

УДК 004.75

doi: 10.18101/2304-5728-2017-2-20-32

**СЦЕНАРИЙ РАСЧЕТА ВРЕМЕННОЙ ДОСТУПНОСТИ ОБЪЕКТОВ ОБРАЗОВАНИЯ<sup>1</sup>****© Федоров Роман Константинович**

кандидат технических наук, ведущий научный сотрудник,  
Институт динамики систем и теории управления  
им. В. М. Матросова СО РАН  
Россия, 664033, Иркутск, ул. Лермонтова, 134  
E-mail: fedorov@icc.ru

**© Шумилов Александр Сергеевич**

аспирант  
Институт динамики систем и теории управления  
им. В. М. Матросова СО РАН  
Россия, 664033, Иркутск, ул. Лермонтова, 134  
E-mail: shumilov@icc.ru

В современном мире информационных технологий активно развивается сервис-ориентированный подход. При решении сложных задач часто возникает необходимость построения композиций — набора взаимодействий между несколькими сервисами. Геопортал ИДСТУ СО РАН предлагает подход, заключающийся в задании композиций в виде программ на языке JavaScript. При выполнении композиций осуществляется распараллеливание входящих в состав композиции сервисов в целях уменьшения общего времени выполнения композиции. Процесс выполнения композиций устойчив к изменениям в вычислительной.

В качестве апробации предлагаемого решается задача расчета временной доступности учреждений образования. Для решения задачи разработаны сервисы для работы с данными и выполнения непосредственных вычислений, задана композиция сервисов в виде сценария на языке JavaScript. Представлен процесс выполнения сценария, рассмотрена реакция Геопортала на изменения, происходящие в среде в процессе выполнения композиции, продемонстрирован результат выполнения сценария и особенности планирования его выполнения.

**Ключевые слова:** SOA; Task scheduling; HEFT; WPS.

---

<sup>1</sup> Работа выполнена при финансовой поддержке РФФИ, номер гранта 16-57-44034\_монг\_а; 16-07-00411\_а; с использованием вычислительных ресурсов Центра коллективного пользования «Иркутский суперкомпьютерный центр СО РАН».

### **Введение**

В современном мире информационных технологий с каждым годом увеличиваются объемы доступных данных. Для обработки и публикации данных все чаще используется сервис-ориентированный подход (SOA — Service-Oriented Architecture), заключающийся в предоставлении вычислительных алгоритмов и источников данных в виде веб-сервисов, доступных через сеть Интернет. Естественным развитием SOA является использование композиций сервисов — наборов взаимодействий между сервисами, нацеленных на решение определенных задач. Так как сервисы могут одновременно выполняться на нескольких вычислительных узлах, возникает проблема распараллеливания выполнения композиций сервисов для уменьшения общего времени выполнения композиции сервисов. Данная работа рассматривает проблему распараллеливания выполнения композиций распределенных сервисов на примере решения задачи расчета временной доступности различных объектов инфраструктуры. Проблему распараллеливания можно представить в виде проблемы формирования такого расписания выполнения сервисов композиции, что общее время выполнения композиции было бы минимальным.

Общепризнанным стандартом определения зависимостей между сервисами в композиции является использование направленного ациклического графа (Directed acyclic graph, DAG), в котором вершинами являются вызовы сервисов, а ребра показывают зависимости между вызовами сервисов по данным. Все алгоритмы нахождения расписания для DAG можно разделить на две группы по способу нахождения расписания — эвристические и метаэвристические алгоритмы.

Одним из наиболее популярных эвристических алгоритмов составления расписания является алгоритм Heterogeneous Earliest Finish Time (HEFT) [1], который является списковым алгоритмом, то есть он сортирует задания на основании заранее вычисленной величины, и назначает их по порядку в списке на менее загруженный на момент назначения узел. Среди метаэвристических алгоритмов наибольшей популярностью пользуется семейство генетических алгоритмов [2]. Они позволяют находить приближенные решения из широкого пространства поиска, применяя эволюционные принципы. Генетические алгоритмы обычно находят более точные решения, нежели эвристические алгоритмы, но проигрывают по времени выполнения.

