

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

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский
подпись

«____ » _____ 2019 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 — Информационные системы и технологии

Модуль управления процессами многоцелевой региональной системы ДЗ3

Руководитель _____ доцент каф. СИИ, канд. техн. наук Р. В. Брежнев
подпись, дата

Выпускник _____ Д. А. Минибаева
подпись, дата

Красноярск 2019

Продолжение титульного листа бакалаврской работы по теме «Модуль управления процессами многоцелевой региональной системы ДЗЗ».

Нормоконтролер

подпись, дата

Р. В. Брежнев

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

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский
подпись

«____ » _____ 2019 г.

**ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы**

Студенту Минибаевой Дарье Андреевне.

Группа КИ15-11Б, направление 09.03.02 «Информационные системы и технологии».

Тема выпускной квалификационной работы «Модуль управления процессами многоцелевой региональной системы ДЗЗ».

Утверждена приказом по университету № 6523/с от 16 мая 2019 г.

Руководитель ВКР Р. В. Брежнев, доцент кафедры систем искусственного интеллекта ИКИТ СФУ, кандидат технических наук.

Исходные данные для ВКР: задание на бакалаврскую работу, полученное в рамках научно-учебной лаборатории "Информационной поддержки космического мониторинга" ИКИТ СФУ.

Перечень разделов ВКР:

- Введение;
- Глава 1. Обзор существующих решений управления процессами в распределенных системах дистанционного мониторинга;
- Выводы по главе 1;
- Глава 2. Разработка модуля управления и автоматического запуска процессов;
- Выводы по главе 2;
- Заключение;
- Список использованных источников;
- Приложения А–Г.

Перечень графического материала: презентация «Разработка модуля управления и автоматического запуска процессов».

Руководитель ВКР

Р. В. Брежнев

подпись, дата

Задание принял к исполнению

Д. А. Минибаева

подпись, дата

«____» _____ 2019 г.

График

выполнения выпускной квалификационной работы студентом направления 09.03.02 «Информационные системы и технологии».

График выполнения выпускной квалификационной работы приведен в таблице 1.

Таблица 1 — График выполнения выпускной квалификационной работы

Наименование этапа	Срок выполнения	Результат выполнения этапа	Примечание руководителя (отметка о выполнении этапа)
Определение темы работы	28.01 — 10.02	Краткое эссе по теме ВКР	Выполнено
Сбор литературных источников	11.02 — 18.02	Список источников литературы	Выполнено
Анализ собранных литературных источников	19.02 — 5.03	Реферат о проблемно-предметной области	Выполнено
Уточнение и обоснование актуальности, цели и задач ВКР	6.03 — 13.03	Окончательная формулировка цели и задач ВКР	Выполнено
Решение первой задачи ВКР	14.03 — 5.04	Доклад и презентация по первой задаче ВКР	Выполнено
Решение второй задачи ВКР	6.04 — 20.04	Доклад и презентация по второй задаче ВКР	Выполнено
Решение третьей задачи ВКР	21.04 — 2.05	Доклад и презентация по третьей задаче ВКР	Выполнено
Решение четвертой задачи ВКР	3.05 — 12.06	Доклад и презентация по четвертой задаче ВКР	Выполнено
Подготовка доклада и презентации по теме ВКР	13.06 — 14.06	Доклад с презентацией по теме ВКР	Выполнено
Компоновка отчета по результатам решения задач ВКР	15.06 — 16.06	Отчет по результатам решения задач ВКР	Выполнено

Окончание таблицы 1.

Наименование этапа	Срок выполнения	Результат выполнения этапа	Примечание руководителя (отметка о выполнении этапа)
Первичный нормоконтроль	17.06	Пояснительная записка, презентация к ВКР	Выполнено
Предварительная защита результатов ВКР	19.06	Доклад и презентация по проделанной работе	Выполнено
Вторичный нормоконтроль	26.06	Пояснительная записка, презентация к ВКР	Выполнено
Итоговый нормоконтроль	28.06	Пояснительная записка, презентация к ВКР	Выполнено
Защита ВКР	01.07	Доклад и презентация по результатам бакалаврской работы	Выполнено

Руководитель

Р. В. Брежнев

подпись, дата

Студент группы КИ15-116

Д. А. Минибаева

подпись, дата

СОДЕРЖАНИЕ

Введение.....	4
1 Обзор существующих решений управления процессами в распределенных программных средах	6
1.1 Вега-PRO.....	6
1.2 ИСДМ Рослесхоз.....	8
1.3 See the Sea	9
1.4 Google Earth Engine.....	10
1.5 NASA GIOVANNI.....	12
1.2 Анализ архитектур рассмотренных систем.....	13
Вывод по главе 1	14
2 Разработка модуля управления и автоматического запуска процессов....	16
2.1 Диаграмма вариантов использования	16
2.2 Диаграмма деятельности.....	19
2.3 Схема взаимодействия.....	21
2.4 Разработка модуля	22
2.4.1 Структура данных для работы модуля	22
2.5 Алгоритм модуля запуска процессов.....	24
2.6 Реализация модуля запуска процессов	26
Вывод по главе 2	26
Заключение	28
Список использованных источников	29
Приложение А Видение.....	32
Приложение Б Глоссарий	37
Приложение В Спецификация требований к ИС.....	40
Приложение Г Плакаты презентации.....	57

ВВЕДЕНИЕ

В рамках многоцелевой региональной системы дистанционного зондирования Земли ИКИТ (*далее МЦРС ДЗЗ или Система*) создаётся подсистема управления потоками работ, для которой требуется разработать модуль управления и автоматического запуска процессов (*далее Модуль*).

Система является гетерогенной, а это значит, что она включает распределенные в пространстве серверы обработки, выполняющие определенные этапы обработки и анализа данных ДЗЗ под управлением различных операционных систем и технологий. Системные компоненты, следовательно, так же распределены между серверами. При таких условиях построение цепочек обработки данных в операторском режиме невозможно.

Архитектура системы построена по модульному принципу. Модульность реализовывает принцип повторного использования системных компонент при решении различных задач, в том числе в параллельном режиме. На физическом уровне такой подход позволяет представить заданную операцию по обработке или анализу данных в виде отдельного модуля, а набор операций в виде совокупности логически взаимосвязанных программных модулей, выполняющихся в заданной последовательности, с заданным набором входных параметров, в заданное время. Поскольку каких-либо унифицированных средств, автоматизирующих процесс многократного запуска программных компонент, широко не представлено, вопрос автоматизации является актуальным, так как эта ситуация приводит к необходимости постоянного участия и контроля со стороны оператора, что в условиях пространственной распределенности и многозадачности является неприемлемым решением.

Актуальность работы так же подтверждают и разработки ИКИ РАН [1], ВЦ СО РАН [2], ИАПУ ДВО РАН [3]. В основном указанные работы были ориентированы на специалистов в области обработки данных ДЗЗ и

одновременно имеющих опыт программирования и разрабатывающих собственное программное обеспечение.

Среди наиболее успешных и передовых решений по управлению процессами обработки данных ДЗЗ следует отметить подход ИКИ РАН, реализованный уже в более чем десятке систем мониторинга, среди которых ВЕГА-PRO, ИСДМ-Рослесхоз, See the Sea, НИЦ «Планета» и др.

Однако унифицированного решения до сих пор не сформировалось. Прежде всего, это связано с различным пониманием коллективов разработчиков внутреннего представления модели процесса, модели потока работ.

Предлагаемое решение, в сравнении с имеющимися, ориентировано на специалистов по обработке данных без опыта программирования. Его внедрение позволит значительно сократить участие оператора и обеспечить рациональное распределение ресурсов между параллельными процессами.

Таким образом, целью данной работы является оптимизация деятельности оператора МЦРС ДЗЗ по запуску процессов обработки данных ДЗЗ. Ожидается, что после внедрения разрабатываемого Модуля роль оператора в запуске процессов получения, распределения, обработки и анализа геопространственных данных будет значительно сокращена или упразднена.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- обзор существующих решений управления процессами в распределенных системах дистанционного мониторинга;
- выявление и анализ требований к Модулю;
- проектирование разрабатываемого Модуля;
- программная реализация Модуля.

1 Обзор существующих решений управления процессами в распределенных системах дистанционного мониторинга

Данные спутниковых систем наблюдения Земли являются одним из основных, а иногда единственным, источником информации о различных процессах в атмосфере и на поверхности планеты. Постоянно возрастающая потребность в данных и развитие технологий в последние десятилетия привели к появлению большого количества высококачественных систем дистанционного зондирования Земли из космоса, таких как Вега, ИСДМ Рослесхоз, See the Sea, Google Earth Engine, NASA GIOVANNI, обеспечивающих стабильное и оперативное получение информации по территории всей поверхности планеты [4].

1.1 Вега-PRO

Спутниковый сервис ВЕГА предназначен для решения задач дистанционного мониторинга растительного покрова [5]. Сервис Вега входит в состав «Созвездие-Вега».

«Созвездие-Вега» — это семейство информационных систем ИКИ РАН, ориентированных на создание на основе единой технологической платформы информационных сервисов, обеспечивающих возможность оперативной удаленной работы различных групп пользователей с данными спутниковых наблюдений для решения широкого круга научных и прикладных задач по исследованию биологических ресурсов. Это семейство систем собирает спутниковую и другую географическую информацию из различных источников и обеспечивают глобальный доступ к ней. При этом обеспечивается работа с данными в режиме реального времени посредством взаимосвязанных динамических сервисов, функционирующих в распределенной среде и позволяющих находить пространственные данные по различным критериям и анализировать их с помощью инструментов web-браузера для решения широкого круга научных и прикладных задач.

На данный момент в сервис Вега интегрированы инструменты обработки и анализа данных, построенные по технологии, разработанной ИКИ РАН. Технология обеспечивает возможность разработки специализированных блоков информационных систем для реализации следующих основных этапов для обработки спутниковых данных:

- выбор данных для обработки;
- управление и настройку операций обработки;
- выполнение операций обработки (подготовка исходных данных, обработка, сохранение и представление результата);
- проведение анализа результатов.

Архитектура рассматриваемого технологического решения представлена на рисунке 1.

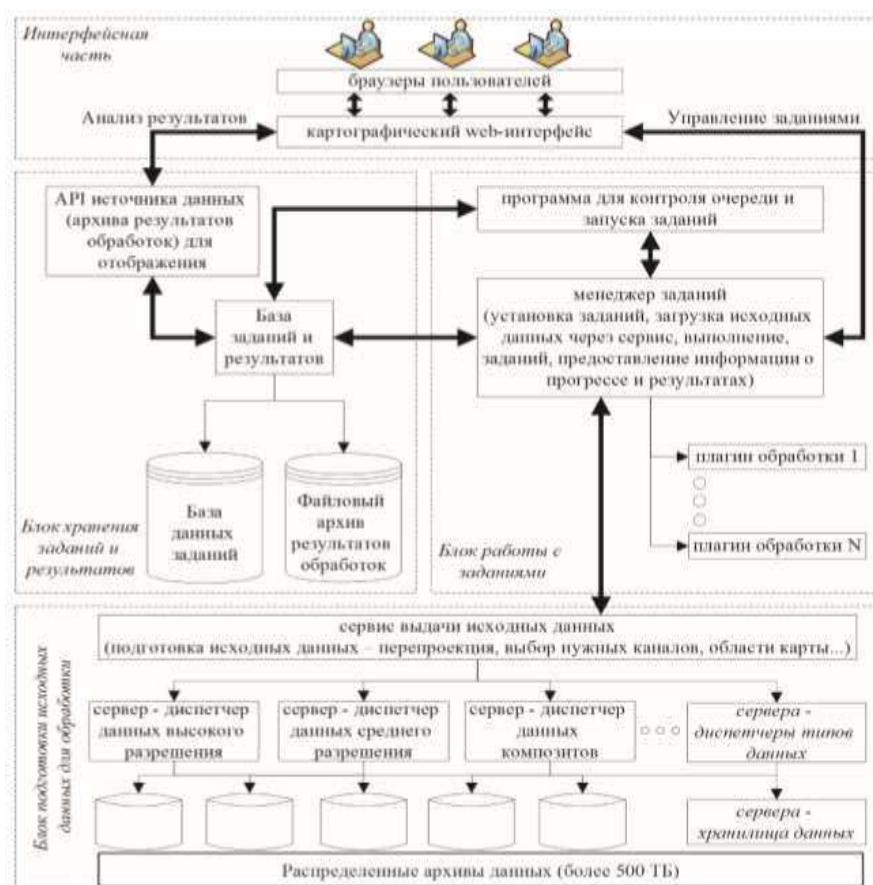


Рисунок 1 — Архитектура технологии

В контексте выпускной квалификационной работы основное внимание будет уделено блоку работы с заданиями.

1.2 ИСДМ Рослесхоз

Информационная система дистанционного мониторинга Федерального агентства лесного хозяйства (ИСДМ Рослесхоз) [6].

Данная система решает следующие основные задачи:

- сбор и хранение спутниковых данных;
- обработка данных;
- интеграция результатов обработки спутниковых данных с информацией, полученной из других источников;
- представление результатов обработки данных пользователю в виде, удобном для анализа и принятия решений.

Архитектура данной системы представлена на рисунке 2.

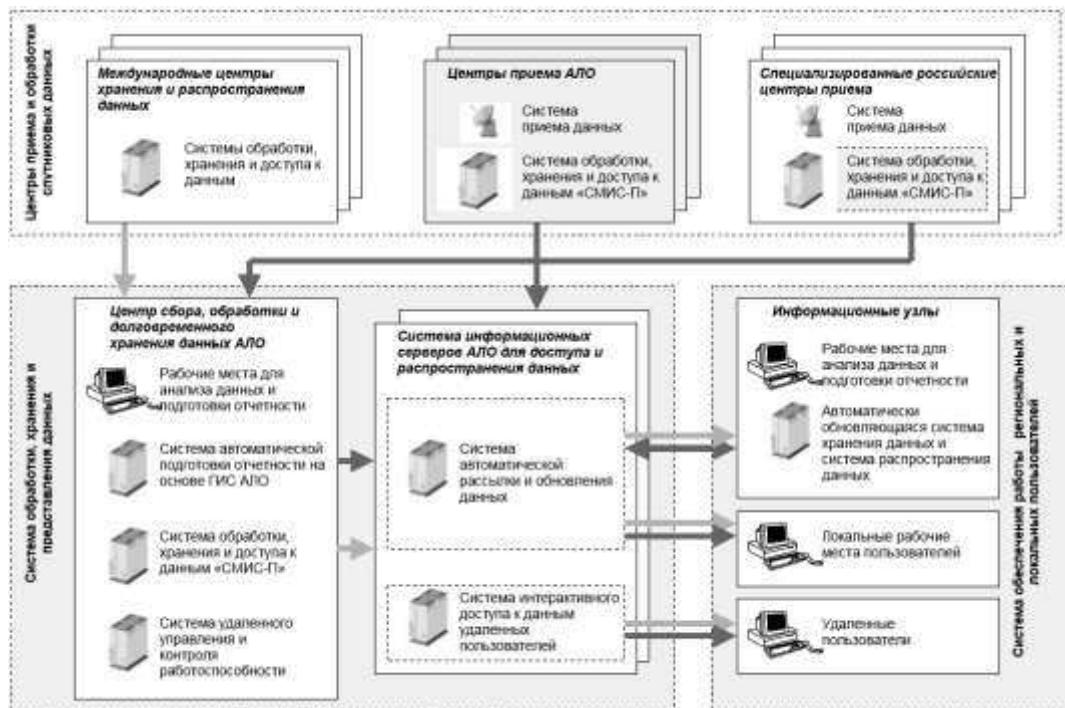


Рисунок 2 — Архитектура построения российской системы спутникового мониторинга лесных пожаров

В системе Рослесхоз все схемы обмена данными рассчитаны на использование публичных каналов российских Интернет-сетей. Естественно, это накладывает некоторые ограничения на объемы передаваемой информации, что учитывалось при планировании потоков данных в системе

и является минусом при том, что работа с данными дистанционных наблюдений рассчитана на работу со сверхбольшими объёмами данных.

