

# ПЕРСПЕКТИВЫ СОЗДАНИЯ СРЕДСТВ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ГЕОДАНЫХ НА ОСНОВЕ СИСТЕМЫ УПРАВЛЕНИЯ ПРОСТРАНСТВЕННЫМИ ДАННЫМИ ГЕОПОРТАЛА

А. Г. Матвеев, С. О. Кондратов, О. Э. Якубайлик

*Институт вычислительного моделирования СО РАН*

andrey.g.matveev@gmail.com, skondratov@gmail.com, oleg@icm.krasn.ru

## Аннотация

*В статье рассматриваются функциональные возможности современных средств разработки клиентской части веб-приложений на примере системы управления пространственными данными геопортала. Приводятся перспективы дальнейших исследований, связанные с распределённой обработкой геопространственных данных с использованием сервис-ориентированной архитектуры.*

Непрерывное развитие технологий и технических характеристик сети Интернет в течение двух десятилетий ее существования обеспечило плавный переход в способе представления информации – от тривиальной презентации простых статичных гипертекстовых документов до сложнейших интерактивных мультимедийных информационных систем. Методология программирования для Интернет постоянно совершенствуется, нарастающими темпами возникают новые средства разработки.

## Основные функциональные возможности современных средств разработки веб-приложений

Разнообразие веб-приложений, необходимость разработки эффективных пользовательских интерфейсов [3], а также необходимость поддержки их стабильной работы не только в последних версиях популярных веб-браузеров, но и в устаревших (но при этом ещё используемых) версиях, делают разработку на JavaScript без использования вспомогательных средств чрезвычайно неэффективной и утомительной. До сих пор существует проблемы различного поведения разных типов браузеров (так называемых «движков» – Trident, Gecko, WebKit, Presto) в области интерпретации JavaScript, поддержки объектной модели документа (DOM) и каскадных таблиц стилей (CSS).

Начиная с середины 2000-х, среди разработчиков клиентской части веб-приложений все популярнее становятся так называемые JavaScript-фреймворки. *JavaScript-фреймворк (или библиотека JavaScript)* – это набор функций и средств, облегчающих реализацию сценариев JavaScript, совместимых с большинством современных браузеров. Каждая из библиотек в процессе своего развития проходит активное тестирование в различных веб-браузерах, и это в определенной степени гарантирует одинаковую работу веб-приложений на основе этих библиотек в различных типах браузеров (кроссбраузерность).

Рассмотрим основные функциональные возможности типичного JavaScript-фреймворка.

*Операции с элементами объектной модели документа (DOM).*

Как правило, в основе библиотек содержится реализация функций поиска и выборки элементов DOM с использованием селекторов CSS (Cascading Style Sheets, каскадные табли-

цы стилей) или XPath (XML Path Language). Данный функционал позволяют получать ссылки на элементы модели документа согласно идентификаторам этих элементов, их классов, значений атрибутов, а также с использованием всевозможных комбинаций этих свойств.

Кроме выборки элементов с использованием селекторов, JavaScript-библиотеки зачастую предоставляют функционал поиска и выборки элементов DOM согласно их относительной позиции в дереве документа, то есть, с использованием отношений: родительский элемент, дочерний элемент, элементы-предки, элементы-потомки и соседние элементы.

Естественным продолжением развития функционала фреймворков JavaScript, связанного с поиском и выборкой элементов, является реализация функций, предназначенных для манипуляций с моделью DOM, то есть: изменение значения элементов, их атрибутов, классов CSS, содержимого, их перемещение, копирование, удаление и так далее. Использование функций, связанных с объектной моделью документа, является одним из основных неоспоримых преимуществ по сравнению с реализацией кода на «чистом» JavaScript, поскольку реализация DOM до сих пор не является полной в различных версиях и типах браузеров.

#### *Поддержка асинхронных HTTP-запросов (AJAX).*

Использование концепции AJAX (Asynchronous JavaScript and XML) уже не раз доказало свою эффективность при реализации веб-приложений. Подобное объединение технологий динамического обращения к серверу (чаще всего, при помощи объекта XMLHttpRequest) и DHTML (Dynamic HTML) для организации интерактивных веб-интерфейсов обладает следующими преимуществами – уменьшение нагрузки на сервер и экономия трафика за счёт существенного уменьшения объёмов передаваемых данных и переноса части нагрузки на сторону веб-клиента, а также повышение интерактивности пользовательских интерфейсов и их ускорение.

