

УТВЕРЖДЕН
БЮЛИ.00131-01 32 01-ЛУ

ПРОГРАММНЫЙ КОМПЛЕКС
«ПРОГРАММНО-ОПРЕДЕЛЯЕМАЯ ЛАБОРАТОРИЯ
«ПОЛАТОР»

РУКОВОДСТВО СИСТЕМНОГО ПРОГРАММИСТА
БЮЛИ.00131-01 32 01
Листов 41

Инв. № подл. А-3848	Подп. и дата [подпись] 20.05.21	Взам. инв. №	Инв. № дубл.	Подп. и дата
------------------------	------------------------------------	--------------	--------------	--------------

АННОТАЦИЯ

Настоящий документ Программный комплекс «Программно-определяемая лаборатория «ПОЛАТОР». Руководство системного программиста» (БЮЛИ.00131-01 32 01) предназначен для описания сведений для установки, проверки, обеспечения функционирования и настройки программы на условия конкретного применения (ГОСТ 19.101-77).

В разделе «Общие сведения о программе» указаны назначение и функции программы и сведения о технических и программных средствах, обеспечивающих выполнение программы.

В разделе «Структура программы» приведены сведения о структуре программы, ее составных частях, о связях между составными частями и о связях с другими программами.

В разделе «Настройка программы» приведено описание действий по настройке программы на условия конкретного применения.

В разделе «Проверка программы» приведено описание способов проверки, позволяющих дать общее заключение о работоспособности программы.

В разделе «Сообщения системному программисту» указаны тексты сообщений, выдаваемых в ходе выполнения настройки, проверки программы, а также в ходе выполнения программы, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ	5
1.1. Назначение и функции программы	5
1.2. Сведения о технических и программных средствах, обеспечивающих выполнение данной программы	6
1.2.1. Аппаратные средства.....	6
1.2.2. Программные средства.....	7
1.2.3. Организационное обеспечение	7
2. СТРУКТУРА ПРОГРАММЫ	8
2.1. Структура программы	8
2.2. Составные части программы	9
2.2.1. Подсистема Информационной Безопасности	9
2.2.2. Подсистема Системного Администрирования	9
2.2.3. Подсистема Управления Ресурсами	10
2.2.4. Подсистема Обмена Данными	10
2.2.5. Подсистема Прикладных Задач	10
2.3. Связи между составными частями	11
2.4. Связи с другими программами	13
3. НАСТРОЙКА ПРОГРАММЫ	16
3.1. Описание установочного комплекта	16
3.2. Описание установки программы	16
3.3. Настройка программы	18
4. ПРОВЕРКА ПРОГРАММЫ.....	19
4.1. Общие сведения.....	19
4.2. Автоматическая проверка	19
4.3. Полуавтоматическая проверка.....	19
4.3.1. Проверка авторизации пользователя	20
4.3.2. Проверка настройки.....	20
4.3.3. Проверка создания проекта и подключения пользователей к нему	20
4.3.4. Проверка сохранения данных проекта	21
4.3.5. Проверка предоставления данных проекта	22
4.3.6. Проверка открытия существующего проекта	22
4.3.7. Проверка включения проекта в список проектов, размещенных на сервере	22
4.3.8. Проверка открытия содержания проекта в виде перечня компонентов, которые включены в проект, с указанием имени каждого компонента.....	23

4.3.9. Проверка одновременного управления всеми компонентами, включенными в проект	23
5. СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ	25
5.1. Общие положения	25
5.2. Функциональная структура.....	26
5.3. Спецификация модели события.....	28
5.3.1. Журнал регистрации событий.....	28
5.3.2. Каталог наборов событий.....	29
5.3.3. Регистр параметров регистрации событий.....	29
5.3.4. Каталог классов событий	30
5.3.5. Регистр системной иерархии.....	30
5.3.6. Уровень значимости	31
5.3.7. Уровень детализации.....	32
5.3.8. Функциональные роли источника события.....	33
5.4. Интерфейс пользователя	34
5.5. Спецификация событий.....	35
5.5.1. Подсистема Обмена Данными	35
5.5.2. Подсистема Прикладных Задач	35
5.5.3. Подсистема Пользователя.....	36
5.5.4. Подсистема Управления Ресурсами	36
ПЕРЕЧЕНЬ ТЕРМИНОВ	37
ПЕРЕЧЕНЬ СОКРАЩЕНИЙ.....	38
ПЕРЕЧЕНЬ РИСУНКОВ	39
ПЕРЕЧЕНЬ ТАБЛИЦ.....	40
ССЫЛОЧНЫЕ ДОКУМЕНТЫ.....	41
6. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	42

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ

1.1. Назначение и функции программы

Программный компонент «ПОЛАТОР-Сервер» входит в состав программного комплекса «Программно-определяемая лаборатория «ПОЛАТОР» и предназначен для обеспечения отказоустойчивого параллельного и масштабируемого совместного доступа к пользовательским ресурсам со стороны программного компонента «ПОЛАТОР-Клиент» в режиме распределенного функционирования.

Программный компонент «ПОЛАТОР-Сервер» предусматривает следующий набор функций:

1. Обеспечение запуска программного компонента «ПОЛАТОР-Сервер» при процессе старта операционной системы с обеспечением последовательности запуска компонент в соответствии с зависимостями;
2. Обеспечение выполнения пользовательских настроек «Сервера Информационной Безопасности». Определение ресурсов пользователя для организации проектов;
3. Обеспечение создания пользовательских проектов;
4. Обеспечение выполнения пользовательских настроек «Сервера Прикладных Задач»; Определение пользователем состава и параметров пользовательских проектов;
5. Обеспечение выполнения пользовательских настроек «Сервера Обмена Данными»; Определение пользователем состава и параметров ввода-вывода, обмена данными в рамках проектов;
6. Обеспечение открытия существующих пользовательских проектов и их загрузки;
7. Обеспечение выполнения пользовательских настроек «Сервера Управления Ресурсами». Определение ресурсов пользователя для организации библиотек;
8. Обеспечение создания пользовательских библиотек;
9. Обеспечение выполнения пользовательских настроек «Сервер Прикладных Задач»; Определение пользователем состава и параметров пользовательских библиотек;
10. Обеспечение подключения существующих пользовательских библиотек;
11. Обеспечение редактирование графической модели на языке GPL;
12. Обеспечение управлением вставкой, редактированием и удалением элементов графической модели на языке GPL;
13. Обеспечение управлением отображением и редактированием параметров элементов графической модели на языке GPL;
14. Обеспечение редактирование кода на языке SPL;
15. Создание пользовательского интерфейса элементов, созданных как в нотации GPL, так и SPL;
16. Обеспечение управления состоянием проекта;

17. Обеспечение управления состоянием библиотеки;
18. Обеспечение управлением журналом событий;
19. Запуск прикладных моделей и программ в режиме "Выполнение" (Циклический режим выполнения). Количество запущенных экземпляров масштабируется производительностью «Сервера Прикладных Задач».
20. Запуск прикладных моделей и программ в режиме "Отладка" (Шаговый режим выполнения). Количество запущенных экземпляров масштабируется производительностью «Сервера Прикладных Задач».
21. Обеспечение нормального останова программного компонента «ПОЛАТОР-Сервер» при процессе останова операционной системы;
22. Обеспечение нормального останова программного компонента «ПОЛАТОР-Сервер» при останове операционной системы при реакции на сигнал от подсистемы мониторинга электропитания;
23. Обеспечение нормального останова программного компонента «ПОЛАТОР-Сервер» при централизованном управлении состоянием аппаратных и программных средств;
24. Обеспечение нормального останова программного компонента «ПОЛАТОР-Сервер» из рабочего аппаратно-программного окружения системного программиста.

1.2. Сведения о технических и программных средствах, обеспечивающих выполнение данной программы

Данный раздел документа определяет рекомендуемые параметры аппаратного и программного окружения, обеспечивающее функционирование программного компонента «ПОЛАТОР-Сервер» в проектном режиме. Специфицируются также рекомендации в части организационного обеспечения в виде требований к квалификации роли «Системный Программист».

1.2.1. Аппаратные средства

Далее определены рекомендуемые требования к аппаратным:

- а) Процессор: не ниже Intel® Core™ i5 не ниже 9-го поколения. Рекомендуется использование аппаратных платформ на базе Intel® Xeon® не ниже 2-го поколения;
- б) Оперативная память: не менее 4 Гб DDR3. Рекомендуется использование DDR4 16 Гб;
- в) Объем жесткого диска: не менее RAID10 512 Гб. Рекомендуется использование локальной СХД объемом 2048 Гб;
- г) ЛВС (LAN): не менее 1 x 1 Мбит/с. Сетевое соединение используется для интеграции с другими экземплярами программного комплекса «Программно-определяемая лаборатория «ПОЛАТОР»»;
- д) Привод DVD с поддержкой формата DVD-R;
- е) Свободный USB-порт для подключения лицензионного ключа;
- ж) Порт IPMI.

