

УДК 519.682.7

**И. В. Бычков, Г. М. Ружников, Р. К. Фёдоров, А. С. Шумилов**

*Институт динамики систем и теории управления СО РАН  
ул. Лермонтова, 134, Иркутск, 664033, Россия*

*bychkov@icc.ru, ruznikov@icc.ru, fedorov@icc.ru, shumisan1011@gmail.com*

## **КОМПОНЕНТЫ СРЕДЫ WPS-СЕРВИСОВ ОБРАБОТКИ ГЕОДАНЫХ**

Описывается разработка среды распределенных вычислений на основе стандарта Web Processing Service, который унифицирует использование Web-сервисов, предоставляющих услуги пространственной обработки растровых и векторных данных.

*Ключевые слова:* OGC, WPS, JavaScript, V8, Calipso.

### **Введение**

Решение сложной научной задачи, как правило, требует применения методов, реализованных в нескольких программных системах (ПС) и разрабатываемых специалистами различных предметных областей. Например, моделирование загрязнения атмосферного воздуха населенных пунктов использует методы инвентаризации источников загрязнений (печное отопление, автотранспорт и т. д.) и моделирования распространения загрязнений. Эти методы достаточно сложные и часто реализуются разными коллективами. Их взаимодействие осуществляется редко, в частности, из-за сложности использования и интеграции программных систем. В итоге при решении сложных задач решается одна или несколько подзадач на должном уровне в зависимости от ресурсов коллектива, а остальные подзадачи решаются упрощенным способом.

Увеличение скорости передачи данных сети Интернет, развитие и стандартизация программного интерфейса браузеров позволяют реализовать многие программные системы на основе Web-технологий, что дает ряд преимуществ, таких как масштабируемость, упрощение распространения, обновления, использование общих информационных ресурсов и т. д. Разработчики многих настольных программных систем создают их Web-версии. Например, Adobe Photoshop, Microsoft office, ArcGIS. По сравнению с настольными системами пользователи получают возможность совместной работы над общими документами, картами и т. д.

Активно развивается взаимодействие между пакетами программ (интероперабельность ПС) через Интернет, используя стандарты Open Geospatial Consortium (OGC)<sup>1</sup>. Одним из перспективных стандартов OGC является Web Processing Service (WPS)<sup>2</sup>, который унифицирует использование через Интернет сервисов обработки пространственных данных (ПД). Например, это могут быть сервисы, реализующие обработку растровых и векторных данных,

---

<sup>1</sup> Open Geospatial Consortium. URL: <http://www.opengeospatial.org/> (дата обращения 04.09.2014).

<sup>2</sup> OpenGIS Web Processing Service (WPS) Implementation Specification, v1.0.0. Release date: June 08, 2007. URL: <http://www.opengeospatial.org/standards/wps> (дата обращения 04.09.2014).

*Бычков И. В., Ружников Г. М., Фёдоров Р. К., Шумилов А. С. Компоненты среды wps-сервисов // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2014. Т. 12, вып. 3. С. 16–24.*

геомоделирование, методы статистики. Достоинства данного стандарта – простота, возможность предоставления метаданных, поддержка длительного выполнения сервисов и т. д.

Развитие современных информационных технологий, в том числе стандартов, позволяет организовать интеграцию специалистов различных предметных областей на новом уровне – взаимодействие с помощью сервисов обработки данных. В статье рассматривается подход, позволяющий объединить распределенные данные и распределенные WPS-сервисы в единые комплексы для решения сложных задач.

### Существующее состояние в области исследования

Отметим несколько успешно развивающихся проектов, реализованных на базе этого стандарта.

- Deegree WPS<sup>3</sup>. Проект позволяет публикацию алгоритма на Java по определенным стандартам с использованием deegree классов и модулей, созданных разработчиками.
- ZOO Project WPS<sup>4</sup>. Проект позволяет пользователям публиковать алгоритмы, написанные на разных языках (Java, C/C++, Python и Perl), и связывать их в цепочки анализа данных с использованием JavaScript.
- GeoServer WPS<sup>5</sup>. В данном проекте не предусмотрена возможность публикации собственного процесса в виде исходного кода, но проект GeoServer объединяет в себе реализацию нескольких стандартов OGC, включая WMS, WFS и WCS, и ориентирован на публикацию данных.
- pyWPS<sup>6</sup>. Проект поддерживает публикацию алгоритмов на языке Python.
- 52° North WPS<sup>7</sup>. В проекте поддерживаются алгоритмы из геоинформационных систем: GRASS GIS, Sextante, ArcGIS.