1.3 See the Sea

Спутниковый сервис See the Sea (далее STS), ориентирован на решение задач, связанных с исследованием различных процессов на поверхности океана [7].

STS решает следующие задачи:

- поиск и выбор нужного для анализа набора данных;
- проведение анализа данных;
- выделение информации о наблюдаемом явлении на поверхности океана;
- создание описания явления, сохранение данного описания в базе данных;
- отображение и анализ информации о различных описанных явлениях.

В системе автоматически организовано поступление информации из различных российских и зарубежных центров сбора и обработки данных. Схема организации поступления данных в спутниковый сервис STS представлена на рисунке 3.

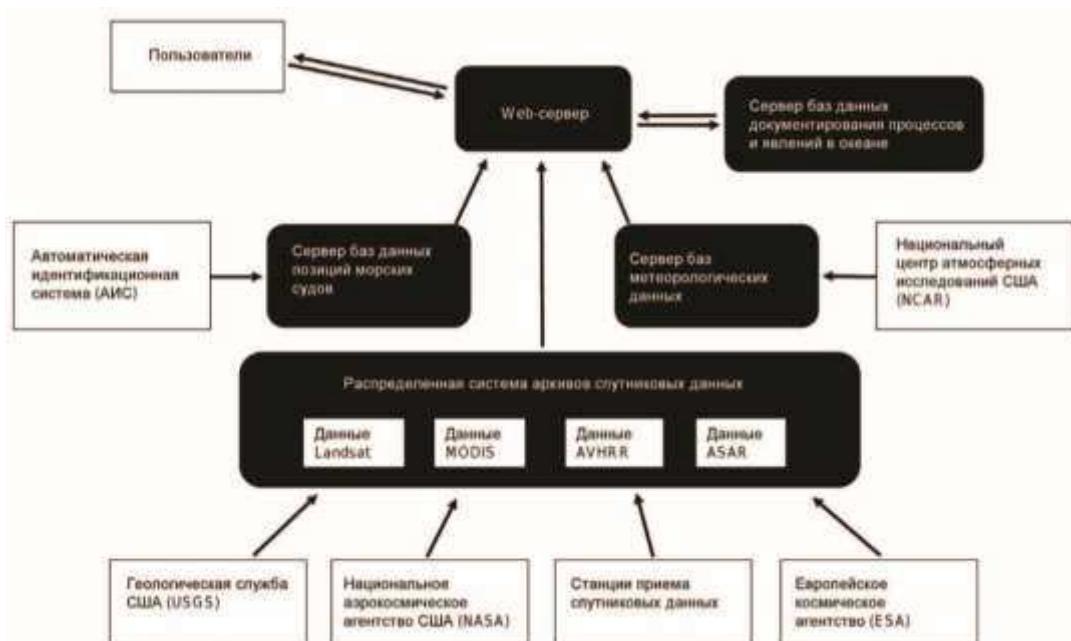


Рисунок 3 — Схема организации поступления данных в сервис STS

Все данные, поступающие в систему, проходят автоматическую обработку и помещаются в специальные архивы данных, из которых возможен их быстрый выбор для проведения онлайн-обработки и предоставления пользователям через интерфейсы STS. На этапе входной обработки данных строятся также различные информационные продукты, ориентированные на проведение анализы различных океанических процессов и явлений.

1.4 Google Earth Engine

Google Earth Engine — быстро развивающаяся облачная платформа для мониторинга состояния Земли.

Система направлена на максимальную интеграцию систем, обеспечивающих ведение сверхбольших распределенных архивов данных, и систем on-line анализа данных. Причем такие системы ориентированы на самый широкий круг специалистов, как использующих для работы с данными дистанционного зондирования различные специализированные программные комплексы и не имеющих глубоких навыков, и знаний программирования, так и имеющих таковые для создания собственных процедур обработки [8].

Доверенные пользователи этой системы имеют возможность запускать любые обработки над всем архивом данных, которые есть в системе, на кластере компании Google. В Google Earth Engine доступен архив спутниковых снимков за последние 40 лет, основу которого составляют данные спутников группировки Landsat и MODIS. Ее пользователи могут использовать как инструменты анализа данных (например, классификация), так и запускать в вычислительной сети свой программный код для обработки любых данных, имеющихся в архивах Google Earth Engine.

Earth Engine построен на основе набора технологий, которые доступны в среде центра обработки данных Google, включая систему управления

кластерами Borg [9]; распределенные базы данных Bigtable [10] и Spanner [11]; Колосс, преемник Файловой системы Google [12]. Earth Engine также взаимодействует с Google Fusion Tables [13], веб-базой данных, которая поддерживает таблицы геометрических данных (точки, линии и полигоны) с атрибутами. Упрощенная архитектура системы показана на рис. 4.

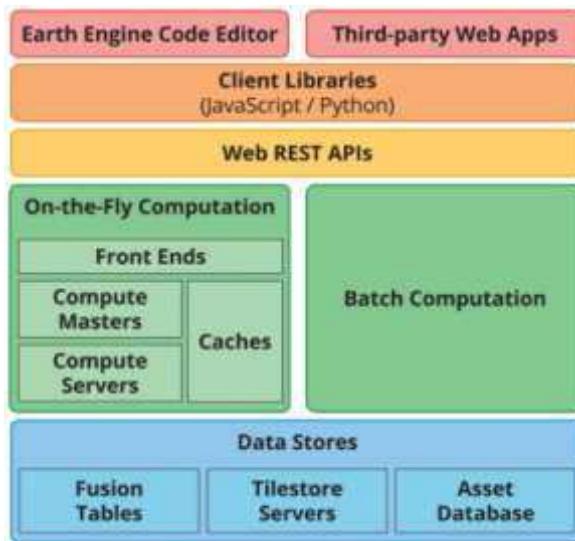


Рисунок 4 — Упрощенная архитектура системы Google Earth Engine

Редактор кода Earth Engine и сторонние приложения используют клиентские библиотеки для отправки интерактивных или пакетных запросов в систему через REST API. Запросы «на лету» обрабатываются серверами переднего плана, которые направляют сложные подзапросы в Compute Masters, которые управляют распределением вычислений среди пула вычислительных серверов. Пакетная система работает аналогичным образом, но использует FlumeJava для управления распространением. Поддержка обеих вычислительных систем представляет собой набор служб данных, включая базу данных активов, которая содержит метаданные для каждого изображения и обеспечивает эффективные возможности фильтрации. Программное обеспечение для управления кластером Borg управляет каждым компонентом системы, и каждый сервис сбалансирован по нагрузке для нескольких сотрудников.

1.5 NASA GIOVANNI

Другим примером системы, предоставляющей возможности доступа к архивам спутниковых данных и результатам их обработки одновременно со средствами, обеспечивающими проведение их обработки и анализа, может служить NASA GIOVANNI [14]. Кроме организации доступа к распределенным архивам данных, эта система предоставляет средства для дополнительной постобработки спутниковых данных и их анализа, при этом набор доступных преобразований сводится к заранее предопределенному набору алгоритмов выборки и агрегации данных.

Основной целью GIOVANNI является предоставление пользователям научного анализа и визуализации с вариантами уточнения. Архитектура GIOVANNI [15] состоит из: подробной схемы XML для описания параметров данных; атомарные (автономные) сервисы для обработки и рендеринга изображений; удобный интерфейс и приемлемая (от нескольких секунд до нескольких минут) производительность обработки; и механизм для определения и установки пользовательских настроек по умолчанию или по предпочтениям пользователя. Упрощенная архитектура системы представлена на рисунке 5.

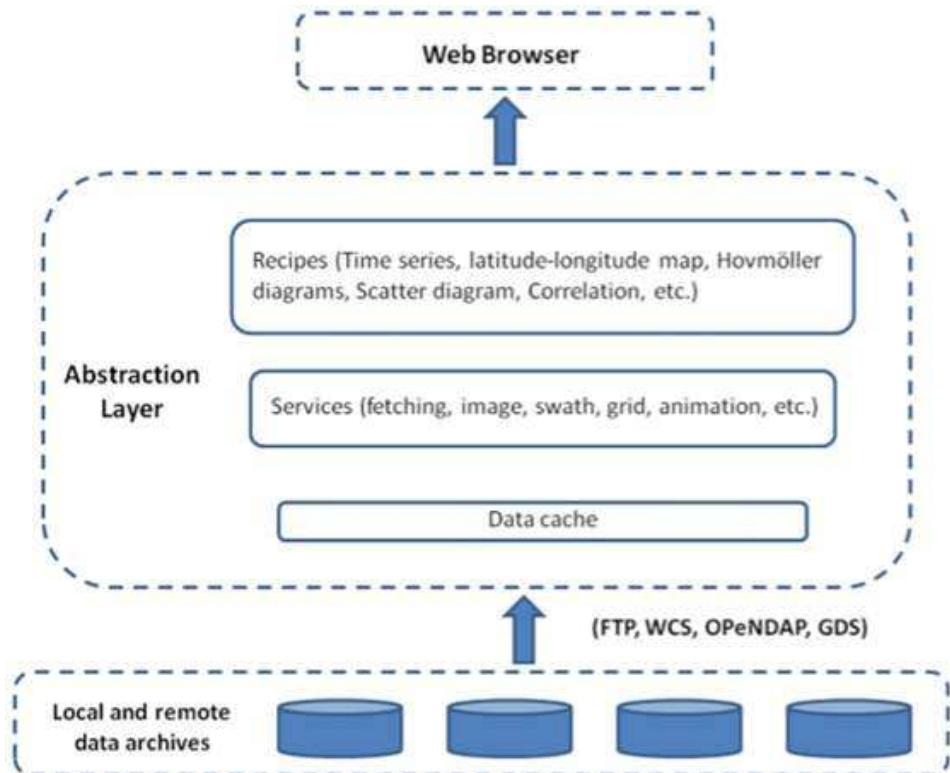


Рисунок 5 — Упрощенная архитектура системы NASA GIOVANNI

Когда пользователь отправляет запрос, GIOVANNI сначала получает данные из локальных или удаленных архивов данных через ftp, OPeNDAP, GDS (сервер данных GrADS), WCS и кэширует их для служб. Конечный результат доставляется в веб-браузер пользователя, где он может настроить выходные данные, загрузить данные для дальнейшего анализа или начать заново новый анализ.

1.2 Анализ архитектур рассмотренных систем

В ходе обзора были рассмотрены несколько существующих систем управления данными ДЗЗ и общая технология, применимая ко всем системам.

В результате выявлено, что у существующей технологии к управлению процессами обработки и анализа данных есть следующие особенности:

- архитектура системы содержит множество серверов и является дорогостоящей;
- все системы являются распределёнными;
- все системы являются программно-аппаратными комплексами, каждый из которых является системой для обработки данных;
- каждая из рассмотренных систем обладает возможностью формировать задания.

У сервиса Vega-PRO отсутствует интерфейс постановки задач, что является крайне неудобным в использовании.

У ИСДМ Рослесхоз есть ограничения на объемы передаваемой информации, что является минусом при том, что работа с данными дистанционных наблюдений рассчитана на работу со сверхбольшими объёмами данных.

Система See the Sea является узконаправленной, ориентирована на исследования различных процессов и явлений, происходящих на поверхности океана. Для расширения возможностей системы в неё следует

интегрировать дополнительные наборы данных. Также, STS доступна не всем исследователям и специалистам.

Системы Google Earth Engine и NASA GIOVANNI являются зарубежными, соответственно, человеку без знания английского языка будет труднее разобраться. Кроме того, возможность анализа данных в системе Google Earth Engine достаточно скромная.

Вывод по главе 1

Как показал анализ существующих зарубежных и российских систем, единого унифицированного подхода, реализующего удаленное управление процессами обработки данными ДЗЗ, состоящими из распределенных программных модулей не сформировано.

Тем ни менее на основе рассмотренных работ можно сформулировать некоторые общие принципы построения систем, требующих удаленного управления процессами обработки значительных архивов данных ДЗЗ, колоссальный, ежедневно увеличивающийся объем которых не позволяет создавать локальные архивы данных для локальной обработки, поскольку передача сотен гигабайт данных по Сети, а затем их обработка и анализ нецелесообразны с точки зрения временных затрат и принципов понятия «мониторинг», которое требует актуализации данных о наблюдаемых объектах с частотой максимально близкой к реальному времени. Таким образом, технология удаленного управления данными ДЗЗ должна базироваться на следующих принципах:

- Все процедуры обработки данных должны выполняться на удаленных серверах, т.е. необходимо исключить ситуации, при которых локальный пользователь (специалист в обработке данных) вынужден был копировать данные на ПК или локально выполнять часть операций обработки.

- Программный модуль для контроля заданий должен запускать на выполнение алгоритмические конструкции.

– Рассмотренные системы отличаются своей моделью представления задачий. В общем случае под заданием понимается запланированное во времени выполнение некоторого системного процесса, содержащего один программный алгоритм.

– Под алгоритмической конструкцией понимается совокупность логически взаимосвязанных программных алгоритмов, представленных отдельными модулями, последовательно выполняющихся в заданном порядке.

Кроме того, технология должна характеризоваться следующими основными элементами:

– Интерфейс пользователя. Технология должна предусматривать достаточный набор диалоговых средств для возможности постановки задачи обработки данных ДЗЗ пользователем. При этом пользователем может быть, как специалист по обработке данных, так и конечный пользователь при условии существования специальных адаптированных диалоговых элементов и схем.

– Источники данных ДЗЗ. Открытый доступ к программам со спутниками, основными спутниками являются Sentinel и Landsat.

– База программных операторов обработки и анализа данных. База должна содержать набор операторов для обработки и анализа данных. Количество серверов, выполняющих обработку подбирается в зависимости от запросов пользователя на выполнение заданий.

– База хранения и предоставления результатов обработки данных. Технология должна предусматривать хранилище, где будут находиться все задания на обработку с полной информацией о прогрессе и файловый архив с результатами проведенных обработок.

– Программный механизм управления процессами обработки данных. Такой механизм должен обладать функциями управления системными процессами (запуск, остановка, изменение приоритета), регламентирования очередей процессов, регламентирования системных ресурсов, выделяемых для процессов.

2 Разработка модуля управления и автоматического запуска процессов

2.1 Диаграмма вариантов использования

Унифицированный язык моделирования (Unified Modeling Language — UML) — стандартный инструмент разработки моделей программного обеспечения. Используется для визуализации, спецификации, конструирования и документирования артефактов программных систем [17].

Диаграмма вариантов использования является одной из 14 видов диаграмм в стандарт UML.

Цели, преследуемые при разработке диаграммы вариантов использования:

- установить для моделируемой предметной области общие границы и контекст на первых этапах проектирования системы;
- изложить единые требования к функциональному поведению проектируемой системы;
- создать первичную концептуальную модель системы для дальнейшей детализации в форме физических и логических моделей;
- подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Прецедент — это описание последовательности действий, включая их варианты, выполняемых системой, достижения определенного результата которых значимо для действующего лица. Применяется для выражения требуемого поведения системы, без описания реализации поведения, то есть определяют внешнее поведение, без объяснения, как его достичь. При этом описывают только существенные аспекты поведения [17].

Актор — сущность, которая взаимодействует с системой извне. При этом действующим лицом может выступать не только человек, но и системы, устройства, программы.