1.2.2. Программные средства

Далее определены рекомендуемые требования к операционной системе и базовым программным средствам:

- а) Предустановленная лицензионная операционная система Microsoft Windows 10 Professional x64;
- б) Предустановленный комплекс базовых программных средств PostgreSQL версии не ниже 11.

1.2.3. Организационное обеспечение

В должностные обязанности роли «Системный Программист» входят:

- а) Поддержание работоспособности аппаратной платформы, на которой установлен программный компонент «ПОЛАТОР-Сервер»;
- б) Процессы установки программного компонента «ПОЛАТОР-Сервер», а также программных модулей из его состава (опционально);
- в) Процессы поддержания работоспособности программного компонента «ПОЛАТОР-Сервер», а также программных модулей из его состава.

Для обеспечения качества выполнения указанных выше работ рекомендуется обеспечить следующие требования в части квалификации и состава компетенций роли «Системный Программист»:

- а) Уровень образования: Специальное высшее образование по специальностям из группы «Информационные Технологии»;
- б) Сертификация по системному администрированию операционных систем Windows;
- в) Сертификация в части администрирования баз данных PostgreSQL.

2. СТРУКТУРА ПРОГРАММЫ

2.1. Структура программы

С целью обеспечения модульности архитектуры программно-технического комплекса «Программно-определяемая лаборатория «ПОЛАТОР» предусмотрено разделение программных компонент из состава Комплекса на выделенные процессы на основе функциональных признаков.

Предполагается возможность выполнения выше указанных процессов программных компонент Комплекса как на одном вычислительном узле, так и отдельных или выделенных вычислительных узлах.

В качестве вычислительных узлов предполагается использование как физических вычислителей, так и виртуальных вычислительных узлов, функционирующих на платформах виртуализации.

С целью обеспечения требований надежности и обеспечения требуемого уровня производительности, физические вычислительные узлы могут быть реализованы в форме кластерных высокопроизводительных отказоустойчивых комплексов.

Решения в части аппаратных и программных решений приведены в соответствующих разделах данного документа.

Приведенная далее функциональная структура представляет схему функционального деления, которая не накладывает требований на структурное деление. Допускается структурное исполнение указанных компонент в виде решений начиная от совмещенных в одном процессе до архитектурных решений, предполагающих выделенные кластеры для каждого из функциональных компонент.

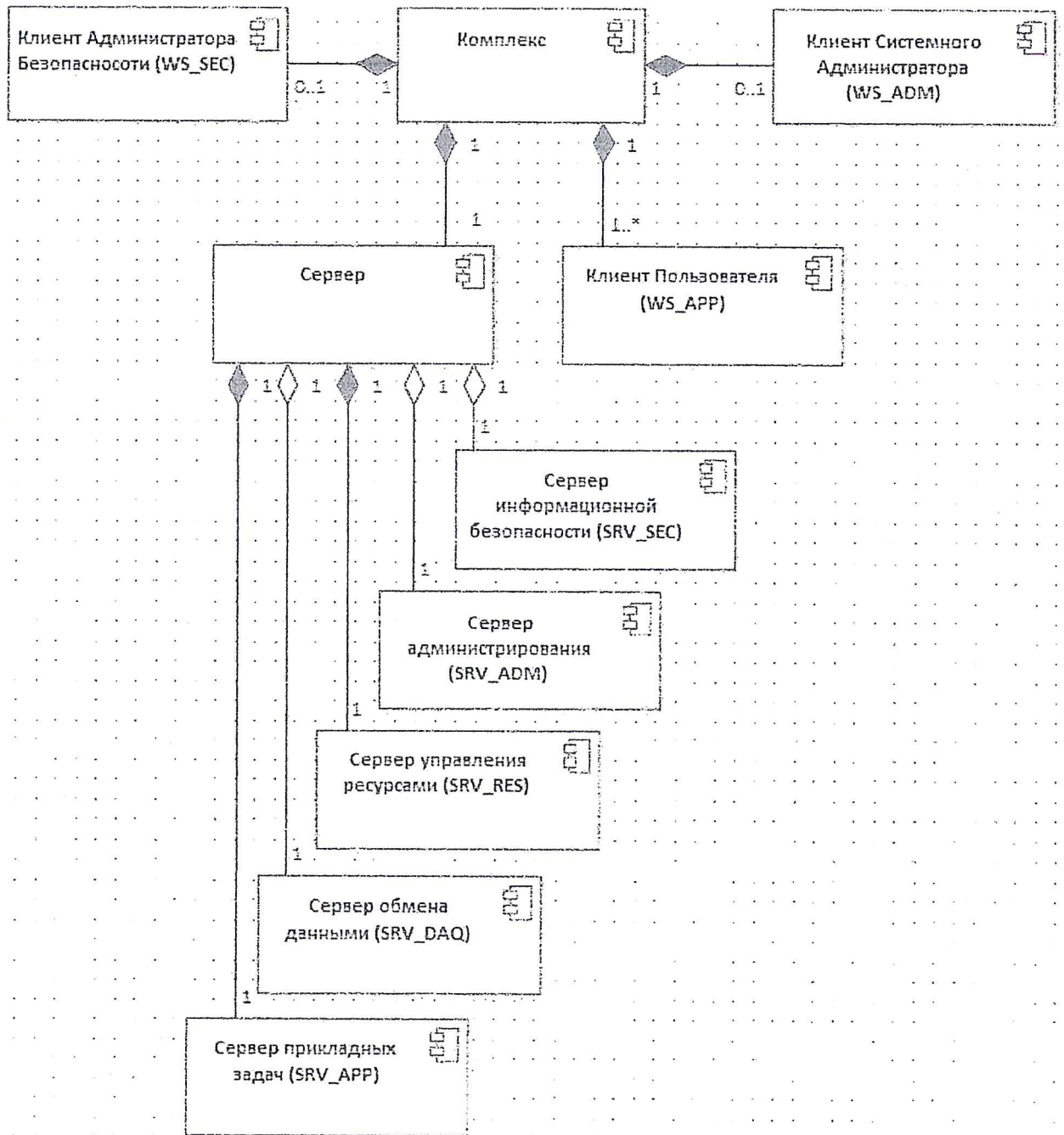


Рисунок 1: Функциональная структура «ПОЛАТОР-Сервер»

2.2. Составные части программы

2.2.1. Подсистема Информационной Безопасности

Подсистема Информационной Безопасности (SRV_SEC) выполняет следующие функции:

- Управление лицензиями;
- Обеспечение локальных сервисов информационной безопасности;
- Обеспечение интеграции в вышестоящую централизованную систему управления информационной безопасностью.

2.2.2. Подсистема Системного Администрирования

Подсистема Системного Администрирования (SRV_ADM) выполняет следующие функции:

- а) Обеспечение локальных сервисов системного администрирования, обслуживания и ремонта аппаратных и программных средств;
- б) Обеспечение интеграции в вышестоящую централизованную систему управления конфигурацией;
- в) Обеспечение интеграции в вышестоящую централизованную систему управления ИТ-инфраструктурой.

2.2.3. Подсистема Управления Ресурсами

Подсистема Управления Ресурсами (SRV_RES) выполняет следующие функции:

- а) Обеспечение интеграции распределенных ресурсов файловых систем;
- б) Обеспечение интеграции распределенных ресурсов систем управления баз данных;
- в) Обеспечение интеграции распределенных ресурсов WEB-сервисов;
- г) Обеспечение централизованного доступа к локальным и интегрированным ресурсам файловой системы со стороны SRV_SEC, SRV_ADM, SRV_DAQ, SRV_APP;
- д) Обеспечение централизованного доступа к локальным и интегрированным ресурсам систем управления базами данных со стороны SRV_SEC, SRV_ADM, SRV_DAQ, SRV_APP;
- е) Обеспечение централизованного доступа к локальным и интегрированным ресурсам WEB-сервисов со стороны SRV_SEC, SRV_ADM, SRV_DAQ, SRV_APP.

2.2.4. Подсистема Обмена Данными

Подсистема Обмена Данными (SRV_DAQ) выполняет следующие функции:

- а) Обеспечение ввода-вывода данных и представление их в форме тегов;
- б) Обеспечение коммуникации по специализированным протоколам и представление данных обмена в форме тегов;
- в) Предоставление доступа к тегам со стороны SRV_APP в различных режимах (циклические чтения-запись, чтение по обновлениям, запись по запросу и т.п.).