В последнее время стали все чаще появляться гибридные алгоритмы, сочетающие в себе разные техники — например, алгоритм Hybrid Evolutionary Workflow Scheduling Algorithm for Dynamic Heterogeneous Distributed Computational Environment [3], сочетающий в себе HEFT и генетический алгоритм. Данный алгоритм способен адаптироваться к изменениям вычислительной среды, что особенно важно в распределенных гетерогенных вычислительных средах, когда работа отдельных вычислительных узлов и каналов передачи данных не гарантируется, а также ко-

гда время выполнения сервисов и передачи данных между ними также изменяется со временем. Этот алгоритм учитывает изменения, происходящие в вычислительной среде, например, отключение вычислительного узла или добавление вычислительных сервисов к композиции ведет к мгновенной перестройке расписания, в то время как несоответствие фактического и ожидаемого времени выполнения сервиса будет учтено только в момент следующей плановой перестройки расписания.

Говоря о подходах к заданию композиций сервисов, необходимо рассмотреть два наиболее известных способа их задания — BPEL (Business Process Execution Language) и XPDL (XML Process Definition Language). Эти оба способа дескриптивно задают граф зависимостей заданий DAG. Несмотря на то, что стандарт XPDL был принят раньше, чаще всего в работе используется стандарт BPEL, так как BPEL более приспособлен для описания взаимодействия сервисов. В свою очередь, XPDL позволяет использовать сервисы, не имеющие веб-интерфейс, а также позволяет определять графическое представление заданного взаимодействия.

Одним из наиболее популярных программных пакетов, позволяющих задавать композиции сервисов с помощью формата BPEL, является Oracle BPEL Process Manager. Данный программный продукт предоставляет удобный графический интерфейс для создания композиций сервисов и средство отладки созданных композиций, поддерживает большое количество форматов интерфейса веб-сервисов (WSDL, WPS и т. д.). BPEL Process Manager не осуществляет планирование выполнения сервисов в композиции, что критично при сложных композициях сервисов с большим количеством участвующих вычислительных узлов.

Также известен для задания композиций сервисов программный продукт Taverna, позволяющий с помощью графического интерфейса определять последовательности выполнения заданий. В отличие от императивного BPEL с явным заданием процесса выполнения, Taverna использует функциональную модель с управлением данными. Taverna распространена в научной среде и часто используется в астрономии, биоинформатике (например, для определения генов, ответственных за определенное заболевание). Однако Taverna не осуществляет планирование выполнения композиций в сложных средах.

Таким образом, можно отметить, что существуют подходы к заданию композиции сервисов, где граф зависимостей заданий DAG полностью определяется до момента выполнения, и разработаны алгоритмы составления расписания для таких DAG в целях обеспечения приближенного к оптимальному времени выполнения. Однако отсутствуют алгоритмы и подходы, позволяющие динамически в процессе выполнения определять граф зависимостей заданий DAG и осуществлять его планирование выполнения с учетом особенностей гетерогенных сред распределенных вычислений.

В качестве апробации разработанной среды выполнения композиций распределенных сервисов рассмотрена задача расчета временной доступности объектов инфраструктуры, часто возникающая в градостроительстве при проектировании развивающихся городских территорий. Расчет временной доступности важен при проектировании дорожной сети и расчете их пропускной способности, организации маршрутов общественного транспорта, расположения объектов инфраструктуры и т. д. В работе рассматривается задача расчета временной доступности объектов среднего образования в пределах города Иркутска. При расчете необходимо учитывать доступность с учетом использования дорожной сети и доступность пешком. Следует учитывать естественные преграды, такие как водные объекты, но в то же время необходимо учитывать путепроводы, проходящие через них, например, мосты. Для расчета временной доступности различных объектов инфраструктуры разработаны отдельные сервисы и их композиция, выполняющаяся на нескольких вычислительных узлах локальной облачной инфраструктуры. Рассмотрим подробнее реализацию композиции и ее планирование.

