

Федеральное государственное бюджетное учреждение науки
Институт динамики систем и теории управления имени В.М. Матросова
Сибирского отделения Российской академии наук
(ИДСТУ СО РАН)

На правах рукописи

Шумилов Александр Сергеевич

**ПРОГРАММНАЯ СИСТЕМА ВЫПОЛНЕНИЯ КОМПОЗИЦИЙ СЕРВИСОВ
ОБРАБОТКИ ПРОСТРАНСТВЕННЫХ ДАННЫХ В РАСПРЕДЕЛЕННОЙ
ГЕТЕРОГЕННОЙ СРЕДЕ**

09.06.01 – Информатика и вычислительная техника

05.13.11 – Математическое и программное обеспечение вычислительных
машин, комплексов и компьютерных сетей

НАУЧНЫЙ ДОКЛАД
об основных результатах научно-квалификационной работы (диссертации)
на соискание ученой степени
кандидата технических наук

Иркутск - 2018

Работа выполнена в лаборатории 4.1. Комплексных информационных систем
отделения 4. Информационных технологий и систем ИДСТУ СО РАН.

Научный руководитель: **Хмельнов Алексей Евгеньевич,**
кандидат технических наук,
первый заместитель директора
ИДСТУ СО РАН по информатизации

Рецензенты: **Сидоров Иван Александрович,**
кандидат технических наук,
научный сотрудник ИДСТУ СО РАН

Бутаков Михаил Игоревич,
кандидат технических наук,
руководитель интеграционных проектов
ООО «Инфотех»

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы диссертации. В последние годы активно развивается сервис-ориентированный подход реализации программных продуктов в виде сервисов и организации их распределенной обработки в вычислительных сетях, обладающий следующими преимуществами: сервисы кроссплатформенны, тестируемы, интерфейс доступа стандартизирован. Распределенные сервисы активно применяются для обработки пространственных данных. Для сервисов обработки пространственных данных характерны манипулирование большими объемами информации и использование специфичных стандартов форматов данных и интерфейсов сервисов, большинство из которых разработаны концерном Open Geospatial Consortium (OGC) и являются открытыми. С развитием и увеличением количества сервисов для решения сложных междисциплинарных задач возникла необходимость создания и выполнения их композиций – объединений существующих сервисов с определенными между ними соотношениями и взаимодействиями, нацеленными на решение конкретных задач.

Композиции могут задаваться в текстовом виде в соответствии с одним из стандартов, а также с помощью программных сред, использующих графические примитивы для построения взаимодействия сервисов. Значительный вклад в задачу построения композиций сервисов внесли Тельнов Ю.Ф., Сухорослов О.В., Фостер И., Папазоглу М., Паутассо К., Тернер М., Будген Д., Пелц К., Рипон С.

Композиции сервисов выполняются в рамках распределенной гетерогенной вычислительной среды, т.е. вычислительные узлы отличаются своим географическим расположением и имеют разные программные и аппаратные характеристики. Количество вычислительных узлов ограничено, поэтому требуется планирование выполнения сервисов с учетом различных критериев, например, уменьшения времени выполнения композиции, равномерного распределения нагрузки по вычислительным узлам и т.д. Данная задача исторически проистекает из проблемы выполнения пакетов прикладных программ (ППП), что делает возможным применение уже существующих подходов и алгоритмов. Ученые, повлиявшие на область выполнения композиций сервисов и пакетов прикладных программ в распределенных средах: Горбунов-Посадов М.М., Опарин Г.А., Бухановский А.В., Насонов Д.А., Топкугло Х., Харири С., Гупта С., Таи Л.

Несмотря на значимые научно-прикладные результаты по разработке методов и теоретических основ организации выполнения композиций сервисов и пакетов прикладных программ в распределенных средах существующие реализации методов построения и выполнения композиций сервисов с помощью процедурных языков программирования не учитывают распределенность и гетерогенность вычислительной среды, что не позволяет использовать существующие наработки по оптимизации выполнения и автоматической адаптации к изменяющимся условиям вычислительной среды. Это определяет актуальность разработки новых методов задания композиций сервисов на процедурных языках программирования, а также реализации программной

системы, выполняющей автоматическое планирование и распараллеливание выполнения композиций сервисов, расположенных на узлах распределенной гетерогенной вычислительной среды.

Цель работы – разработка и реализация программной системы организации выполнения заданных на процедурных языках программирования композиций сервисов обработки пространственных данных в распределенной гетерогенной среде.

Основными задачами исследования являются:

1. Анализ существующих подходов, методов и программных средств организации выполнения композиций распределенных сервисов обработки пространственных данных, а также особенностей гетерогенной среды;
2. Построение концептуальной модели выполнения композиций сервисов в распределенной гетерогенной среде;
3. Разработка методов задания и планирования выполнения процедурных композиций распределенных сервисов в распределенной гетерогенной среде;
4. Реализация и апробация программного средства организации выполнения процедурных композиций сервисов.

Объект исследования – теоретические и информационные аспекты задания и выполнения процедурных композиций сервисов обработки пространственных данных, распределенная гетерогенная информационно-вычислительная среда.

Предмет исследования: методы задания композиций сервисов; методы планирования и выполнения процедурных композиций сервисов в распределенной гетерогенной вычислительной среде.

Методы исследования. В работе использовались методы информационного моделирования, теории графов, системного и объектно-ориентированного программирования, проектирования баз данных, построения распределенных комплексов проблемно-ориентированных программ, веб-технологий, планирования выполнения в статических и динамических средах.

Научная новизна диссертационной работы:

1. Оригинальный метод задания процедурных композиций сервисов обработки пространственных данных с помощью языка программирования JavaScript, позволяющий производить обработку промежуточных данных с помощью средств языка и его библиотек, использовать промежуточные данные в управляющих конструкциях языка. Метод упрощает подключение новых сервисов и передачу пространственных данных между сервисами.

2. Впервые реализован метод выполнения процедурных композиций сервисов обработки пространственных данных, позволяющий применить существующие методы планирования выполнения композиций и параллельной обработки пространственных данных в гетерогенной динамической вычислительной среде.

3. Реализована не имеющая аналогов многопользовательская интернет-система организации выполнения сервисов обработки пространственных данных, использующая существующие методы планирования, выполнения и параллельной обработки для процедурных композиций в распределенной гетерогенной среде.