2.2.5. Подсистема Прикладных Задач

Подсистема Прикладных Задач (SRV_APP) выполняет следующие функции:

- а) Обеспечение работы с моделями в режиме дизайна;
- б) Обеспечение выполнения моделей в различных режимах (шаговый, циклический и т.п.);
- в) Обеспечение режима отладки моделей;
- г) Обеспечение отображения результатов выполнения модели;

- д) Обмен данными с автоматизированными рабочими местами «ПОЛАТОР-Клиент».

2.3. Связи между составными частями

Обмен данными между программными модулями из состава «ПОЛАТОР-Сервер», а также между «ПОЛАТОР-Сервер» и «ПОЛАТОР-Клиент» осуществляется посредством архитектурного решения на основе разделяемой базы данных. Применена архитектура класса «ON-HOST»:

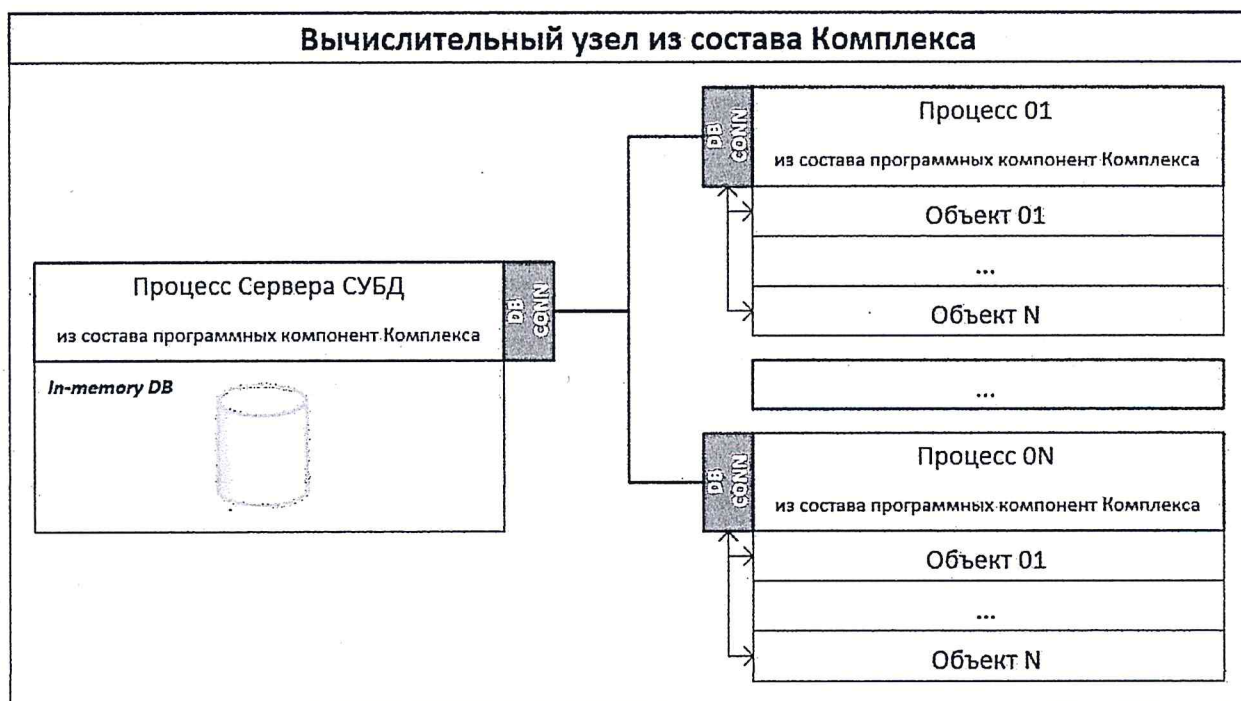


Рисунок 2: Обмен данными через БД

Для обмена данными организуются следующее структурное разделение на группу доменов:

- Домен метаданных системы. Спецификация структуры, конфигурации и параметров программных компонент из состава Комплекса;
- Домен информационных ресурсов прикладных проектов. Содержит в структурированной форме данные прикладных проектов и библиотек;
- Домен данных в режиме реального времени. Содержит в структурированной форме необходимый набор параметров и результатов выполнения SPL-программ и GPL-моделей;
- Домен данных обмена сообщениями в режиме реального времени;
- Домен архивных данных.

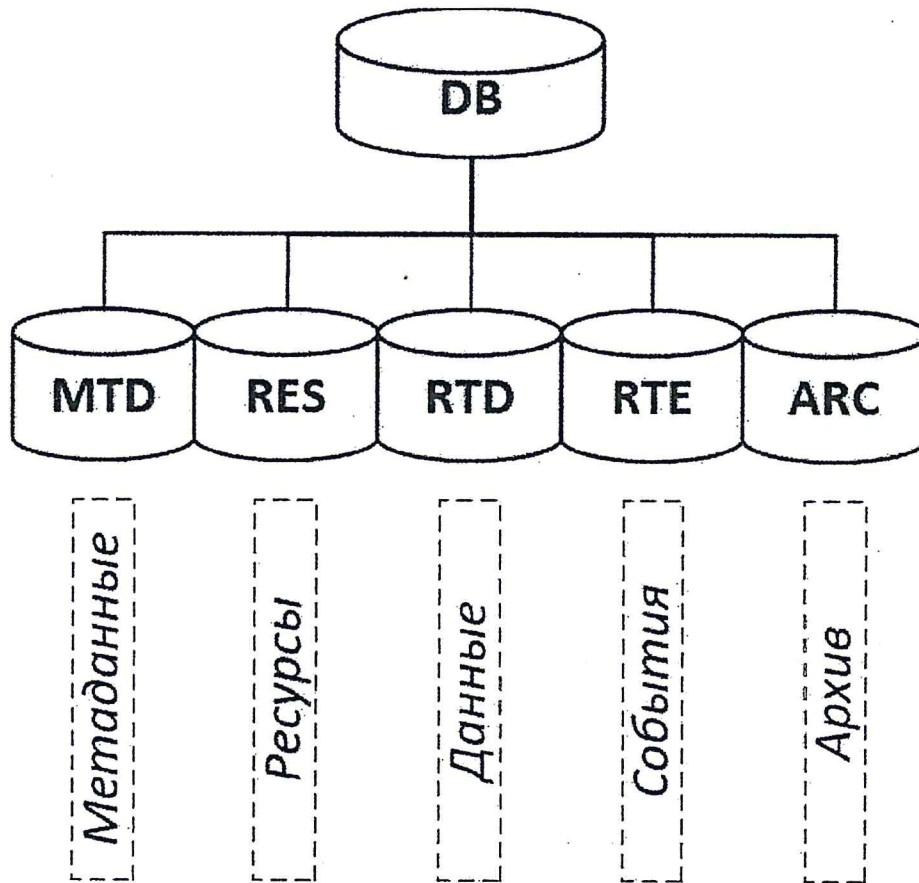


Рисунок 3: Структура доменов БД

Деление производится на основе комплекса признаков с целью оптимизации структуры под требования специфической функциональности.

Данное архитектурное решение обеспечивает:

- а) Применение современных подходов на базе технологий ORM (Object-Relational Mapping - объектно-реляционное отображение), которые позволяют гибко и надежно реализовать двунаправленное преобразование объектно-ориентированного подхода для представления сущностей предметной области в памяти прикладных процессов к реляционной модели представления в СУБД;
- б) Применение современных методологий, компонент и инструментов с высокими значениями технических показателей, а также высокая доступность пула ресурса разработчиков.
- в) Параллельное многопользовательское функционирование программных компонент из состава Комплекса обеспечивается посредством встроенных программных механизмов ACID (Atomicity, Consistency, Isolation, Durability) - транзакционный многопользовательский доступ с гарантией изоляции, при котором транзакции одного пользователя не влияют на транзакции другого пользователя;
- г) Высокую пропускную способность. Интеграция программных компонент из состава Комплекса посредством прямых подключений к

- базе данных обеспечивает обработку больших объемов данных с гарантированным временем доступа. При необходимости, масштабируемость решения обеспечивается путем организации выделенного сервера СУБД с применением к нему соответствующих методов вертикального и горизонтального масштабирования;
- д) Требуемый уровень показателей надежности на уровне программно-технических решений реализуется посредством применения аппаратных и программных средств с соответствующими значениями показателей, на организационном уровне реализуется путем применения на стадии эксплуатации проектных плановых и внеплановых регламентов ТОиР. На уровне архитектурного решения в части СУБД как единичной точки отказа в качестве СУБД применена система соответствующего класса;
 - е) Требуемый уровень показателей информационной безопасности реализуется посредством развитых методов централизованного применения основанных на корпоративных и юридических нормах политик информационной безопасности со сбалансированной гранулярностью на уровнях базы данных, таблиц и объектов данных. Предусмотрена возможность применения шифрования данных;
 - ж) Требуемый уровень показателей обслуживаемости и контролепригодности реализуется посредством наличия соответствующего инструментария в составе СУБД, наличия API для интеграции в смежные и вышестоящие уровни системного администрирования, а также высокой степени документированности и наличия развитой инфраструктуры поддержки программного продукта СУБД.

