 м [13].

Для внутренней отделки интерьера помещений, где расположены ПЭВМ, должны использоваться диффузно отражающие материалы с коэффициентом отражения для потолка - 0,7-0,8; для стен - 0,5-0,6; для пола - 0,3-0,5 [13].

#### **6.4.2.2 Требования к микроклимату, содержанию аэроионов и вредных химических веществ в воздухе на рабочих местах, оборудованных ПЭВМ**

В помещениях, оборудованных ПЭВМ, проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ЭВМ [13].

Уровни положительных и отрицательных аэроионов в воздухе помещений, где расположены ПЭВМ, должны соответствовать действующим санитарно-эпидемиологическим нормативам [13].

#### **6.4.2.3 Требования к освещению на рабочих местах, оборудованных ПЭВМ**

Рабочие столы следует размещать таким образом, чтобы видеодисплейные терминалы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева [13].

Искусственное освещение в помещениях для эксплуатации ПЭВМ должно осуществляться системой общего равномерного освещения [13].

Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300-500 лк. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк [13].

Следует ограничивать прямую блескость от источников освещения, при этом яркость светящихся поверхностей (окна, светильники и др.), находящихся в поле зрения, должна быть не более 200 кд/м [13].

Следует ограничивать неравномерность распределения яркости в поле зрения пользователя ПЭВМ, при этом соотношение яркости между рабочими поверхностями не должно превышать 3:1-5:1, а между рабочими поверхностями и поверхностями стен и оборудования 10:1 [13].

#### **6.4.2.4 Общие требования к организации рабочих мест пользователей ПЭВМ**

При размещении рабочих мест с ПЭВМ расстояние между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора), должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м [13].

Конструкция рабочего стола должна обеспечивать оптимальное размещение на рабочей поверхности используемого оборудования с учетом его количества и конструктивных особенностей, характера выполняемой работы. При этом допускается использование рабочих столов различных конструкций, отвечающих современным требованиям эргономики. Поверхность рабочего стола должна иметь коэффициент отражения 0,5-0,7 [13].

#### **6.4.5 Требования к организации и оборудованию рабочих мест с ПЭВМ для взрослых пользователей**

Высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680-800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм [13].

Модульными размерами рабочей поверхности стола для ПЭВМ, на основании которых должны рассчитываться конструктивные размеры, следует считать: ширину 800, 1000, 1200 и 1400 мм, глубину 800 и 1000 мм при нерегулируемой его высоте, равной 725 мм [13].

Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной - не менее 500 мм, глубиной на уровне колен - не менее 450 мм и на уровне вытянутых ног - не менее 650 мм [13].

Конструкция рабочего стула должна обеспечивать:

- ширину и глубину поверхности сиденья не менее 400 мм;
- поверхность сиденья с закругленным передним краем;
- регулировку высоты поверхности сиденья в пределах 400-550 мм и углов наклона вперед до 15° и назад до 5°;
- угол наклона спинки в вертикальной плоскости в пределах  $\pm 30^\circ$ ;
- регулировку расстояния спинки от переднего края сиденья в пределах 260-400 мм;
- стационарные или съемные подлокотники длиной не менее 250 мм и шириной - 50-70 мм;
- регулировку подлокотников по высоте над сиденьем в пределах  $230 \pm 30$  мм и внутреннего расстояния между подлокотниками в пределах 350-500 мм [13].

Клавиатуру следует располагать на поверхности стола на расстоянии 100-300 мм от края, обращенного к пользователю, или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы [13].

## ЗАКЛЮЧЕНИЕ

В процессе выполнения выпускной квалификационной работы изучена модульная архитектура приложения *DailyReport* и базы данных исходных и обработанных показаний качества воды.

В результате выполнения выпускной квалификационной работы разработан программный модуль для преобразования данных инженерных служб для приложения *DailyReport* на языке *Delphi*, а также создан набор хранимых процедур и функций на сервере СУБД *Oracle* на языке *PL/SQL*.

В рамках работы рассмотрены способы загрузки данных из нескольких источников, их преобразования и сохранения. Спроектированы алгоритмы передачи преобразованных данных из программного модуля в СУБД *Oracle*. Произведено сравнение быстродействия алгоритмов.

Результаты выпускной квалификационной работы внедрены на предприятии ООО «Водоканал» г.Новокузнецка в эксплуатацию, о чем свидетельствует справка о внедрении программного модуля (Приложение Б). В дальнейшем планируется расширение функциональности созданного программного модуля возможностью запуска процесса преобразования данных по расписанию.