#### **Среда выполнения композиций распределенных сервисов**

Среда выполнения композиций распределенных сервисов — программная система, позволяющая организовывать совместное выполнение веб-сервисов в виде композиций, заданных в виде сценариев на языке программирования JavaScript. Среда интегрирована в Геопортал — программную веб-систему, разработанную в ИДСТУ СО РАН и предоставляющую инструменты для работы с геоинформационными данными. Основными компонентами среды являются:

- веб-интерфейс задания композиций;
- веб-интерфейс ввода параметров запускаемой композиций, её выполнения и просмотра результатов работы;
- диспетчер выполнения композиций сервисов.

Рассмотрим каждый компонент среды распределенных сервис-ориентированных вычислений применительно к задаче, решаемой в данной работе.

*Веб-интерфейс задания композиций* позволяет задавать композиции в виде сценариев на языке программирования JavaScript в специальной веб-форме, упрощающей и корректирующей данный процесс.

Перед заданием кода сценария пользователю необходимо определить входные и выходные параметры. В случае рассматриваемой задачи это будут векторные файлы с точечными объектами инфраструктуры в формате геоинформационных данных SHP. Вторым параметром следует файл с указанием естественных границ (водные объекты) с линейными объектами в формате SHP. Третьим параметром является векторный файл с линейными объектами в формате SHP, содержащий объекты дорожной сети. Четвертым параметром является область производимого расчета (расчет

производится для тех объектов инфраструктуры, которые попадают в выделенную область).

В качестве выходного параметра пользователь указывает папку, в которую необходимо сохранить результаты работы, которые должны быть загружены с удаленного сервера.

После определения входных и выходных параметров сценария пользователь задает код сценария композиции сервисов, представленный ниже в сокращенном виде. При задании кода сценария пользователь использует специальное текстовое поле с подсветкой синтаксиса языка и пользователю предоставляется список доступных для использования сервисов.

```
function School_availability(input, mapping){
  shp_to_geojson_converter({Source: input.schools},
    { ResultJSON: schools});           (1)
  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]});       (2)
      bufferize_vector({Source: shp [i], Buffer: 50},
        {Result: buffered[i]});     (3)
      road_analysis({Cities: buffered [i], Roads: in-
        put.roads}, {Result: resultValueStores[i]}); (4)
    }
  }
}
```

Всего в композиции используется четыре сервиса:

- **shp\_to\_geojson\_converter** (сноска 1 в приведенном коде сценария) — сервис конвертирует SHP файл в формат GeoJSON. В рамках решаемой задачи данный сервис осуществляет преобразование входного файла с точечными объектами образовательной инфраструктуры в строку с данными формата GeoJSON. GeoJSON впоследствии преобразовывается в стандартный объект JavaScript;

- **geojson\_to\_shp\_converter** (сноска 2) — сервис конвертирует данные в формате GeoJSON в SHP файл. Полученный в результате выполнения предыдущего сервиса объект, содержащий все объекты инфраструктуры, разбивается на отдельные объекты, и каждый из полученных объектов преобразовывается в SHP файл для дальнейшей работы;

- **bufferize\_vector** (сноска 3) — сервис создает буферные зоны вокруг точечных объектов переданного SHP файла. Результаты выполнения предыдущего сервиса преобразовываются из точечных в полигональные с определенным радиусом буфера;

- **road\_analysis** (сноска 4) — сервис выполняет анализ доступности точечных объектов на основании данных о дорожной сети и естественных границах. В качестве объектов, для которых необходимо выполнить анализ, используются результаты работы предыдущего сервиса. Ре-

результатом работы сервиса является регулярная сетка в растровом формате данных GeoTIFF, где в каждой ячейке содержится время достижения этой точки от определенного объекта образовательной инфраструктуры.

