**Внедрение вычислений на стороне клиента при реализации веб-сервисов обработки пространственных данных на основе стандарта OGC WPS.**

Паниди Е.А., Казаков Э.Э., Капралов Е.Г., Терехов А.В.

Санкт-Петербургский государственный университет

Email: evgeny.panidi@spbu.ru, e.kazakov@spbu.ru, eugeniy.kapralov@spbu.ru, a.terehov@spbu.ru

**Аннотация**

В статье рассмотрены вопросы применения вычислений на стороне клиента при реализации веб-сервисов обработки пространственных данных. Предложен подход, заключающийся в создании веб-сервисов, предусматривающих возможность запуска вычислений как на стороне сервера, так и на стороне клиента, по выбору пользователя. Предложено название для таких сервисов – гибридные веб-сервисы обработки пространственных данных. Описаны результаты разработки программного обеспечения в соответствии с предложенной концепцией.

Исследования выполнены при финансовой поддержке РФФИ в рамках научного проекта № 13-05-12079 офи\_м.

**Abstract**

The paper describes ways of application of the client-side computing when implementing Web services for spatial data processing. The approach described that consists in implementation of Web service capable to run computations on server side or client side by user choice. A title suggested for such service type, which is Hybrid Geoprocessing Web Service. Described the results of the software development in accordance with the proposed concept.

The study was partially supported by Russian Foundation for Basic Research (RFBR), research project No. 13-05-12079 ofi\_m.

**Ключевые слова:** обработка пространственных данных, гибридные веб-сервисы обработки пространственных данных, WPS, обработка данных на стороне клиента

**ВВЕДЕНИЕ**

Всего несколько лет назад основными направлениями применения веб средств при работе с пространственными данными были доставка данных конечному пользователю и графическая визуализация данных в интерфейсе веб-браузера (в отдельных случаях в интерфейсе настольного приложения) и интерфейсе настольного геоинформационного программного обеспечения. На современном этапе развития данной мультидисциплинарной области знания на первый план выходят новые тренды, которые берут своё начало, отчасти, в геоинформационной науке и технологии, отчасти в веб-технологиях.

Наиболее значимым является разворот от пассивных операций с пространными данными (хранение, передача и визуализация) к активным операциям (извлечение производных данных, обработка и анализ данных). Истоком этого разворота является в том числе смена парадигмы в области веб-технологий, а именно появление концепции Веб 2.0 [13], которая подразумевает появление интерактивного взаимодействия пользователя с веб-средой и переход от создания статических сайтов к созданию веб-приложений.

Следует заметить, также, что современные технологии работы с пространственными данными в веб-среде находятся в рамках парадигмы облачных вычислений. Современные стандарты публикации веб-сервисов для обработки и анализа пространственных данных ориентированы на сервер-стороннюю обработку данных, и представление пользователю только конечного результата [5, 8, 9, 12, 16, 17]. Такие веб-сервисы подразумевают использование облачных решений или вычислительных кластеров, в которых обработка данных, с точки зрения управления процессом, производится централизованно на стороне сервера.

В то же время, при исполнении небольших и среднеразмерных производственных проектов и проектов научных исследований использование существующих технологий разработки веб-геосервисов становится малоприменимым. Использование сервисов, ориентированных на обработку данных на стороне сервера, не только ведёт к необходимости создавать некоторую серверную вычислительную инфраструктуру, но и влечёт временные затраты на передачу данных на сервер для обработки.

Возможным решением проблемы является разработка технологии и средств, которые позволили бы дополнить существующие подходы к обработке пространственных данных с использованием веб-сервисов, используя обработку на стороне клиента. При этом, использование ресурсов клиента при создании и внедрении веб-сервисов для обработки и анализа пространственных данных большинством исследователей и разработчиков не рассматривается.