Положения, выносимые на защиту:

1. Метод задания композиций распределенных сервисов обработки пространственных данных в распределенной гетерогенной среде с помощью процедурного языка программирования JavaScript.
2. Метод выполнения процедурных композиций сервисов обработки пространственных данных в распределенной гетерогенной вычислительной среде.
3. Реализация программной системы задания и выполнения процедурных композиций сервисов обработки пространственных данных в распределенной гетерогенной среде в виде многопользовательской интернет-системы.

Теоретическая и практическая значимость. Основные научные результаты по теме работы получены в рамках Программы № I.33П фундаментальных исследований Президиума РАН, проект 0348-2015-0007; Программы фундаментальных исследований государственных академий на 2013-2020 годы, проект IV.38.2.3 «Новые методы, технологии и сервисы обработки пространственных и тематических данных, основанные на декларативных спецификациях и знаниях»; Интеграционной программы ИНЦ СО РАН, проект 0341-2016-0001; проектов РФФИ 17-47-380007_p_a, 16-57-44034 монг_a, 16-07-00411-a, 16-07-00554-a, 16-37-00110 мол_a, а также ведущей научной школы НШ-8081.2016.9.

Созданное программное обеспечение зарегистрировано в Федеральной службе по интеллектуальной собственности, патентам и товарным знакам (№ 2017617913, № 2016663524, № 2014610274).

Практическая значимость результатов подтверждена актами о внедрении, а также их использованием в учебном процессе в ИГУ в рамках курсов «Разработка программного обеспечения».

Соответствие специальности. В соответствии с паспортом специальности 05.13.11 «Математическое и программное обеспечение вычислительных комплексов и компьютерных сетей» квалификационная работа охватывает исследование моделей, методов и алгоритмов проектирования и анализа программных систем; программные инструменты организации взаимодействия программ и их систем; создание программных систем параллельной и распределенной обработки данных. Отраженные в квалификационной работе положения соответствуют пунктам 1, 3, 8 и 9 области исследований специальности 05.13.11.

Достоверность полученных результатов. Достоверность подтверждается обоснованным использованием методов информационного моделирования, теории графов, системного и объектно-ориентированного программирования, проектирования баз данных, построения распределенных комплексов проблемно-ориентированных программ, веб-технологий, планирования выполнения в статических и динамических средах. Результаты апробации программной системы выполнения композиций сервисов в распределенной гетерогенной среде при решении тестовых и прикладных задач опубликованы в открытой печати.

Апробация результатов работы. Основные результаты работы докладывались и обсуждались на научных конференциях: Международная конференция «Second Russia and Pacific Conference on Computer Technology and Applications» (г. Владивосток, 2017 г.); Международная научно-практическая конференция «Использование современных информационных технологий в ботанических исследованиях» (г. Апатиты, 2017 г.); Всероссийская конференция «Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов» (г. Новосибирск, 2017 г.); XVIII Всероссийская конференция молодых ученых по математическому моделированию и информационным технологиям (г. Новосибирск, 2017 г.); Международная конференция «Информационные технологии и математическое моделирование в науке, технике и образовании» (г. Бишкек, 2016 г.); Национальный суперкомпьютерный форум (НСКФ-2016) (г. Переславль-Залесский, 2016 г.); Байкальская Всероссийская конференция и школа-семинар научной молодежи (г. Иркутск, 2016, 2017 гг.); Международная конференция «Вычислительные и информационные технологии в науке, технике и образовании» (CITech-2015) (г. Алматы (Казахстан), 2015 г.); Российско-монгольская конференция молодых ученых по математическому моделированию, вычислительно-информационным технологиям и управлению (г. Иркутск-Ханх (Монголия), 2013, 2015, 2016 гг.); Всероссийская конференция «Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов» (с. Усть-Сема, Республика Алтай, 2014 г.); XXXVIII Дальневосточная математическая школа-семинар имени акад. Е.В. Золотова (г. Владивосток, 2014 г.); Всероссийская конференция «Обработка пространственных данных и дистанционный мониторинг природной среды и масштабных антропогенных процессов» (г. Барнаул, 2013 г.); III Всероссийская конференция «Математическое моделирование и вычислительно-информационные технологии в междисциплинарных научных исследованиях» (г. Иркутск, 2013 г.); «Ляпуновские чтения» (г. Иркутск, 2014-2017 гг.).

Личный вклад соискателя и других. Результаты квалификационной работы опубликованы в 39 печатных работах, в том числе 7 статей в рецензируемых журналах, рекомендованных ВАК для опубликования результатов диссертаций, 1 коллективная монография, 26 публикаций в трудах международных и всероссийских конференций, 3 свидетельства о государственной регистрации программ для ЭВМ.

Все выносимые на защиту научные положения получены соискателем лично. В основных научных работах по теме работы, опубликованных в соавторстве, лично соискателем разработаны: в [1,3,4,5] – метод задания процедурных композиций сервисов в виде сценариев на языке программирования JavaScript; в [1] – метод организации планирования выполнения процедурных композиций сервисов в распределенной гетерогенной динамической вычислительной среде; в [2,6,7] – реализация программной системы задания и выполнения процедурных композиций сервисов в гетерогенной динамической среде в виде интернет-системы.

Благодарности. Автор выражает глубокую благодарность научному руководителю к.т.н. А.Е. Хмельнову за руководство научно-исследовательской работой, к.т.н. Р.К. Федорову за постановку и поддержку при решении задач, совместную работу, вдохновение и помощь, д.т.н. Г.М. Ружникову и академику И.В. Бычкову за обсуждение и полезные замечания при выполнении работы.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

В главе 1 производится анализ текущего состояния развития сервис-ориентированных сред, технологий обнаружения и методов задания композиций сервисов, рассматриваются различные методы планирования выполнения композиций распределенных сервисов обработки пространственных данных.

В главе 2 рассматривается проблема задания и планирования выполнения композиций сервисов обработки пространственных данных в гетерогенной среде на процедурном языке программирования. Существующие методы задания композиций сервисов – графический (используя графические примитивы) и текстовый (с использованием одного из существующих текстовых стандартов) имеют свои ограничения, рассмотренные в главе 1. Предлагается задавать композиции сервисов в виде программ на одном из распространенных языков программирования. Планирование выполнения композиций должно стремиться к наименьшему времени выполнения, при этом учитывая изменения, происходящие с вычислительными узлами и сетевой инфраструктурой вычислительной среды. Решением данной проблемы является разработанный в рамках диссертационной работы алгоритм.