На рисунке 6 изображена модель прецедентов для акторов «Оператор» и «ПМЗП».

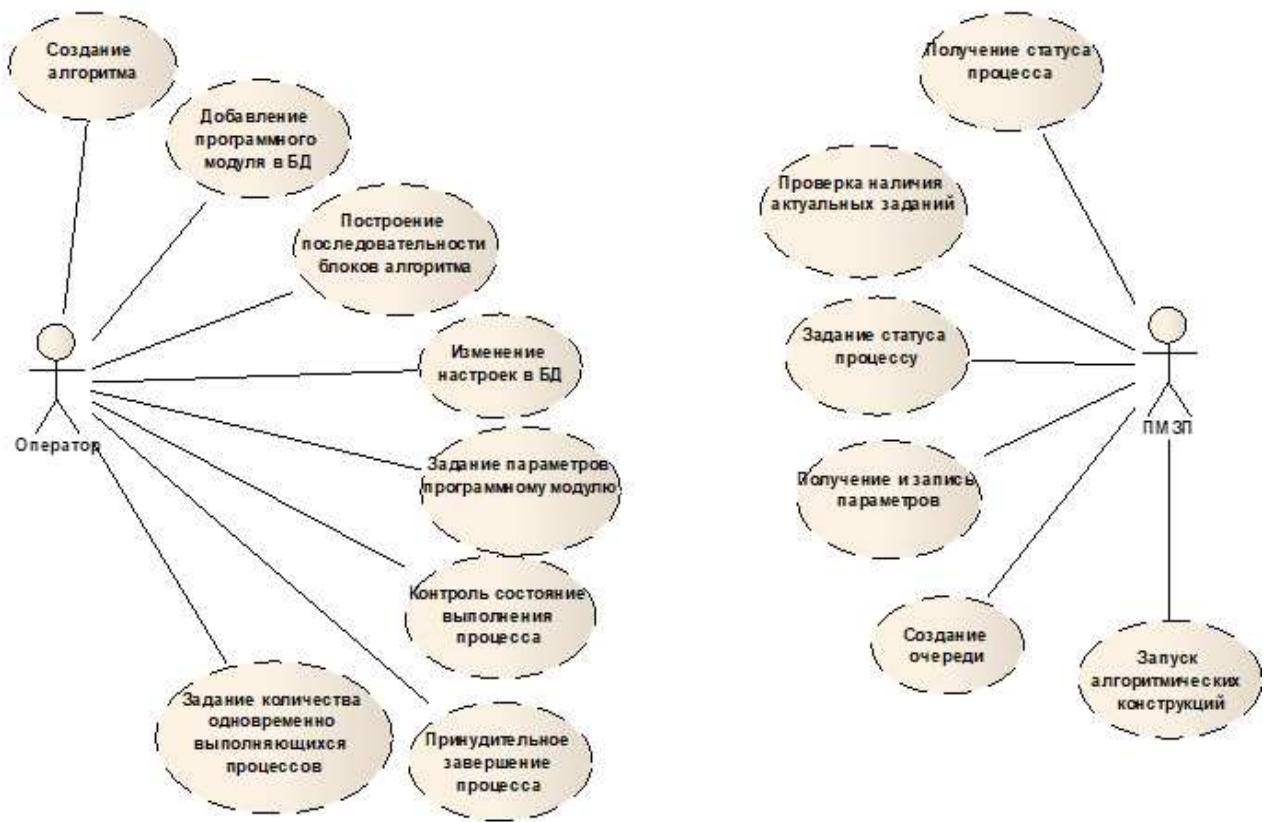


Рисунок 6 — Диаграмма вариантов использования для оператора и ПМЗП

В таблице 1 отражены акторы, прецеденты и их описания для модуля автоматического запуска процессов. Более подробно рассмотрено в Приложении В.

Таблица 1 — Прецеденты модуля запуска процессов

Код	Актор	Наименование	Формулировка
O1	Оператор	Создание алгоритма	Оператор добавляет в БД информацию об алгоритме: Название алгоритма, дата создания, описание назначения алгоритма
O2	Оператор	Добавление программного модуля в БД	Оператор вносит информацию о программных модулях, которые используются для построения цепочек обработки данных. Информация включает: название модуля, расположение модуля в файловой системе, параметры по умолчанию, IP-адрес сервера расположения.

Окончание таблицы 1

Код	Актор	Наименование	Формулировка
O3	Оператор	Построение последовательности блоков алгоритма	Оператор может построить программные модули в заданной последовательности
O4	Оператор	Изменение настроек в БД	Оператор может откорректировать алгоритм путём добавления/удаления модуля в базе данных.
O5	Оператор	Задание параметров программному модулю.	Оператор при построении последовательности блоков алгоритма задает входные параметры программным модулям.
O6	Оператор	Контроль состояния выполнения процесса	Оператор просматривает состояния процессов обработки данных.
МП1	ПМЗП	Получение статуса процесса	ПМЗП посылает информационные запросы со статусом процесса.
МП2	ПМЗП	Проверка наличия актуальных заданий	ПМЗП ежеминутно считывает статусы заданий. В случае обнаружения статуса «now» или «planed» определяет ID соответствующей конструкции. Определяет статус конструкции, если она не удалена, то считывает операторы, входящие в ее состав.
МП3	ПМЗП	Задание статуса процессу	ПМЗП задает статус процессу (inprogress-выполнение, now-готов к выполнению, ready-завершен).
МП4	ПМЗП	Получение и запись параметров	ПМЗП записывает в строку команды запуска параметры оператора и получает множество
МП4	ПМЗП	Получение и запись параметров	Заданных параметров оператора.
МП5	ПМЗП	Создание очереди	ПМЗП создает очередь и назначает каждому оператору из алгоритмической конструкции поток.
МП6	ПМЗП	Запуск алгоритмических конструкций	ПМЗП запускает программные модули обработки данных по расписанию, которое до этого сформировал оператор.
O7	Оператор	Принудительное завершение процесса	Оператор может принудительно завершить процесс до его окончания.
O8	Оператор	Задание количества одновременно выполняющихся процессов	Оператор может задавать количество параллельно выполняющихся процессов.

2.2 Диаграмма деятельности

Диаграмма деятельности — диаграмма для моделирования динамических аспектов систем. Эти динамические аспекты могут включать деятельность на любом уровне абстракции в любом представлении системной архитектуры, включая классы (в том числе активные), интерфейсы, компоненты и узлы. Представляет собой блок-схему, которая показывает, как поток управления переходит от одной деятельности к другой, однако в отличие от традиционной блок-схемы позволяет отображать параллелизм [16].

Рассмотрим более детально прецедент «Запуск алгоритмических конструкций».

Прецедент МП6: Запуск алгоритма

Краткое описание. Модуль управления процессами запускает на выполнение задания, которым соответствуют алгоритмические конструкции, представленные в БД и имеющие статусы 0 — удален или 1 — не удален. Задания имеют иные статусы: plan — означает, что процесс запланирован на определенную дату и время; now — означает, что процесс должен быть выполнен немедленно; canceled — процесс отменен пользователем; ready — процесс завершен; inprogress — процесс выполняется.

Основное действующее лицо — ПМЗП.

Предусловия

1. Частота запуска алгоритмов модулем управления процессами определена и задана в планировщике CRON.
2. Последовательность модулей и их входные и выходные параметры определены оператором.
3. В БД Системы сформирован соответствующий заданной последовательности алгоритм.

Поток событий. Прецедент инициируется планировщиком по заданному расписанию (1 раз в минуту).

Базовый поток — запуск модуля

1. CRON осуществляет запуск модуля управления процессами.
2. Модуль выполняет запрос к БД на выборку подготовленных алгоритмов.
3. БД возвращает результат запроса.
4. Модуль получает последовательность модулей алгоритма.
5. Модуль формирует команду для добавления алгоритма в очередь.
6. Модуль создаёт системную очередь.
7. Модуль создаёт системный процесс.
8. Модуль ожидает завершения работы процесса.
9. Модуль задаёт процессу статус 3 — завершён.
10. БД обновляет статус процесса.

Альтернативные потоки

При выполнении пункта 1 базового потока возникает точка принятия решения в случае, когда подготовленных алгоритмов для выполнения не найдено. В этом случае модуль переходит в режим ожидания следующего запуска по расписанию.

Оператор может запускать модули вручную по расписанию, используя интерфейс командной строки и контролируя их выполнение.

Постусловия

При успешном окончании прецедента, будут последовательно выполнены все процессы, занесенные в базу данных.

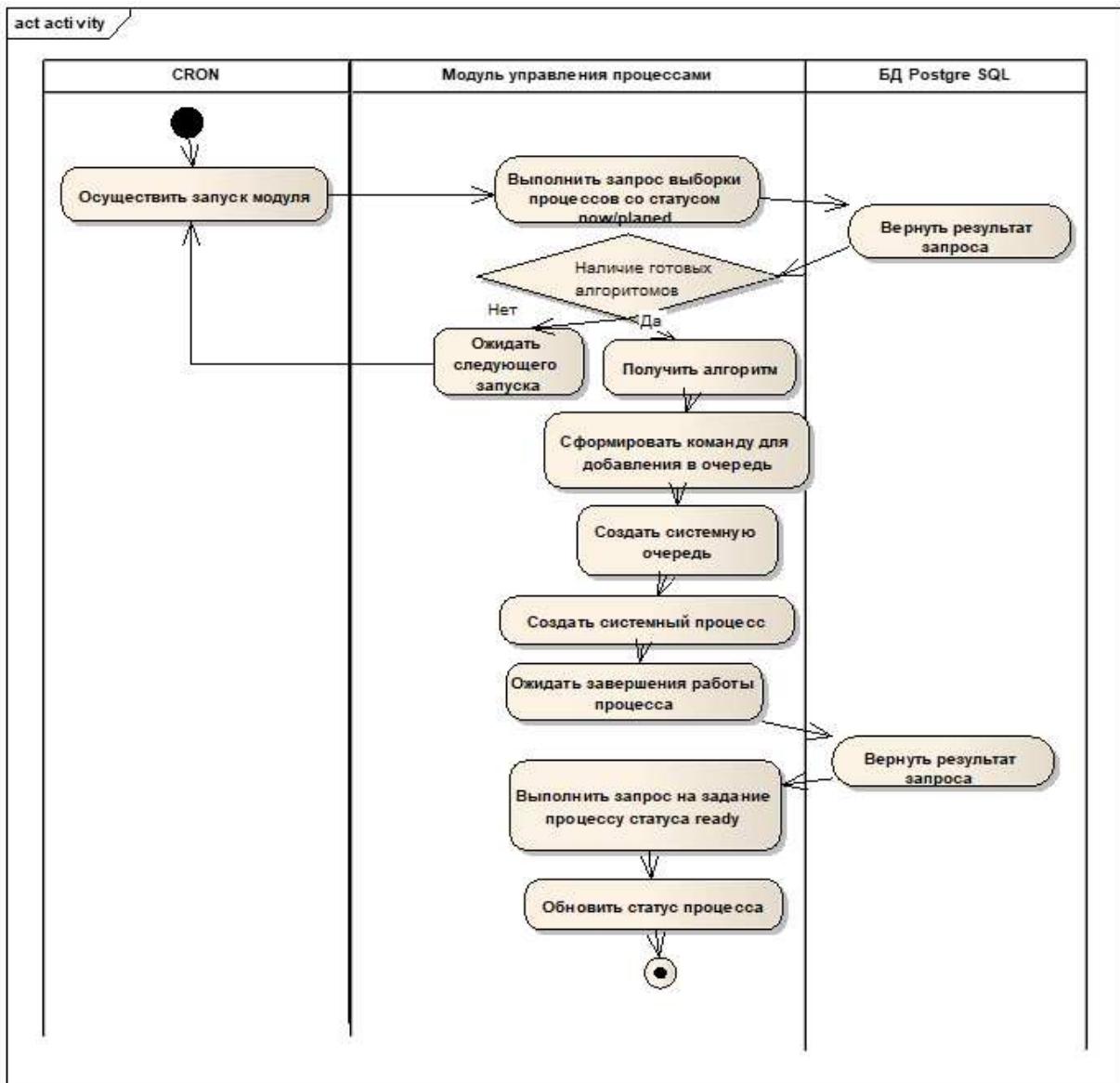


Рисунок 7 — Диаграмма деятельности precedента «Запуск модуля»

2.3 Схема взаимодействия

Компонент — замещаемая часть системы, которая соответствует набору интерфейсов и обеспечивает его реализацию.

Диаграмма компонентов описывает особенности физического представления системы. Такая диаграмма позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами [17].

На диаграмме развертывания демонстрируется взаимодействие компонентов с узлами в физической системе, а также соединение узлов

собой. На рисунке 8 изображена схема взаимодействия компонентов с узлами в физической системе для функционирования модуля визуализации.

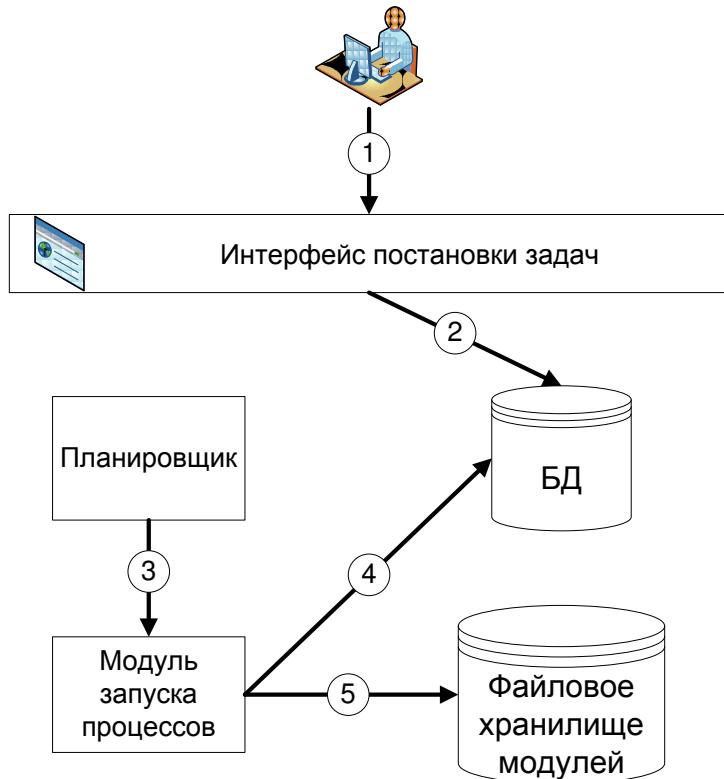


Рисунок 8 — Схема запуска алгоритмических конструкций

В схеме изображены компоненты, необходимые для функционирования модуля. Оператор добавляет задачу в БД, Модуль запуска процессов с помощью планировщика в заданное время обращается к БД и запускает задачи на выполнение. Файловое хранилище модулей содержит программные модули и хранит на отдельном сервере.

2.4 Разработка модуля

Для функционирования модуля потребовалось создать дополнительную структуру данных.

2.4.1 Структура данных для работы модуля

Данные, с которыми работает модуль запуска процессов собраны в таблице «Представление алгоритмических конструкций», состав которой отражен на рисунке 9.