Вместе с тем, наряду с общепринятым конструированием облачных систем, всё ярче начинают проявляться тенденции, связанные с созданием децентрализованных систем. Подобные системы имеют достаточно давнюю историю, в том числе связанную с применением систем распределённых вычислений (англ. Grid Computing), которое практически не нашло отражения в геоинформационных технологиях, за исключением единичных проектов [4, 7]. Однако рассмотрение такой современной концепции, как концепция туманных вычислений (англ. Fog Computing) [3] в совокупности с концепцией семантических веб-сервисов (англ. Semantic Web Services) [6], позволяет сделать предположение, о том, что в ближайшем будущем вопрос применения клиент-сторонних вычислений в веб-сервисах обработки пространственных данных будет актуален.

Всё более широкое развитие упомянутых концепций позволяет сделать выводы о том, что с точки зрения проектирования и внедрения веб-сервисов обработки пространственных данных:

* не следует пренебрегать вычислительными ресурсами, имеющимися на стороне клиента, которые оказываются достаточными при решении многих вычислительных задач и задач обработки данных;
* при внедрении веб-сервисов, реализующих обработку данных (в том числе пространственных данных) на стороне клиента, необходимо обеспечить автоматизацию поиска и использования таких сервисов, а значит стандартизацию средств запуска обработки на стороне клиента, их описания и порядка их вызова.

Использование веб-сервисов для обработки пространственных данных на стороне клиента подразумевает передачу на клиент исполняемых файлов, содержащих программный код некоторых инструментов для обработки. Идея передачи исполняемого кода вполне очевидна и в целом не нова. Применительно к средствам обработки пространственных данных она упоминалась и другими авторами [11]. Однако широко известной и доведённой до состояния стандарта реализации подобного решения авторам данной статьи не известно. В рамках освещённого в статье исследования авторы изучает пути и возможности унификации и стандартизации методов и средств передачи исполняемого кода веб-сервисов обработки пространственных данных (геообработки) [1, 2, 10, 14, 15]. Разрабатываемые при проведении исследования реализации таких сервисов, подразумевают возможность выполнения обработки на обеих сторонах, серверной и клиентской. Такие сервисы авторы называют гибридными веб-сервисами геообработки (сокращённо, гибридными сервисами).

В исследовании и разработке предполагались:

* минимизация требований (программных и аппаратных) к серверной части системы путём передачи пользователю только компактных исполняемых компонент приложения при обработке данных;
* обеспечение совместимости с существующими стандартами, и прежде всего, обеспечение возможности создания сервисов обработки обратно совместимых со стандартом OGC WPS [12, 16], то есть позволяющих выполнять обработку по выбору на клиентской или серверной стороне, учитывая доступные ресурсы и контекст проекта;
* предоставление пользователю возможности более гибкого управления хранящимися у него пространственными данными при обработке с использованием веб-сервисов, путём обеспечения возможности выбора серверной или клиентской стороны при запуске обработки.

**СРЕДСТВА РАЗРАБОТКИ**

Для апробации выбранного подхода в исследовании использована связка из веб-сервера Apache HTTP Server (*http://httpd.apache.org/*) и WPS-сервера PyWPS (*http://pywps.wald.intevation.org/*). Выбор веб-сервера обусловлен лишь наличием опыта администрирования Apache и не подразумевал каких-либо специальных требований к производительности. Отсутствие этих требований вызвано тем, что на текущий стадии проект реализуется как частное исследование и прототип, а не как промышленное решение. Выбор PyWPS обусловлен тем, что на момент запуска проекта данный WPS-сервер являлся единственным самостоятельным, не интегрированным в иное, более комплексное серверное программное обеспечение, решением.

В качестве основного средства разработки программного кода в рамках проекта использован язык программирования Python (*https://www.python.org/*).

При реализации проекта особое внимание уделяется исключению жёсткой привязки к средствам разработки, в целях выработки независимой архитектуры, пригодной для выработки абстрактной спецификации.