Функционал для работы с AJAX в JavaScript-фреймворках представляет собой набор удобных в использовании методов, позволяющих выбрать тип ответа (XML, JSON, HTML или текст), назначить функции, запускаемые при успешном и неудачном запросе, выбрать метод запроса (GET или POST), настроить синхронную передачу данных или асинхронную и так далее. Возможности конфигурирования отличаются у различных библиотек, но все они существенно облегчают работу с AJAX, существенно сокращая объёмы кода, увеличивая его читаемость и прозрачность, а также избавляют от необходимых проверок на совместимость с текущим браузером, поскольку, например, браузер Internet Explorer версии 6 использует ActiveX вместо объекта XMLHttpRequest. Кроме того, существуют различия в реализации XMLHttpRequest в других типах браузеров.

#### *Обработка событий.*

Традиционный способ назначения событий браузера путём внедрения в HTML-код элемента специальных атрибутов является очень неэффективным, в особенности, учитывая функциональную сложность интерфейсов современных веб-приложений. Кроме того, отчасти существует проблема кроссбраузерности. Практически во всех современных JavaScript-фреймворках реализован собственный функционал обработки событий, который является простой и удобной альтернативой стандартному способу.

Помимо непосредственного назначения обработчиков событий клиента (таких как клик мышью по элементу, двойной клик, наведение курсора, перемещение мыши и так далее) элементам, уже созданным в модели документа, современные фреймворки позволяют осуществлять делегирование событий. Делегирование событий в JavaScript – это способ

назначения элементу DOM обработки события путём передачи управления от обработчика родительского элемента. Иными словами, функции-обработчики назначаются не самим элементам, а их родителям, что приводит к благоприятным последствиям:

- В коде становится меньше функций, которыми нужно оперировать;
- Занимается меньше памяти;
- Уменьшается количество связей между JavaScript-кодом и DOM;
- Исчезает необходимость удаления ненужных событий при изменении DOM;
- Становится возможным сквозное добавление событий к элементам DOM, создаваемым на лету.

#### *Визуальные эффекты*

Сочетание работы с таймерами JavaScript и стилями CSS позволяет реализовывать множество визуальных эффектов, таких как анимированное движение элементов, изменение их пропорций, смена цвета, поддержка прозрачности и так далее. Такого рода функционал играет большую роль при реализации интерактивных пользовательских интерфейсов веб-приложений.

#### *Служебные функции*

Практически каждый JavaScript-фреймворк обладает набором вспомогательных (служебных) функций, которые призваны облегчить разработку. Часто среди таких функций можно увидеть:

- определение «движка» и версии браузера;
- обход массива с выполнением функции на каждой итерации;
- поиск значения в массиве или объекте;
- объединение массивов или объектов;
- обработка строк (синтаксический анализ и пр.);
- и другие.

Помимо описанных выше возможностей, библиотеки JavaScript часто имеют возможность расширения посредством модулей или плагинов. Наряду с расширениями от разработчиков библиотек, существует огромное количество сторонних проектов. Поскольку большинство из JavaScript-фреймворков поставляется с открытым исходным кодом по лицензии свободного/открытого ПО (BSD, GPL или MIT), с течением времени вокруг этих продуктов обычно возникают сообщества активных веб-разработчиков, многие из которых создают собственные модули и выкладывают их в свободный доступ.

В целом, использование библиотек JavaScript дает огромные преимущества при разработке веб-приложений. Среди очевидных плюсов – уменьшение объёма кода, повышение удобочитаемости, поддержка кроссбраузерности. В дополнение к перечисленному, разработчик получает возможность правильно структурировать программный код, так как необходимость физически смешивать HTML и JavaScript практически исчезает.

Рассмотрим наиболее популярные на сегодняшний день JavaScript-фреймворки. Это – Dojo, Mootools, YUI, ExtJS, jQuery. Некоторые их характеристик перечисленных средств разработки приведены в таблице 1.