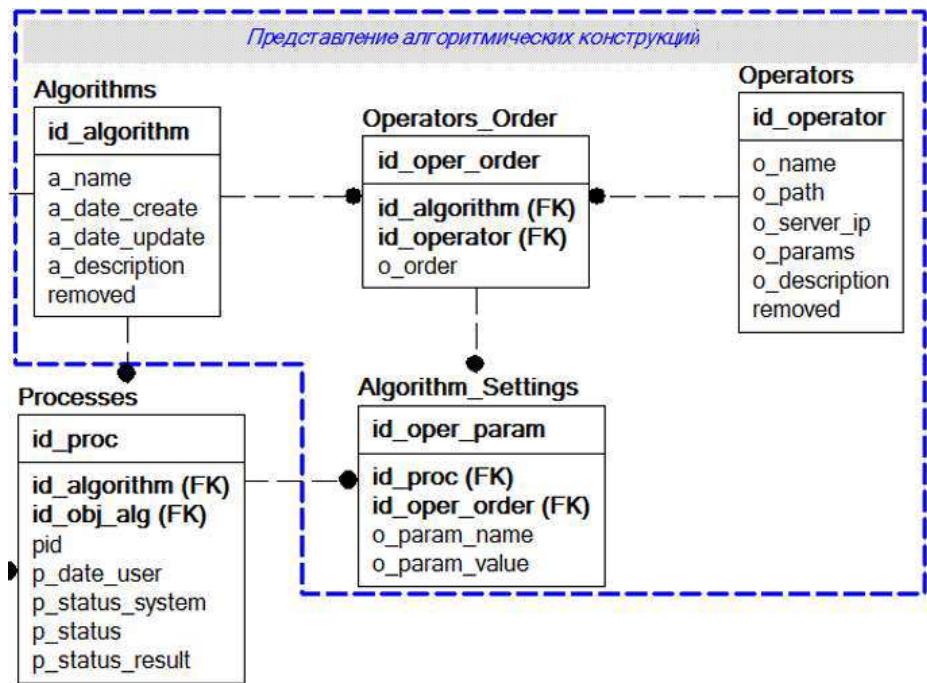


Рисунок 9 — Структура и состав таблицы «Представление алгоритмических конструкций»

Таблица «Processes» является системной таблицей, позволяющей подсистеме управления отслеживать статусы системных процессов. Также, в ней представлен план мониторинга. Для операторов конструкции предусмотрена технология очередей (Queue) потоков (Thread), поскольку требование многозадачности подразумевает, что операторы разных алгоритмических конструкций могут выполняться в параллельном режиме и каждый оператор выполнялся строго в соответствии с порядком, заданным специалистом. С другой стороны, подсистема управления контролирует системную память и не допускает одновременного выполнения более 5 конструкций.

Рассмотрим пример, который иллюстрирует структуру плана мониторинга. Пусть задан некоторый пространственный объект, для которого оператором определено множество временных точек: 01.05.2009, 03.05.2009, 05.05.2009. Для этого объекта задан алгоритм мониторинга состояния растительности, $id_algorithm = 4$. В результате постановки задачи формируется план мониторинга, который учитывает интервал дат между начальной и конечной датой жизненного цикла объекта и временной сдвиг,

который связан с задержкой получения данных (для Landsat-8 — 8 дней, для Sentinel-2 — 1 день). План можно представить в виде фрагмента таблицы Processes.

Таблица 2 — фрагмент таблицы Processes

pid	id_obj_alg	p_status	p_dateuser	p_datesystem	p_status_result
	2	plan	01.05.2009	02.05.2009	
	2	plan	01.05.2009	09.05.2009	
	2	plan	02.05.2009	03.05.2009	
	2	plan	02.05.2009	10.05.2009	
	2	plan	03.05.2009	04.05.2009	
	2	plan	03.05.2009	11.05.2009	
	2	plan	04.05.2009	05.05.2009	
	2	plan	04.05.2009	12.05.2009	
	2	plan	05.05.2009	06.05.2009	
	2	plan	05.05.2009	13.05.2009	

2.5 Алгоритм модуля запуска процессов

Модуль запуска процессов включает два основных этапа работы (рис. 11):

1. Подготовка последовательностей команд, соответствующих алгоритмическим конструкциям. Модуль обращается к БД с запросом на наличие новых заданий, имеющих статус «немедленный запуск» (now) или «запланированный запуск» (plan) — шаг 1, и, в случае обнаружения, создает ассоциативный массив (шаг 2) для хранения всех алгоритмических конструкций и определяет непосредственно сами алгоритмические конструкции, соответствующие полученным заданиям (цикл «i»). Затем определяет все операторы конструкции в соответствии с заданным порядком (order). Затем модуль инициализирует массив (шаг 7), в который по порядку добавляет строки команд запуска операторов (цикл «j») и их параметров с заданными значениями (цикл «k»).

2. Запуск команд на исполнение в многопоточном режиме. Следующим этапом алгоритм модуля инициализирует очередь потоков (цикл «l»), количество которых равно числу операторов конструкции. Каждый оператор будет назначен в отдельный поток (цикл «m») и будет выполняться

последовательно в соответствии с порядком в очереди до полного завершения (цикл while).

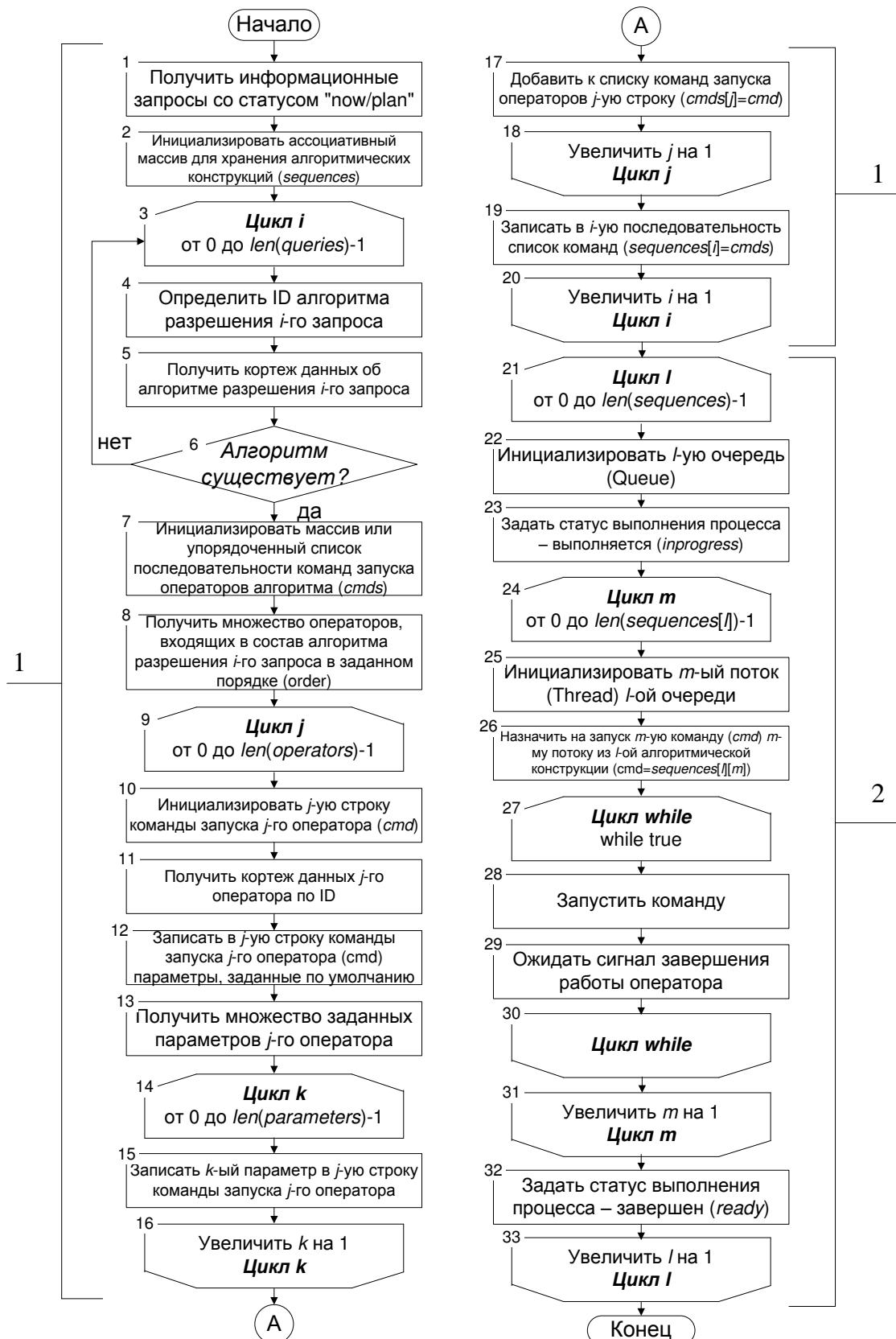


Рисунок 11 — Алгоритм модуля управления и запуска процессов

2.6 Реализация модуля запуска процессов

Типовым примером процесса является создание RGB изображения и его публикация на Геопортале системы мониторинга, представленный на рисунке 12.

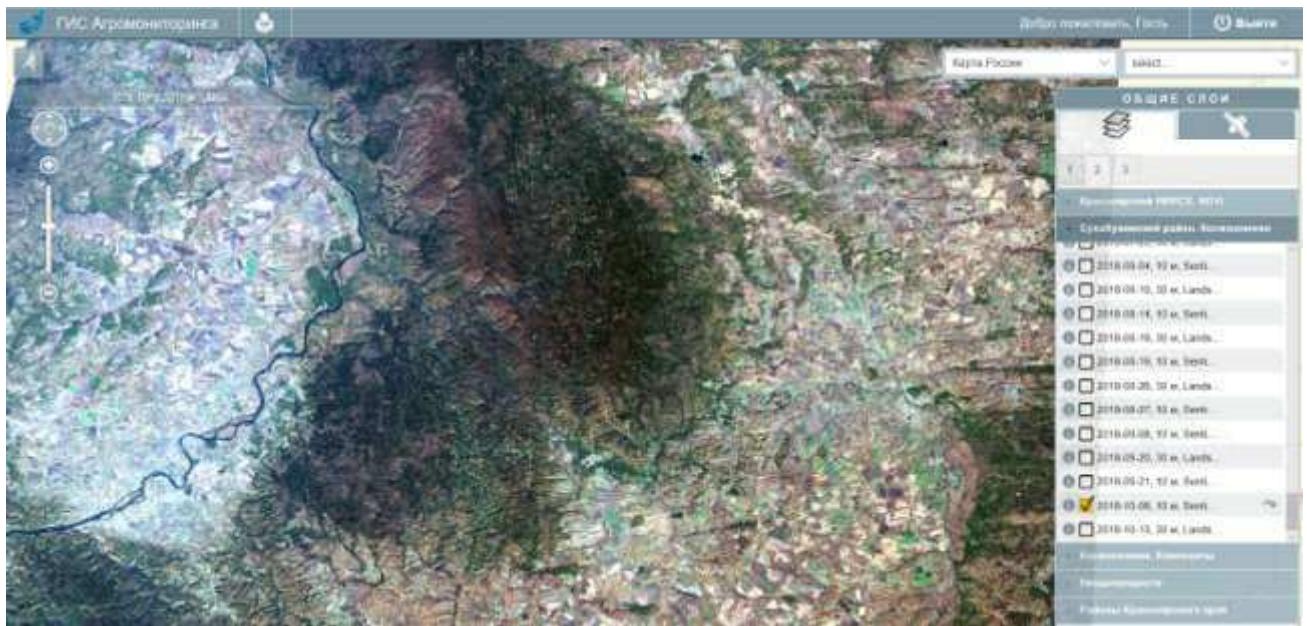


Рисунок 12 — RGB изображение на Геопортале

За каждый функцию процесса отвечает отдельный модуль, эти модули выполняются последовательно в заданном порядке: модуль «Извлечение архива», модуль «Сбор мозаики» (создается мозаика из каналов Red, Green, Blue), модуль «Объединение каналов» (каналы объединяются в единый RGB-файл), модуль «Хранилище Геосерверов», модуль «Web-публикация» (RGB-снимок публикуется на Геопортале).

Вывод по главе 2

В ходе второй главы спроектирован модуль управления и автоматического запуска процессов. Проект включает в себя диаграмму вариантов использования, отражающую возможность взаимодействия оператора и ПМЗП с системой. С помощью диаграммы компонентов системы показано взаимодействие компонентов внутри системы. Диаграмма

деятельности отражает как модуль выполняет запуск на выполнение заданий.

На основе разработанного проекта реализован модуль управления и автоматического запуска процессов.

ЗАКЛЮЧЕНИЕ

В результате выпускной квалификационной работы были выполнены поставленные задачи:

- Выполнен и проведен анализ существующих решений управления распределенными программными средами. Были выведены общие принципы построения систем и их основные элементы. После обзоров других систем были сформулированы требования к разрабатываемому модулю и составлено техническое задание.
- При проектировании Модуля создана диаграмма вариантов использования, описывающие модель взаимодействия пользователя с модулем запуска программных алгоритмов, диаграмма компонентов системы отражает взаимодействие компонентов внутри системы;
- Разработан модуль управления и автоматического запуска процессов. Модуль способен запускать алгоритмические конструкции, задавать статус процессу, проверять наличие актуальных заданий.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Балашов И.В., Бурцев М.А., Галеев А.А., Ефремов В.Ю., Крашенинникова Ю.С., Лупян Е.А., Мазуров А.А., Московский А.А., Прошин А.А., Сергеева К.О., Флитман Е.В. Разработка эффективных высокопроизводительных решений для создания систем динамической обработки спутниковых данных и результатов их анализа // Третья международная научная конференция «Суперкомпьютерные системы и их применения» (SSA' 2010). 25-27 мая 2010. Минск. Сборник докладов, ОИПИ НАН Белоруси, 2010. Т. 1. С. 30-34.
2. Шокин Ю.И., Антонов В.Н., Добрецов Н.Н., Кихтенко В.А., Лагутин А.А., Смирнов В.В., Чубаров Д.Л., Чубаров Л.Б. Распределенная система приема и обработки спутниковых данных Сибири и Дальнего Востока. Текущее состояние и перспективы развития // Современные проблемы дистанционного зондирования Земли из космоса, 2012. Т. 9. №. 5. С. 45-54.
3. Бабяк П. В., Недолужко И. В., Фомин Е. В. Подход к предоставлению услуг по обработке спутниковых данных в Центре коллективного пользования регионального спутникового мониторинга окружающей среды ДВО РАН // Материалы конф. «Интернет и современное общество», 2014. С. 27-32.
4. Лупян Е.А., Саворский В.П., Шокин Ю.И., Алексанин А.И., Назиров Р.Р., Недолужко И.В., Панова О.Ю. Современные подходы и технологии организации работы с данными дистанционного зондирования Земли для решения научных задач // Современные проблемы дистанционного зондирования Земли из космоса, 2012. Т. 9. № 5. С. 21-44.
5. Лупян Е.А., Савин И.Ю., Барталев С.А., Толpin В.А., Балашов И.В., Плотников Д.Е. Спутниковый сервис мониторинга состояния растительности («ВЕГА») // Современные проблемы дистанционного зондирования Земли из космоса, 2011. Т. 8. № 1. С. 190-198.