В п. 2.1 рассматривается проблема планирования выполнения композиций сервисов и приводится математическая модель проблемы планирования. Композиция сервисов представляется в виде направленного ациклического графа. Вершинам графа соответствуют вызовы сервисов (задания), а ребрам – зависимости между вызовами сервисов по данным. Пример такого графа приведен на рисунке 1.

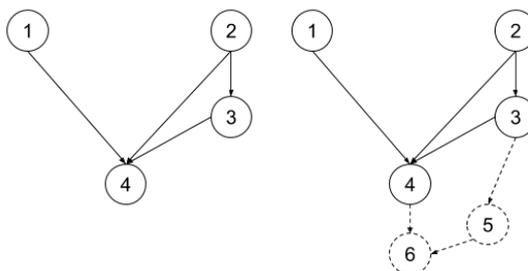


Рисунок 1 – Добавление заданий в граф

В процессе выполнения сценария направленный ациклический граф может расширяться. На рисунке 1 представлена ситуация, когда граф дополняется еще двумя заданиями с идентификаторами 5 и 6.

Задача поиска расписания является NP-полной, т.е. нахождение оптимального решения в некоторых случаях для такого рода задач является практически невыполнимым. Таким образом, следует осуществлять поиск приемлемого решения.

Математической моделью задачи поиска расписания выполнения композиции сервисов является $M = (N, S, T, E, W)$, где N – множество вычислительных узлов, которое может меняться с течением времени, например, из-за изменения квоты на ресурсы, поломок узлов и т.д. Предполагается, что эти изменения трудно спрогнозировать. S – множество сервисов, имеющихся в среде. W – множество пар (s_i, n_j) , показывающее, что сервис s_i установлен на узле n_j . Множество W может меняться в процессе выполнения композиции. T – множество заданий, каждое задание выполняется определенным сервисом, задание может быть выполнено на любом узле, на котором установлен сервис. Множество T может увеличиваться в процессе выполнения композиции, в частности, из-за ветвления алгоритма в зависимости от результатов работы сервисов композиции могут добавляться разные задания. E – множество ребер графа, представляющих собой зависимости между заданиями по данным, соответственно, (T, E) представляет собой ациклический направленный граф (DAG) зависимостей заданий по данным.

Определим $P(T, N)$ как некое расписание выполнения заданий, назначенных на какие-либо узлы. Расписанием $P(T, N)$ является набор упорядоченных последовательностей заданий вида (t_i, \dots, t_k) для каждого узла n_j . Множеством допустимых расписаний называется набор всех расписаний для заданий T и вычислительных узлов N , для которых выполняется отношение предшествования заданий и возможности выполнения заданий на узлах.

Определим вспомогательную функцию $busy(n_i, P)$ – время освобождения узла, т.е. момент времени, когда на вычислительный узел можно назначать новые задания. Если на узел не назначены задания, то значение $busy(n_i, P)$ равно текущему времени. Если на вычислительный узел назначены какие-либо задания, то значение $busy(n_i, P)$ равно либо времени завершения последнего назначенного задания, либо значению текущего времени, если оно больше, чем время завершения последнего назначенного задания.

Определим целевую функцию $evaluate(P) = \max_{0 \leq i < N} \{busy(n_i, P)\}$, определяющую время завершения работы всего расписания. Задача планирования заключается в построении такого расписания, которое бы минимизировало время выполнения композиции сервисов $evaluate(P) \rightarrow \min$ на множестве допустимых расписаний.

В п. 2.2 рассматривается задание композиций сервисов на процедурном языке программирования и динамическое формирование графа заданий DAG. В процессе выполнения программы на процедурном языке программирования оно имеет ряд преимуществ:

1. Наполнение DAG происходит в автоматическом режиме по мере выполнения программ с учетом управляющих конструкций, имеющихся в теле программ;

2. Создание заданий DAG осуществляется посредством вызова заранее заданных функций, входные параметры которых являются параметрами вызываемых сервисов.

Программный код формирования графа заданий DAG называется сценарием композиции. Для выбранного языка программирования формируется библиотека, в которой для каждого зарегистрированного в системе сервиса сформирована уникальная функция-обертка, с помощью которой осуществляется вызов сервиса в коде сценария. При вызове данных функций происходит добавление вершины в граф композиции с указанием сервиса, его характеристик и параметров, производится определение его зависимости от других заданий.

Вызов функций-оберток не требует блокирования выполнения сценария. Блокирование необходимо, если ветвление алгоритма сценария зависит от данных, получаемых в процессе выполнения сервисов. Соответственно, граф DAG формируется по мере выполнения сценария и завершения работы сервисов.

В соответствии с теоремой Бема-Якопини любой исполняемый алгоритм может быть преобразован к структурированному виду, т.е. ход его выполнения определяется при помощи трех структур управления: последовательностей, ветвлений и повторов. Данные структуры влияют на составление DAG следующим образом:

1. Последовательности представляют собой упорядоченные вызовы команд, в данном случае это могут быть функции-обертки, вызывающие соответствующие сервисы. При выполнении последовательностей DAG заполняется последовательно;

2. Ветвления представляют собой условные операторы, которые выполняют код в зависимости от определенных условий, наполнение DAG будет зависеть от результата ветвления;

3. Повторяющиеся вызовы определенного кода – если отсутствует блокирующая обработка результатов выполнения вызванных сервисов, то DAG наполняется последовательными вызовами повторяющихся сервисов.

II. 2.3 определяет алгоритм планирования (рисунок 2), который учитывает текущее состояние вычислительной среды, т.е. степень выполнения и временные характеристики сервисов, доступность вычислительных узлов. Алгоритм при каждом планировании стремится к минимизации времени выполнения композиции сервисов.

Событием называется изменение вычислительной среды, которое может потребовать перепланирования выполнения композиции. К событиям относятся изменения в доступности вычислительных узлов (включение или отключение узлов) и времени выполнения заданий (сервисы могут завершать свою работу раньше или позже ожидаемого времени), ошибки при выполнении заданий.