На данный момент существует несколько методов решения сложных геоинформационных задач путем комбинации различных WPS-сервисов. Xiaoliang Meng, Yichun Xie, Fuling Bian в статье [1] приводят три способа решения сложных задач. Первый способ – использование BPEL-сценариев для построения цепочек сервисов. BPEL<sup>8</sup> – стандарт, основанный на XML, который позволяет описывать и выполнять задачи в распределенном окружении. Данный способ не определяет формат передачи бинарных данных между сервисами, а также сложен в освоении в силу своей природы (требуется достаточно свободное владение языком разметки XML). Второй способ – это программная реализация комбинации сервисов в виде приложения, имеющего интерфейс, соответствующий стандарту WPS, таким образом, данный сервис можно использовать в дальнейшем вместе с другими WPS-сервисами. Третий способ – цепочка выполнения сервисов в виде последовательности GET-запросов и обработки XML результатов работы методов. Из всех представленных в статье методов наибольший интерес представляет способ использования BPEL и комбинации нескольких сервисов в новый сервис. Предложенный в настоящей статье способ берет преимущества у обоих вышеназванных способов – полнота возможностей написания сценариев (различные управляющие конструкции, распространенность JavaScript) наравне с BPEL, а также возможность публикации сценариев в виде отдельных WPS-сервисов наравне со вторым способом.

Отдельно следует упомянуть об обязательности замыкания среды WPS-сервисов, т. е. необходимо обеспечивать возможность использования результатов работы одного сервиса в качестве входных результатов для работы другого сервиса. Пример создания замкнутой среды подробно описан в статье «Towards The Transactional Web Processing Service» [2] под авторством Bastian Schaeffer, где для написания сервисов предлагается использовать стандарт BPEL, а входные данные описывать с помощью стандарта XML Schema. Автор предла-

<sup>3</sup> Deegree open source software for spatial data infrastructures and the geospatial web. URL: <http://www.deegree.org> (дата обращения 04.09.2014).

<sup>4</sup> ZOO Project Documentation / Ed. by Nicolas Bozon, ZOO Community. URL: <http://zoo-project.org/docs/ZOOProject.pdf> (дата обращения 04.09.2014).

<sup>5</sup> Open source software server // GeoServer. URL: <http://geoserver.org> (дата обращения 04.09.2014).

<sup>6</sup> Python Web Processing Service. URL: <http://pywps.wald.intevation.org> (дата обращения 04.09.2014).

<sup>7</sup> Geospatial Open Source Software GmbH. URL: <http://52north.org> (дата обращения 04.09.2014).

<sup>8</sup> URL: <http://searchsoa.techtarget.com/definition/BPEL> (дата обращения 04.09.2014).

гает определять любой BPEL-сценарий как отдельный WPS-сервис, в то же время определяя замкнутость данных внутри системы. В качестве примера приводится сервис, позволяющий рассчитывать загрязнение воздуха, реализованный на платформе 52° North WPS. Данная статья, кроме определения замкнутости, предлагает мощный инструмент для валидации входных данных – стандарт XML Schema.

Говоря о распараллеливании обработки данных на основе WPS-сервисов, следует отметить статью Bastian Baranski [3], в которой рассматривается проблема гридификации – построения распределенной системы выполнения определенного WPS-сервиса. Автор приводит два способа построения такой системы: написание отдельных скриптов распараллеливания WPS-сервисов и непосредственную интеграцию WPS-сервисов в специальное программное обеспечение. Распараллеливание потребуется по мере работы над средой распределенных вычислений на основе WPS-сервисов, и предложенные Bastian Baranski способы будут, несомненно, рассмотрены в применении к среде.