6. Ефремов В.Ю., Балашов И.В., Котельников Р.В., Лупян Е.А., Мазуров А.А., Прошин А.А., Толпин В.А., Уваров И.А., Флитман Е.В. Объединенный картографический интерфейс для работы с данными ИСДМ Рослесхоз // Современные проблемы дистанционного зондирования Земли из космоса, 2011. Т. 8. № 3. С. 129-139.
7. Лупян Е.А., Матвеев А.А., Уваров И.А., Бочарова Т.Ю., Лаврова О.Ю., Митягина М.И. Спутниковый сервис See the Sea – инструмент для изучения различных явлений на поверхности океана // Современные проблемы дистанционного зондирования Земли из космоса, 2012. Т. 9. № 2. С. 251-262.
8. Moore R. T., Hansen M. C. Google Earth Engine: a new cloud-computing platform for global-scale Earth observation data and analysis, AGU Fall Meeting, Abstracts, 2011, Vol.1, pp. 2.
9. Коровин Г.Н., Андреев Н.А. Авиационная охрана лесов. М.: Агропромиздат. 1988, 220 с.
10. F. Chang, J. Dean, S. Ghemawat, W.C. Hsieh, D.A. Wallach, M. Burrows, T. Chandra, *et al.* Bigtable: a distributed storage system for structured data. ACM Trans. Comput. Syst., 26 (2) (2008), p. 4
11. J.C. Corbett, J. Dean, M. Epstein, A. Fikes, C. Frost, J.J. Furman, S. Ghemawat, *et al.* Spanner: Google's globally distributed database. ACM Trans. Comput. Syst., 31 (3) (2013), p. 8
12. A. Verma, L. Pedrosa, M. Korupolu, D. Oppenheimer, E. Tune, J. Wilkes Large-scale cluster management at Google with Borg. Proc. EuroSys 10, 18. ACM (2015)
13. H. Gonzalez, A.Y. Halevy, C.S. Jensen, A. Langen, J. Madhavan, R. Shapley, *et al.* Google fusion tables: web-centered data management and collaboration, ACM SIGMOD (2010), pp. 1061-1066
14. Acker J. G., Leptoukh G. Online analysis enhances use of NASA earth science data, Eos, Transactions American Geophysical Union, 2007, Vol. 88, No. 2, pp. 14-17.

15. Berrick, S.W., Leptoukh, G., Farley, J.D., Rui, H., 2009. Giovanni: a web service workflow-based data visualization and analysis system. *IEEE Trans. Geosci. Remote Sensing* 47 (1), 106–113.
16. Буч Г., Рамбо, Д., Якобсон, И. Язык UML. Руководство пользователя. / Г. Буч, Д. Рамбо, И. Якобсон: [пер. с англ]. –Москва: ДМК Пресс, 2006. –496 с.
17. Леонков, А. В. Самоучитель UML / А. В. Леонков. –Санкт-Петербург: БХВ-Петербург, 2007. –376 с.

ПРИЛОЖЕНИЕ А

Видение

1 Введение

1.1 Цель

Цель создания этого документа состоит в том, чтобы собрать, проанализировать и определить высокоуровневые потребности и возможности модуля управления и автоматического запуска процессов. Документ акцентирует внимание на возможностях, необходимых совладельцам и целевым пользователям, и на том, почему эти потребности существуют. Подробности того, как модуль запуска выполняет эти потребности, будут детализированы в прецедентах и дополнительных спецификациях.

1.2 Контекст

Настоящий документ разрабатывается в рамках программно-аппаратного комплекса геоинформационной системы агромониторинга Сибирского Федерального Университета (ГИС СФУ).

1.3 Определения, акронимы и сокращения

Основные определения приведены в Приложении Б.

1.4 Краткое содержание

Документ описывает высокоуровневые требования к модулю управления и автоматического запуска процессов. Указаны основные деловые преимущества рассматриваемого в Видении решения, сформулированы ключевые проблемы и способы их решения, приведены характеристики пользователей системы, возможности системы, ограничения, показатели качества и другие требования к продукту.

2 Позиционирование

2.1 Деловые преимущества

В настоящее время система построена по модульному принципу. Модульность реализовывает принцип повторного использования системных компонент при решении различных задач в том числе в параллельном режиме. На физическом уровне такой подход позволяет представить заданную операцию по обработке или анализу данных в виде отдельного модуля, а набор операций в виде совокупности логически взаимосвязанных программных модулей, выполняющих в заданной последовательности, с заданным набором входных параметров, в заданное время. Поскольку каких-либо средств, автоматизирующих процесс многократного запуска программных компонент, не существует, то эта ситуация приводит к необходимости постоянного участия и контроля со стороны оператора, что в условиях пространственной распределенности и многозадачности является неприемлемым решением. В сравнении с имеющимся, новое решение позволит оптимизировать деятельность оператора и обеспечить рациональное распределение ресурсов между параллельными процессами путём создания модуля автоматизированного запуска процессов.

Также, система гетерогенная, а это значит, что она включает различные распределенные в пространстве серверы обработки. Системные компоненты, следовательно, тоже могут быть распределены по различным серверам. При таких условиях построение цепочек обработки данных в операторском режиме невозможно. Новое решение обеспечит работу оператора в интерактивном графическом режиме.

2.2 Определение проблемы

Проблема	Отсутствие средств, автоматизирующих процесс многократного запуска программных компонент.
затрагивает	Оператора.
Ее следствием является	Необходимость постоянного участия и контроля со стороны оператора.
Успешное решение	Позволит оптимизировать деятельность оператора и обеспечить рациональное распределение ресурсов между параллельными процессами путём создания модуля автоматизированного запуска

	процессов.
--	------------

2.3 Определение позиции изделия

Для	Многоцелевой региональной системы ДЗЗ ИКИТ СФУ
которой	Требуется автоматизировать процесс запуска программных модулей в заданной последовательности, с заданным набором входных параметров, в заданное время.
(Название продукта)	Модуль запуска процессов
который	Основан на языке Python и СУБД PostgreSQL
В отличие от	Существующего решения на основе операторского режима работы
наш продукт	Обеспечивает автоматизированный последовательный запуск процессов с настройками, заданными в базе данных.

3 Описания пользователей

3.1 Сведения о пользователях

У системы существуют один основной пользователь: оператор. Оператор — занимается построением цепочек модулей обработки данных, запускает модули по расписанию, следит за их выполнением.

В настоящее время в программно-аппаратном комплексе работает один оператор. Система будет работать на платформе Linux. Операционная система: Linux.

3.2 Профили пользователей

Типичный представитель	Оператор
------------------------	----------

Описание	Пользователь программы, наделенный правами на чтение информации, построение последовательностей алгоритмов, изменение настроек.
Тип	Пользователь
Ответственност ти	1. Строит модели процесса, его инициализации и запуска. 2. Сохраняет настройки в базу данных. 3. Осуществляет контроль выполнения процессов.
Критерий успеха	Возможность автоматизированного последовательного запуска процессов.

3.3 Ключевые потребности пользователей

Оператор сталкивается с такой трудностью, как затрата большого количества времени на составление очереди запуска модулей по расписанию. Предприятие нуждается в программе, которая бы автоматизировала запуск программных модулей для обработки данных.

4 Возможности продукта

4.1 Автоматизированный процесс многократного запуска компонент

Возможность автоматизированного запуска набора операций в виде совокупности логически взаимосвязанных программных модулей, выполняющихся в заданной последовательности, с заданным набором входных параметров, в заданное время.

4.2 Интеграция информации

Возможность интегрирования на основе единой БД информации обо всех модулях гетерогенной системы.

4.3 Оперативная корректировка алгоритмов

Возможность изменить последовательность алгоритма путём добавления/удаления модуля в базе данных.

4.4 Оптимизация ресурсов

Возможность рационального распределения ресурсов между параллельными процессами путём создания модуля автоматического запуска процессов.

4.5 Отсутствие постоянного участия и контроля оператора

Возможность многократного запуска программных компонент без постоянного участия и контроля со стороны оператора в условиях пространственной распределённости и многозадачности.

5 Ограничения

В ядре системы должна быть представлена промышленная СУБД реляционного доступа.

6 Другие требования к изделию

6.1 Системные требования

Минимальные системные требования:

- Оперативная память 16 ГБ, 10 ГБ свободного дискового пространства
- Виртуальная машина VMWare ESXi50 на аппаратной базе IBM Blade Server E HS21XM с двумя виртуальными процессорами Intel Xeon E5420 по 2.5GHz
- Операционная система Linux.

ПРИЛОЖЕНИЕ Б

Глоссарий

1 Введение

1.1 Цель

Глоссарий содержит описания терминов, используемых при проектировании модуля управления и автоматического запуска процессов. Определяются основные понятия, непосредственно связанные с дистанционным зондированием Земли и управлением данными.

1.2 Контекст

Глоссарий создан в рамках многоцелевой региональной системы дистанционного зондирования Земли ИКИТ

1.3 Ссылки

Сопутствующая информация представлена в Приложении А, Приложении В.

2 Определения

2.1 Понятия, используемые при описании исходной информации

2.1.1 ДЗЗ

ДЗЗ (дистанционное зондирование Земли) — наблюдение поверхности Земли космическими средствами, оснащенными съемочной аппаратурой.

2.1.2 Модуль

Модуль — программа, предназначенная для хранения, трансляции, объединения с другими модулями и загрузки в оперативную память.

Работа является реализацией некоторого типа работ и относится к конкретному заказу.

2.1.3 Оператор

Оператор — человек, который осуществляет управление удаленными процессами, также создает и добавляет новые модули.

2.1.4 Алгоритмическая конструкция

Алгоритмическая конструкция — совокупность логически взаимосвязанных программных алгоритмов, представленных отдельными модулями, последовательно выполняющихся в заданном порядке.

2.1.5 Задание

Задание — запланированное во времени выполнение некоторого системного процесса, содержащего один программный модуль.

2.1.6 Статус задания

Статус задания. Модуль управления процессами запускает на выполнение задания, которым соответствуют алгоритмические конструкции, представленные в БД и имеющие статусы 0 – удален или 1 – не удален. Задания имеют иные статусы: plan – означает, что процесс запланирован на определенную дату и время; now – означает, что процесс должен быть выполнен немедленно; canceled – процесс отменен пользователем; ready – процесс завершен; inprogress – процесс выполняется.

2.1.7 ПМЗП

ПМЗП (программный модуль запуска процессов) — механизм, обладающий функциями управления системными процессами, (запуск, остановка, изменение приоритета), регламентирования очередей процессов, регламентирования системных ресурсов, выделяемых для процессов.

2.1.8 План запуска процессов

План запуска процессов — график запуска процессов во времени, составляемый оператором.

2.1.9 Очередь процессов

Очередь процессов — метод планирования процессов, ориентированный на эффективную загрузку ресурсов.

ПРИЛОЖЕНИЕ В

Спецификация требований к АИС

1 Введение

1.1 Цель

Цель этого документа — в том, чтобы сформулировать требования к разрабатываемому модулю управления и автоматического запуска процессов. Данные требования описаны в форме прецедентов, кратких описаний функциональных требований и описаний нефункциональных требований.

1.2 Определения, акронимы и сокращения

Основные определения приведены в Приложении Б.

1.3 Ссылки

Сопутствующая информация представлена в следующих документах:

- видении (Приложение А) и
- глоссарии (Приложение Б).

Обзор системы

1.4 Обзор прецедентов

Краткое описание акторов представлено в таблице 1.

Таблица 1 — Акторы системы

<i>Актор</i>	<i>Краткое описание</i>
Оператор	Занимается построением цепочек модулей обработки данных, сохраняет настройки в базу данных, следит за их выполнением.

Список вариантов использования показан в таблице 2.

Таблица 2 — Реестр вариантов использования

<i>Код</i>	<i>Основной актор</i>	<i>Наименование</i>	<i>Формулировка</i>
O1	Оператор	Создание алгоритма	Оператор добавляет в БД информацию об алгоритме: Название алгоритма, дата создания, описание назначения алгоритма
O2	Оператор	Добавление программного модуля в БД	Оператор вносит информацию о программных модулях, которые используются для построения цепочек обработки данных. Информация включает: название модуля, расположение модуля в файловой системе, параметры, IP-адрес сервера расположения.
O3	Оператор	Построение последовательности блоков алгоритма	Оператор может построить программные модули для обработки данных в заданной последовательности.
O4	Оператор	Изменение настроек в базе данных	Оператор может откорректировать алгоритм путём добавления/удаления модуля в базе данных.
O5	Оператор	Задание параметров программному модулю.	Оператор при построении последовательности блоков алгоритма задает входные параметры программным модулям.
O6	Оператор	Контроль состояния выполнения процесса	Оператор просматривает состояния процессов обработки данных.
МП1	ПМЗП	Получение статуса процесса	ПМЗП посылает информационные запросы со статусом процесса.

Таблица 2 — Реестр вариантов использования

МП2	ПМЗП	Проверка наличия актуальных заданий	ПМЗП ежеминутно считывает статусы заданий. В случае обнаружения статуса «now» или «planed» определяет ID соответствующей заданию алгоритмической конструкции. Определяет статус конструкции, если она не удалена, то считывает операторы, входящие в ее состав.
МП3	ПМЗП	Задание статуса процессу	ПМЗП задает статус процессу (inprogress — выполнение, now — готов к выполнению, ready — завершен).
МП4	ПМЗП	Получение и запись параметров	ПМЗП записывает в строку команды запуска параметры оператора и получает множество заданных параметров оператора.
МП5	ПМЗП	Создание очереди	ПМЗП создает очередь и назначает каждому оператору из алгоритмической конструкции поток.
МП6	ПМЗП	Запуск алгоритмических конструкций	ПМЗП запускает программные модули обработки данных по расписанию, которое до этого сформировал оператор.
O7	Оператор	Принудительное завершение процесса	Оператор может принудительно завершить процесс до его окончания.
O8	Оператор	Задание количества одновременно выполняющихся процессов	Оператор может задавать количество параллельно выполняющихся процессов.

1.5 Предположения и зависимости

Программно-аппаратная инфраструктура многоцелевой системы ДЗЗ ИКИТ предполагает изменение конфигурации серверов, следствием чего может быть увеличение серверных единиц и размещение на них отдельных программных компонентов обработки данных ДЗЗ. Таким образом, модуль запуска процессов предполагает возможность конфигурирования цепочек обработки на основе распределенных программных модулей.

2 Описание прецедентов

2.1 Прецеденты, выполняемые оператором, как предусловия

В данном параграфе рассмотрены основные прецеденты, выполняемые оператором МЦС ДЗЗ ИКИТ при построении алгоритмов. Перечисленные ниже прецеденты являются входными и обязательными предусловиями для функционирования ПМЗП.

2.1.1 Прецедент О2: Добавление программного модуля в БД

Краткое описание

Оператор вносит информацию о программных модулях, которые используются для построения цепочек обработки данных.

Основное действующее лицо — оператор.

Предусловия

1. Оператор имеет доступ к БД.
2. База данных Системы содержит необходимые поля для ввода информации о модуле.

Поток событий

Прецедент начинается, когда Оператор выбирает деятельность «Добавить модуль».

Базовый поток — добавление программного модуля в БД

1. Оператор в БД выбирает «Добавить модуль»
2. БД Системы отображает список доступных полей для ввода информации.

3. Оператор выбирает одно из полей.
4. Оператор вводит информацию о модуле.
5. Выполнять пункты 3-4, пока все необходимые поля не будут заполнены.
6. Оператор нажимает кнопку «Сохранить модуль».
7. БД сохраняет модуль с информацией о нем, внесенной в таблицы.

Постусловия

При успешном окончании прецедента программный модуль, готовый к запуску, хранится в базе данных.