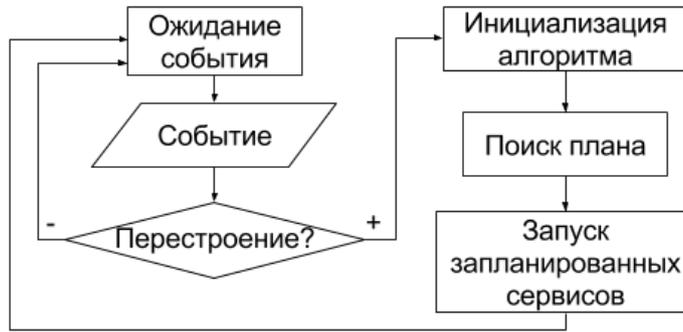


Рисунок 2 – Алгоритм планирования

При инициализации алгоритма происходит составление DAG, при этом учитывается текущее состояние множества заданий A и множества вычислительных узлов N . Для каждого задания определяется время начала выполнения (если оно запущено) и завершения работы (если оно завершило свою работу), список зависимых заданий, текущее состояние выполнения, набор вычислительных узлов, на которых оно может быть запущено. Для каждого вычислительного узла определяется его статус (включен или выключен). На основании информации о заданиях и вычислительных узлах производится построение DAG.

Поиск плана осуществляется на основании созданного DAG и информации о текущем моменте времени. Результатом планирования является список заданий, для каждого из которых определено время запуска и предположительное время окончания их работы.

Для нахождения расписания, обеспечивающего приближенное время выполнения композиции заданий, используется модификация спискового эвристического алгоритма HEFT (Heterogeneous Earliest Finish Time), реализующего метод поиска в глубину. Особенности алгоритма, служащими для ускорения нахождения приемлемого расписания, являются сортировка заданий, узлов и отсечение неперспективных ветвей графа поиска решения, использование информации о текущем состоянии вычислительной среды.

Для отсечения неперспективных ветвей поиска используется эвристическая функция $aggregatedCost(t_i)$, вычисляющая суммарное время выполнения задания t_i и всех ее зависимых заданий путем прохождения графа DAG, начиная с выходного задания t_{end} до самого задания t_i :

если $t_i \in T^{exit}$, **то** $aggregatedCost(t_i) = w(t_i) + dt(t_i)$

иначе $aggregatedCost(t_i) =$

$= w(t_i) + dt(t_i) + \max(aggregatedCost(depends(t_i)))$,

где w – минимальное время выполнения задания среди всех поддерживающих его вычислительных узлов, dt – минимальное ненулевое время получения результатов выполнения данного задания (data transmission).

На вход алгоритму поиска расписания поступает описание DAG задачи. Между заданиями определены зависимости. Алгоритм планирования включает несколько шагов:

1. На основании описания DAG составляется список заданий, которые начали выполняться или завершили свою работу (фиксированные задания). Строится фактически выполняющаяся часть плана, что ускоряет и уточняет процесс планирования для оставшихся заданий.

2. Сортировка по убыванию значения функции $aggregatedCost(a_i)$ для каждого нефиксированного задания происходит перед началом работы алгоритма, таким образом, алгоритм поочередно выбирает задания для анализа, начиная с самого затратного по времени.

3. После каждого назначения задания на определенный узел происходит сортировка узлов по возрастанию значения времени их освобождения.

4. Сокращение пространства перебора решений с помощью метода пропуска ветвей графа, реализуемое с помощью алгоритма A^* , в соответствии с которым на каждом шаге решение о рассмотрении какой-либо ветви принимается на основании сравнения значения эвристической функции и наилучшего решения, достигнутого на предыдущих шагах. Функция эвристики складывается из текущего значения загруженности узла и значения $aggregatedCost(a_i)$ для текущего задания.

5. Перебор вариантов расписания осуществляется до значения таймаута L , при достижении L планировщик возвращает самое лучшее из построенных на данный момент расписаний.

Алгоритм планирования выполнения композиции сервисов формирует расписание, занимающее наименьшее время выполнения.

Для увеличения скорости работы алгоритма используется метод фактического воспроизведения плана при перестроении расписания. Для заданий, закончивших свою работу, указывается время начала и завершения их работы. Для заданий, которые выполняются в момент планирования, указывается время начала их работы и предполагаемое время завершения. При перестроении расписания учитывается текущее состояние среды и не тратится время на рассмотрение вариантов построения фактически выполняющейся части плана.

Процесс поиска наилучшего плана графически представлен на рисунке 3. На представленном графе поиска плана вершины $V_{i,j}$ – это назначения задания t_i на вычислительный узел n_j . Вершина V_0 не является назначением какого-либо задания на узел, она соединяет начальные варианты назначений для более удобной работы с графом. Нахождение плана происходит путем прохода графа в глубину, сначала происходит назначение задания t_1 на узел n_1 ($V_{1,1}$), затем назначение задания t_2 на вычислительный узел n_1 ($V_{2,1}$) и т.д. Найденным планом будет называться подграф, представляющий собой цепочку назначений от задания V_0 до задания $V_{M,n}$, где M – количество заданий, $n \in n_0, \dots, n_N$, где N – количество вычислительных узлов.

При перестроении плана возникает ситуация, когда некоторые из заданий уже приступили к выполнению или уже выполнены. В таком случае процесс перестроения плана можно ускорить, учитывая фиксированные задания. При фиксации заданий уменьшается пространство поиска возможных планов. На

рисунке 4 представлен вариант графа поиска плана, когда задания V_0 и $V_{1,1}$ зафиксированы (V_0 зафиксирован всегда, $V_{1,1}$ уже начал свое выполнение). При переборе вариантов построения плана учитывается только та часть графа, которая остается при переборе вариантов после задания $V_{1,1}$ (обозначена пунктиром).

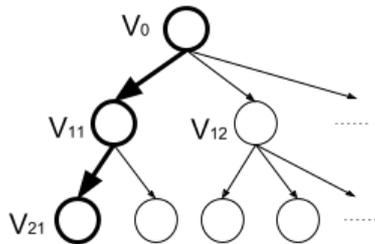


Рисунок 3 – Поиск наилучшего плана

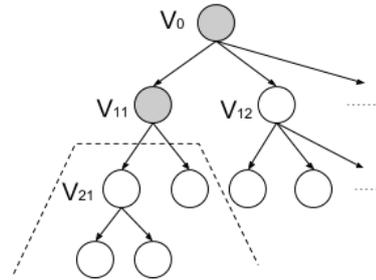


Рисунок 4 – Поиск плана с фиксированными заданиями

Воспроизведение подграфа графа поиска плана на основании фактического состояния вычислительной среды в данный момент времени ускоряет процесс нахождения плана, так как пространство перебора вариантов плана сокращается. При перестроении расписания время выполнения уже выполненных сервисов имеет фактическое значение, полученное в процессе выполнения композиции сервисов, таким образом, перестраиваемое расписание более точно.