Недостатки стандарта WPS при реализации в существующих проектах:

- требования к параметрам WPS-сервисов слишком общие, необходима более детальная спецификация. В частности, нельзя установить требования к входным данным в виде таблиц определенной структуры. Например, для сервиса расчета загрязнений от точечных источников необходима таблица с определенной структурой и данными, в которой имеются координаты источников, количество выделяемых вредных веществ на единицу времени. При этом набор веществ и единицы измерения должны быть строго фиксированы;

- данные для WPS-сервисов предоставляются в виде файлов, доступных через HTTP или с помощью сервисов на базе стандартов WMS, WMF, WMC. Часто доступ предоставляется только на чтение. Применение WPS-сервисов не обладает свойством замкнутости, т. е. к результатам выполнения WPS-сервиса нельзя сразу применить другие сервисы. Пользователю необходимо результаты предыдущих вычислений опубликовать с помощью перечисленных способов доступа. Автоматическая публикация отсутствует. Соответственно обработка данных на основе нескольких WPS-сервисов довольно трудоемкий процесс;

- использование пользователем данных в виде файлов в обработке WPS-сервисами представляет определенную сложность. При обращении к WPS-сервису передаются URL-ссылки на файлы, которые WPS-сервис загружает через протокол HTTP. Доступ к файлам в соответствии со стандартом не регламентируется и, следовательно, должен быть открытым. Кроме того, необходима организация доступа к файлам пользователя через HTTP.

Результаты исследования текущих теоретических разработок и программных реализаций различных средств, а также возможности и недостатки стандарта WPS и распределенного выполнения сервисов в целом обуславливают объективную необходимость в создании среды распределенных вычислений на основе WPS-сервисов.

### **Организация среды WPS-сервисов**

Отмеченные недостатки обосновывают актуальность создания распределенной вычислительной среды, которая могла бы обеспечить интеграцию распределенных WPS-сервисов, ориентированных на решение широкого класса задачи. Среда распределенных вычислений должна обеспечивать как хранение пользовательских данных, так и выполнение WPS-сервисов над этими данными. Для организации WPS-сервисов требуется обеспечить их инвентаризацию, запуск, обмен данными, контроль выполнения WPS-сервисов. В Институте динамики систем и теории управления СО РАН разрабатываются основные компоненты среды.

### **Хранение данных среды**

Одна из важных функций среды – это обеспечение передачи и получения данных WPS-сервисами через Интернет. Следовательно, необходимо использование систем хранения данных, обеспечивающих регламентированный доступ через Интернет не только на чтение, но и на запись. В качестве входных данных чаще всего используют файлы и реляционные таблицы. Для WPS-сервисов применяются и могут потребоваться следующие протоколы

и стандарты доступа к данным: HTTP, WFS, WFS-T, WCS, WMS. В связи с большим объемом данных использование стандартов WFS и WFS-T неэффективно из-за текстового представления данных. Стандарты WCS, WMS не позволяют производить изменение данных. Поэтому необходимо в первую очередь поддержать протокол HTTP и стандарт Simple Features Specification For SQL. Также отметим, что существующие в Интернете файловые системы хранения данных позволяют чтение данных, но для их записи требуется авторизация, которая WPS-стандартом не предусмотрена. Поэтому требуются компоненты для обеспечения регламентированного доступа к данным в соответствии с WPS. В рамках среды разработаны:

1) файловая система хранения данных (СХД), в которой пользователю предоставляется директория, и доступ к файлам регламентируется в соответствии с запуском WPS-сервисов;

2) СУБД PostgreSQL (с модулем расширения PostGIS) для хранения реляционных данных, в котором для каждого пользователя создается схема данных.

Для работы в браузере с этими компонентами предоставляется файловый менеджер и подсистема редактирования и отображения реляционных данных.

### Каталог WPS-сервисов

Для организации выполнения WPS-сервисов необходимо учитывать их распределение в сети Интернет. Соответственно необходимо обеспечить инвентаризацию и поиск WPS-сервисов. Кроме того, требуется более детальная спецификация входных и выходных параметров, чем это определено стандартом WPS. Детальная спецификация позволит упростить пользовательский интерфейс и провести верификацию входных данных еще до запуска WPS-сервиса. Инвентаризацию и поиск WPS-сервисов реализует разработанный в рамках проекта каталог WPS-сервисов. Каталог WPS-сервисов разработан в виде модуля системы управления контентом Calipso<sup>9</sup>. Регистрация WPS-сервиса в каталоге происходит в несколько этапов. На первом этапе пользователь вводит название сервиса, его описание, данные для обращения к WPS-службе. На втором этапе каталог запрашивает, в соответствии со стандартом WPS, по введенному адресу метаданные и отображает список имеющихся WPS-сервисов. После выбора нужного сервиса выполняется запрос на получение метаданных о параметрах WPS-сервиса. На последнем этапе пользователь дополняет информацию о параметрах: используемый для ввода элемент управления и его свойства, пользовательское название параметра, поясняющий текст. Данная информация применяется для генерации пользовательского интерфейса, верификации параметров и запуска WPS-сервиса. Поиск в каталоге может производиться по названию и описанию работы WPS-сервиса. Планируется добавить поиск по категории WPS-сервиса и описанию входных и выходных данных.