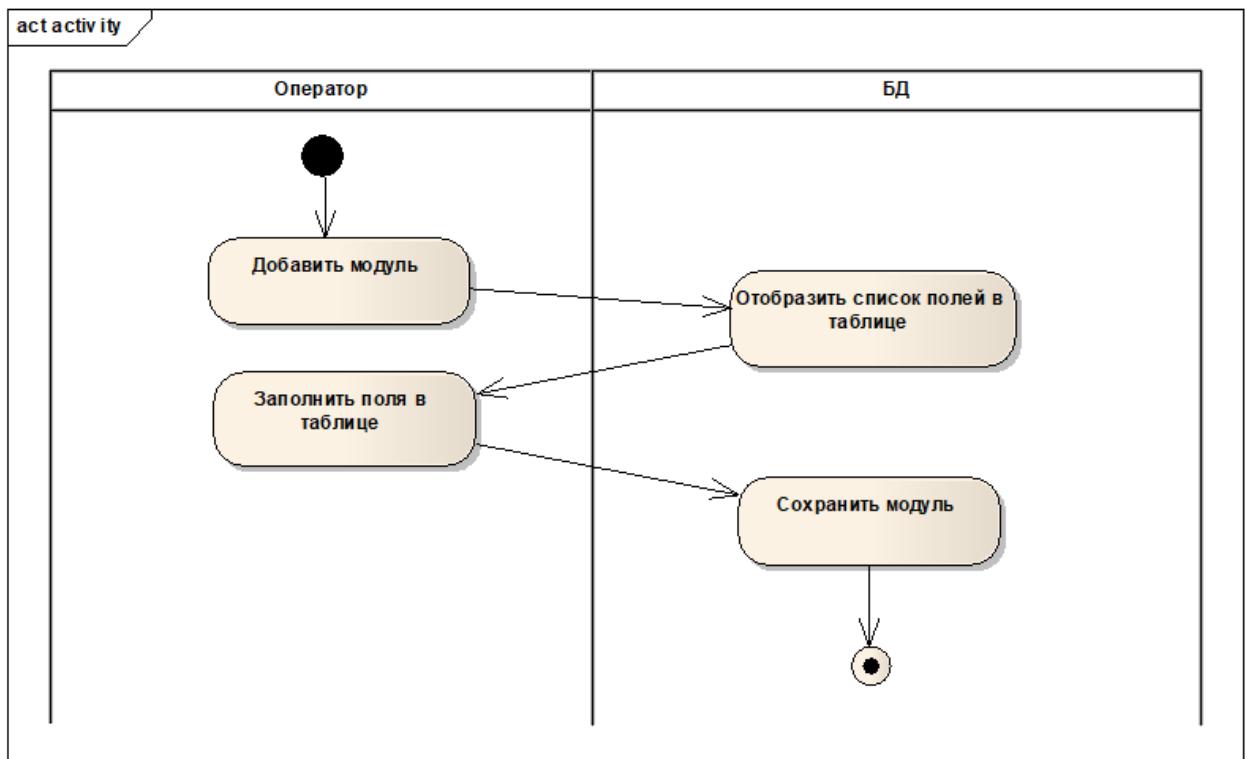


Рисунок В. 1 — Диаграмма деятельности прецедента «Добавление модуля в БД»

2.1.2 Прецедент О3: Построение последовательности блоков алгоритма

Краткое описание

Оператор строит из возможных программных модулей для обработки данных заданную последовательность и сохраняет это в базу данных.

Основное действующее лицо — оператор.

Предусловия

1. Оператор авторизован в Системе.
2. База данных Системы содержит необходимые программные модули, составляющие последовательность алгоритма.
3. Программный модуль имеет интерфейс управления через командную строку.

Поток событий

Прецедент начинается, когда Оператор выбирает деятельность «Построить алгоритм».

Базовый поток – построение последовательности блоков алгоритма

1. Оператор выбирает «Построить алгоритм»
2. Система отображает список доступных программных модулей.
3. Оператор выбирает один из модулей.
4. Система графически отображает блок, характеризующий выбранный программный модуль обработки и анализа данных.
5. Оператор нажимает кнопку «Параметры» на блоке модуля.
6. Система визуализирует диалоговое окно с элементами ввода данных.
7. Оператор вводит имя параметра и его значение (при наличии).
8. Оператор нажимает кнопку «Добавить».
9. Система сохраняет введенные параметры в таблицу БД.
10. Выполнять п. 2 – 9, пока алгоритм не будет сформирован.
11. Определить порядок выполнения программных модулей в составе алгоритма.
12. Оператор нажимает кнопку «Сохранить алгоритм».
13. Система сохраняет алгоритм, как совокупность модулей, в таблицы БД.

Альтернативные потоки

Оператор может выстроить последовательность модулей вручную, используя интерфейс командной строки на каждой отдельной серверной платформе, где размещен тот или иной модуль. Такой подход является неприемлемым.

Постусловия

При успешном окончании прецедента алгоритм, готовый к запуску, хранится в базе данных.

Точки расширения

При достижении п. 3 базового потока переходим к рассмотрению следующего прецедента «Составить алгоритм из предложенных блоков».

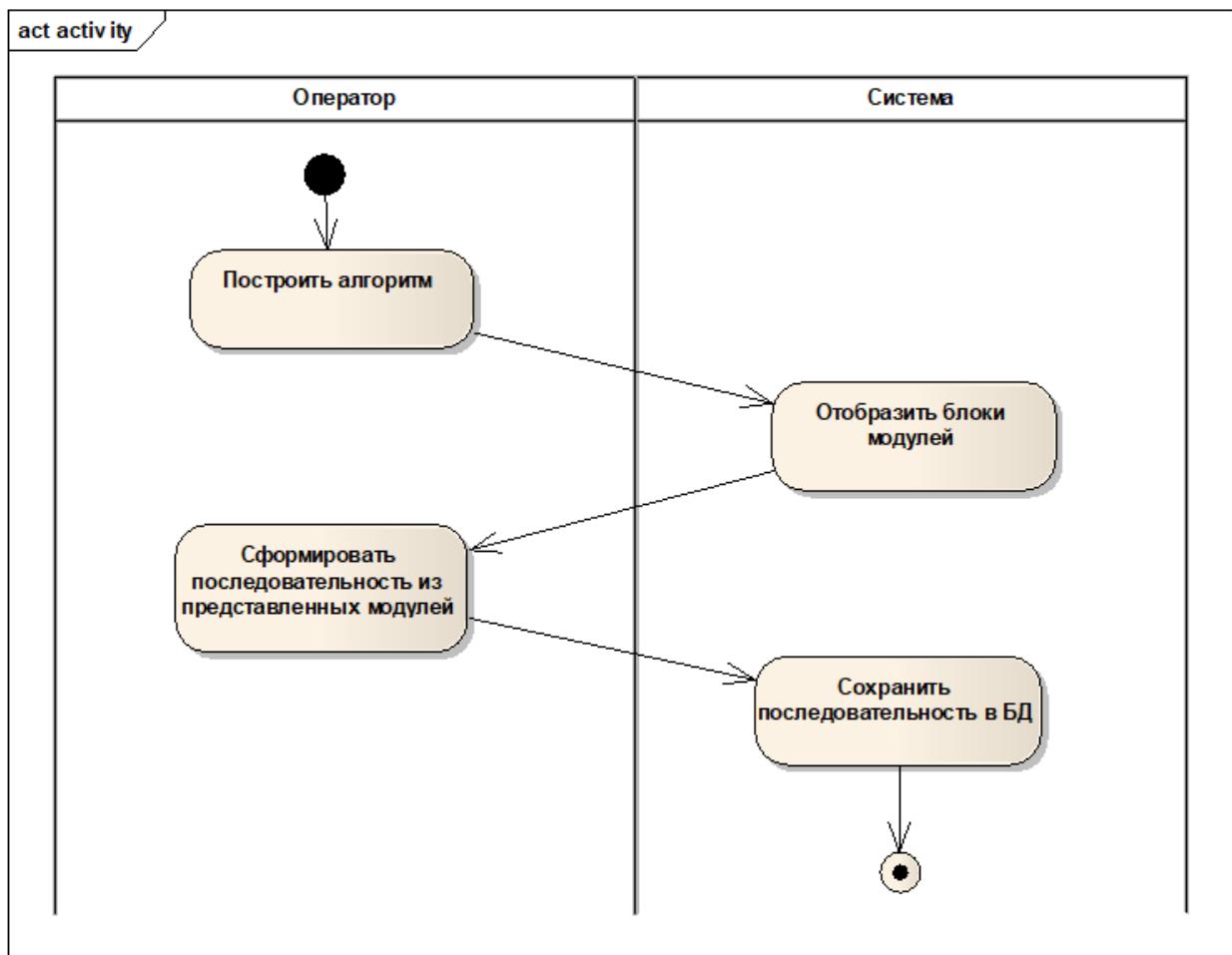


Рисунок В. 2 — Диаграмма деятельности прецедента «Построение последовательности блоков алгоритма»

При достижении п. 11 базового потока следует рассмотреть прецедент «Задание порядка выполнения вычислительных операций».

2.1.3 Прецедент О7: Принудительное завершение процесса

Краткое описание

Оператор может принудительно завершить процесс до его окончания.

Основное действующее лицо — Оператор.

Предусловия

1. В БД Системы хранятся соответствующие задания.
2. Оператор авторизован в Системе.

Поток событий

Прецедент начинается, когда оператор задает статус процессу “canceled”.

Базовый поток — завершение процесса

- 1 Оператор заходит в БД
- 2 Оператор по ID находит нужный программный модуль в алгоритмической конструкции.
- 3 Оператор присваивает программному модулю статус “canceled”.

Постусловия

При успешном окончании precedента, процесс отменен и не будет выполняться.

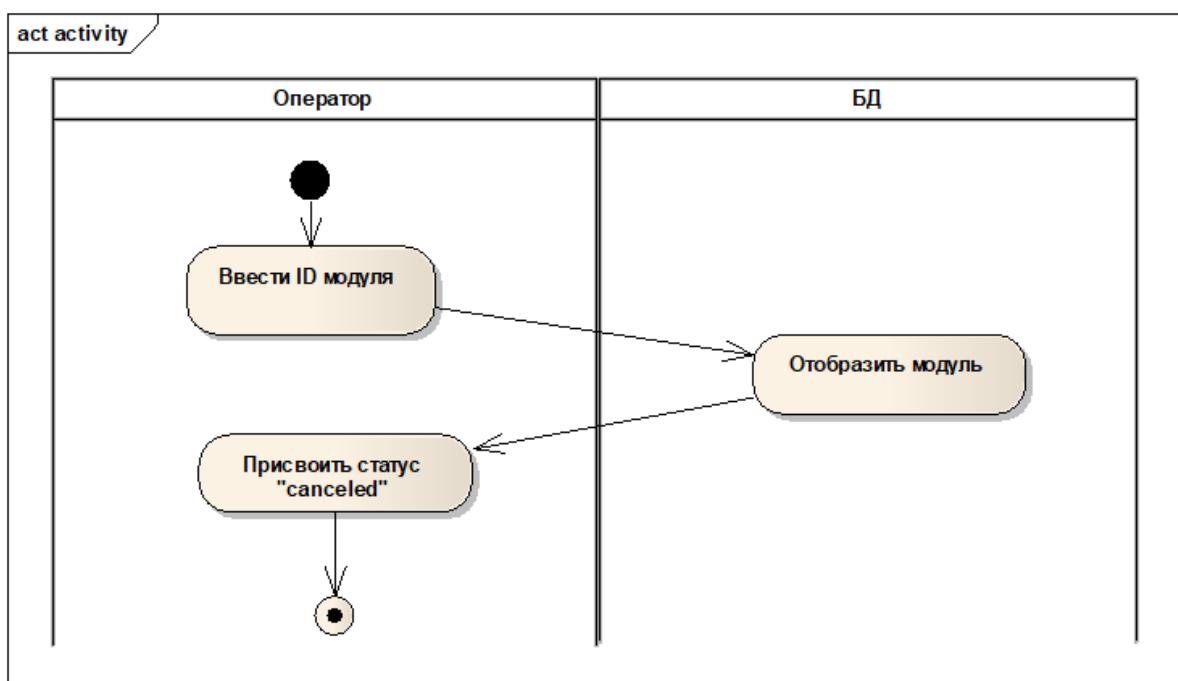


Рисунок В. 3 — Диаграмма деятельности precedента «Принудительное завершение процесса»

2.1.4 Прецедент О8: Задание количества одновременно выполняющихся процессов

Краткое описание

Оператор задает количество параллельно выполняющихся процессов.

Основное действующее лицо — Оператор

Предусловия

1. Оператор авторизован в системе
2. В БД хранятся необходимые модули.

Поток событий

Прецедент начинается, когда оператор создает задание.

Базовый поток — задание количества одновременно выполняющихся процессов

1. Оператор создает задания.
2. CRON считывает задание и запускает программные модули алгоритма.
3. Повтор п.2, пока все задания не будут запущены.

Постусловия

При успешном окончании прецедента, процессы работают одновременно, пока не завершатся.

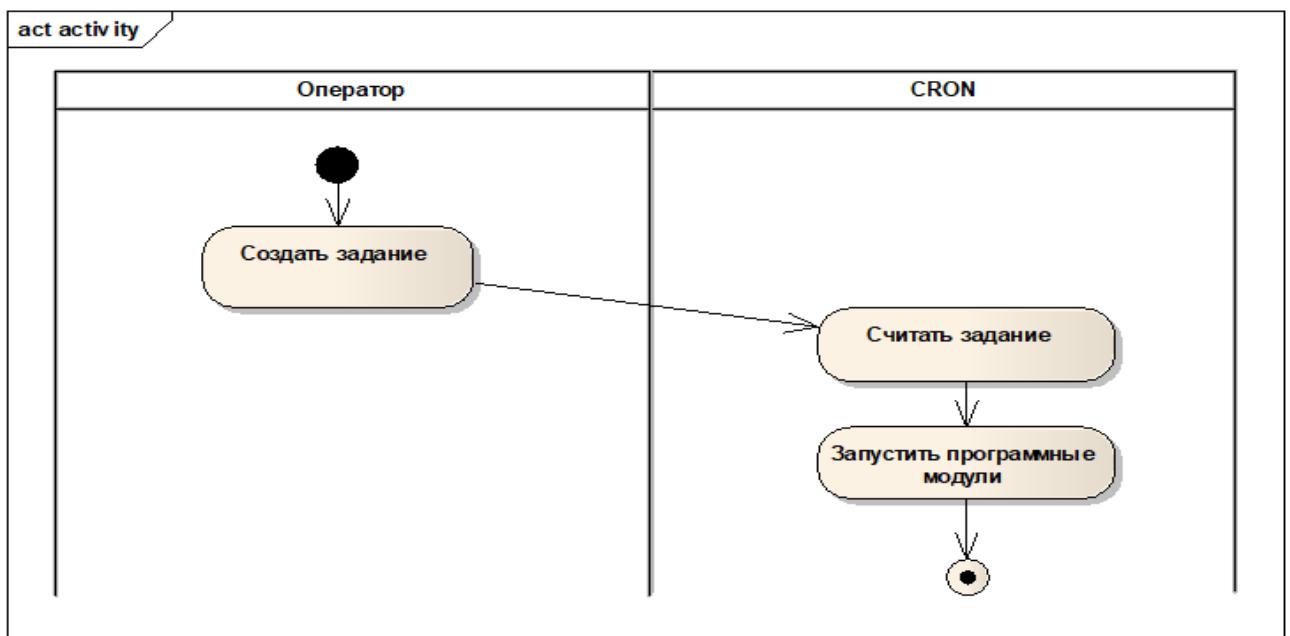


Рисунок В. 4 — Диаграмма деятельности прецедента «Задание количества одновременно выполняющихся процессов»

2.2 Прецеденты программного модуля запуска процессов

В данном параграфе рассмотрены основные прецеденты, выполняемые ПМЗП, входящим в МЦРС ДЗЗ ИКИТ.

2.2.1 Прецедент МП4: задание статуса процессу

Краткое описание

ПМЗП задает статус процессу. Задания имеют иные статусы: plan — означает, что процесс запланирован на определенную дату и время; now — означает, что процесс должен быть выполнен немедленно; canceled — процесс отменен пользователем; ready — процесс завершен; inprogress — процесс выполняется.

Основное действующее лицо — ПМЗП.

Предусловия

1. В БД Системы хранятся соответствующие задания.