В разделе финансовый менеджмент обоснована экономическая целесообразность разработки программного модуля преобразования данных.

В разделе социальная ответственность разработаны меры безопасности, обеспечивающие предотвращение несчастных случаев, защиту здоровье работников. Рассмотрены факторы, оказывающие вредное воздействие на окружающую среду и меры по их предотвращению.

## CONCLUSION

During execution of the final qualifying work the modular architecture of the application DailyReport and databases of original and processed data of water quality were studied

As a result of execution of the final qualifying work a Delphi software module to convert the data of engineering services from the application DailyReport was developed, and also a set of PL/SQL stored procedures and functions on the DBMS Oracle server was created.

This work studies the ways to load data from multiple sources, to transform and to save it. Algorithms to transfer the converted data from the software module to DBMS Oracle were designed. A comparison of the algorithms performance was conducted.

The results of the final qualifying work are deployed by the Novokuznetsk "Vodokanal", as evidenced by the certificate of the software module implementation (Annex B). In the future we plan to expand the functionality of the generated software module to run the data conversion process on schedule.

Section financial management justifies economic feasibility of the software data conversion module development.

Section social responsibility describes implemented security measures that ensure the prevention of accidents and protection of workers health. The factors that have an adverse impact on the environment and measures to prevent them were also considered.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Методические рекомендации по определению потребности в электрической энергии на технологические нужды в сфере водоснабжения, водоотведения и очистке сточных вод. [Электронный ресурс] URL: <http://www.gosthelp.ru/text/Metodicheskierekomendacii393.html>, свободный. – Загл. С экрана. – Яз. рус. Дата обращения: 14.02.2017.
- 2 Медведев А.Н. Руководство пользователя. Приложение Суточный рапорт. [Текст]: Руководство пользователя / Медведев А.Н. – 26 стр.
- 3 Direct Oracle Access 4.1. [Электронный ресурс] URL: <http://maintracker.org/forum/viewtopic.php?t=1154815>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 05.04.2017.
- 4 Oracle для профессионалов: архитектура, методики программирования и особенности версий 9i, 10g и 11g, 2-е изд.: Пер. с англ. – М.: ООО «И.Д.Вильямс», 2011. – 848 с.
- 5 Гетерогенный сервис. [Электронный ресурс] URL: [http://www.sql.ru/faq/faq\\_topic.aspx?fid=163](http://www.sql.ru/faq/faq_topic.aspx?fid=163), , свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 05.04.2017.
- 6 Компонент TSQLDataSet. [Электронный ресурс] URL: <http://delphi.support.uz/index.php?type=tema&id=232>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 22.03.2017.
- 7 Официальная справка компонента Direct Oracle Access.
- 8 Рубанцев В. Большой самоучитель Delphi XE3. – С-П.: ООО «Я + R », 2012. – 1274 с.
- 9 Oracle для профессионалов: технологии и решения для достижения высокой производительности и эффективности, 3-е изд.: Пер с англ. – М.: ООО «И.Д.Вильямс», 2015. – 960 с.
- 10 Секреты Oracle SQL.: Пер с англ. – М.: ООО «И.Д.Вильямс», 2006. – 368 с.



- 11 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение: учебно-методическое пособие / И.Г. Видяев, Г.Н. Серикова, Н.А. Гаврикова, Н.В. Шаповалова, Л.Р. Тухватулина З.В. Крицына; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2014. – 36 с.
- 12 Межгосударственный стандарт. Система стандартов безопасности труда. Опасные и вредные производственные факторы. Классификация. [Электронный ресурс] URL: <http://docs.cntd.ru/document/5200224>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 05.05.2017.
- 13 Постановление от 13 июня 2003 года №118 «О введении в действие санитарно-эпидемиологических правил и нормативов СанПиН 2.2.2/2.4.1340-03» (с изменениями на 21 июня 2016 года). [Электронный ресурс] URL: <http://docs.cntd.ru/document/901865498>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 05.05.2017.
- 14 Национальный стандарт Российской Федерации «Освещение рабочих мест внутри зданий». [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200105707>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 06.05.2017.
- 15 СанПиН 2.2.4.1191-03 Электромагнитные поля в производственных условиях. [Электронный ресурс] URL: <http://docs.cntd.ru/document/901853847>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 06.05.2017.
- 16 ГОСТ 12.4.124-83 Система стандартов безопасности труда (ССБТ). Средства защиты от статического электричества. Общие технические требования. [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200006078>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 06.05.2017.
- 17 Федеральный закон от 10.01.2002 N 7-ФЗ (ред. от 25.06.2012, с изм. от 05.03.2013) "Об охране окружающей среды". [Электронный ресурс] URL:

- <http://www.consultant.ru/cons/cgi/online.cgi?req=doc;base=LAW;n=131664;div=LAW;dst=100376>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 06.05.2017.
- 18 Методика проведения работ по комплексной утилизации драгоценных металлов из отработанных средств вычислительной техники, утвержденной Государственным комитетом РФ по телекоммуникациям (от 19 октября 1999 г.). [Электронный ресурс] URL: <http://nenuda.ru/государственный-комитет-рф-по-телекоммуникациям.html>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 07.05.2017.
- 19 Действия при пожаре. МЧС России. [Электронный ресурс] URL: <http://38.mchs.gov.ru/document/1396914>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 07.05.2017.
- 20 Государственный стандарт Российской Федерации «Безопасность в чрезвычайных ситуациях. Основные положения». [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200001531>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 08.05.2017.
- 21 Государственный стандарт Российской Федерации «Безопасность в чрезвычайных ситуациях. Защита населения». [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200001521>, свободный. - Загл. С экрана. – Яз. рус. Дата обращения: 08.05.2017.

## Приложение А (справочное)

### Руководство пользователя

Для запуска модуля преобразования данных на главном окне приложения *DailyReport* необходимо выбрать Инструменты -> Импорт данных (рисунок А.1).

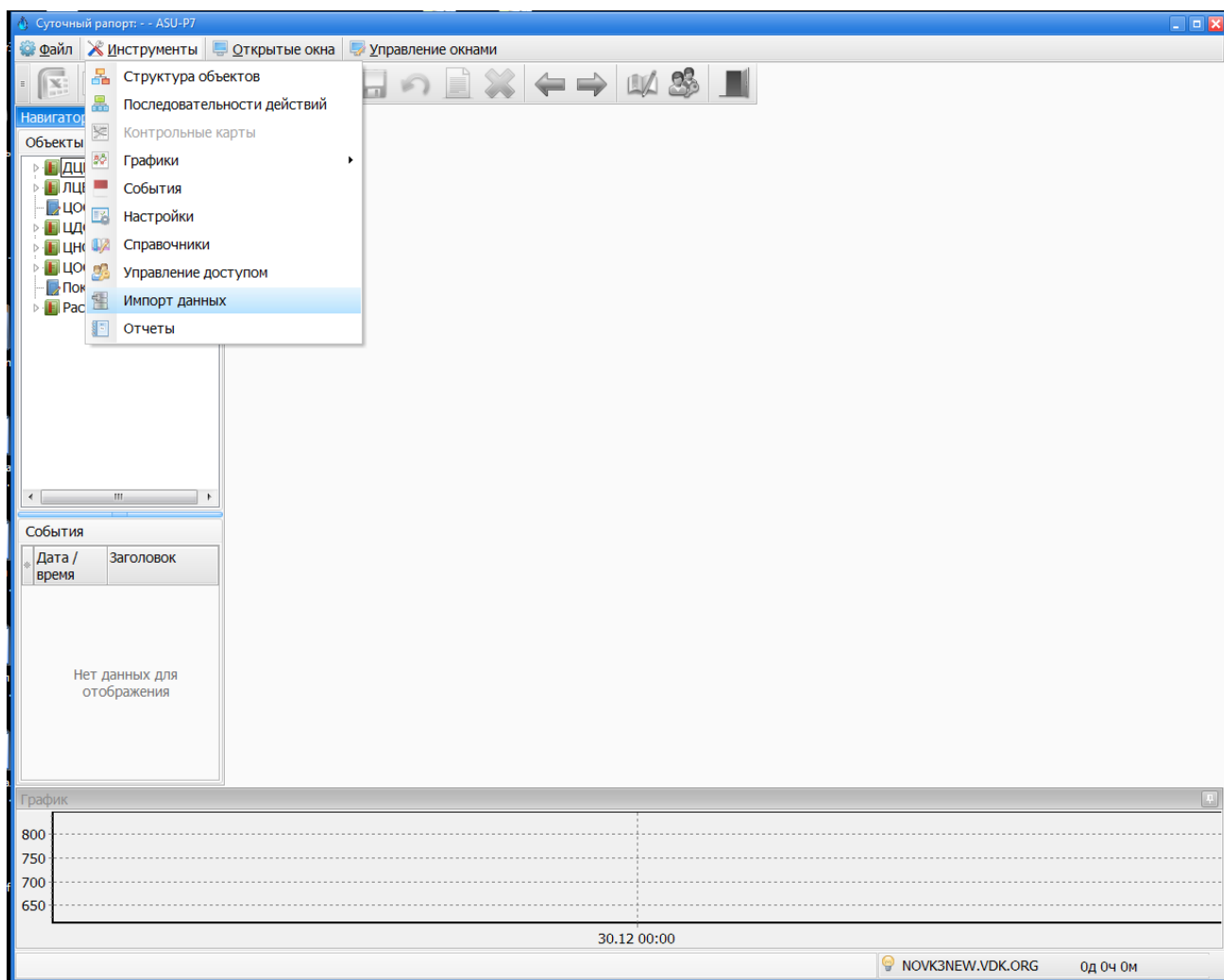


Рисунок А.1 – Главное окно приложения *DailyReport*

Откроется окно настроек параметров преобразования данных (рисунок А.2). Для запуска процесса преобразования данных необходимо выбрать объект и указать корректно временной период. Затем нажать на кнопку «ОК».

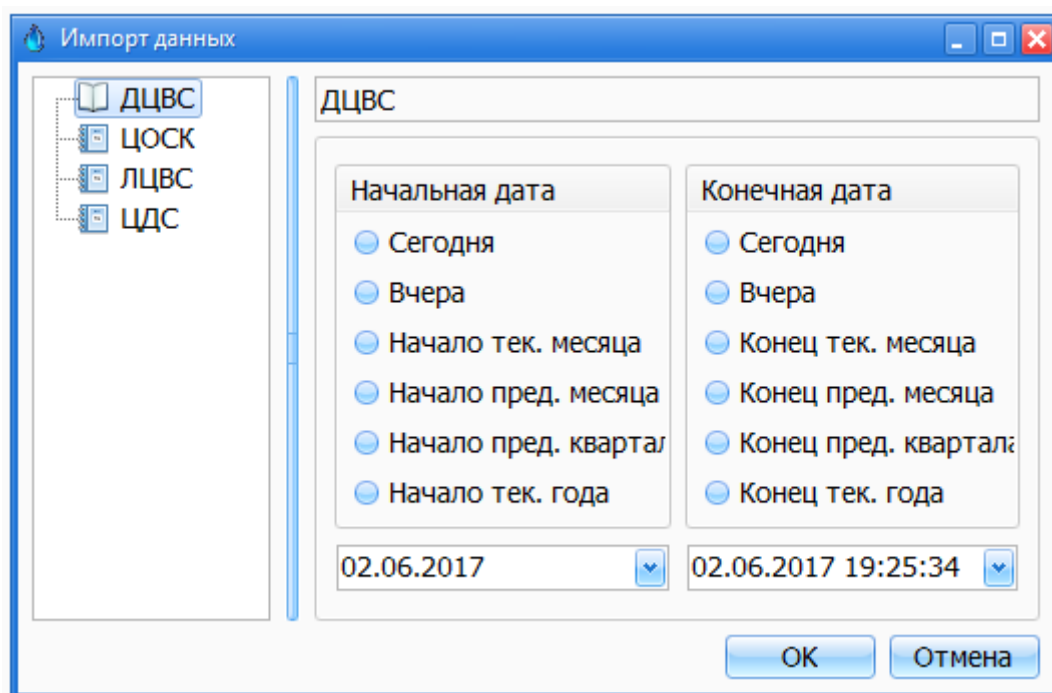


Рисунок А.2 – Окно настроек параметров преобразования данных

Во время выполнения процесса преобразования данных приложение сообщит о наличии ошибки или после выполнения – об успешном завершении (рисунок А.3).

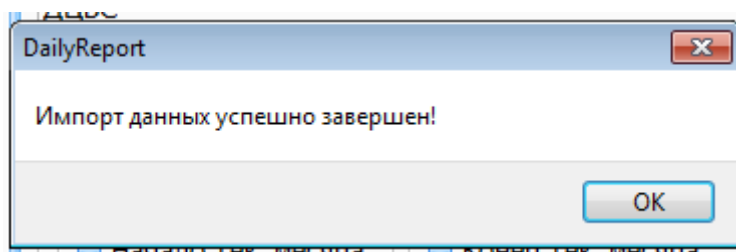


Рисунок А.3 – Результат процесса преобразования данных

Для отображения результатов процесса преобразования данных необходимо открыть стандартное средства отображения отчетов в табличной форме (Рисунок А.4).