**ГИБРИДНЫЕ ВЕБ-ГЕОСЕРВИСЫ**

С точки зрения архитектуры, реализованный прототип системы публикации и запуска гибридных сервисов включает в себя два базовых блока, серверный, который представляет собой набор серверного программного обеспечения, реализующего доступ к программным средствам обработки пространственных данных и сервер-стороннюю обработку данных, и клиентский, который представляет собой комплекс средств, обеспечивающих запуск необходимых процессов обработки с клиентского компьютера и при необходимости загрузку на клиент необходимых программных компонент.

Принципиальная схема организации (Рис. 1) системы включает две взаимосвязанных структуры, в соответствии с концепцией гибридных веб-сервисов реализующих сервер-сторонний и клиент-сторонний механизмы работы. Серверная сторона включает собственно веб-сервер, принимающий пользовательские запросы и управляющий потоками данных, WPS-сервер, сервер гибридных сервисов, позволяющий обрабатывать запросы на передачу на клиент программных компонент процессов обработки пространственных данных, базы данных, предназначенные для хранения и программных компонент процессов обработки, предполагающихся к передаче на клиент, а также пользовательских данные.



Рисунок 1. Система публикации и запуска гибридных сервисов обработки пространственных данных.

Пользователь может работать с сервером гибридных сервисов в двух режимах: через веб-интерфейс, осуществляя загрузку данных на сервер и удаленную их обработку, получая ссылки на результат, либо через специализированное настольное приложение – клиентскую среду исполнения гибридных сервисов, получая на локальное устройство необходимые программные компоненты и там же запуская их.

**СХЕМА ЗАПРОСОВ К СЕРВЕРУ**

Расширение стандарта WPS для перехода к гибридному типу сервиса, в первую очередь, подразумевает добавление нового типа запросов, который будет использоваться для получения на клиент программных компонент, необходимых для локальной реализации выбранного процесса. В разрабатываемом прототипе такой запрос получил имя getProcess. Пример записи такого запроса при использовании метода GET:

*http://some.host/server?*

*Request=getProcess&*

*Service=HGWS&*

*Version=1.0.0&*

*Platform=Win32&*

*Identifier=processName*

Запрос предусматривает указание ряда параметров, перечисленных в таблице 1.

Таблица 1. Параметры запроса getProcess.

|  |  |  |
| --- | --- | --- |
| Название | Присутствие | Описание |
| Request | Обязательный | Идентифицирует тип запроса, всегда «getProcess» |
| Service | Обязательный | Идентифицирует тип сервиса, всегда «HGWS» |
| Version | Обязательный | Идентифицирует версию |
| Identifier | Необязательный | Указывает конкретный сервис |
| Platform | Обязательный | Идентифицирует операционную систему пользователя, например «Win32», «Win64», «Mac», «Linux Debian» и т.д. |

Этот тип запросов занимает своё место в многоступенчатом процессе взаимодействия среды исполнения и сервера гибридных сервисов (Рис. 2). Все запросы, за исключением запроса getProcess, при этом, выполняются в соответствии со стандартом WPS. Пользователь выбирает сервер и подключается к нему, настольное приложение через отправляет на выбранный пользователем сервер стандартный WPS-запрос getCapabilities (запрос перечны доступных WPS-сервисов), ответ которого расширен атрибутом «desktop-available». Разбирая ответ, приложение получает список процессов, доступных для настольного исполнения.



Рисунок 2. Цепочка запросов к серверу гибридных сервисов.

Далее пользователь выбирает тот процесс из отобранных, который ему интересен и запрашивает его. Выполняется стандартный WPS-запрос describeProcess (запрос метаданных выбранного процесса), ответ которого стандартен. На основании ответа генерируются графический интерфейс будущего запускаемого модуля, а также стандартная точка входа программного интерфейса, подключающая созданный графический интерфейс и вызывающая стандартно именованную функцию, ожидая, что среди загружаемых далее программных компонент будет стандартно именованный (также, как сам процесс) модуль с такой входной функцией.