Каждый из фреймворков является мощным инструментом для веб-разработчика, со своими положительными и отрицательными сторонами. Все они обладают поддержкой рассмотренного набора функциональных возможностей – поддержка асинхронной передачи данных AJAX, обработка событий, и проч. Помимо базовой функциональности, которая в общих чертах совпадает у всех перечисленных библиотек, рассматриваемые JavaScript-

фреймворки отличаются наличием и возможностями подключаемых модулей и элементов веб-интерфейса (виджетов). Например, ExtJS обладает мощным набором функций для создания графического пользовательского интерфейса веб-приложения, так называемого "оконного интерфейса" на веб-странице, в стиле настольной операционной системы.

Таблица 1

### Сравнительные характеристики JavaScript-фреймворков

	Dojo	Mootools	YUI	ExtJS	jQuery
Версия	1.6.0 (15.03.2011)	1.3.1 (24.02.2011)	3.0 (29.09.2009)	4 (26.04.2011)	1.6 (3.05.2011)
Размер	28 КБ (gzip), 123 КБ (исх.)	7.3–65 КБ 101 КБ (исх.)	Ядро ~ 31 КБ	84–502 КБ	24 КБ (gzip), 155 КБ (исх.)
Лицензия	BSD и AFL	MIT	BSD	Коммерческая и GPL v3	MIT и GPL
Поддержка браузеров					
IE	6+	6+	6+	6+	6+
Firefox	3+	2+	3+	1,5+	2+
Safari	4	3+	4,0	3+	3+
Opera	10	9+	10,0+	9+	9+
Chrome	3	1+	Нет данных	1+	1+

Самым популярным фреймворком, согласно статистике Google Trends, является jQuery, причем его отрыв от конкурентов все больше растет, начиная с 2008 года. Популярность jQuery связана с тем, что, с одной стороны, данный фреймворк является превосходным средством для решения повседневных задач, связанных с манипуляциями с объектами DOM, обработкой событий, реализацией визуальных эффектов, работой с AJAX и так далее, и, с другой стороны, обладает довольно высокой производительностью, малым размером, хорошей документацией, и как следствие – низким порогом вхождения. Возможность расширения при помощи официальных и авторских модулей в совокупности с активным сообществом разработчиков, интеграция данного фреймворка в популярные системы управления веб-контентом (Drupal, Wordpress, TikiWiki), а также библиотека jQuery UI, содержащая набор примитивов графического интерфейса пользователя – все это сделало данный продукт очень качественным универсальным инструментом веб-разработчика [5].

Однако стоит также отметить, что выбор JavaScript-фреймворка – сложный вопрос, который должен быть рассмотрен в контексте решаемых задач. Иногда целесообразно использовать сразу несколько подобных инструментов.

### Система управления пространственными данными геопортала

Рассмотренный выше анализ технологий разработки веб-приложений стал составной частью исследований, направленных на формирование системы управления пространственными данными геопортала ИВМ СО РАН – распределенной геоинформационной системы, предназначенной для организации хранения, доступа к географической информации, навига-

ции и поиску по геопространственным данным, связанным с ними географическим сервисам (для доступа, редактирования, анализа, и т.д.) [4]. Предполагается, что основной задачей геопортала ИВМ СО РАН является поддержка научных исследований, обеспечение информационного взаимодействия специалистов, занимающихся обработкой и формированием тематических картографических, пространственных данных

Будем рассматривать конкретную задачу – разработку клиентской части системы управления пространственными данными геопортала – подсистемы «Каталог метаданных». Опыт разработки веб-сайтов и веб-приложений различной сложности и назначения показывает, что в большинстве случаев для ускорения и облегчения процесса создания данного типа программного обеспечения можно использовать специализированные системы, так называемые системы управления содержимым веб-сайтов (CMS, Content Management System). На данный момент наиболее популярными, согласно статистике «Рейтинга Рунета» (<http://www.ratingruneta.ru/cms/opensource/>), свободными CMS являются Joomla, Drupal, Wordpress, MODx, TYPO3. Проведенный анализ данных систем показал нецелесообразность их использования в качестве базовых средств для решения данной задачи, поскольку они по своему основному целевому назначению далеки от систем каталогизации, и, соответственно, излишне громоздки, т.к. перегружены ненужным в нашем случае функционалом [2]. Поэтому было принято решение создания специальной системы разработки геоинформационных веб-приложений.

При проектировании данной системы был определён ряд функциональных требований.