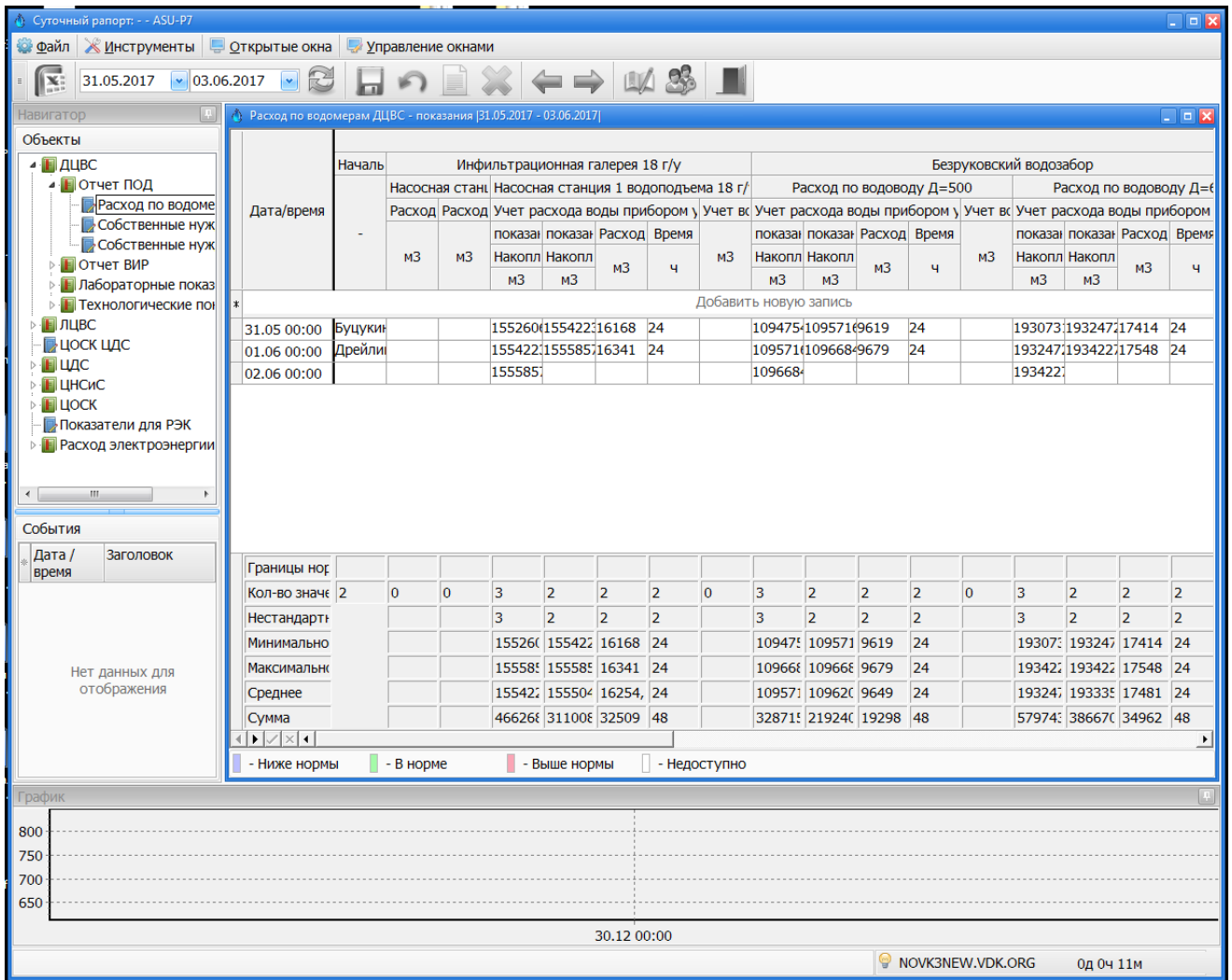


Рисунок А.4 – Результат преобразования данных

## Приложение Б



Общество с ограниченной ответственностью  
**«ВОДОКАНАЛ»**

654005, г.Новокузнецк Кемеровской области, пр.Строителей, 98.  
Тел.: (3843) 900-700, факс: (3843) 79-04-94, e-mail: [office@vdk.ru](mailto:office@vdk.ru)  
[www.vdk.ru](http://www.vdk.ru)

### Справка о внедрении программного модуля

Настоящая справка дана Носковой Алене Викторовне о том, что результат выпускной квалификационной работы – программный модуль преобразования данных успешно внедрен в промышленную эксплуатацию на предприятии.

Начальник отдела  
информационных технологий



Н.Ю. Симкина