Поток событий

Прецедент начинается, когда модуль получает информационные запросы со статусом “now/plan”.

Базовый поток – задание статуса

1. Модуль получает информационные запросы со статусом «now/plan».
2. Инициализируется массив для хранения алгоритмических конструкций.
3. Проверяется наличие существующего алгоритма.
4. Инициализируется список последовательности команд запуска операторов алгоритма.
5. Модуль получает параметры операторов.
6. Инициализирует очередь
7. Модуль задает статус процессу “inprogress”.
8. Запустить команду.
9. Ожидается сигнал завершения работы оператора.
10. Модуль задает статус выполнения “ready”.

Постусловия

При успешном окончании прецедента, все программные модули будут иметь статус “ready”, “inprogress”, “canceled”, “now”, “plan”.



Рисунок В. 5 — Диаграмма деятельности прецедента «Задание статуса процессу»

2.2.2 Прецедент МПЗ: получение и запись параметров

Краткое описание

ПМЗП записывает в строку команды запуска параметры оператора и получает множество заданных параметров оператора.

Основное действующее лицо — ПМЗП.

Предусловия

1. В БД Системы хранятся необходимые программные модули.
2. В БД созданы условия для записи параметров оператора.

Поток событий

Прецедент начинается, когда модуль получает данные оператора по ID.

Базовый поток — получение и запись параметров.

1. Модуль получает кортеж данных об алгоритме.
2. Модуль получает множество операторов, входящих в состав алгоритма.
3. Модуль получает кортеж данных оператора по ID.
4. Модуль записывает в строку команды запуска параметра оператора, заданные по умолчанию.
5. Модуль получает множество заданных параметров.

Постусловия

При успешном окончании precedента, параметры всех операторов записаны и получены.

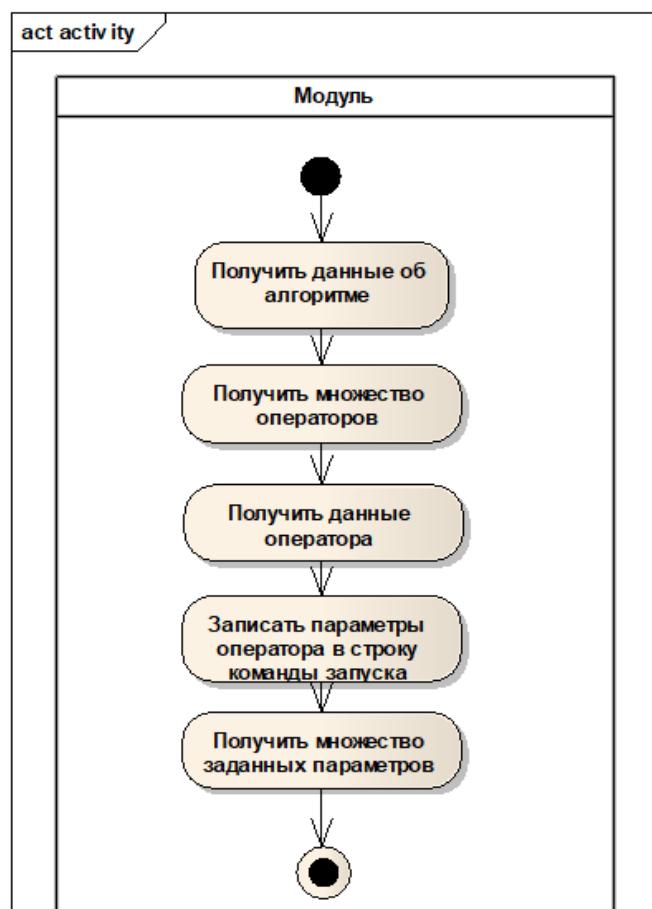


Рисунок В. 6 — Диаграмма деятельности precedента «Получение и запись параметров»

2.2.3 Прецедент МП5: создание очереди

Краткое описание

ПМЗП создает очередь и назначает каждому оператору из алгоритмической конструкции поток.

Основное действующее лицо — ПМЗП.

Предусловия

1. В БД Системы хранятся необходимые программные модули.

Поток событий

Прецедент начинается, когда модуль инициализирует очередь.

Базовый поток — создание очереди

1. Модуль получает алгоритм.
2. Модуль получает множество операторов алгоритма.
3. Модуль инициализирует очередь из операторов.

Постусловия

При успешном окончании precedента, выстроена очередь из операторов алгоритма.

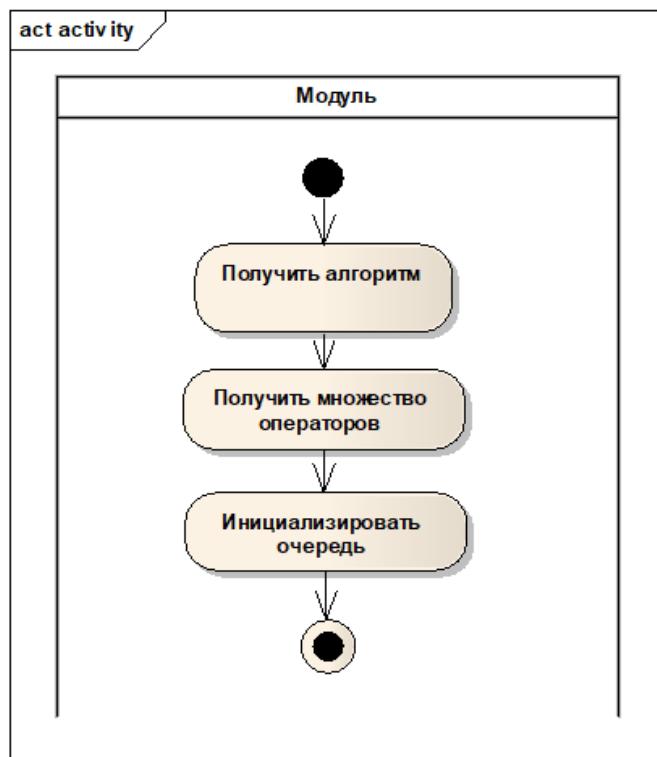


Рисунок В. 7 — Диаграмма деятельности precedента «Создание очереди»

2.2.4 Прецедент МП6: запуск алгоритма

Краткое описание

Модуль управления процессами запускает на выполнение задания, которым соответствуют алгоритмические конструкции, представленные в БД и имеющие статусы 0 — удален или 1 — не удален. Задания имеют иные статусы: plan — означает, что процесс запланирован на определенную дату и время; now — означает, что процесс должен быть выполнен немедленно; canceled — процесс отменен пользователем; ready — процесс завершен; inprogress — процесс выполняется.

Основное действующее лицо — ПМЗП.

Предусловия

1. Частота запуска алгоритмов модулем управления процессами определена и задана в планировщике CRON.
2. Последовательность модулей и их входные и выходные параметры определены оператором.
3. В БД Системы сформирован соответствующий заданной последовательности алгоритм.

Поток событий

Прецедент инициируется планировщиком по заданному расписанию (1 раз в минуту).

Базовый поток — запуск модуля

1. CRON осуществляет запуск модуля управления процессами.
2. Модуль выполняет запрос к БД на выборку подготовленных алгоритмов.
3. БД возвращает результат запроса.
4. Модуль получает последовательность модулей алгоритма.
5. Модуль формирует команду для добавления алгоритма в очередь.
6. Модуль создаёт системную очередь.
7. Модуль создаёт системный процесс.
8. Модуль ожидает завершения работы процесса.
9. Модуль задаёт процессу статус 3 — завершён.
10. БД обновляет статус процесса.

Альтернативные потоки

При выполнении пункта 1 базового потока возникает точка принятия решения в случае, когда подготовленных алгоритмов для выполнения не найдено. В этом случае модуль переходит в режим ожидания следующего запуска по расписанию.

Оператор может запускать модули вручную по расписанию, используя интерфейс командной строки и контролируя их выполнение.

Постусловия

При успешном окончании прецедента, будут последовательно выполнены все процессы, занесенные в базу данных.

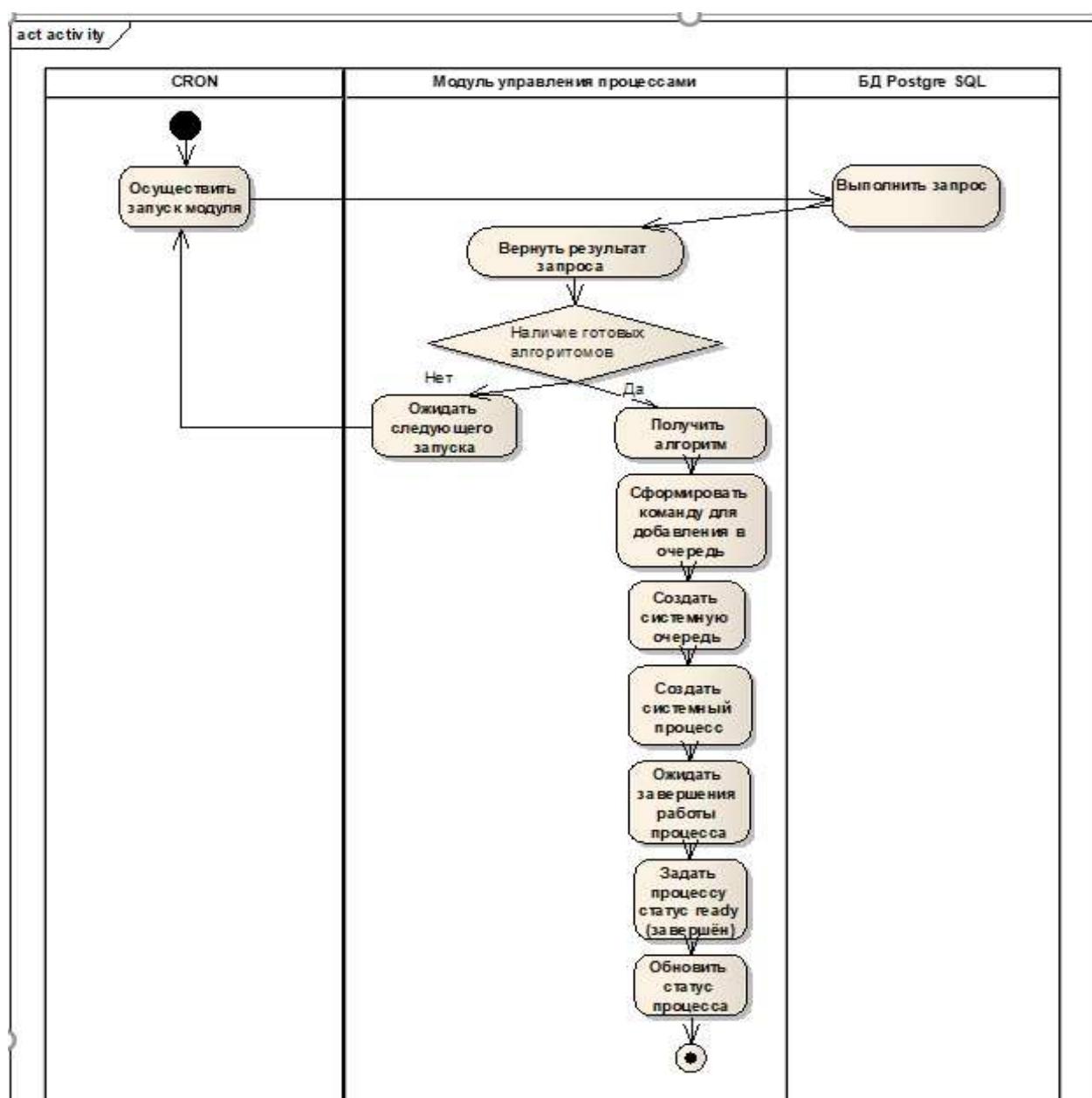


Рисунок В. 8 — Диаграмма деятельности прецедента «Запуск модуля»

2.3 Специальные требования

2.3.1.1 Автоматизированный процесс многоократного запуска компонент

Модуль должен автоматически запускать набор операций в виде совокупности логически взаимосвязанных программных модулей, выполняющихся в заданной последовательности, с заданным набором входных параметров, в заданное время.

2.3.1.2 Интеграция информации

Информации обо всех модулях гетерогенной системы должна быть интегрирована на основе единой базы данных.

2.3.1.3 Гибкое создание процессов

В графическом интерактивном режиме должны быть условия выбора модулей из имеющихся и возможность составления цепочек программных модулей обработки информации при условии, что системные компоненты могут быть распределены по различным серверам.

2.3.1.4 Оперативная корректировка алгоритмов

В базе данных должна быть возможность изменения последовательности алгоритма путём добавления/удаления модуля в базе данных.

2.3.1.5 Оптимизация ресурсов

После создания модуля должно быть рациональное распределение ресурсов между параллельными процессами. Количество одновременно выполняющихся процессов обработки данных ДЗЗ при системных характеристиках (Виртуальная машина VMWare ESXi50 на аппаратной базе IBM Blade Server E HS21XM с двумя виртуальными процессорами Intel Xeon E5420 по 2.5GHz, Оперативная память 16Gb, Дисковое пространство 10Gb, 2Gb, 35Tb на аппаратной базе IBM Blade Server E HS21XM) не более 5.

2.3.1.6 Отсутствие постоянного участия и контроля оператора

Со стороны оператора должно быть отсутствие постоянного контроля и участия при многократном запуске программных компонент в условиях пространственной распределенности и многозадачности.

2.3.2 Ограничения проектирования

2.3.2.1 X1. Применяемые стандарты

Система должна соответствовать всем стандартам интерфейса пользователя Linux® Windows®.

2.3.2.2 X2. Требования к среде выполнения

Системные требования

Система должна удовлетворять вышеуказанным требованиям на компьютере в следующей минимальной комплектации:

- 16 Гб оперативной памяти
- 10 ГБ свободного дискового пространства
- Виртуальная машина VMWare ESXi50 на аппаратной базе IBM Blade Server E HS21XM с двумя виртуальными процессорами Intel Xeon E5420 по 2.5GHz
- Операционная система Linux.

2.3.2.3 X3. Требования к СУБД и доступу к данным.

В ядре системы должна быть представлена промышленная СУБД PostgreSQL.

ПРИЛОЖЕНИЕ Г

Плакаты презентации



Рисунок Г.1 — Плакат презентации №1

Актуальность

В рамках многоцелевой региональной системы дистанционного зондирования Земли ИКИТ (далее МЦРС ДЗЗ или Система) создаётся подсистема управления потоками работ, для которой требуется разработать модуль управления и автоматического запуска процессов (далее Модуль).

- Необходимость параллельного выполнения процессов обработки данных.
- Необходимость привлечения оператора для запуска модулей процесса.