#### *Универсальное хранение данных.*

В рассмотренных популярных системах управления содержимым веб-сайтов в качестве хранилища данных используются СУБД (MySQL, PostgreSQL, MS SQL), причём использование систем управления базами данных является обязательным условием создания приложения на основе CMS. Учитывая тот факт, что геоинформационные веб-приложения могут обладать достаточно разнообразным функционалом — от небольших клиентских приложений, работающих с данными посредством технологий веб-сервисов, до крупных многофункциональных комплексов управления и анализа пространственной информации — было выдвинуто следующее требование: система разработки геоинформационных веб-приложений должна предоставлять возможность свободного выбора способа хранения данных (файловые архивы, СУБД, комбинированные способы).

#### *Гибкость формирования пользовательского интерфейса.*

Традиционно, CMS обладают достаточно развитыми средствами формирования (X)HTML-кода страниц, для облегчения создания различных элементов пользовательских интерфейсов, таких как гиперссылки, каскадные меню, формы и пр. Формирование индивидуальных интерфейсов обеспечивается в данных системах посредством создания тем, которые должны быть разработаны с использованием данных функций по определённой технологии. Данный способ является вполне удобным при создании традиционных интерфейсов веб-сайтов (учитывая современные возможности технологии CSS — каскадных таблиц стилей), но он накладывает существенные ограничения на разработчиков, перед которыми стоит задача формирования сложных интерфейсов веб-приложений. Поэтому разрабатываемая система должна обеспечить максимально гибкий способ реализации пользовательских интерфейсов.

### *Возможность интеграции различных JavaScript-фреймворков.*

Как было сказано выше, в популярных системах управления содержимым веб-сайтов в качестве JavaScript-фреймворка чаще всего используется библиотека jQuery. Несмотря на то, что данное программное средство является достаточно мощным и включает в себя множество функций для создания интерактивных пользовательских интерфейсов, геоинформационные приложения имеют особую специфику, которая требует подключения специализированных библиотек, предназначенных для формирования интерактивных карт, создания Windows-подобных интерактивных интерфейсов и т.д. Поэтому функционал разрабатываемой системы должен позволять свободную интеграцию дополнительных библиотек.

### *Поддержка технологии AJAX.*

Использование технологии асинхронных запросов (AJAX) является достаточно эффективным способом ускорения работы веб-приложения, уменьшения нагрузки на веб-сервер, уменьшения объёмов передаваемых данных, уменьшения времени реакции элементов пользовательского интерфейса, поэтому при создании системы разработки геоинформационных веб-приложений необходимо предусмотреть достаточно развитые и гибкие средства для применения данной технологии, а также предоставить возможность выбора возможного формата передачи данных (XML, JSON и т.п.).

### *Обеспечения маршрутизации HTTP-запросов и ответов.*

Одним из основных функциональных требований является обеспечение базовых функций работы с HTTP-запросами, поступающих от веб-браузера на веб-сервер, и ответов в виде (X)HTML-кода.

Созданная система разработки геоинформационных веб-приложений состоит из клиентской и серверной частей, реализуя тем самым технологию «клиент – сервер» [1]. Применение в основе серверной части приложения шаблона проектирования MVC (модель – представление – контроллер) предоставляет широкие возможности для решения поставленных задач. Использование данной архитектуры предполагает разделение данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента (рис. 1). Модификация каждого компонента может осуществляться независимо. Помимо стандартных элементов MVC, ключевыми блоками рассматриваемой системы также являются шаблоны представления страниц и AJAX-обработчики. На основе данной системы разработки была построена подсистема визуализации каталога пространственных данных – приложение «Каталог метаданных».

Связь веб-приложения с каталогом ресурсов ведётся на основе сервис-ориентированной архитектуры, реализованной при помощи протокола SOAP/XML [6]. Созданный набор функций, доступный в виде прикладного программного интерфейса (API), предоставляет возможности поиска ресурсов, их фильтрации, управления, редактирования, копирования, перемещения и так далее.

Существует два набора программных интерфейсов: общий (клиентский) интерфейс для пользовательских приложений и расширенный интерфейс (серверный) для приложений, имеющих возможность управления каталогом ресурсов, его объектами и отношениями между объектами.