2.4. Связи с другими программами

Программный компонент «ПОЛАТОР-Сервер» предусматривает опциональный программно-технический модуль, который реализует интерфейс открытых систем на базе семейства протоколов OPC:

- а) OPC UA;
- б) OPC DA;
- в) OPC AE;
- г) OPC HDA.

Указанный программный модуль обеспечивает интерфейс со сторонними программными системами и функционально входит в состав Подсистемы Обмена Данными (SRV_DAQ).

Дополнительно к функциям управления конфигурацией и мониторингом состояния со стороны специализированных компонент из состава программного комплекса «Программно-определяемая лаборатория «ПОЛАТОР», «ПОЛАТОР-Сервер» предусматривает интерфейсы интеграции в соответствующие системы уровня

предприятия с целью обеспечения централизованного группового управления и мониторинга.

Для обеспечения и управления безопасностью аппаратных и программных компонент из состава «ПОЛАТОР-Сервер» предусмотрен интерфейс интеграции в системы управления информационной безопасностью (Security Capsule, RuSIEM, Microsoft SCCM):

- а) Анализ и управление рисками безопасности;
- б) Сбор, обработка и анализ событий безопасности;
- в) Обнаружение атак и нарушений критериев и политик безопасности;
- г) Централизованное управление аутентификацией и контролем доступа;
- д) Антивирусная защита и защита от вредоносного кода;
- е) Межсетевое экранирование;
- ж) Формирование отчетных документов безопасности.

Для управления конфигурацией аппаратных и программных компонент из состава «ПОЛАТОР-Сервер» предусмотрен интерфейс интеграции в системы управления конфигурацией ИТ-инфраструктур (Microsoft SCCM, Ansible, Chef):

- а) Централизованное управление конфигурацией системных ресурсов сервера. Возможность интеграции с распределенными ресурсами систем хранения данных (iSCSI, Samba, Ceph, HDFS) с целью обеспечения требований производительности и надежности;
- б) Централизованное управление конфигурацией системных ресурсов сервера. Возможность интеграции с распределенными ресурсами систем баз данных с целью обеспечения требований производительности и надежности;
- в) Централизованное управление обновлениями программных компонент операционной системы;
- г) Централизованное управление обновлениями компонент базового программного обеспечения;
- д) Централизованное управление обновлениями программных компонент из состава программного комплекса «Программно-определяемая лаборатория «ПОЛАТОР».

Для мониторинга состояния аппаратных и программных компонент из состава «ПОЛАТОР-Сервер» предусмотрен интерфейс интеграции в системы управления ИТ-инфраструктурой (Microsoft SCCM, Zabbix, Nagios):

- а) Централизованное управление состоянием аппаратных компонент на базе SNMP;
- б) Мониторинг состояния аппаратных компонент на базе SNMP и SNMP-ловушек;
- в) Централизованное управление состоянием программных компонент на базе SNMP;

- г) Мониторинг состояния программных компонент на базе SNMP и SNMP-ловушек;
- д) Централизованный мониторинг журналов системных событий;
- е) SLA-мониторинг, формирование отчётов и тенденций;
- ж) Комплексная реакция на события с возможностью расширения за счёт выполнения внешних скриптов.

3. НАСТРОЙКА ПРОГРАММЫ

3.1. Описание установочного комплекта

Все требуемые установочные комплекты для установки программного компонента «ПОЛАТОР-Сервер» находятся на съемном носителе из состава программного изделия «Изделие «Программно-определяемая лаборатория «ПОЛАТОР»»:

- а) SERVER_INSTALL.EXE - Стартовый установочный комплект «ПОЛАТОР-Сервер»;
- б) INSTALL_SRV_SEC.EXE - Установочный комплект Сервера Информационной Безопасности;
- в) INSTALL_WS_SEC.EXE - Установочный комплект Автоматизированного Рабочего Места Администратора Информационной Безопасности;
- г) INSTALL_SRV_ADM.EXE - Установочный комплект Сервера Системного Администрирования;
- д) INSTALL_WS_ADM.EXE - Установочный комплект Автоматизированного Рабочего Места Системного Администратора;
- е) INSTALL_SRV_RES.EXE - Установочный комплект Сервера Управления Ресурсами;
- ж) INSTALL_SRV_DAQ.EXE - Установочный комплект Сервера Обмена Данными;
- з) INSTALL_SRV_APP.EXE - Установочный комплект Сервера Прикладных Задач;
- и) INSTALL_WS_APP.EXE - Установочный комплект Автоматизированного Рабочего Места Пользователя;
- к) POSTGRESQL.EXE - Установочный комплект СУБД PostgreSQL.

В комплект поставки входит Лицензионное Соглашение, которое поставляется на отдельном специализированном носителе с интерфейсом USB.

3.2. Описание установки программы

Предполагается, что перед началом установки выполнены следующие условия:

- а) Произведено развертывание и аттестация функционирования аппаратных средств;
- б) Наличие учетной записи пользователя с привилегированными правами в операционной системе, на которую производится установка конфигурации;
- в) Наличие полного установочного комплекта в соответствии с разделом 3.1 «Описание установочного комплекта»;

г) Наличие съемного носителя с Лицензионным Соглашением.

Далее специфицирован процесс установки программного обеспечения программного компонента «ПОЛАТОР-Сервер». Для установки необходимо выполнить шаги, описанные ниже:

- а) Произведите установку прилагаемого лицензионного ключа в свободный USB-порт компьютера.
- б) Произведите установку съемного носителя из состава программного изделия «Изделие «Программно-определяемая лаборатория «ПОЛАТОР» в устройство считывания DVD-дисков.
- в) В программе «Проводник» операционной системы Windows откройте съемный носитель Изделия. «Проводник» выведет файлы, находящиеся на съемном носителе.
- г) Запустите программу SERVER_INSTALL.EXE.
- д) Дождитесь открытия окна программы установки Сервера.
- е) В открывшемся окне внимательно ознакомьтесь с лицензионным соглашением на использование программы, после чего подтвердите свое согласие на использование программы путем установки отметки в поле «Я согласен с условиями лицензионного соглашения». Выполните одно из следующих действий:
 - ж) нажмите кнопку «Далее» в случае согласия с условиями лицензии, сценарий установки перейдет к следующему шагу;
 - з) нажмите «Отмена» в случае несогласия с условиями лицензии, программа установки будет закрыта, установка будет прервана.
- и) Программа установки предложит выбор установочного пути Сервера. При необходимости изменения пути нажмите кнопку «Обзор» и выберите каталог установки Сервера. Для подтверждения выбранного пути нажмите кнопку «Далее».
- к) Программа установки предложит указать для учетной записи, обладающей исключительным набором прав, имя и пароль, а также запросит его подтверждение. Обратите внимание на выдвигаемые требования к имени и паролю:
 - л) при вводе имени допускается использование латинских букв и цифр;
 - м) регистр букв в имени не учитывается;
 - н) имя должно содержать не менее 3 символов;
 - о) при вводе пароля допускается использование букв, цифр и служебных символов;
 - п) регистр букв в пароле учитывается;
 - р) пароль должен содержать не менее 8 символов, в том числе:
 - с) буквы в верхнем и нижнем регистрах;
 - т) как минимум одну цифру;
 - у) как минимум один служебный символ.
- ф) Укажите имя и пароль учетной записи, произведите подтверждение пароля, затем нажмите кнопку «Далее».
- х) Программа установки запросит подтверждение создания учетной записи в диалоговом окне. Произведите подтверждение, нажав кнопку

«Да», или вернитесь к указанию имени и пароля учетной записи, нажав кнопку «Нет».

- ц) Программа установки начнет копирование данных, сопровождаемое выводом информации в журнал установки.
- ч) По завершению копирования программа уведомит об успешной установке Сервера. Нажмите кнопку «Завершить».

3.3. Настройка программы

Данный раздел специфицирует состав задаваемых на стадии эксплуатации параметров функционирования и процессы их задания для программного компонента «ПОЛАТОР-Сервер».

Настройка программного компонента «ПОЛАТОР-Сервер» осуществляется с использованием программного компонента «ПОЛАТОР-Клиент» и включает в себя настройку учетных записей Сервера. Для этого необходимо обратиться к Руководству оператора (БЮЛИ.00131-01 34 01) и произвести настройку учетных записей и параметров согласно данного руководства.