Учитывая специфику среды распределенных сервисов, алгоритм построения расписаний выполнения заданий должен перестраивать ранее созданное и частично выполняющееся расписание при наступлении следующих событий:

1. Происходит добавление или удаление вычислительного узла – в случае выхода узла из строя все задания, выполняющиеся на нем, считаются еще не запущенными и ставятся в очередь на запуск, все завершившиеся на отключившемся узле задания при планировании остаются на нем;

2. Наполнение DAG – во время выполнения композиции в DAG добавляются новые задания по мере интерпретации сценария композиции. Перестроение всего расписания не происходит, если добавленный сервис при назначении на самый свободный узел не меняет общее время выполнения сценария;

3. Завершенное задание выполнялось меньше ожидаемого времени – если разница между фактическим и ожидаемым временем меньше значения L , то перестроение не производится, так как оно потребует больше времени, нежели получившийся по времени выполнения данного сервиса выигрыш;

4. Выполняющееся задание не закончилось в ожидаемый срок – происходит перестроение расписания с предположением, что время рассматриваемого задания становится $t = (1 + d) * c(t) + L$, где d – некоторая константа от 0 до 1;

5. Выполняющееся задание завершилось с ошибкой – копия задания добавляется в список заданий, ожидающих назначения на вычислительные узлы, но с изменением в составе узлов, на которых оно может выполняться – узел, на котором оно завершилось с ошибкой, не учитывается при планировании рассматриваемого задания.

Глава 3 рассматривает вопросы реализации предложенного метода задания композиций в виде сценариев на языке программирования JavaScript и метода планирования выполнения композиций сервисов в гетерогенной среде. Реализация разработанных методов требует разработки вспомогательных модулей, которые рассмотрены в данной главе.

В начале главы приводится общая схема системы динамического выполнения композиций сервисов, представленная на рисунке 5. Система состоит из модуля выполнения сценариев, каталога сервисов и сценариев, системы хранения данных (СХД) геопортала, облачной инфраструктуры.

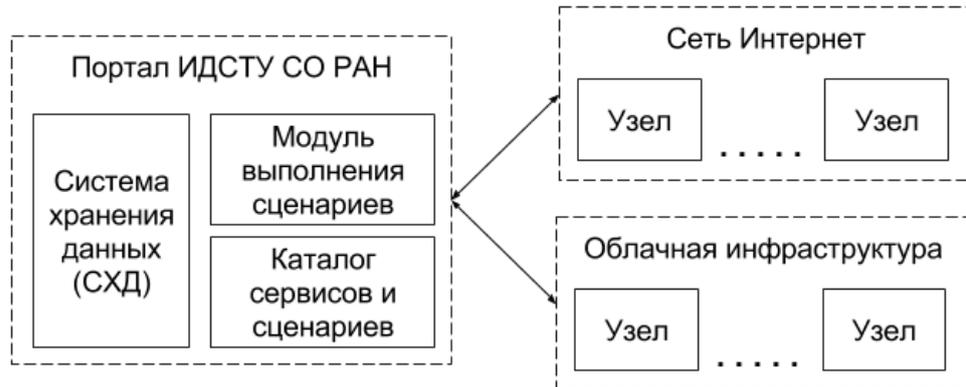


Рисунок 5 – Схема системы динамического выполнения композиций сервисов

Модуль выполнения сценариев, рассмотренный в п. 3.1, отвечает за выполнение сценариев сервисов, построение DAG в процессе выполнения сценария, контроль вычислительной среды, планирование выполнения сервисов, распараллеливание обработки больших массивов данных.

В п. 3.1.1 рассматриваются особенности реализации метода задания композиций распределенных сервисов в виде сценариев на языке программирования JavaScript, а также приводится пример сценария сервисов. Реализация основана на использовании интерпретатора с открытым исходным Google V8. Вводится понятие функции-обертки в рамках сценариев на языке JavaScript.

В п. 3.1.2 описывается вложенность сценариев – способ вызова сценариев внутри других сценариев посредством вызова соответствующей функции-обертки.

В п. 3.1.3 рассматривается механизм установления зависимостей между вызовами сервисов внутри сценариев. Определение зависимостей между вызовами необходимо, так как на основе данных о зависимостях по данным строится DAG.

В п. 3.1.4 рассматриваются преимущества неблокирующего вызова функций-оберток. По сравнению с блокирующим вызовом, при котором выполнение сценария блокируется до тех пор, пока не получен результат вызова сервиса, неблокирующий вызов позволяет выполнять несколько сервисов одновременно, так как при вызове соответствующей сервису функции-обертки сценарий продолжает выполняться.

В п. 3.1.5 рассматриваются основные аспекты реализации и работы с динамически изменяющимся DAG. DAG реализован в виде связного списка структур, для каждой из которых определены ссылки на зависящие и зависимые элементы DAG.

В п. 3.1.6 описывается механизм контроля состояния вычислительной среды, что необходимо для своевременного перестроения расписаний выполнения сервисов.

В п. 3.1.7 рассматривается реализация модуля выполнения сценариев в виде приложения, принимающего на вход код и входные параметры сценария и выполняющего переданный сценарий. Принятый на вход модулем скрипт сценария компилируется с помощью встроенного JavaScript движка и запускается на выполнение. Для учета времени выполнения сервисов на определенных вычислительных узлах имеется модуль статистики.

В п. 3.1.8 рассматривается возможность распараллеливания обработки больших массивов данных в рамках программной модели MapReduce. Использование модели MapReduce повышает скорость обработки больших массивов данных.

Каталог сервисов и сценариев, рассмотренный в п. 3.2, отвечает за регистрацию сервисов, создание сценариев, запуск сервисов и сценариев, хранение результатов выполнения сервисов и сценариев.

В п. 3.2.1 демонстрируется механизм регистрации сервиса в системе динамического выполнения композиций сервисов посредством специальной веб-формы. Данная веб-форма упрощает ввод информации о сервисе, а также позволяет определить элементы управления для каждого входного параметра.