На третьем шаге используется запрос getProcess. После его выполнения на клиент загружаются те программные компоненты выбранного процесса, которые не были загружены ранее. Они помещаются в стандартные директории и оказываются готовым к работе.

Результатом выполнения запроса getProcess является, по аналогии со стандартными WPS-запросами, XML-документ, содержащий все необходимые сведения о программных компонентах, загрузка которых требуется для запуска процесса геообработки на клиентском компьютере. Возвращаемый документ передает неограниченное число описаний программных компонент, содержащих ссылки для загрузки. Пример подобного XML-документа:

*<!--Module parameters-->*

*<complexType name="FModule">*

 *<complexContent>*

 *<name>sampleFModule</name>*

 *<annotation>sampleFModule annotation</annotation>*

 *<compability>*

 *<!-- Description of platform compatibility for module-->*

 *Win32-common*

 *</compability>*

 *<type>*

 *<!--Type of software component-->*

 *DLL*

 *</type>*

 *<downloadLink>URL</downloadLink>*

 *<usedExtraApps>*

 *<complexContent>*

 *<name>SampleExtraApp</name>*

 *<annotation>sampleExtraApp annotation</annotation>*

 *<compability>Win32-common</compability>*

 *<type>Binary</type>*

 *<downloadLink>URL</downloadLink>*

 *</complexContent>*

 *</usedExtraApps>*

 *</complexContent>*

*</complexType>*

**КЛИЕНТСКАЯ СРЕДА ИСПОЛНЕНИЯ**

Для управления выполнением обработки пространственных данных на клиентском компьютере должна присутствовать среда исполнения. В простейшем случае роль среды исполнения может играть программный клиент – настольное приложение или веб-браузер, отображающие графический пользовательский интерфейс.

В рамках данного проекта, учитывая то, что исполняемый код может передаваться на клиент в виде набора исполняемых файлов и библиотек, среда исполнения в узком смысле включает в себя настольное приложение, которое помимо отображения пользовательского интерфейса реализует задачи загрузки, хранения и удаления на клиентском компьютере файлов, содержащих исполняемый код, а также проверку соответствия параметров клиентского компьютера требованиям, необходимым для запуска геообработки и контроль запуска и завершения геообработки. В широком смысле в состав среды исполнения дополнительно входят внешние программные средства, необходимые для запуска исполняемых файлов, в частности интерпретатор Python.

**ВЕБ-ИНТЕРФЕЙС**

Для публикации и тестирования гибридных сервисов был создан прототип геопортального интерфейса (*http://195.70.211.131/*). Реализованный интерфейс предельно прост, и включает в себя несколько веб-страниц, таких как стартовая страница, страница со справочной информацией, страница со списком публикаций, и две основных страницы, предназначенных для работы пользователя, страница персонального аккаунта пользователя и страница параметризации и запуска процессов обработки (Рис. 3).



Рисунок 3. Веб-страница параметризации и запуска процессов обработки (слева) и веб-страница персонального аккаунта пользователя (справа).

Данный ресурс позволяет обрабатывать запросы на скачивание программных компонент процессов обработки данных, в соответствии с порядком использования гибридных сервисов геообработки, и выполнять обработку данных на стороне сервера, в соответствии со стандартом WPS, ресурс также предусматривает возможность хранения некоторого объема пользовательских данных на сервере, устраняя тем самым необходимость многократно загружать один и тот же файл для выполнения над ним нескольких операций. Сохраняя результаты обработки на сервере и пользуясь, например, функцией предварительного просмотра растровых данных в окне браузера можно отбраковать неудовлетворительные результаты, не скачивая их на свой компьютер и сэкономив время и трафик. Для загрузки, просмотра хранящихся на сервере файлов и некоторых других операций над ними предназначен раздел «Личный кабинет», для пользования которым необходимо быть зарегистрированным в системе.