Таблица 1 - Параметры SRV_ADM

Параметр	Спецификация и процесс установки
Уровень детализации	Задание уровня детализации регистрации событий в журнале событий. Область значений параметра специфицирована в разделе 5.3.7 данного документа
Время жизни зарегистрированного события	Задание длительности периода нахождения зарегистрированного события в журнале событий. Единицы измерения: Час; Область значений: 1 – 100 000
Время жизни события в архиве	Задание длительности периода нахождения зарегистрированного события в архиве. Единицы измерения: Час; Область значений: 1 – 100 000

4. ПРОВЕРКА ПРОГРАММЫ

4.1. Общие сведения

- 1) Проверка программы ПОЛАТОР-Сервер производится с целью получить общее заключение о работоспособности программы на всех этапах исполнения программы.
- 2) Проверка программы ПОЛАТОР-Сервер может осуществляться в двух режимах:
 - Автоматическая проверка,
 - Полуавтоматическая проверка.
- 3) В случае, если проверка обнаружила неисправность программы ПОЛАТОР-Сервер, дальнейшие выводы о причинах неисправности делаются путём изучения журнала подсистемы регистрации событий сервера и журнала подсистемы регистрации событий операционной системы.
- 4) Перед проведением проверки программы ПОЛАТОР-Сервер необходимо убедиться, что программа запущена, функционирует и отсутствуют сообщения операционной системы о невозможности выполнения программы.

4.2. Автоматическая проверка

- 1) Автоматическая проверка программы ПОЛАТОР-Сервер производится путём периодического запуска цепочки тестов и сравнения результата тестов с ожидаемым. Запуск тестов производит программа ПОЛАТОР-Сервер автоматически в соответствии с заложенным в ней алгоритмом.
- 2) В случае, если тесты завершились успешно, сообщения не выводятся.
- 3) В случае, если какой-либо тест завершился неудачно, на экране компьютера появится следующее сообщение операционной системы: «<Время> ПОЛАТОР-Сервер – Тест <Номер теста> неудачен», где
 - <Время> - текущее время,
 - <Номер теста> - номер теста, который завершился неудачно.
- 4) Перечень и функциональная направленность тестов приведены в п. 4.3 настоящего документа.

4.3. Полуавтоматическая проверка

- 1) Полуавтоматическая проверка проводится системным программистом путём ручного запуска тестов и сравнения результатов тестов с ожидаемыми.

- 2) Ниже описаны порядок проведения тестов, описание тестов и описание ожидаемых результатов.

4.3.1. Проверка авторизации пользователя

- 1) Проверка авторизации пользователя производится путём создания новой учетной записи пользователя и авторизации его.
- 2) Чтобы проверить возможность авторизации пользователя, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test1» и выполнить её;
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test1: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появились записи:
«<Время> Пользователь Иванов И.И. создан»,
«<Время> Пользователь Иванов И.И. авторизован»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.2. Проверка настройки

- 1) Проверка настройки сервера производится путём добавления и удаления учетной записи пользователя.
- 2) Чтобы проверить возможность изменения настроек сервера, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test2» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test2: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появились записи:
«<Время> Пользователь Фаридов Ф.Ф. добавлен»,
«<Время> Пользователь Фаридов Ф.Ф. удалён»,
«<Время> Пользователь Фаридов Ф.Ф. добавлен»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.3. Проверка создания проекта и подключения пользователей к нему

- 1) Проверка создания проекта и подключения пользователей к проекту производится путём создания 3-х новых проектов, добавления новых компонентов в проект и предоставления одному из пользователей права на доступ к проекту.
- 2) Чтобы проверить возможность создания проекта и подключения пользователей к нему, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test3» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test3: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появились записи:
«<Время> Проект Сервер Тест 1 создан пользователем Иванов И.И.»,
«<Время> Проект Сервер Тест 2 создан пользователем Иванов И.И.»,
«<Время> Проект Сервер Тест 3 создан пользователем Иванов И.И.»,
«<Время> Пользователю Фаридов Ф.Ф. предоставлено право на доступ к проекту Сервер Тест 1»,
«<Время> Пользователь Иванов И.И. добавил Компонент GPL 1 в проект Сервер Тест 1»,
«<Время> Пользователь Иванов И.И. добавил Компонент GPL 2 в проект Сервер Тест 1»,
«<Время> Пользователь Иванов И.И. добавил Компонент SPL в проект Сервер Тест 1»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.4. Проверка сохранения данных проекта

- 1) Проверка сохранения данных проекта производится путём исполнения запроса пользователя на сохранение проекта.
- 2) Чтобы проверить возможность сохранения данных проекта, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test4» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test4: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появились записи:
«<Время> Проект Сервер Тест 1 сохранён пользователем Иванов И.И.»,

где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.5. Проверка предоставления данных проекта

- 1) Проверка предоставления данных проекта производится путём исполнения запроса пользователя на предоставление доступа к данным проекта.
- 2) Чтобы проверить возможность предоставления данных проекта, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test5» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test5: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появилась запись:
«<Время> Пользователю Фаридов Ф.Ф. предоставлен доступ к данным проекта Сервер Тест 1»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.6. Проверка открытия существующего проекта

- 1) Проверка открытия существующего проекта производится путём исполнения запроса пользователя на открытие проекта пользователем для редактирования.
- 2) Чтобы проверить возможность предоставления данных проекта, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test6» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test6: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появилась запись:
«<Время> Пользователь Фаридов Ф.Ф. открыл проект Сервер Тест 1»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

4.3.7. Проверка включения проекта в список проектов, размещенных на сервере

- 1) Проверка включения проекта в список проектов, размещенных на сервере, производится путём просмотра списка созданных во время предыдущих проверок проектов.
- 2) Чтобы проверить возможность предоставления данных проекта, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test7» и выполнить её.
 - Убедиться, что сообщения операционной системы соответствуют следующим:
«<Время> server-polator-test7: Список проектов:
Сервер Тест 1,
Сервер Тест 2,
Сервер Тест 3»,
«<Время> server-polator-test7: выполнено успешно»,
где <Время> - время запуска теста.

4.3.8. Проверка открытия содержания проекта в виде перечня компонентов, которые включены в проект, с указанием имени каждого компонента

- 1) Проверка открытия содержания проекта в виде перечня компонентов, которые включены в проект, с указанием имени каждого компонента, производится путём просмотра списка созданных во время предыдущих проверок компонентов одного из проектов.
- 2) Чтобы проверить возможность открытия содержания проекта в виде перечня компонентов, которые включены в проект, с указанием имени каждого компонента, следует выполнить следующие шаги:
 - В командной строке операционной системы ввести команду «server-polator-test8» и выполнить её.
 - Убедиться, что сообщения операционной системы соответствуют следующим:
«<Время> server-polator-test8: Компоненты проекта Сервер Тест 1:
Компонент GPL 1,
Компонент GPL 2,
Компонент SPL»,
«<Время> server-polator-test8: выполнено успешно»,
где <Время> - время запуска теста.

4.3.9. Проверка одновременного управления всеми компонентами, включенными в проект

- 1) Проверка одновременного управления всеми компонентами, включенными в проект, производится путём одновременного изменения компонентов проекта двумя разными пользователями.

- 2) Чтобы проверить возможность открытия содержания проекта в виде перечня компонентов, которые включены в проект, с указанием имени каждого компонента, следует выполнить следующие шаги:
- В командной строке операционной системы ввести команду «server-polator-test9» и выполнить её.
 - Убедиться, что сообщение операционной системы соответствует следующему:
«<Время> server-polator-test9: выполнено успешно»,
где <Время> - время запуска теста.
 - Открыть журнал подсистемы регистрации событий сервера и убедиться, что там появились записи:
«<Время> Пользователь Фаридов Ф.Ф. изменил Компонент GPL 1 в проекте Сервер Тест 1»,
«<Время> Пользователь Иванов И.И. изменил Компонент GPL 2 в проекте Сервер Тест 1»,
«<Время> Пользователь Фаридов Ф.Ф. изменил Компонент SPL в проекте Сервер Тест 1»,
где <Время> - время запуска теста и оно совпадает со временем в сообщении операционной системы.

5. СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ

5.1. Общие положения

В составе программного продукта «ПОЛАТОР-Сервер» предусмотрена «Система Управления Событиями» (EVN_CTRL). «Система Управления Событиями» функционально входит в состав подсистемы «Администрирования» (SRV_ADM).

Данная подсистема предназначена для обеспечения:

- а) Сопровождаемости как программного продукта в целом, так и входящих в него компонент;
- б) Наблюдаемости и контролируемости программного продукта;
- в) Оптимизации значений показателя «Среднее время восстановления» после сбоя или отказа за счет сокращения времени обнаружения;
- г) Оптимизации значений показателя «Среднее время между сбоями или отказами» за счет управления проведения превентивных алгоритмов и регламентов обслуживания.