В п. 3.2.2 приводится описание интерфейса, позволяющего регистрировать, выполнять сценарии и просматривать результаты выполнения сценариев с помощью существующих средств Геопортала ИДСТУ СО РАН.

В п. 3.2.3 рассматривается возможность публикации созданных композиций сервисов в виде WPS-сервисов, что делает возможным вызов созданных композиций в виде сервисов извне.

В п. 3.2.4 демонстрируется возможность хранения и просмотра результатов запуска сценариев посредством специального интерфейса.

Система хранения данных геопортала, отвечающая за передачу и хранение данных, получаемых в процессе выполнения сервисов, рассмотрена в п. 3.3.

Облачная инфраструктура, представляющая собой набор вычислительных узлов с развернутыми на них сервисами, рассмотрена в п. 3.4. Для облегчения процесса разработки сервисов была реализована возможность создания виртуальных машин с заранее установленным программным обеспечением для разработки сервисов.

В главе 4 рассматривается три междисциплинарных задачи, решенных с помощью системы динамического выполнения композиций сервисов в гетерогенной среде.

В п. 4.1 в целях апробации устойчивости работы композиции сервисов к происходящим изменениям в гетерогенной вычислительной среде рассмотрена

задача поиска наиболее подходящих растров для классификации видов растительности методом опорных векторов (SVM, Support Vector Machine).

Для решения данной задачи создан сценарий, использующий сервис SVM_Learn. Сервис развернут на 4 вычислительных узлах локальной облачной инфраструктуры и принимает на вход наборы данных в виде векторных и растровых файлов. При выполнении сценария воспроизводятся следующие ситуации:

- отключение половины узлов – происходит перепланирование с учетом того, что два узла становятся недоступными. Через определенное время узлы включаются, система динамического выполнения композиций обнаруживает включение узлов и перестраивает расписание;

- переменное время выполнения сервисов – происходит перепланирование с учетом изменения времени выполнения одного из сервисов на определенном вычислительном узле. Рассматривается случай раннего завершения работы сервиса и случай с временем выполнения больше ожидаемого;

- завершение выполнения заданий с ошибкой – происходит перепланирование с учетом того, что сервис, завершивший работу с ошибкой, больше не назначается на вычислительный узел, на котором ошибка произошла, подразумевается, что копия сервиса на данном узле некорректна и не должна быть вызвана в будущем.

При завершении выполнения композиции сервисов во всех случаях в пользовательскую консоль выводится результат выполнения. Результаты работы сервисов (файлы моделей классификации, которые могут быть впоследствии использованы) автоматически загружены на СХД геопортала.

В п. 4.2 в целях апробации предложенного подхода к планированию выполнения композиции с зависимостями между сервисами по данным рассмотрена задача расчета загрязнения определенной территории загрязняющими веществами. Сценарий учитывает загрязнения от точечных и линейных объектов при известном среднем загрязнении за определенный период от точечного объекта и одного километра линейного объекта, соответственно.

Задача решена с помощью сценария, работающего с тремя сервисами – vectorToGrid, roadToGrid, raster_sum. Сервисы развернуты на 4 вычислительных узлах, причем vectorToGrid выполним только на первых двух. При решении задачи демонстрируется процесс построения графа зависимости сервисов по данным (рисунок б), а также схема построения расписания с учетом ограничений сервисов.

Результатом выполнения композиции, в соответствии с постановкой задачи, является набор растровых файлов, каждый из которых содержит в ячейках суммарное значение загрязнения от точечных и линейных объектов для определенного загрязняющего вещества.

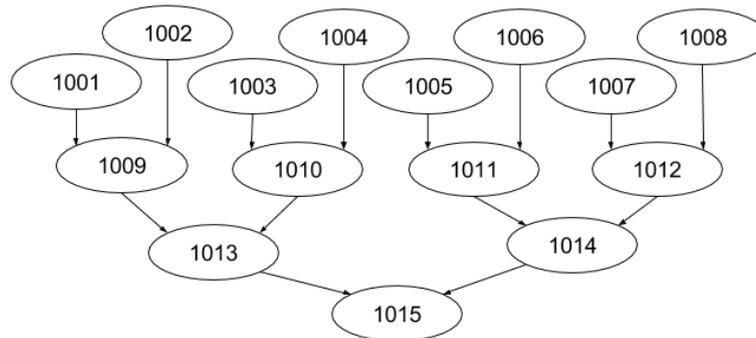


Рисунок 6 – Направленный ациклический граф (DAG) композиции

В п. 4.3 в целях апробации возможности работы с промежуточными результатами работы вычислительных сервисов в рамках выполняемых сценариев была решена задача расчета временной доступности образовательных учреждений для определенного района города с учетом дорожной сети и естественных преград. Данная задача возникает при планировании развития городской инфраструктуры.

В сценарии, приведенном ниже, используются сервисы `shp_to_geojson_converter`, `geojson_to_shp_converter`, `bufferize_vector`, `road_analysis`. Демонстрируется обработка результатов выполнения сервиса `shp_to_geojson_converter` – возвращаемая строка текста обрабатывается JavaScript методом `JSON.parse()` в приведенном ниже фрагменте кода, т.е. результат выполнения сервиса становится доступным внутри сценария в виде стандартного объекта JavaScript. Далее происходит выполнение последовательности оставшихся сервисов для каждого точечного объекта, полученного в результате обработки результата работы сервиса `shp_to_geojson_converter` внутри сценария композиции.

```

function School_availability(input, mapping){
  shp_to_geojson_converter({Source: ...}, { ResultJSON: schools});
  if (schools.get().length > 0) {
    var parsedData = JSON.parse(schools.get());
    for (var i = 0; i < limit; i++) {
      geojson_to_shp_converter({Source: parsedData[i]},
        {ResultFile: shp[i]});
      bufferize_vector({Source: shp [i], Buffer: 50},
        {Result: buffered[i]});
      road_analysis({Cities: buffered [i], Roads: input.roads},
        {Result: resultValueStores[i]});
    }
  }
}

```

Результатом выполнения композиции сервисов является набор растров, каждый из которых содержит данные о времени, необходимом для того, чтобы добраться до каждого из объектов образовательной инфраструктуры пешком.

В **заключении** сформулированы основные выводы и результаты диссертационной работы.