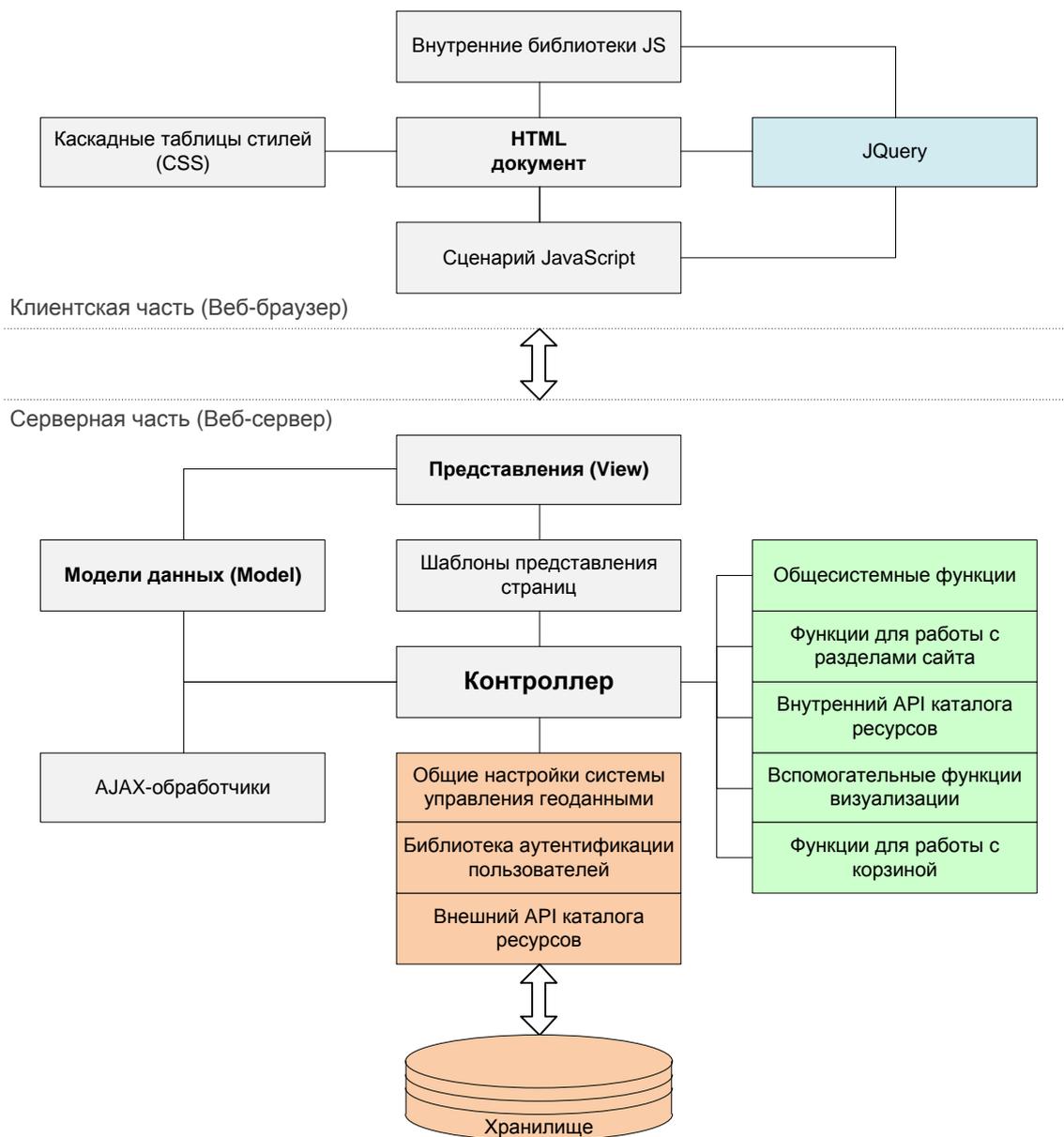


Рисунок 1. Архитектура веб-приложения для визуализации каталога пространственных данных

Основные функции общего прикладного программного интерфейса:

- Аутентификация и авторизация пользователя посредством создания (восстановление) сессии в каталоге ресурсов;
- Получение списка объектов каталога либо получение атрибутов одного объекта по его идентификатору;
- Удаление объекта;
- Создание фильтра поиска объектов и поиск объектов по фильтру;
- Получение списка доступных серверов либо вывод описания одного сервера;
- Получение списка доступных классификаторов, получение списка узлов (категорий) выбранного классификатора;
- Операции поиска, перечисления, редактирования, создания объектов типа «Папка»;
- Подсчёт количества ресурсов с пространственной привязкой с заданным фильтром;

- Операции поиска, перечисления, редактирования, создания геоинформационных ресурсов;
- Получение метаданных одного геоинформационного ресурса;
- Получение стилового описания выбранного геоинформационного ресурса;
- Работа с библиотеками стилей;
- Вывод информации о пользователях и правах.