Работа раздела обработки данных устроена следующим образом:

* при загрузке пользователем соответствующей страницы, на сервер отправляется запрос GetCapabilities (WPS-запрос описания доступных на данном сервере WPS-процессов), ответом на который является список доступных для выполнения сервисов обработки данных, появляется список с названиями этих алгоритмов;
* пользователю становится доступен доступен поиск, по ключевым словам, в описаниях доступных сервисов и выбор одного из них в соответствующем выпадающем списке на веб-страницы;
* после выбора пользователем требуемого сервиса, на веб-странице автоматически конструируется форма выбора исходных данных, при этом, для разных алгоритмов количество исходных данных и их типы различны (например, числовые параметры или файлы пространственных данных, в этом случае пользователю будет доступен список имеющихся в его личном кабинете файлов требуемого формата);
* после заполнения всех требуемых полей и запуска вычислений предоставленные пользователем данные передаются на сервер; результат обработки (в случае успешного выполнения алгоритма), являющийся файлом пространственных данных, появляется в личном кабинете пользователя.

Для растровых данных доступен предварительный просмотр в окне браузера либо на интерактивной карте. Все файлы в личном кабинете доступны для скачивания.

**РЕЗУЛЬТАТЫ И ВЫВОДЫ**

Подводя итог, необходимо отметить, что в целом выбранный подход оказался работоспособным и позволил решить поставленную в рамках проекта задачу переноса вычислений на сторону клиента при использовании веб-сервисов обработки пространственных данных. Безусловно ещё существуют вопросы, требующие дальнейшего изучения и проработки, самыми актуальными из которых мы обозначаем вопросы обеспечения безопасности при запуске программного кода, полученного со стороннего сервера и вопросы создания цепочек обработки данных, как при выполнении обработки под контролем пользователя, так и при реализации интерфейса машина-машина.

Вместе с тем, сформулированный подход и разработанные прототипы программных средств уже сейчас могут быть использованы при выполнении локальных проектов используя традиционные преимущества веб-сервисов.

Выполненные авторами исследования вскрыли, также, ряд вопросов теоретического и практического характера, требующих дополнительного рассмотрения:

* необходимость обеспечения безопасного запуска исполняемого кода на стороне клиента, полученного со стороннего сервера, детальное обсуждение которого, однако, в абстрактной спецификации или стандарте, регламентирующем создание подобных сервисов вряд ли возможно; вопрос безопасного запуска исполняемого кода должен рассматриваться индивидуально и детально в контексте каждого конкретного проекта;
* вопрос оркестровки сервисов и построения комплексных систем обработки пространственных также является актуальной задачей, но не имеет универсального решения, по причине сложности и нелинейности многих процедур обработки данных; частично проблема оркестровки состоит в обеспечении согласованного описания и структурирования входных и выходных данных обработки, что может быть решено путём разработки стандарта, описывающего допустимые структуры данных и правила их описания;
* комплекс задач, связанных с обеспечением возможностей автоматического взаимодействия серверов данных, серверов обработки данных и серверов, которые являются брокерами сервисов и процессов обработки данных; реализация подобного автоматического взаимодействия между серверами может быть востребована в распределённых системах обработки пространственных данных, в том числе больших данных;
* концептуальный пробел в технологии внедрения стандарта WPS, который связан с выделением и распределением ресурсов на стороне сервера при выполнении обработки на стороне сервера; в данном проекте проблема распределения ограниченных ресурсов сервера между пользователями была решена с помощью механизма пользовательских аккаунтов, однако, такая система является крайне неудобной как с точки зрения администрирования так и с точки зрения автоматизации запуска обработки и получения её результатов; решением, вероятно, может стать динамическое перераспределение ресурсов и введение в качестве атрибута данных временного периода их существования на сервере;
* результаты данного проекта требуют детального обсуждения как пользователями, так и разработчиками.