Приложения содержат скан-копии свидетельств о регистрации программ для ЭВМ, являющихся частями данной работы.

ОСНОВНЫЕ ВЫВОДЫ И РЕЗУЛЬТАТЫ РАБОТЫ

Выполненная работа посвящена созданию системы динамического выполнения композиций сервисов обработки пространственных данных в гетерогенной среде. В результате проведенного исследования были решены следующие задачи:

1. Изучены существующие средства организации и выполнения композиций распределенных сервисов, проанализированы методы адаптации процесса выполнения композиций к изменениям в гетерогенной среде распределенных сервисов;

2. Построена концептуальная модель динамического выполнения композиций в гетерогенной среде, определяющая основные процессы и взаимодействия между участниками и компонентами среды;

3. Разработан метод задания композиций распределенных сервисов обработки пространственных данных в гетерогенной среде в виде программ на языке JavaScript, что позволило обрабатывать промежуточные данные внутри композиций с помощью средств языка и его библиотек, а также использовать промежуточные данные в управляющих конструкциях языка;

4. Разработан метод выполнения композиций сервисов обработки пространственных данных в гетерогенной изменяющейся среде, позволяющий применить существующие методы планирования выполнения композиций и параллельной обработки пространственных данных в гетерогенной динамической вычислительной среде;

5. Реализовано инструментальное средство динамического выполнения композиций в гетерогенной среде, создана технология применения разработанного инструментального средства для организации междисциплинарных исследований с участием сервисов обработки пространственных данных;

6. Выполнена апробация разработанного программного средства и в целях демонстрации основных возможностей системы приведены три практические задачи, возникшие в процессе проведения различных междисциплинарных исследований с использованием сервисов обработки пространственных данных.

Все задачи, поставленные перед исследованием, были успешно выполнены. Полученные в рамках квалификационной работы результаты успешно применяются при проведении различных междисциплинарных исследований, требующих использования вычислительных модулей в виде сервисов в сервис-ориентированной среде. Получены и выносятся на защиту следующие результаты:

1. Предложен метод разработки композиций сервисов обработки пространственных данных в виде сценариев на языке программирования

JavaScript. Метод был успешно реализован, поставленные перед методом задачи решены: обработка промежуточных данных с помощью средств языка и его библиотек, использование промежуточных данных в управляющих конструкциях языка, упрощение подключения новых сервисов и передача пространственных данных между сервисами, автоматическое построение графа зависимостей сервисов по данным, распространенность самого языка, на котором реализуются сценарии композиций сервисов;

2. Реализован метод выполнения процедурных композиций сервисов обработки пространственных данных, позволяющий применить существующие методы планирования выполнения композиций и параллельной обработки пространственных данных в гетерогенной динамической вычислительной среде. Выполнение композиций устойчиво к событиям, происходящим в вычислительной среде (изменение доступности вычислительных узлов, изменение времени выполнения отдельных сервисов, ошибки выполнения сервисов);

3. Реализована интернет-система динамического выполнения композиций сервисов обработки пространственных данных в гетерогенной среде, позволяющая как создавать, так и выполнять композиции сервисов для решения задач, возникающих в процессе проведения междисциплинарных исследований. Разработанное программное средство было апробировано в процессе исследований различной направленности.

Разработанная программная система ориентирована на работу в сервис-ориентированной среде распределенных сервисов обработки пространственных данных. Дальнейшим развитием данной системы является увеличение спектра типов регистрируемых в системе сервисов (поддержка WSDL, REST), внедрение на основе технологий виртуализации масштабирования по требованию, реализация диспетчера выполнения сценариев в виде распределенного сервиса с RESTAPI, а также разработка и внедрение новых, более совершенных, подходов к планированию выполнения композиций сервисов в распределенной сервис-ориентированной среде.

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

Монография:

1. Бычков И.В. Инфраструктура информационных ресурсов и технологии создания информационно-аналитических систем территориального управления / И.В. Бычков, Г.М. Ружников, Р.К. Федоров, А.С. Шумилов [и др.] — Новосибирск: Изд-во СО РАН, 2016. — 242 с.

Статьи в журналах из перечня ВАК:

1. Шумилов А.С. Система планирования и выполнения композиций веб-сервисов в гетерогенной динамической среде / Р. К. Федоров, И.В. Бычков, А.С. Шумилов, Г.М. Ружников // Вычислительные технологии. — 2016. — Т. 21, № 6. — С. 18–35.

2. Шумилов А.С. Интернет-система ввода и редактирования пространственных данных «Фарамант» / А.В. Верховина, И.В. Бычков, Г.М.

Ружников, Р.К. Федоров, А.С. Шумилов, А.А. Михайлов // Вестник компьютерных и информационных технологий. — 2015. — № 9. — С. 21–25.

3. Шумилов А.С. Обработка векторных данных с помощью спецификаций в соответствии с моделью MapReduce / Р.К. Федоров, А.С. Шумилов, Ю.В. Авраменко // Вестник Бурятского гос. ун-та. — 2017. — Т. 2. — С. 12–19.

4. Шумилов А.С. Компоненты среды WPS-сервисов обработки геоданных / Р.К. Федоров, И.В. Бычков, А.С. Шумилов, Г.М. Ружников // Вестник Новосибирского гос. ун-та. Сер. Информационные технологии. — 2014. — Т. 12, № 3. — С. 16–24.

5. Шумилов А. С. Сценарий расчета временной доступности объектов образования / Федоров Р. К., Шумилов А. С. // Вестник Бурятского гос. ун-та. — 2017. — Т. 2. — С. 20–32.

6. Шумилов А.С. Информационно-аналитическая система по фиторазнообразию Байкальской Сибири / А.В. Верховина, Р.К. Федоров, С.Г. Казановский, А.С. Шумилов, Д.А. Кривенко, В.В. Мурашко // Известия Иркутского гос. ун-та. Сер. Биология. Экология. — 2016. — Т. 9, № 3. — С. 11–28.

7. Шумилов А.С. Создание и публикация WPS-сервисов на основе облачной инфраструктуры / Р.К. Федоров, А.С. Шумилов // Вестник Бурятского гос. ун-та. — 2015. — № 4. — С. 29–35.

Свидетельства о регистрации программ для ЭВМ:

1. Среда выполнения сервисов и их сценариев: Свидетельство о государственной регистрации программы для ЭВМ № 2017617913 / Р.К. Федоров, И.В. Бычков, Г.М. Ружников, А.С. Шумилов. – Заявка № 2017613155. Дата поступления 10 апреля 2017 г. Зарегистрировано в Реестре программ для ЭВМ 17 июля 2017 г.