Расширенный прикладной программный интерфейс предоставляет дополнительные возможности управления пользователями, ролями, правами доступа, серверами размещения исходных данных и другими элементами каталога ресурсов.

### **Распределенная обработка геопространственных данных**

Модульный подход к разработке программного обеспечения, основанный на использовании распределенных компонентов в сервис-ориентированной архитектуре, становится все более популярным в настоящее время. Концепция «облачных вычислений» стала одной из самых модных тенденций ИТ-индустрии. Появление новых типов аппаратных платформ (планшетов и ультрабуков с сенсорным вводом, смартфонов, и т.п.), операционных систем и пользовательских интерфейсов ставит новые задачи перед исследователями.

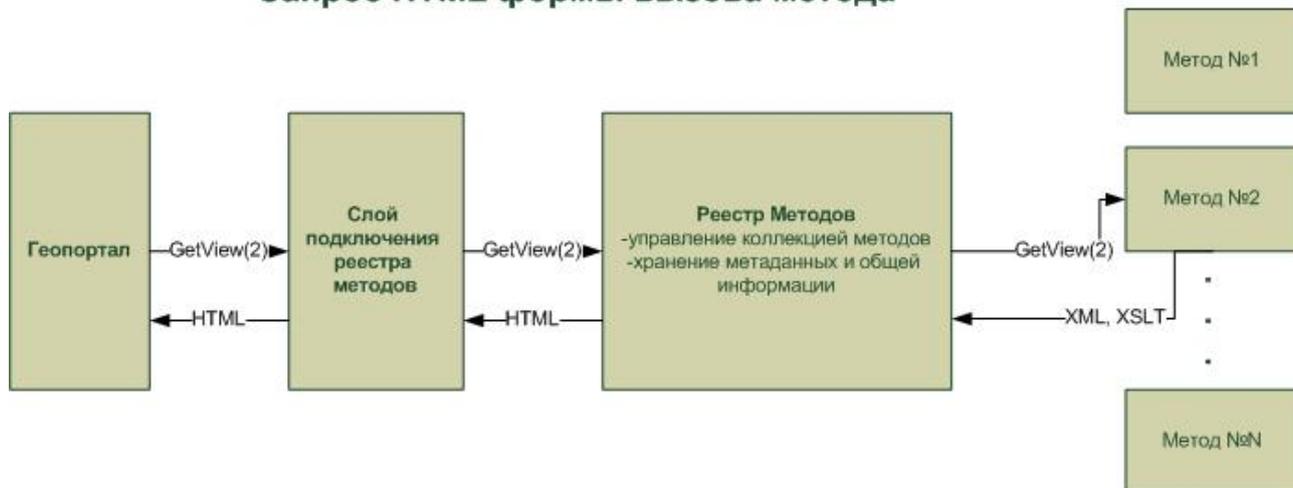
Объем геопространственной информации и ее использование стремительно нарастает. Как в правительственных кругах, так и в бизнес-сфере, растет осознание того, что использование информации о месторасположении – жизненно важный компонент эффективного принятия решений. Граждане, не являющиеся экспертами в геопространственной информации, и которые вряд ли даже знакомы с этим термином, также все чаще используют ее, взаимодействуют с ней, и во многих случаях даже вносят свой вклад в ее сбор.

В этом контексте все большее внимание сегодня уделяется формированию инфраструктур пространственных данных, как на государственном, национальном и международном, так и отраслевом уровне. Исследование и развитие глобальных, региональных и национальных инфраструктур геопространственных данных (ИПД) в настоящее время очень актуально. Наиболее существенными инициативами в этом направлении являются действующие международные программы: INSPIRE (Infrastructure for Spatial Information in Europe), NSDI (National Spatial Data Infrastructure), GSDI (Global Spatial Data Infrastructure) и GMES (Global Monitoring for Environment and Security). Общими целями большинства действующих программ ИПД является обеспечение координированного распределенного многопользовательского доступа к геопространственным ресурсам, поддержка решения фундаментальных и прикладных задач.