Для программного продукта «ПОЛАТОР-Сервер» на платформе Windows «Система Управления Событиями» реализована на базе технологии Трассировка Событий для Windows (ETW - Event Tracing for Windows). Данная технология обеспечивает высокие показатели в части безопасности, надежности и производительности.

Для программного продукта «ПОЛАТОР-Сервер» на платформах Linux «Система Управления Событиями» реализована на базе технологий SysLog в соответствии с требованиями RFC 5424. Данная технология также обеспечивает высокие показатели в части безопасности, надежности и производительности на данной операционной платформе.

5.2. Функциональная структура

В данном разделе специфицирована функциональная структура «Система Управления Событиями».

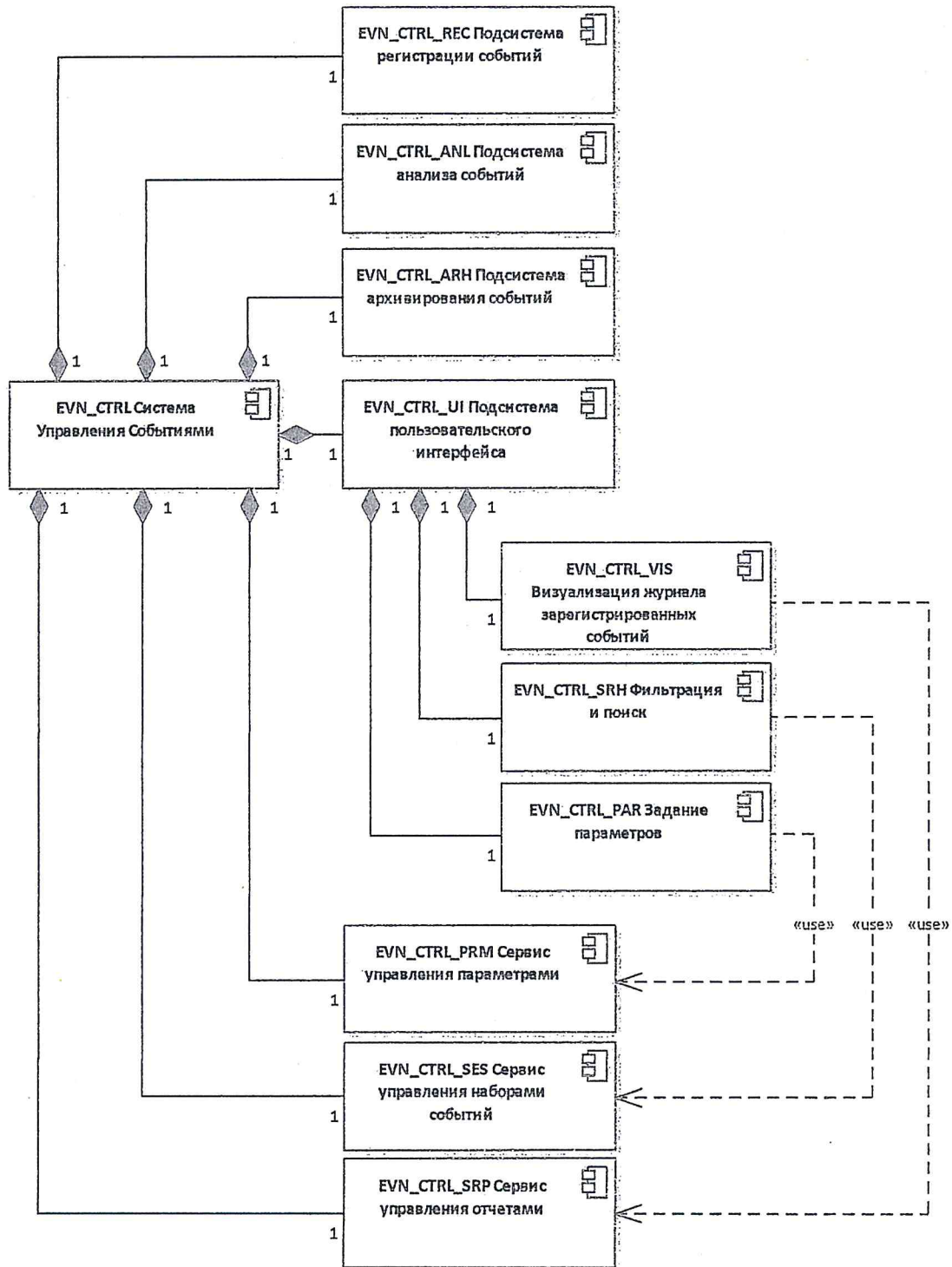


Рисунок 4: Функциональная структура «Управления Событиями».

ID	Описание
EVN_CTRL_REC	Подсистема регистрации событий.

EVN_CTRL_ANL	Подсистема анализа событий.
EVN_CTRL_ARH	Подсистема архивирования событий.
EVN_CTRL_UI	Подсистема пользовательского интерфейса.
EVN_CTRL_VIS	Визуализация журнала зарегистрированных событий.
EVN_CTRL_SRH	Фильтрация и поиск.
EVN_CTRL_PAR	Задание параметров.
EVN_CTRL_PRM	Сервис управления параметрами.
EVN_CTRL_SES	Сервис управления наборами событий.
EVN_CTRL_SRP	Сервис управления отчетами.

5.3. Спецификация модели события

Информационная модель регистрации событий приведена на приведенном ниже рисунке.

Применение технологий ORM (Object-Relational Mapping – объектно-реляционное отображение) позволяет частично использовать применяемые в реляционной модели подходы.

Для исключения избыточности данных при регистрации событий приведенная выше информационная модель обеспечивает соответствие требованиям третьей нормальной формы.

Следующие разделы специфицируют элементы приведенной выше информационной модели.

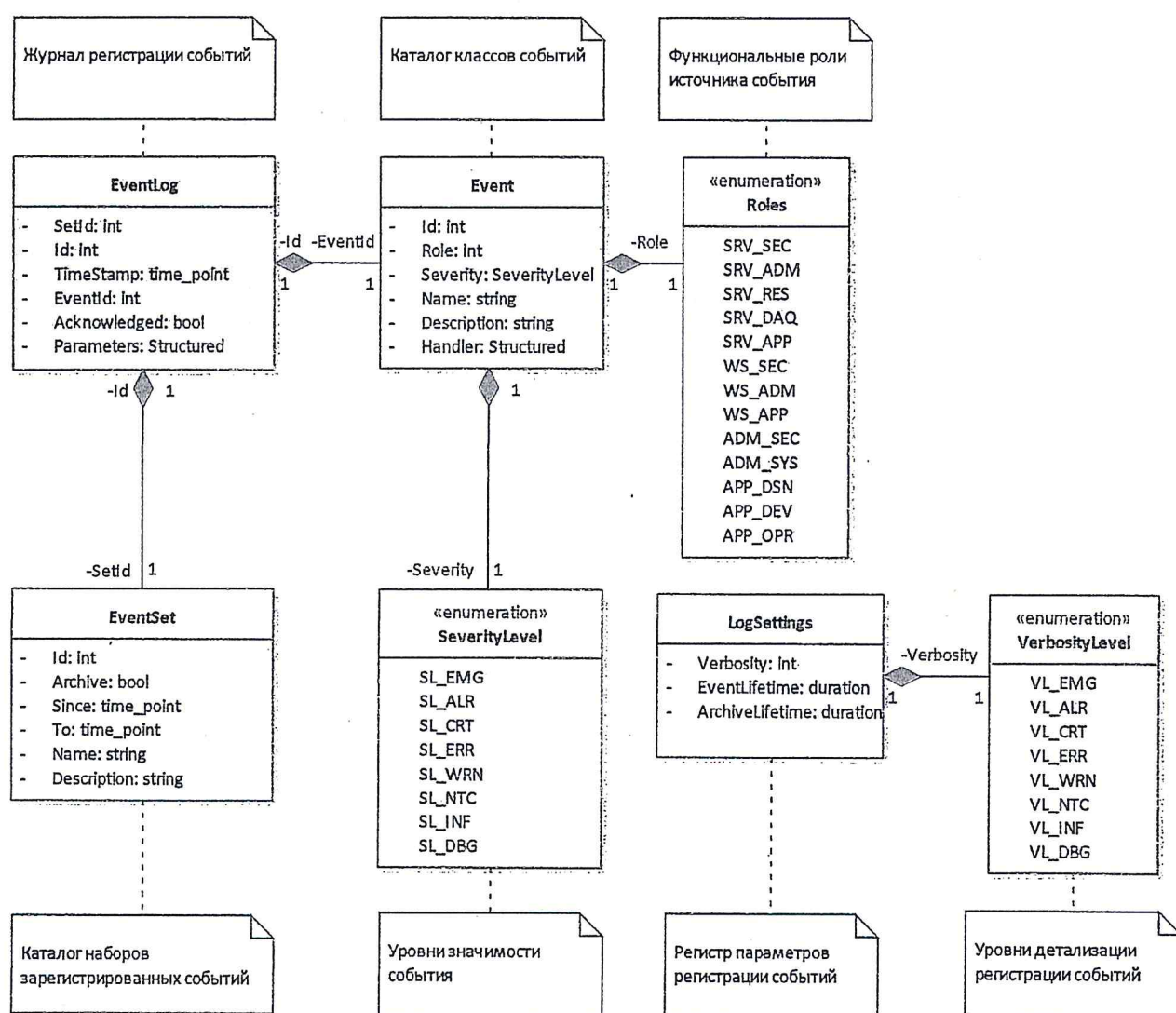


Рисунок 5. Информационная модель «Управления Событиями»

5.3.1. Журнал регистрации событий

Основной компонент «Система Управления Событиями». Является общесистемным журналом регистрации событий.