*Веб-интерфейс ввода параметров запускаемой композиции, ее выполнения и просмотра результатов работы.* Для удобства работы с композициями сервисов разработан специальный интерфейс, его интерфейс разделен на три части:

1. Форма ввода входных параметров композиции — список входных параметров с соответствующими элементами ввода информации, определенными для каждого параметра. Например, входные параметры с файлами объектов образовательной инфраструктуры, дорог и естественных преград имеют элемент управления «Выбор файла», параметр области расчета имеет элемент управления «Выделение области на карте».

2. Консоль выполнения композиции сервисов (доступно после запуска композиции) — отображение процесса выполнения композиции. Сервисная информация, которая отображается в данной консоли, включает в себя характеристики выполняющихся сервисов, результаты запросов к удаленным серверам, планы назначения сервисов на вычислительные узлы.

3. Отображение состояния выполнения композиции (доступно после запуска композиции) — отображение списка вычислительных узлов, их статуса и списка назначенных сервисов, либо графа зависимостей между вызовами сервисов по данным.

*Диспетчер выполнения композиций сервисов* принимает на вход код сценария, а также введенные пользователем параметры и приступает к его выполнению. При выполнении сценария диспетчер:

1. Строит направленный ациклический граф зависимостей между вызовами сервисов по данным по мере выполнения сценария.

2. Выполняет планирование выполнения сервисов внутри сценария в соответствии с происходящими в распределенной среде изменениями.

3. Отслеживает состояние вычислительной среды и перестраивает расписание выполнения сценария на при наступлении определенных событий.

Общепризнанным стандартом постановки задачи планирования композиции сервисов является определение зависимостей между заданиями с помощью направленного ациклического графа (Directed acyclic graph, DAG). В таком графе вершинами являются вызовы сервисов, а ребрами — зависимости между сервисами. Сервисы, которые имеют только выходные дуги, называются входными сервисами, соответственно, те сервисы, которые имеют только входные дуги, называются выходными (результатирующими сервисами). В композиции может быть несколько как входных, так и выходных сервисов.

При выполнении композиций сервисов возникает задача планирования, то есть такого назначения вызовов сервисов композиции на вычисли-

тельные узлы таким образом, что общее время выполнения композиции было бы минимальным. Для нахождения расписания, обеспечивающего приближенное время выполнения композиции заданий, используется модификация спискового эвристического алгоритма составления расписания 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 более быстрого и точного нахождения расписания [4].

Среда распределенных сервисов является гетерогенной, то есть вычислительные узлы, на которых развернуты сервисы, имеют различные характеристики. Вычислительные узлы могут менять свою доступность, возможно изменение в сетевой инфраструктуре, обеспечивающей доступ к узлам. Таким образом, алгоритм построения расписаний выполнения заданий должен перестраивать ранее созданное и частично выполняющееся расписание при наступлении следующих событий:

1. Происходит добавление или удаление вычислительного узла — во время выполнения композиции вычислительные узлы могут выходить из строя и возвращаться в рабочее состояние. В случае выхода узла из строя все задания, выполняющиеся на нем, считаются еще не запущенными и снова ставятся в очередь на выполнение, в то же время все уже выполненные на отключившемся узле задания при планировании остаются на нём (при условии, что результаты выполнения заданий остаются доступными)

2. Происходит добавление задания в DAG — во время выполнения композиции добавляются новые задания по мере интерпретации сценария композиции. Перестроение всего расписания происходит, когда добавляемое задание изменяет прежде рассчитанное время выполнения всего сценария. В случае, когда добавленный сервис при назначении на самый свободный узел не меняет общее время выполнения сценария, перестроение плана не происходит и производится запуск задания на выполнение.

3. Завершенное задание выполнялось меньше ожидаемого времени — если разница между фактическим и ожидаемым временем меньше значения  $L$ , то перестроение не производится, так как перестроение 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 — если в текущем расписании на вычислительном узле после рассматриваемого задания назначены какие-либо другие задания, которые уже начали выполне-

ние, то данное задание помечается как невыполненное. Копия задания добавляется в список заданий, ожидающих планирование, но с изменением в составе узлов, на которых оно может выполняться — узел, на котором оно завершилось с ошибкой, не учитывается при планировании рассматриваемого задания.