При запуске WPS-сервисов необходимо обеспечить дружелюбный пользовательский интерфейс, который, в частности, позволяет упростить указание данных из СХД и СУБД в качестве значений входных и выходных параметров. Поэтому при выборе WPS-сервиса из каталога и генерации пользовательского интерфейса (формы) для ввода параметров (рис. 1) используются элементы управления, указанные при регистрации WPS-сервиса. Элементы управления применяются для упрощения ввода и формирования корректных значений параметров. Например, ввод значения па-

Execute method (hide)

Source: testinter

Result: /check/test.tif

AttrName: val (val)

EXTENT (Use Shift button to select a rect on the map.):

099°30'25" 051°59'11" 096°31'07" 049°59'43"

save open

CELLSIZE: 0.001

Execute

13:9:17.196 - Executing wps method interpolation with following params.  
gls\_admin testinter /check/test.tif, val, MULTIPOLYGON(((99.50694444444444, 49.99527777777778, 99.50694444444444, 51.98638888888889, 96.51861111111111, 51.98638888888889, 96.51861111111111, 49.99527777777778, 99.50694444444444, 49.99527777777778))), 0.001  
13:9:17.201 - WPS interpolation starts its execution

Рис. 1. Сгенерированная форма для сервиса интерполяции

<sup>9</sup> <http://calip.so/>.

раметра «Source» проводится элементом управления file, который отображает файловую систему СХД, доступную пользователю.

Разработаны следующие элементы управления: edit – для ввода строковых значений; number – для ввода числовых значений; checkbox – для ввода булевых значений; rectangle – для указания экстента (прямоугольной области на карте) и т. д. Разработаны элементы управления, которые работают с данными пользователя в СХД и СУБД: file – для выбора файла из СХД; file\_save – для сохранения файла в СХД; select\_table – для выбора таблицы из СУБД; select\_table\_attr – для выбора атрибута таблицы и т. д. При наличии на форме элементов управления file, file\_save, rectangle создается карта, на которой отображаются данные, связанные с этими элементами. Набор элементов управления является расширяемым.

### **Подсистема управления WPS-сервисами**

После ввода пользователем в форме Web-браузера данных среда должна обеспечить запуск WPS-сервисов и передачу, получение этих данных в соответствии со стандартом WPS и политиками регламентации доступа. В форме WPS-сервиса формируется список значений параметров и передается запрос подсистеме управления WPS-сервисами на выполнение. Подсистема управления сервисами производит обработку параметров для их передачи в соответствии со стандартом WPS, регистрирует экземпляр выполняемого WPS-сервиса, запускает WPS-сервис, получает и сохраняет результаты работы. В зависимости от типа параметров значения обрабатываются следующим образом.

1. Файл. Если передаваемый параметр является файлом, расположенным в директории пользователя в СХД, то необходима передача данного файла WPS-сервису. В соответствии со стандартом WPS-сервису передается URL-адрес файла, используя который, WPS-сервис должен по протоколу HTTP скачать файл. Файл должен находиться в открытом доступе. Так как данные пользователей могут содержать информацию ограниченного доступа, то любой файл, хранимый в СХД, по умолчанию не может быть свободно доступным. Таким образом, необходимо одновременно защитить и предоставить WPS-сервисам доступ к хранимым в СХД файлам. Механизм доступа и контроля обращений к файлам построен следующим образом: каждому экземпляру выполнения WPS-сервиса присваивается уникальный идентификатор. Если выполняемый WPS-сервис должен получить в качестве параметров файлы, то для каждого из файлов создается уникальная ссылка, привязываемая именно к выполняемому экземпляру. Таким образом, WPS-сервис может свободно скачать требуемый файл из СХД по сгенерированной ссылке определенное число раз, причем все ссылки, созданные для конкретного экземпляра, уничтожаются, как только метод завершит свою работу. Если файл является результатом работы WPS-сервиса, то подсистема выполнения сценариев загружает в СХД в соответствующую пользовательскую директорию по протоколу HTTP. URL-адреса файлов берутся из метаинформации в ExecuteResponse-документе, который формируется по окончании работы WPS-сервиса.