Обеспечивает соответствие требованиям третьей нормальной формы.

ID	Описание
SetId	Идентификатор набора событий: [0] - Оперативный журнал; [1 : MaxLong] - Выборка или архив
Id	Идентификатор события. Применяется при квитировании события
TimeStamp	Метка времени события
EventId	Идентификатор класса события
Acknowledged	Признак подтверждения события
Parameters	Опциональные параметры события

«Система Управления Событиями» предусматривает наличие произвольного числа экземпляров «Журнала Регистрации Событий».

Экземпляр «Журнала Регистрации Событий» с SetId = 0 является оперативным журналом регистрации событий, в котором регистрируются текущие события, для которых разница между текущей меткой времени и TimeStamp не превышает значения параметра «Время жизни события».

Зарегистрированные события, для которых разница между текущей меткой времени и TimeStamp превышает значение параметра «Время жизни события» находятся в «Архиве Зарегистрированных Событий», структура которого полностью соответствует EventLog.

В «Системе Управления Событиями» предусмотрена возможность сохранения произвольной выборки событий, которая сформирована применением соответствующих предусмотренных методов фильтрации, в отдельном идентифицированном хранилище, структура которого полностью соответствует EventLog.

5.3.2. Каталог наборов событий

Каталог наборов зарегистрированных событий (EventSet).

ID	Описание
Id	Идентификатор набора событий: [0] - Оперативный журнал; [1 : MaxLong] - Выборка или архив
Archive	Признак архива зарегистрированных событий
Since	Минимальная метка времени набора
To	Максимальная метка времени набора
Name	Опциональное название набора зарегистрированных событий
Description	Опциональное описание набора зарегистрированных событий

5.3.3. Регистр параметров регистрации событий

Регистр параметров регистрации событий (LogSettings).

ID	Описание
Verbosity	Общесистемный уровень детализации регистрации событий
EventLifetime	Время жизни зарегистрированного события в оперативном журнале регистрации событий
ArchiveLifetime	Время жизни зарегистрированного события в архиве событий

С целью оптимизации использования системного ресурса в «Системе Управления Событиями» предусмотрен задаваемый из пользовательского интерфейса параметр «Время жизни события». Данный параметр специфицирует максимальное время жизни записи зарегистрированного события в Общесистемном Журнале Регистрации событий (EventLog). После истечения специфицированного данным параметром промежутка времени зарегистрированное событие автоматически перемещается в Архив Событий. Архив Событий предусматривает пользовательский интерфейс для просмотра.

С целью оптимизации использования системного ресурса в «Системе Управления Событиями» предусмотрен задаваемый из пользовательского интерфейса параметр «Время жизни архива события». Данный параметр специфицирует максимальное время жизни записи в Архиве Событий. После истечения специфицированного данным параметром промежутка времени зарегистрированное событие автоматически удаляется.

5.3.4. Каталог классов событий

Применяется для спецификации класса событий (Event).

Обеспечивает соответствие требованиям третьей нормальной формы.

ID	Описание
Id	Идентификатор класса события
SystemId	Идентификатор системы-источника класса события
Severity	Уровень значимости класса события
Name	Название класса события
Description	Описание класса события
Handler	Опциональный обработчик события. Интерпретируемый сценарий на языке SPL. Выполняется с параметрами EventLog.Parameters

5.3.5. Регистр системной иерархии

Применяется для спецификации системной иерархии (System) программных компонент (систем и подсистем) из состава программного продукта «ПОЛАТОР-Сервер».

Обеспечивает соответствие требованиям третьей нормальной формы.

ID	Описание
Id	Идентификатор системы
Role	Идентификатор функциональной роли системы
Name	Название системы
Description	Описание системы

5.3.6. Уровень значимости

Для каждого события определяется «Уровень значимости» (Severity Level).
В «Системе Управления Событиями» предусмотрены следующие уровни

значимости:

ID	Название	Описание
SL_EMG	Чрезвычайная ситуация (Emergency)	Дальнейшее функционирование системы невозможно
SL_ALR	Тревога (Alert)	Дальнейшее функционирование системы в полном проектом объеме невозможно. Частичная потеря функциональности без возможности автоматического или автоматизированного восстановления
SL_CRT	Критическая ошибка (Critical)	Дальнейшее функционирование системы в полном проектом объеме невозможно. Частичная потеря функциональности, выполнение алгоритмов и регламентов в автоматическом или автоматизированном режиме с целью восстановления функциональности
SL_ERR	Ошибка (Error)	Отказ функционирования
SL_WRN	Предупреждение (Warning)	Сбой при функционировании
SL_NTC	Уведомление (Notice)	Информационное сообщение. Требуется квитирования – действия со стороны эксплуатирующего персонала по причине того, что дальнейшее выполнение алгоритмов в автоматическом режиме невозможно и требуется решение от персонала. Пример: Запрос решения оператора по продолжению выполнения прикладной программы на языке SPL при выходе значения параметра за диапазон
SL_INF	Информационное сообщение (Informational)	Информационное сообщение. Не требует квитирования. Пример: регистрация факта добавления в проект элемента на языке GPL

ID	Название	Описание
SL_DBG	Сообщение отладки (Debug)	<p>Применяется в следующих стадиях и режимах функционирования:</p> <ol style="list-style-type: none"> 1. «Эксплуатация. Режим плановых регламентов технического обслуживания»; 2. «Эксплуатация. Режим внеплановых регламентов ремонта и восстановления после сбоев и отказов»; 3. «Эксплуатация. Режим модификации»; 4. «Модернизация» <p>при выполнении алгоритмов и регламентов с целью регистрации факта их выполнения и полученных результатов.</p>

5.3.7. Уровень детализации

Для каждой из функциональных подсистем задается в качестве параметра «Уровень детализации» (Verbosity Level), который используется при регистрации события в «Общесистемном журнале регистрации событий» EventLog. Задание значения уровня детализации производится как при помощи графического пользовательского интерфейса WS_ADM, так и при помощи пользовательского интерфейса командной строки WS_ADM. Значение параметра «Уровень детализации» задает состав событий, которые подлежат регистрации в соответствии с их уровнем значимости. Состав событий для регистрации для каждого из значений уровня детализации специфицирован в далее приведенной таблице.

ID	Название	Описание
VL_EMG	Чрезвычайная ситуация (Emergency)	Регистрируются события уровней: SL_EMG.
VL_ALR	Тревога (Alert)	Регистрируются события уровней: SL_EMG; SL_ALR.
VL_CRT	Критическая ошибка (Critical)	Регистрируются события уровней: SL_EMG; SL_ALR; SL_CRT.
VL_ERR	Ошибка (Error)	Регистрируются события уровней: SL_EMG; SL_ALR; SL_CRT; SL_ERR.
VL_WRN	Предупреждение (Warning)	Регистрируются события уровней: SL_EMG;

ID	Название	Описание
		SL_ALR; SL_CRT; SL_ERR; SL_WRN.
VL_NTC	Уведомление (Notice)	Регистрируются события уровней: SL_EMG; SL_ALR; SL_CRT; SL_ERR; SL_WRN; SL_NTC.
VL_INF	Информационное сообщение (Informational)	Регистрируются события уровней: SL_EMG; SL_ALR; SL_CRT; SL_ERR; SL_WRN; SL_NTC; SL_INF.
VL_DBG	Сообщение отладки (Debug)	Регистрируются события уровней: SL_EMG; SL_ALR; SL_CRT; SL_ERR; SL_WRN; SL_NTC; SL_INF; SL_DBG.