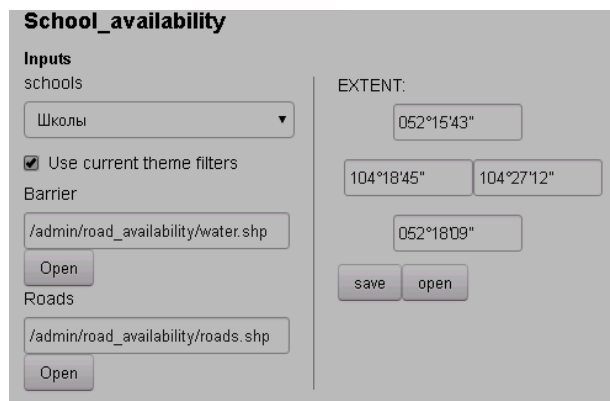
### **Расчет доступности образовательных учреждений**

Для расчета доступности образовательных учреждений необходимо выполнить сценарий композиции сервисов, решающий данную задачу. Перед выполнением необходимо задать входные параметры.

На рис. 2 показана форма ввода входных параметров сервиса.

- Параметр `schools` определяет какую тему следует выбрать для файла с прецедентами. В рамках Геопортала темой называется реляционная таблица, для которой доступно редактирование через специализированный веб-интерфейс. В рассматриваемом случае в качестве темы указывается таблица, содержащая точечные объекты, соответствующие образовательным учреждениям. Тема при передаче в качестве входного параметра в сценарий преобразуется в файл в формате SHP встроенными средствами СУБД PostgreSQL. Количество объектов, для которых выполняется расчет, возможно ограничить, воспользовавшись инструментом выделения области на интерактивной карте. В рассматриваемом случае взяты пять школ, ограниченных прямоугольной областью на карте, представленной на рис. 3.
- Входные параметры `barrier` и `roads` являются файлами в формате SHP, содержавшие линейные объекты с объектами дорожной сети и естественных преград соответственно.
- Параметр `extent` определяет область, для которой выполняется расчет временной доступности объектов.





**School\_availability**

**Inputs**

schools

Школы

Use current theme filters

Barrier

/admin/road\_availability/water.shp

Open

Roads

/admin/road\_availability/roads.shp

Open

**EXTENT:**

052°15'43"

104°18'45" 104°27'12"

052°18'09"

save open

Рис. 2. Входные параметры сценария



Рис. 3. Выбор объектов на карте

Результатом выполнения сценария является набор растровых файлов, соответствующих рассматриваемым объектам образовательной инфраструктуры. Полученные растровые файлы должны быть отображены на интерактивной веб-карте Геопортала.

На рис. 4 представлен автоматически создаваемый ациклический граф зависимости заданий по данным. DAG строится по мере выполнения сценария. Например, задание с идентификатором 1001 соответствует вызову функции `shp_to_geojson_converter`, результаты которой необходимы для запуска функции `geojson_to_shp_converter` в теле цикла. В теле цикла вызываются задания `geojson_to_shp_converter`, `bufferize_vector`, `road_analysis`, что соответствует последовательностям заданий 1002 – 1003 – 1004, 1005 – 1006 – 1007 и т. д.

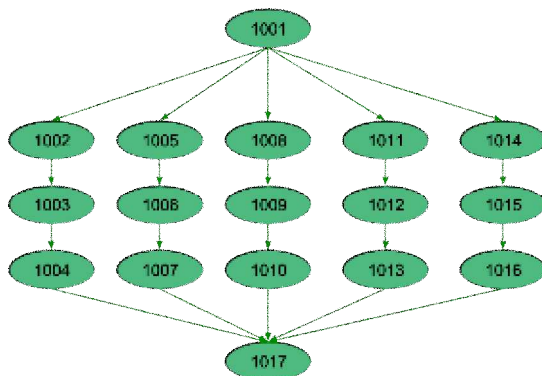


Рис. 4. Направленный ациклический граф зависимости заданий по данным

Так как среда, в пределах которой развернуты распределенные вычислительные сервисы, является гетерогенной, в процессе выполнения будут искусственно отключаться некоторые вычислительные узлы и сервисы в целях имитации изменений вычислительной среды. При наступлении событий, перечисленных выше, должно осуществляться перепланирование выполнения сервисов.