2. Экстент (прямоугольная область на карте). На клиентской части (браузере) получение экстента производится в формате WKT [1]. Для передачи WPS-сервису такого типа данных осуществляется преобразование в соответствии со стандартом WPS.

3. Таблица базы данных PostgreSQL. При передаче ссылки на какую-либо таблицу в БД формируется строка соединения драйвера GDAL для непосредственного соединения WPS-сервиса с базой данных. Данные для соединения напрямую недоступны клиенту из соображений безопасности. В текущей версии таблицы PostgreSQL доступны только для локальных WPS-сервисов, но в дальнейшем предусмотрена реализация для регламентированного доступа к базе данных извне.

### **Выполнение WPS-сервисов**

Существующие подходы к выполнению WPS-сервисов предлагают только одиночное или линейную последовательность выполнения WPS-сервисов. Решение сложной задачи часто требует нетривиального применения распределенных WPS-сервисов, которая может включать обработку параметров, итерационное применение WPS-сервисов, ветвление в зависимо-

сти от промежуточных результатов и т. д. Поэтому необходим механизм, позволяющий реализовать сложную логику применения WPS-сервисов. Наиболее гибким подходом является использование некоторого языка программирования, на котором можно задать сценарий применения WPS-сервисов. Выбран язык JavaScript в силу своей широкой распространенности. Поскольку выполнение WPS-сервисов может быть длительным, асинхронное выполнение и сохранение контекста выполнения JavaScript позволит выполнять WPS-сервисы параллельно.

Для реализации сложной логики применения WPS-сервисов разработан интерпретатор WPS-сценариев. При выполнении WPS-сервиса или сценария формируется код JavaScript программы, который передается интерпретатору. В сценарии обращение к WPS-сервисам производится с помощью специальных функций Javascript-оберток. Javascript-обертки создаются автоматически для всех зарегистрированных в каталоге WPS-сервисов. В каталоге создан компонент разработки сценариев в Web-браузере, в которой пользователь может определить название, все его параметры, используемые элементы управления и программный код. При сохранении сценария он регистрируется в каталоге и доступен пользователю как WPS-сервис. Для пользователя в компоненте разработки сценариев отображается список доступных в системе Javascript-оберток WPS-сервисов и сценариев.

### **Интерпретатор сценариев WPS-сервисов**

Использование отдельного интерпретатора на стороне сервера в отличие, например, от выполнения кода JavaScript программы в браузере пользователя позволяет обходить ограничение на HTTP-запросы к другим хостам в браузерах, а также делает выполнение сценария независимым от таких факторов, как, например, непреднамеренное закрытие или нехватка ресурсов браузера пользователя. Подсистема управления поддерживает длительное выполнение сервисов (например, несколько суток) в соответствии со стандартом WPS. При выполнении таких сервисов происходит периодический опрос WPS-службы на предмет степени завершения работы WPS-сервиса в процентном соотношении, что положительно влияет на информативность и уровень контроля за выполнением сервисов. В случае непосредственного встраивания JavaScript-интерпретатора в WPS-службу (как это реализовано в ZOO Project) возникают следующие проблемы: набор WPS-сервисов ограничен только теми, что работают в рамках локальной WPS-службы, выполнение сценариев ограничивается предоставляемыми мощностями (сценарии выполняются в рамках CGI-скрипта веб-сервера) и напрямую зависит от стабильности и времени допустимого таймаута HTTP-запросов браузера клиента (возникновение проблем при запуске длительных процессов). Для непосредственной интерпретации сценариев и обращения к WPS-сервисам разработан специальный модуль, написанный на C++ с использованием JavaScript-интерпретатора Google V8. Применение интерпретатора Google V8 позволяет интегрировать распределенные WPS-сервисы в единую программу на JavaScript. Google V8 компилирует JavaScript в машинный код и выполняет его, в результате чего достигается высокая производительность. При разработке интерпретатора делался акцент на безопасность и стабильность системы. Программный интерфейс интерпретатора JavaScript ограничен и полностью контролируется политиками безопасности геопортала. Интерпретатор JavaScript, по сути, является отдельной виртуальной машиной, доступ из которой к внешним ресурсам регламентируется. Реализован контроль доступа к WPS-сервисам и промежуточным данным.

Рассмотрим расширение интерпретатора – предоставляемый программный интерфейс. Реализованы следующие функции.