5.3.8. Функциональные роли источника события

Спецификация функциональных ролей систем, которые инициировали событие.

ID	Описание
SRV_SEC	источником события является программный модуль из состава Подсистемы информационной безопасности
SRV_ADM	источником события является программный модуль из состава Подсистемы Системного Администрирования
SRV_RES	источником события является программный модуль из состава Подсистемы Управления Ресурсами
SRV_DAQ	источником события является программный модуль из состава Подсистемы Обмена Данными
SRV_APP	источником события является программный модуль из состава Подсистемы Прикладных Задач

ID	Описание
WS_ADM	источником события является программный модуль из состава Рабочего Места Администратора Информационной Безопасности
WS_APP	источником события является программный модуль из состава Рабочего Места Системного Администратора
WS_SEC	источником события является программный модуль из состава Рабочего Места Пользователя
ADM_SEC	иницировано действиями Администратора Информационной Безопасности
ADM_SYS	иницировано действиями Системного Администратора
APP_DSN	иницировано действиями Инженера-Проектировщика прикладных моделей на языке GPL
APP_DEV	иницировано действиями Инженера-Программиста прикладных программ на языке SPL
APP_OPR	иницировано действиями Оператора

5.4. Интерфейс пользователя

«Система Управления Событиями» обеспечивает интерфейс пользователя для:

1. Рабочее место Администратора Информационной Безопасности (WS_SEC);
2. Рабочее место Системного Администратора (WS_ADM);
3. Рабочее место Пользователя (WS_APP).

Рабочее место Системного Администратора (WS_ADM) является основным интерфейсом пользователя «Системы Управления Событиями» и реализует полный комплекс специфицированной функциональности.

5.5. Спецификация событий

В данном разделе специфицированы события. Спецификация событий сгруппирована по функциональным подсистемам.

5.5.1. Подсистема Обмена Данными

ID	Значимость	Описание
EMG101	SL_EMG	Не обнаружен путь к файлу.
EMG102		Нет доступа к хранилищу.
ALR101	SL_ALR	Нет связи с хранилищем.
ALR102		Нет доступа к элементу.
CRT101	SL_CRT	Сбой сохранения файла.
CRT102		Недостаточно памяти на хранилище.
ERR101	SL_ERR	Недостаточно информации для сохранения.
ERR102		Связь недоступна.
WRN101	SL_WRN	Неправильный формат сохранения.
WRN102		Файл не доступен.
NTC101	SL_NTC	Превышен лимит обращений к базе данных.
NTC102		Недостаточно прав оператора при обращении к базе данных.
INF101	SL_INF	Произведено сохранение файла.
INF102		Произведено удаление файла.
DBG101	SL_DBG	Производится обновление системы.
DBG102		Восстановление системы после внепланового выключения.

5.5.2. Подсистема Прикладных Задач

ID	Значимость	Описание
EMG201	SL_EMG	Библиотека функций недоступна.
EMG202		Недостаточно памяти на хранилище.
ALR201	SL_ALR	Функция работает некорректно.
ALR202		Элемент недоступен.
CRT201	SL_CRT	Нет связи между элементами.
CRT202		Функция недоступна.
ERR201	SL_ERR	Присутствуют неопределенные входные параметры.
ERR202		Обрыв цепи с оборудованием.
WRN201	SL_WRN	Возможно деление на ноль.
WRN202		Библиотека не подключена.
NTC201	SL_NTC	Число доступных значений превышено.
NTC202		Превышено допустимое значение по диапазону.
INF201	SL_INF	Добавлен элемент на языке GPL.
INF202		Удаление элемента в диаграмме.
DBG201	SL_DBG	Обновление диаграммы.
DBG202		Производится подключение библиотеки.

5.5.3. Подсистема Пользователя

ID	Значимость	Описание
EMG301	SL_EMG	Файлы программы повреждены.
EMG302		Ошибка обращения к памяти.
ALR301	SL_ALR	Нет подключения.
ALR302		Лицензия недоступна.
CRT301	SL_CRT	Недостаточно оперативной памяти.
CRT302		Не установлен лицензионный ключ.
ERP301	SL_ERR	Недостаточно прав пользователя.
ERP302		Лицензия истекла.
WRN301	SL_WRN	Проект редактируется другим пользователем.
WRN302		Пароль введен неверно.
NTC301	SL_NTC	Истекает срок действия лицензии.
NTC302		Истекает срок действия пароля учетной записи.
INF301	SL_INF	Имя учетной записи изменено
INF302		Добавлена новая учетная запись.
DBG301	SL_DBG	Производится обновление системы.
DBG302		Производится настройка системы.

5.5.4. Подсистема Управления Ресурсами

ID	Значимость	Описание
EMG401	SL_EMG	Ресурс недоступен.
EMG402		Нет прав доступа к ресурсу.
ALR401	SL_ALR	Определение функции изменилось.
ALR402		Ресурс поврежден.
CRT401	SL_CRT	Недостаточно памяти.
CRT402		Повреждение информационной ресурса.
ERP401	SL_ERR	Ресурс не найден.
ERP402		Зависимости ресурса не найдены.
WRN401	SL_WRN	Функция не найдена в ресурсе.
WRN402		Конфликт функций.
NTC401	SL_NTC	Ресурс не содержит функций.
NTC402		Ошибка параметров вызова.
INF401	SL_INF	Ресурс добавлен.
INF402		Ресурс удален.
DBG401	SL_DBG	Ресурс загружен.
DBG402		Получен список функции.

ПЕРЕЧЕНЬ ТЕРМИНОВ

Применяемые в руководстве термины и их определения приведены в таблице ниже (Таблица 2).

Таблица 2 - Термины

Термины	Значение
Изделие	«Изделие «Программно-определяемая лаборатория «ПОЛАТОР»». По тексту документа - Изделие
Клиент	Программный компонент «ПОЛАТОР-Клиент». По тексту документа - Клиент
Комплекс	Программный комплекс «Программно-определяемая лаборатория «ПОЛАТОР»». По тексту документа - Комплекс
Носитель	Покупное изделие - съемный носитель информации DVD-R
Программный комплекс	Программа, состоящая из двух или более компонентов и (или) комплексов, выполняющих взаимосвязанные функции, и применяемая самостоятельно или в составе другого комплекса
Программный компонент	Программа, рассматриваемая как единое целое, выполняющая законченную функцию и применяемая самостоятельно или в составе комплекса
Радиоэлектронная аппаратура (РЭА)	Изделие, предназначенное для передачи, приёма, информации на расстояние по радиоканалу при помощи электромагнитных сигналов
Сервер	Программный компонент «ПОЛАТОР-Сервер». По тексту документа - Сервер

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Применяемые в руководстве сокращения и их расшифровка приведены в таблице ниже (Таблица 3).

Таблица 3 - Сокращения

Сокращение	Расшифровка
DVD	Цифровой многоцелевой диск
ЕСПД	Единая система программной документации
КД	Конструкторская документация
НТС	Научно-технический совет
ОКР	Опытно-конструкторская работа
ОО	Опытный образец
ОС	Операционная система
ОТД	Отдел технической документации
ОТК	Отдел технического контроля
ПИ	Предварительные испытания
ПСИ	Приемочные испытания
РКД	Рабочая конструкторская документация
РЭА	Радиоэлектронная аппаратура
СЗ	Служебная записка
ТЗ	Техническое задание
ТП	Технический проект
ЭД	Эксплуатационная документация

ПЕРЕЧЕНЬ РИСУНКОВ

Рисунок 1: Функциональная структура «ПОЛАТОР-Сервер»	9
Рисунок 2: Обмен данными через БД.....	11
Рисунок 3: Структура доменов БД.....	12
Рисунок 4: Функциональная структура «Управления Событиями».....	26
Рисунок 5. Информационная модель «Управления Событиями»	28

ПЕРЕЧЕНЬ ТАБЛИЦ

Таблица 1 - Параметры SRV_ADM.....	18
Таблица 2 - Термины.....	37
Таблица 3 - Сокращения	38

ССЫЛОЧНЫЕ ДОКУМЕНТЫ

- [1] ГОСТ 19.004-80 - «Единая система программной документации. Термины и определения»;
- [2] ГОСТ 19.201-78 - «Единая система программной документации. Техническое задание. Требования к содержанию и оформлению»;
- [3] ГОСТ 19.505-79 - «Единая система программной документации. Руководство оператора. Требования к содержанию и оформлению»;
- [4] ГОСТ 19.301-79 - «Единая система программной документации. Программа и методика испытаний. Требования к содержанию и оформлению»;
- [5] ГОСТ 19.101-77 - «Единая система программной документации. Виды программ и программных документов»;
- [6] ГОСТ 19.105-78 - «Единая система программной документации. Общие требования к программным документам»;