2. Система планирования и выполнения композиций веб-сервисов в гетерогенной динамической среде: Свидетельство о государственной регистрации программы для ЭВМ № 2016663724 / Р.К. Федоров, И.В. Бычков, Г.М. Ружников, А.С. Шумилов. – Заявка № 2016660937/69. Дата поступления 18 октября 2016 г. Зарегистрировано в Реестре программ для ЭВМ 15 декабря 2016 г.

3. Интернет-система ввода и редактирования пространственных данных «Фарамант»: Свидетельство о государственной регистрации программы для ЭВМ № 2014610274 / А.С. Шумилов, Р.К. Федоров, А.А. Михайлов, Г.М. Ружников, А.А. Ветров. – Заявка № 2013660173. Дата поступления 6 ноября 2013 г. Зарегистрировано в Реестре программ для ЭВМ 9 января 2014 г.

Статьи в других изданиях:

1. Шумилов А.С. Элементы виртуальной исследовательской среды для анализа больших данных об окружающей среде / И.В. Бычков, Г.М. Ружников, В.В. Парамонов, Р.К. Федоров, С. Будням, А.С. Шумилов // Сб. тр. всерос. конф. «Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов». — Новосибирск, 2017. — С. 7–10.

2. Шумилов А.С. Облако геопорталов / Р.К. Федоров, Г.М. Ружников, А.С. Шумилов // Сб. тр. всерос. конф. «Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов». — Новосибирск, 2017. — С. 305–309.

3. Шумилов А.С. Расчет временной доступности для географических объектов с помощью системы распределенных сервис-ориентированных вычислений / Р.К. Федоров, А.С. Шумилов // Информационные и математические технологии в науке и управлении. — 2017. — Т. 4. — С. 80–90.

4. Шумилов А.С. Облачная среда для научных исследований // Р.К. Федоров, И.В. Бычков, Г.М. Ружников, А.В. Верхозина, А.С. Шумилов // Сб. тр. междунар. конф. «Использование современных информационных технологий в ботанических исследованиях». — Апатиты, 2017. — С. 123–126.

5. Шумилов А.С. Спецификация распараллеливания обработки векторных данных в модели MapReduce / Ю.В. Авраменко, А.С. Шумилов // Информационные и математические технологии в науке и управлении. — 2017. — Т. 4. — С. 71–79.

6. Шумилов А.С. Сервис идентификации объектов на спутниковых снимках / Ю.В. Авраменко, А.С. Шумилов // Сб. тр. XVIII Всерос. конф. молодых ученых по математическому моделированию и информационным технологиям». — Новосибирск: ИВТ СО РАН, 2017. — С. 64 – 65.

7. Шумилов А.С. Инфраструктурный подход обработки пространственных данных в задачах управления территорией / Бычков И.В., Ружников Г.М., Парамонов В.В., Федоров Р.К., Шумилов А.С., Будням С. // Сб. тр. всерос. конф. «Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов». — Новосибирск, 2017. — С. 7–9.

8. Шумилов А.С. Динамическое выполнение композиций сервисов в гетерогенной среде / Р.К. Федоров, И.В. Бычков, Г.М. Ружников, А.С. Шумилов // Сб. тр. междунар. конф. «Информационные технологии и математическое моделирование в науке, технике и образовании». — Бишкек: Изд-во КГТУ им. Раззакова, 2016. — С. 31–42.

9. Шумилов А.С. Метод обработки растровых изображений в рамках модели MAPREDUCE / Ю.В. Авраменко, А.С. Шумилов // Информационные и математические технологии в науке и управлении. — 2016. — Т. 4, № 2. — С. 110–115.

10. Шумилов А.С. Планирование выполнения композиции сервисов в гетерогенной среде / Р.К. Федоров, А.С. Шумилов // Информационные и математические технологии в науке и управлении. 2016. — Т. 4, № 1. — С. 127–136.

11. Шумилов А.С. Спецификации методов разбиения и сборки растровых изображений в рамках программной модели MapReduce / Ю.В. Авраменко, А.С. Шумилов // География и природные ресурсы. — 2016. — № 6 (спец. выпуск). — С. 156–159.

12. Шумилов А.С. Задание графа зависимостей для композиций сервисов с помощью JavaScript сценариев / Р.К. Федоров, А.С. Шумилов // География и природные ресурсы. — 2016. — № 6 (спец. выпуск). — С. 160–163.

13. Shumilov A.S. Integration technologies of heterogeneous software systems in the information-computing environment for mathematical modeling and data analysis / I.V. Bychkov, R.K. Fedorov, G.M. Ruzhnikov, I.A. Sidorov, A.S. Shumilov, V.P. Potapov // Conference proceedings «Computational and Informational Technologies in Science, Engineering and Education» (CITech-2015). — Almaty (Kazakhstan), 2015. — Vol. 20. — P. 93–98.

14. Шумилов А.С. Информационно-аналитическая система мониторинга биологического разнообразия Байкальской природной территории / И.В. Бычков, Г.М. Ружников, Р.К. Федоров, А.В. Верхозина, А.А. Михайлов, В.В. Парамонов, А.С. Шумилов // Актуальные проблемы науки Прибайкалья. — 2015. — № 1. — С. 62–67.

15. Шумилов А.С. WPS-сервисы пространственного анализа состояния окружающей среды и природных ресурсов / Р.К. Федоров, А.С. Шумилов // Труды IV Всероссийского симпозиума «Инфраструктура научных информационных ресурсов и систем». — СПб., 2014. — Т. II. — С. 66–74.

16. Шумилов А.С. Сервисы ввода и редактирования реляционных данных на основе базовых пространственных данных / Федоров Р.К., Федорова Е.Н., Шумилов А.С. // Труды IV Всероссийского симпозиума «Инфраструктура научных информационных ресурсов и систем». — СПб., 2014. — Т. II. — С. 144–152.

17. Шумилов А.С. Среда распределенных вычислений на основе WPS-сервисов // Сб. тр. XXXVIII Дальневосточной матем. школы-семинара имени акад. Е.В. Золотова. — Владивосток: ИАПУ ДВО РАН, 2014. — С. 484 – 489.