**ЛИТЕРАТУРА**

1. Казаков Э.Э. Подходы к созданию Веб-сервисов обработки и анализа материалов дистанционного зондирования. // Дистанционное зондирование Земли из космоса: алгоритмы, технологии, данные. Материалы молодёжной школы-семинара (г. Барнаул, 2-6 октября 2013 г.). Барнаул, 2013. С. 82-86.
2. Паниди Е.А. Геосервисы для онлайн-обработки данных дистанционного зондирования. // Дистанционное зондирование Земли из космоса: алгоритмы, технологии, данные. Материалы молодёжной школы-семинара (г. Барнаул, 2-6 октября 2013 г.). Барнаул, 2013. С. 74-81
3. Abdelshkour M., 2015. IoT, from Cloud to Fog Computing. Available at: http://blogs.cisco.com/perspectives/iot-from-cloud-to-fog-computing
4. Aktas M.S., Aydin G., Fox G.C., Gadgil H., Pierce M. and Sayar A., 2005. Information Services for Grid/Web Service Oriented Architecture (SOA) Based Geospatial Applications. Community Grids Lab, Indiana University, Technical Report. Bloomington, Indiana, USA
5. Baumann P. (ed.), 2009. Web Coverage Processing Service (WCPS) Language Interface Standard. OGC 08-068r2. Version: 1.0.0, 2009-03-25
6. Berners-Lee T., Hendler J., Lassila O., 2001. The Semantic Web. Scientific American Magazine. May 2001
7. Buyya R., Broberg J., Goscinski A. (eds.), 2011. Cloud Computing: Principles and Paradigms. John Wiley & Sons Inc., 637 p., DOI: 10.1002/9780470940105.fmatter
8. Coene Y., Marchetti P.G., Smolders S., 2007. Architecture and Standards for a Distributed Digital Library of Geospatial Services. Third Italian Research Conference on Digital Library Systems (IRCDL 2007) Padova 29–30 January 2007 Proceedings pp. 52–60
9. Doyle A. (ed.), 2000. OpenGIS Web Map Server Interface Implementation Specification. OGC 00-028. Version 1.0.0, 2000-04-19
10. Evans J. (ed.), 2003. OGC Web Coverage Services (WCS). OGC 03-065r6. Version 1.0.0, 2003-08-27
11. Kazakov E., Terekhov A., Kapralov E., and Panidi E., 2015. WPS-based technology for client-side remote sensing data processing. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-7/W3, pp. 643-649, doi:10.5194/isprsarchives-XL-7-W3-643-2015
12. Keens S. (ed.), 2007. Discussions, findings, and use of WPS in OWS-4. OGC Discussion Paper. OGC 06-182r1. Version 0.9.1, 2007-05-10
13. Mueller M., Pross B. (eds.), 2015. OGC WPS 2.0 Interface Standard. OGC Implementation Standard. OGC 14-065. Version 2.0, 2015-03-05
14. O'Reilly T., 2005. What Is Web 2.0 Design Patterns and Business Models for the Next Generation of Software. Available at: http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html
15. Panidi E.A., 2014. Towards Client-Side Web Processing Services. Proceedings of OSGeo's European Conference on Free and Open Source Software for Geospatial (FOSS4G-Europe 2014) July 15-17 2014 Jacobs University, Bremen, Germany. 4 p. Issued online. Available at: http://europe.foss4g.org/2014/content/toward-client-side-web-processing-services.html
16. Panidi E., Kazakov E., Kapralov E., Terekhov A., 2015. Hybrid Geoprocessing Web Services. In SGEM2015 Conference Proceedings (15th International Multidisciplinary Scientific GeoConference SGEM, Albena, Bulgaria, June 18-24, 2015), Book 2 Vol. 1, pp. 669-679
17. Schut P. (ed.), 2007. OpenGIS Web Processing Service. OpenGIS Standard. OGC 05-007r7. Version 1.0.0, 2007-06-08
18. Vretanos P.A., (ed.), 2002. Web Feature Service Implementation Specification. OGC 02-058. Version 1.0.0, 2002-05-17