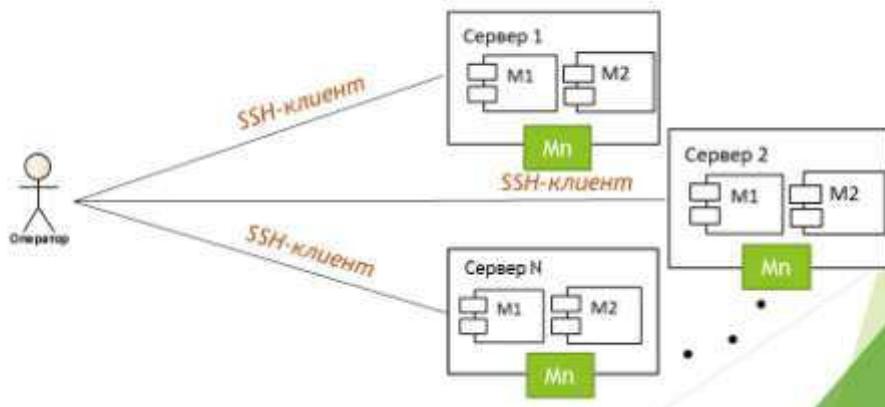


Рисунок Г.2 — Плакат презентации №2

Пример процесса обработки данных

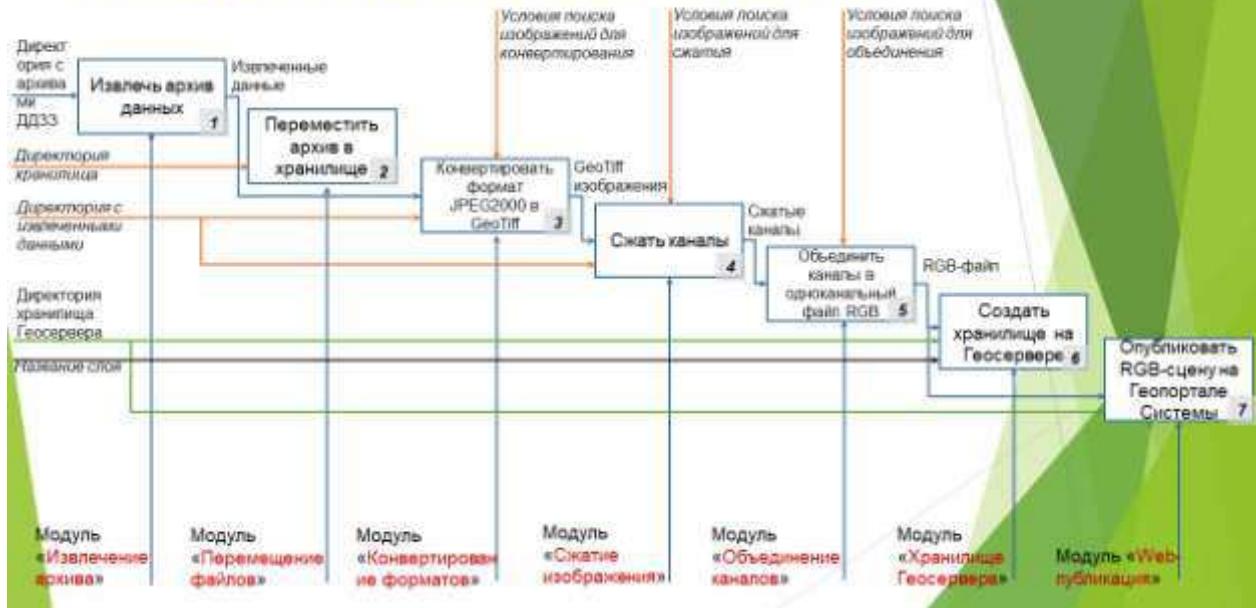


Рисунок Г.3 — Плакат презентации №3

Цели и задачи:

Цель:

- Оптимизировать диалоговое взаимодействие оператора МЦРС ДЗЗ по запуску процессов обработки данных ДЗЗ.

Задачи:

- обзор существующих решений управления распределенными программными средами;
- выявление и анализ требований к Модулю;
- проектирование разрабатываемого Модуля;
- программная реализация Модуля.

Рисунок Г.4 — Плакат презентации №4

Основные элементы подсистемы удаленного управления данными

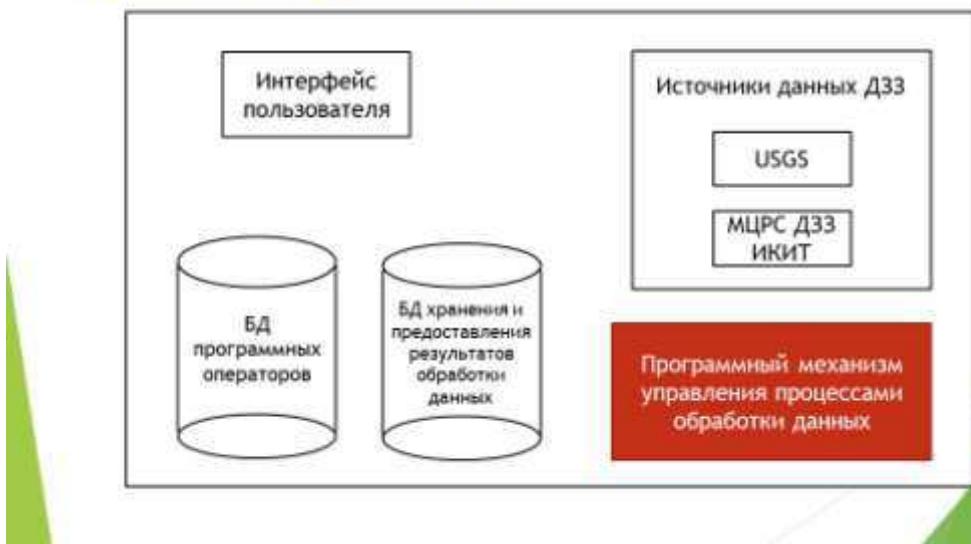


Рисунок Г.5 — Плакат презентации №5

Диаграмма вариантов использования для оператора и ПМЗП

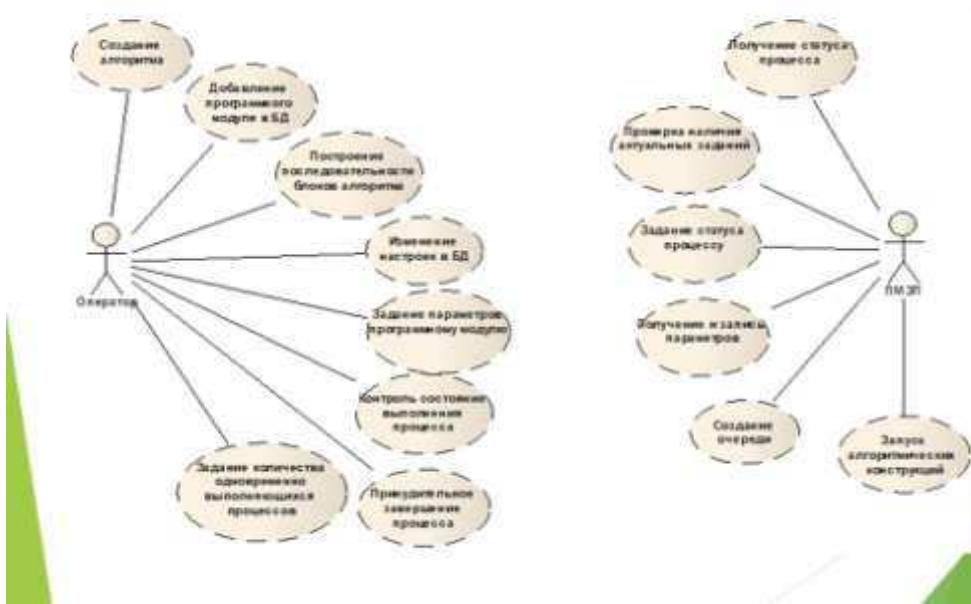


Рисунок Г.6 — Плакат презентации №6



Рисунок Г.7 — Плакат презентации №7



Рисунок Г.8 — Плакат презентации №8

Фрагмент модели данных для работы Модуля

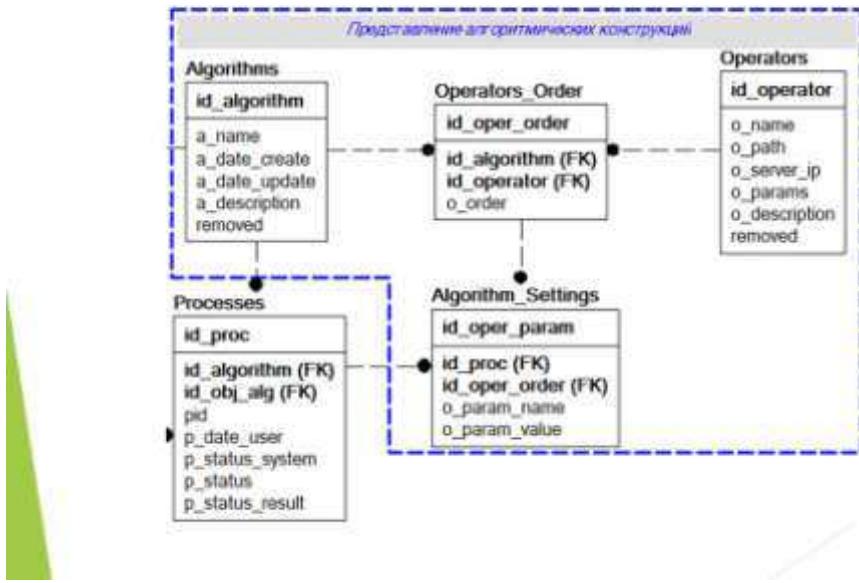


Рисунок Г.9 — Плакат презентации №9



Рисунок Г.10 — Плакат презентации №10

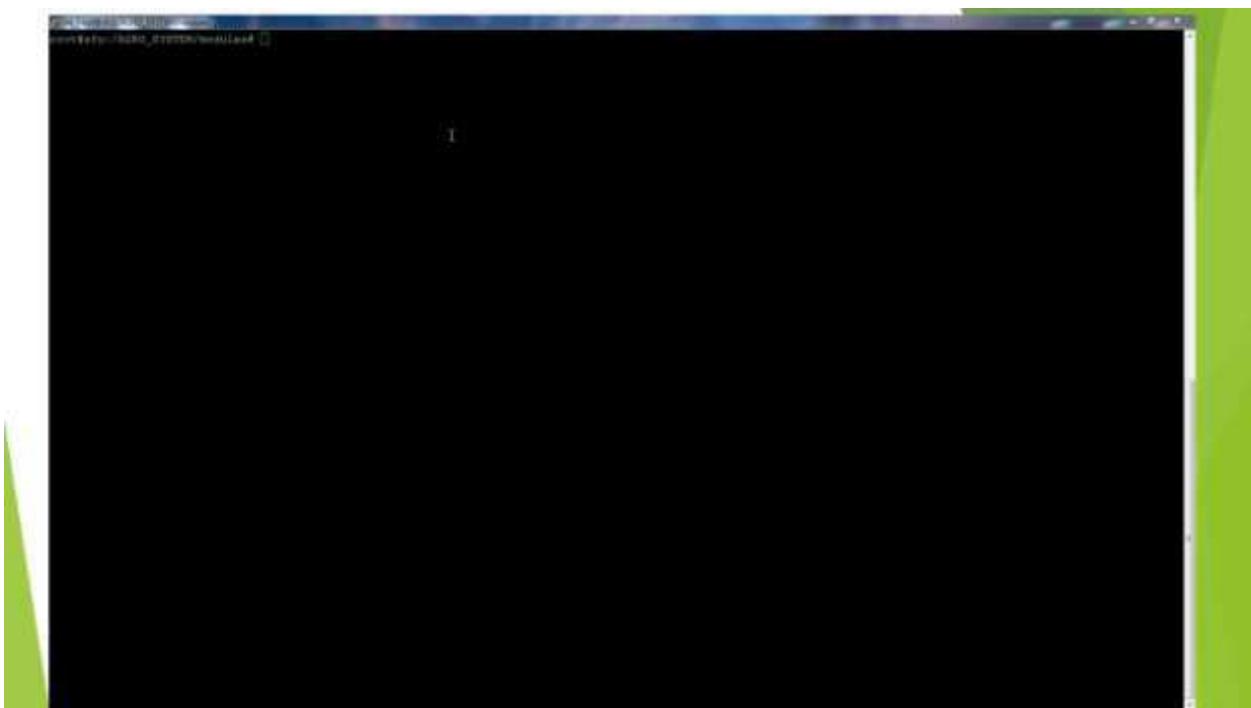


Рисунок Г.11 — Плакат презентации №11

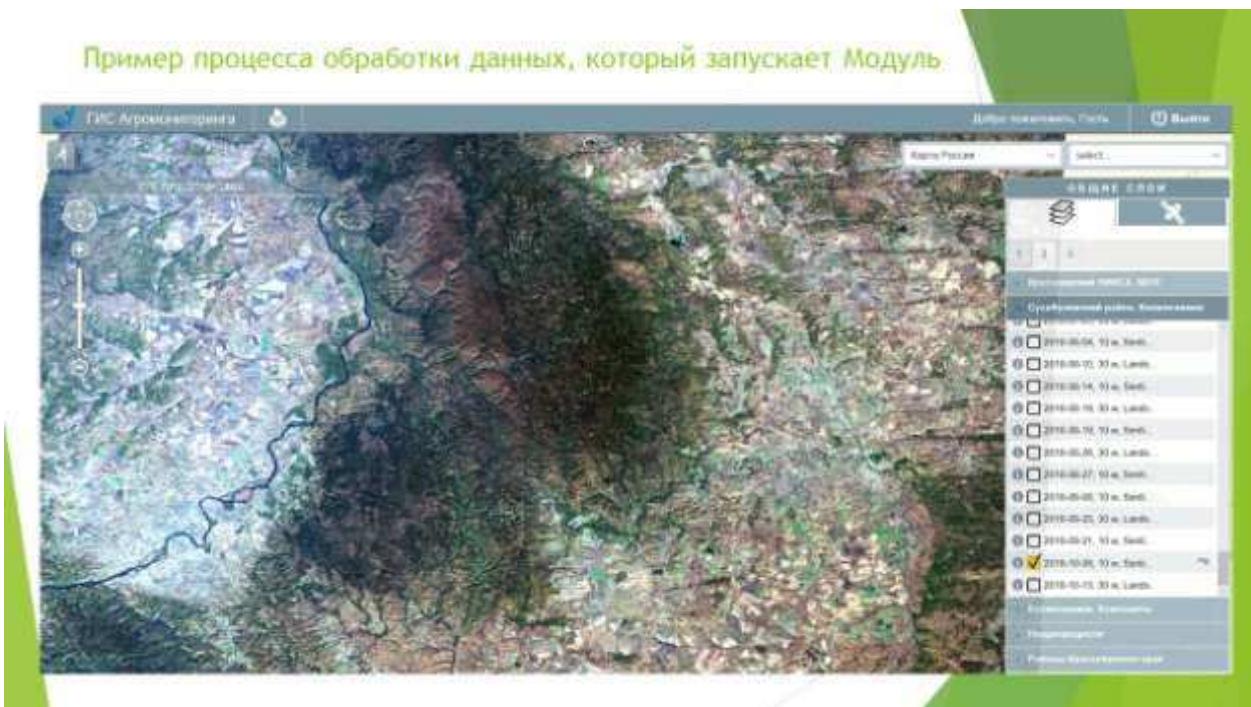


Рисунок Г.12 — Плакат презентации №12

Заключение

В результате выпускной квалификационной работы были выполнены поставленные задачи:

- Выполнен и проведен анализ существующих решений управления распределенными программными средами. Были выведены общие принципы построения систем и их основные элементы.
- При проектировании Модуля создана диаграмма вариантов использования и деятельности, описывающие модель взаимодействия пользователя с модулем запуска программных алгоритмов, диаграмма компонентов системы отражает взаимодействие компонентов внутри системы;
- Разработан модуль управления и автоматического запуска процессов. Модуль способен запускать алгоритмические конструкции, задавать статус процессу.

Рисунок Г.13 — Плакат презентации №13

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

УТВЕРЖДАЮ

Заведующий кафедрой

Г. М. Цибульский

подпись

« 10 » 2019 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 — Информационные системы и технологии

Модуль управления процессами многоцелевой региональной системы ДЗЗ

Руководитель Р. В. Брежнев доцент каф. СИИ, канд. техн. наук Р. В. Брежнев
подпись, дата

Выпускник Д. А. Минибаева Д. А. Минибаева
подпись, дата

Красноярск 2019