1. Функция CallWPS – принимает на вход параметры WPS-службы, метаописание WPS-сервиса и список входных параметров в виде JavaScript-объектов. Формирует строку запроса на выполнение WPS-сервиса, затем выполняет его. Поддерживает долгосрочное выполнение WPS-сервисов. Операции с запросами выполняет с помощью библиотеки с URL.

2. Библиотека методов для работы с матрицами – предоставляет набор JavaScript-функций, позволяющих создавать матрицы в виде JavaScript-объектов, в реальности представляющих собой структуры, создаваемые с помощью специализированной библиотеки C++.

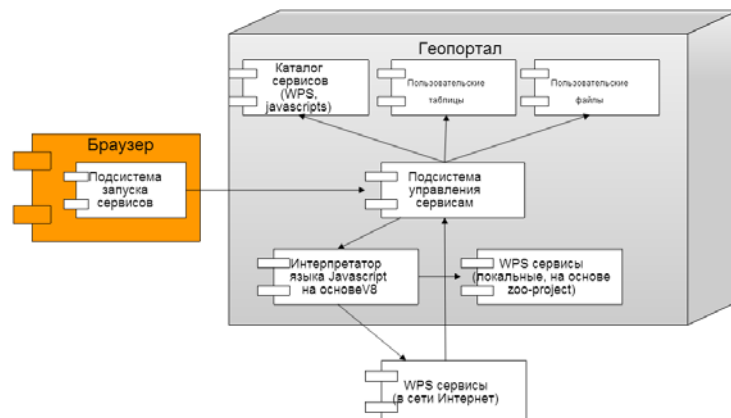


Рис. 2. Архитектура подсистемы запуска и учета состояния выполнения WPS-сервисов

Таким образом, достигается сочетание высокой производительности библиотек на языке C++ и простоты написания кода для JavaScript.

Для каждого зарегистрированного сервиса на языке Javascript генерируется функция-обертка, которая отвечает за подготовку параметров и запуск сервиса из интерпретатора. Функции-обертки необходимы для разработки сценариев использования нескольких WPS-сервисов. В функциях обертки обычно вызывается описанный метод CallWPS с соответствующими параметрами.

Для каждого экземпляра выполнения WPS-сценария запускается отдельный серверный процесс, состояние выполнения которого, а также выходной лог (консоль выполнения) сохраняется в базе данных по мере обновления, что делает разработку, тестирование и отладку проще и эффективнее.

### Общая архитектура среды

Общая архитектура распределенной среды (рис. 2) в рамках геопортала состоит из следующих подсистем:

- каталог WPS-сервисов, предназначенный для регистрации, поиска и запуска WPS-сервисов;
- подсистема управления WPS-сервисами, осуществляющая общее управление процессом выполнения WPS-сервисов;
- интерпретатор WPS-сценариев;
- пакеты WPS-сервисов (локальных и распределенных).

### Апробация

Рассмотрим сценарий `road_pnt_pol`, зарегистрированный в каталоге. Данный сценарий предназначен для подсчета выделяемых вредных веществ от точечных источников и автотранспорта в ячейках регулярной сетки. Представленный сценарий использует распределенные по разным серверам WPS-сервисы: `vector2grid`, `road2grid`, `g_sum`.

```
function road_pnt_pol(housefile, roadsources, commonresult, sumpol){
    var houseresult='/tmp/hr.tif';
    var roadResult ='/tmp/hr.tif';
    vector2grid(housefile, houseresult);
    road2grid(roadsources, roadResult, sumpol);
    g_sum(roadResult, houseresult, commonresult);
    return true;
}
```

WPS-сервис `vector2grid` производит подсчет загрязнений от точечных источников. На входе WPS-сервиса точечный векторный файл, описывающий положение источников и количество выделяемых веществ. На выходе WPS-сервис создает файл в формате GeoTIFF, содержащий суммарные выбросы по ячейкам. WPS-сервис `road2grid` производит подсчет загрязнений от автотранспорта. На входе WPS-сервиса векторный файл, описывающий расположение автодорог и их загруженность, и суммарное количество выделяемых веществ населенного пункта. WPS-сервис производит распределение суммарных загрязнений с учетом расположения автодорог и их загруженности. На выходе WPS-сервис также создает файл в формате GeoTIFF, содержащий суммарные выбросы по ячейкам. WPS-сервис `g_sum` выполняет сложение выбросов от точечных объектов и автотранспорта.