В случае отсутствия изменений в вычислительной среде по мере выполнения сценария расписание выглядит согласно представленному на рис. 5. На рисунке 5 видно, что вычислительные узлы не простаивают, задания компактно выполняются на всех четырех вычислительных узлах.

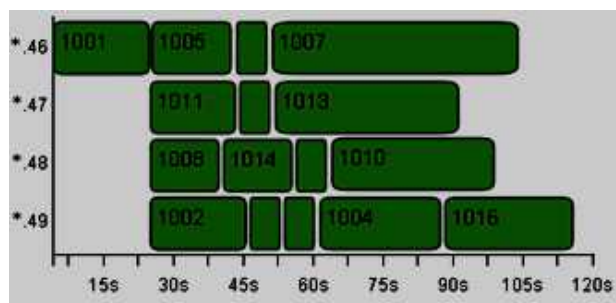


Рис. 5. Расписание выполнения композиции сервисов

При отключении двух из четырех вычислительных узлов алгоритм планирования обрабатывает данную ситуацию и перестраивает расписание. На расписании, представленном на рис. 6, видно, что два вычислительных узла были отключены некоторое время. По истечению некоторого времени с момента обнаружения неисправности вычислительного узла алгоритм пробует выполнить запрос на удаленном вычислительном узле,

и в случае ответа узла, сервисы снова назначаются на этот вычислительный узел.

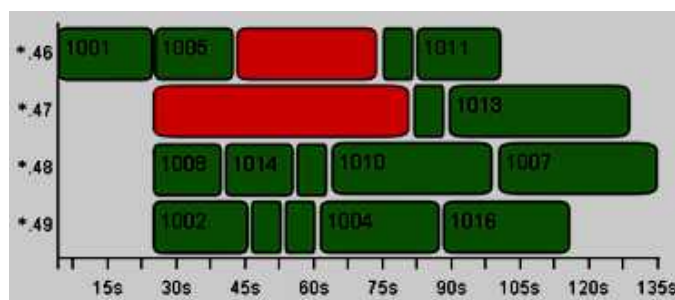


Рис. 6. Расписание в случае отключения вычислительных узлов

В случае, когда вычислительный узел не в состоянии выполнить определенный сервис (например, когда узлы .46 и .47 имеют некорректные копии сервиса `bufferize_vector`, выполняющегося наименьшее время), расписание перестраивается с учетом невыполнимости определенного сервиса на некоторых узлах. На рис. 7 видно, что узлы .46 и .47 не выполняют сервис `bufferize_vector`.

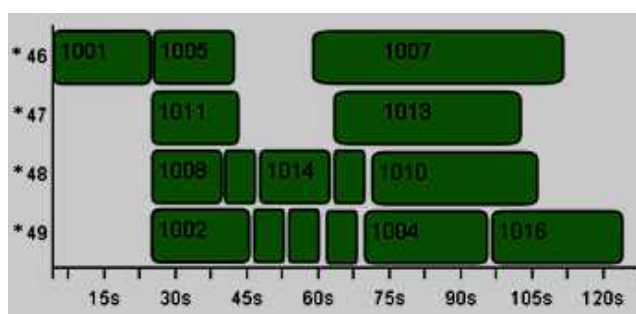


Рис. 7. Расписание в случае невыполнимости сервиса на некоторых узлах

По мере завершения сценария результаты расчета загружаются на Геопортал и визуализируются с помощью интерактивной веб-карты. На рис. 8 представлен скриншот демонстрации результатов расчета временной доступности для двух образовательных учреждений. С помощью инструментов по работе с интерактивной картой Геопортала ячейки отображаемых растровых файлов были окрашены в соответствии с попаданием их значений в интервалы 0–10 минут, 10–20 минут и т. д. Стоит отметить, что на растровых файлах присутствуют пустые ячейки, которые соответствуют естественным преградам (водоемы).