В соответствии с этой идеологией, при проектировании и разработке геопортала ИВМ СО РАН в настоящее время анализируются различные архитектуры системы распределенных вычислений. Рассматриваются различные методы построения прикладных систем с использованием международных стандартов открытого геопространственного консорциума OGC, в частности – Web Processing Service (WPS). WPS – это веб-сервис, предоставляющий услуги пространственной обработки растровых и векторных данных, возможности доступа к пакетам геомоделирования, инструментам статистики и обработки через Интернет. WPS является протоколом, обеспечивающим правила обмена данными, построения запросов к удаленным вычислительным процедурам. Однако эти запросы должны выполняться в соответствующем программном окружении. Должна быть сформирована среда поддержки програм-

мирования, нацеленная на решение задач информационного обеспечения и вычислительного геопространственного моделирования, определены правила и спецификации для расширения ее функциональных возможностей, в соответствии с формируемой сервис-ориентированной архитектурой инфраструктурой распределенных вычислений.

### Запрос HTML формы вызова метода



### Вызов метода



Рисунок 2 – Схема работы подсистемы распределенных вычислений геопортала.

На текущем этапе проектирования и разработки средств распределенной обработке геопространственных данных исследуется относительно простой пример формирования программного интерфейса, который включает описание набора входных параметров вычислительной процедуры с использованием XML/XSLT, средства обработки метаданных этой процедуры на геопортале, и прочие вспомогательные элементы. Формируемое приложение проектируется как неотъемлемая составная часть геопортала ИВМ СО РАН, который является источником всех исходных пространственных данных. Сервисы геопортала также обеспечивают визуализацию результатов вычислительных процедур на удаленных вычислительных системах – специально сконфигурированных серверах обработки геоданных.

Подсистема визуализации процедуры обработки геопространственных данных содержит базовый класс GMethod, который возвращает соответствующие XML/XSLT-данные, преобразуемые, в свою очередь, в HTML-форму вызова данного метода. Возврат результата

будет происходить примерно таким же образом, за исключением сохранения результирующего объекта. Разрабатываемая подсистема проектируется как автономный модуль геопортала, и поэтому модернизация ее структуры в процессе дальнейшего проектирования и разработки не будет нарушать его работоспособность (Рис. 2).

## ЛИТЕРАТУРА

- [1] Матвеев А.Г., Якубайлик О.Э. Разработка веб-приложения для обработки и представления пространственных метаданных геопортала // Вестник СибГАУ - 2012. - Вып. 2(42). - С. 48-54.
- [2] Якубайлик О.Э., Гостева А.А., Ерунова М.Г., Кадочников А.А., Матвеев А.Г., Пятаев А.С., Токарев А.В. Разработка средств информационной поддержки наблюдений за состоянием окружающей природной среды // Вестник КемГУ. - 2012. - № 3/1(51) - С. 136-142.
- [3] Матвеев А.Г. Формирование веб-интерфейса клиентской части геоинформационного интернет-портала. – В кн.: Геоинформационные технологии и математические модели для мониторинга и управления экологическими и социально-экономическими системами: ред. кол.: Ю.И. Шокин [и др.]; под ред. И.Н. Ротановой; Рос. акад. наук, Сиб. отделение, Ин-т водных и экологич. проблем. – Барнаул: Пять плюс, 2011. – С. 115-121.
- [4] Якубайлик О.Э. Проблемы формирования информационно-вычислительного обеспечения систем экологического мониторинга // Вестник СибГАУ - 2012. - Вып. 3(43). - С. 96-102.
- [5] Матвеев А.Г. Анализ JavaScript-фреймворков для создания динамичных веб-интерфейсов для задач поиска и фильтрации пространственных ресурсов // Перспективы развития информационных технологий: сборник материалов VII Международной научно-практической конференции / Под общ. ред. С.С. Чернова. – Новосибирск: Изд-во НГТУ, 2012. - С. 81-86.
- [6] Якубайлик О.Э., Кадочников А.А., Попов В.Г., Токарев А.В. Модель геоинформационной аналитической Интернет-системы для анализа состояния и презентации региона // Вестник СибГАУ. - 2009. - № 4. - С. 61-66.