В этом примере компоненты среды обеспечивают:

- выполнение сценария;
- передачу удаленным WPS-сервисам файловых данных, определенных в переменных `housefile`, `roadsource`, из СХД;
- получение результатов работы WPS-сервисов и помещение их в СХД по путям, определенным в переменных `houserresult` `roadResult` `commonresult`;
- контроль выполнения, вывод сообщений WPS-сервисов.

Автоматизация применения WPS-сервисов на основе JavaScript позволила значительно сократить время расчета множества вариантов улучшения воздуха населенного пункта, уменьшить количество пользовательских ошибок.

## Заключение

Комбинация WPS-сервисов и подсистем геопортала СХД, СУБД PostgreSQL позволяет упростить обработку пространственных данных в Интернете. Для передачи данных WPS-сервисам достаточно разместить их в геопортале с помощью подсистем ввода / редактирования данных, файлового менеджера и FTPS-сервера. Результаты работы WPS-сервисов также размещаются в геопортале и доступны для скачивания.

Разработчикам WPS-сервисов расширяемый набор элементов управления позволяет установить дополнительные требования относительно стандарта WPS к входным данным, которые проверяются на этапе ввода входных данных пользователем. WPS-сервисы могут поставляться с набором элементов управления для специфичных для WPS-сервиса параметров. Кроме того, генерируется удобный пользовательский интерфейс вызова WPS-сервисов.

Архитектура среды позволяет расширять её функциональность за счет WPS-сервисов. В дальнейшем каталог WPS-сервисов может выступать как общедоступный банк WPS-сервисов, решающих задачи разных предметных областей. Возможность интеграции сервисов посредством специальных Javascript-функций дает возможность разработки новых методов поддержки междисциплинарных научных исследований.

## Список литературы

1. *Xiaoliang Meng, Yichun Xie. Fuling Bian Distributed Geospatial Analysis through Web Processing Service: A Case Study of Earthquake Disaster Assessment // Journal Of Software. 2010. Vol. 5. No. 6.*
2. *Schaeffer B. Towards The Transactional Web Processing Service. URL: <http://www.gi-tage.de/archive/2008/downloads/acceptedPapers/Papers/Schaeffer.pdf> (дата обращения 04.09.2014).*
3. *Baranski B. Grid Computing Enabled Web Processing Service: Grid Computing Enabled Web Processing Service. URL: [http://www.gi-tage.de/archive/2008/downloads/Material/Presentation-Grid\\_Computing\\_Enabled\\_Web\\_Processing\\_Service-Baranski.pdf](http://www.gi-tage.de/archive/2008/downloads/Material/Presentation-Grid_Computing_Enabled_Web_Processing_Service-Baranski.pdf) (дата обращения 04.09.2014).*



**I. V. Bychkov, G. M. Rugnikov, R. K. Fedorov, A. S. Shumilov**

*Institute of System Dynamics and Control Theory of SB RAS  
134 Lermontov Str., Irkutsk, 664033, Russian Federation*

*bychkov@icc.ru, rugnikov@icc.ru, fedorov@icc.ru, shumsan1011@gmail.com*

## **COMPONENTS OF WPS ENVIRONMENT FOR GEOPROCESSING**

The article describes the development of distributed computing environment based on standard Web Processing Service, which unifies the use of Web-services for spatial processing raster and vector data.

*Keywords:* OGC, WPS, JavaScript, V8, Calipso.

### **References**

1. Xiaoliang Meng, Yichun Xie. Fuling Bian Distributed Geospatial Analysis through Web Processing Service: A Case Study of Earthquake Disaster Assessment. *Journal of Software*, 2010, vol. 5, no. 6.
2. Schaeffer B. *Towards The Transactional Web Processing Service*. URL: <http://www.gi-tage.de/archive/2008/downloads/acceptedPapers/Papers/Schaeffer.pdf>
3. Baranski B. *Grid Computing Enabled Web Processing Service: Grid Computing Enabled Web Processing Service*. URL: [http://www.gi-tage.de/archive/2008/downloads/Material/Presentation-Grid\\_Computing\\_Enabled\\_Web\\_Processing\\_Service-Baranski.pdf](http://www.gi-tage.de/archive/2008/downloads/Material/Presentation-Grid_Computing_Enabled_Web_Processing_Service-Baranski.pdf)