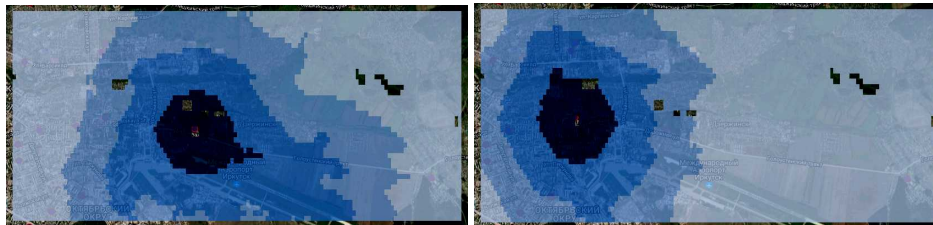


Рис. 8. Визуализация расчета для двух образовательных учреждений

### **Вывод**

В результате применения разработанного подхода к заданию и планированию выполнения композиций веб-сервисов в распределенной гетерогенной среде удалось произвести распараллеливание выполнения композиции сервисов, решающей задачу построения карт временной доступности для определенного набора образовательных учреждений. Для решения задачи потребовалась разработка соответствующих сервисов и организация их совместной работы с помощью задания композиции в виде сценария на языке программирования JavaScript. Разработанная композиция сервисов была выполнена в рамках распределённой гетерогенной среды. При выполнении композиции была апробирована устойчивость планирования выполнения композиций к изменениям, происходящим в гетерогенной среде. Решение задачи было полностью выполнено в рамках инфраструктуры Геопортала ИДСТУ СО РАН.

Достоинством реализации среды выполнения композиций распределенных сервисов является устойчивость к изменениям, происходящим в гетерогенной среде, отсутствие каких-либо специфических конструкций в теле сценария композиции, влияющих на организацию распараллеливания или процесс планирования выполнения сервисов, а также построение взаимодействия между сервисами в виде программ на языке JavaScript для решения сложных задач.

### **Литература**

1. H. Zhao, R. Sakellariou. An Experimental Investigation into the Rank Function of the Heterogeneous Earliest Finish Time Scheduling Algorithm // *Lecture Notes in Computer Science*. 2003. Vol. 2790. P. 189–194.
2. O. Sinnen. *Task scheduling for parallel systems* // Wiley-Interscience, 2007. P. 108.
3. Hybrid Evolutionary Workflow Scheduling Algorithm for Dynamic Heterogeneous Distributed Computational Environment / D. Nasonov [et al] // *Advances in Intelligent Systems and Computing*. 2014. Vol. 299. P. 83–92.
4. 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.

---

A SCRIPT FOR CALCULATING TEMPORAL AVAILABILITY OF EDUCATIONAL ORGANIZATIONS

*Roman K. Fedorov*

Cand. Sci. (Engineering), Leading Researcher,  
Matrosov Institute of System Dynamics and Control Theory, SB RAS,  
134 Lermontova St., Irkutsk 664033, Russia  
E-mail: fedorov@icc.ru

*Aleksandr S. Shumilov*

Research Assistant,  
Matrosov Institute of System Dynamics and Control Theory, SB RAS,  
134 Lermontova St., Irkutsk 664033, Russia  
E-mail: shumilov@icc.ru

The service-oriented approach is actively developing nowadays. While solving complex problem, the need for service composition often arises. The composition is a set of interactions between several services. Geoportal of SB RAS Institute of System Dynamics and Control Theory proposes to define service compositions as JavaScript programs. When implementing the composition, its services are parallelized in order to minimize the overall time for composition execution. This process is stable to the changes that occur in the computational environment.

We propose to solve the problem of calculating the temporal availability of educational institutions. To solve the problem we have developed the services for data processing and performing direct calculations, defined the composition services as a JavaScript program. The process of script execution is presented, as well as the reaction of Geoportal to environment changes, the results and features of planning execution of the script.

*Keywords:* SOA; Task scheduling; HEFT; WPS.