

**РАЗРАБОТКА WEB-АДАПТАЦИИ GIS-СИСТЕМЫ WINMAP НА ОСНОВЕ
ТРЕХУРОВНЕВОЙ АРХИТЕКТУРЫ**

dzast@sfnedu.ru

К настоящему времени Геоинформационные системы (GIS-системы) получили широкое распространение, как в профессиональной деятельности, в первую очередь в виде традиционных Windows-приложений, ориентированных на профессиональную деятельность, так и в сфере некоммерческих информационных сайтов Internet. Особый интерес представляет создание версий профессиональных GIS, которые были бы доступны как online-приложения, в особенности на мобильных устройствах, но не использовали бы средства HTTP-протокола, как это принято в современных Web-адаптациях GIS-системы. В данной работе предложена архитектура распределенного Web-приложения, основанного на использовании специфического протокола обмена данными между GIS-клиентом и сервером. Кроме того, излагаемый подход используется трехуровневую архитектуру, разделяющую функции Web-сервера, выполняющего функции приема и обработки запросов клиентов, и GIS-сервера, который реализует традиционные функции генерации картографического изображения и выборки из атрибутивных таблиц. Это потенциально позволит повысить производительность online-системы, а так же упростить некоторые вопросы, связанные, например, с авторизацией и кешированием данных.

Ключевые слова: Web-приложения, GIS-системы, трехуровневая архитектура.

Введение. Современные GIS-системы [1] можно с функциональной точки зрения разделить на две категории. К первой категории относятся профессиональные системы, которые предназначены для управления данными и ресурсами, имеющими картографическую структуру, и которые используются в таких традиционных сферах, как ведение земельного кадастра и управление ресурсами недвижимости, коммунальное хозяйство, экологическая область, и т.д. Подобные системы строятся на основе хорошо известных пакетов, как ArcGIS, MAPInfo, и с архитектурной точки зрения являются традиционными оконными приложениями. Ко второй категории относятся online Web-системы (напр. GoogleMap и многочисленные другие источники), которые с функциональной точки зрения обычно являются некоммерческими информативными сайтами, не имеющими профессиональной ориентированности.

Следует так же отметить современную тенденцию по созданию профессионально-ориентированных Web-приложений, напр. ArcGIS Online, или отечественную разработку GIS WebToolKit [2]. Последняя использует технологию ASP.NET для визуализации картографических и атрибутивных данных и, таким образом, является надстройкой над Web-сервером [3].

В данной статье представлен подход к построению online-приложений для GIS-системы WinMAP [4]; характерными особенностями выбранного архитектурного решения являются:

1. Отказ от использования традиционной для Web-приложений схемы, основанной на использовании HTTP-протокола, и модулей серверных расширений. В качестве альтернативы разработаны новая архитектура клиент-серверного взаимодействия, оптимизированная для выполнения традиционных функций GIS-систем, и соответствующий протокол.

2. Разделение функций серверной части на собственно сервер, обеспечивающий взаимодействие с клиентами, и GIS-компонент, которые реализуют традиционные функции генерации картографического изображения и выборки из атрибутивных таблиц. Подобная архитектура обусловлена, в первую очередь, необходимостью интеграции с существующей GIS-системой, функционирующей в качестве отдельной программной компоненты.

3. Ориентированность клиентской части GIS-приложения на мобильные устройства, с учетом специфических особенностей их интерфейса.

Основная часть.

Рассмотрим причины отказа от традиционной архитектуры Web-приложений. В действительности, эта архитектура, основанная на протоколе HTTP и серверных и клиентских расширениях, является по существу зрелой и адекватной основой для разработки сетевых приложений в среде Web. Одним из основных ее достоинств – универсальность по отношению к клиентской части, что обусловило ее широчайшее применение. Так же следует отметить

изобилие доступных инструментальных средств для развития приложений. Однако у нее существуют и недостатки, среди которых в первую очередь следует отнести отсутствие поддержки полноценных авторизированных сессий (хотя это компенсируется при помощи средств надстройки, например, при помощи cookies), отсутствие развитых средств для оперирования с общими данными на уровне приложения.

Возвращаясь к первоначальной задаче, – создание средств удаленного доступа к существующей GIS-системе, можно использовать два принципиально различных архитектурных решений. Первое решение состоит в разработке специфических серверных расширений для некоторого Web-сервера, например, по аналогии с компонентами ASP.NET, которые можно было бы далее внедрять в код HTML-страниц. Эти компоненты были бы ответственны за взаимодействие с GIS-системой, и формировали бы данные в виде html-тегов - изображений и таблиц. Этот способ не лишен привлекательности, в первую очередь по причине возможности интеграции таких компонент с уже существующим изобилием инструментальных средств. Кроме того, поскольку результатом является по-прежнему HTML-страница, очевидным образом отпадает необходимость разработки клиентской части.

Тем не менее, при разработке системы был выбран альтернативный подход, по следующим причинам. Во-первых, автор полагает, что разработка отдельного протокола позволила бы более адекватно учесть специфику использования GIS-систем. Во-вторых, как говорилось выше,

следует использовать более удобные и функциональные средства для поддержки авторизации и сессий. Полноценное рассмотрение используемых в Web-приложениях механизмов для решения этих задач существенно выходит за рамки данной статьи, но следует сказать, что, например, для такого традиционного Web-приложения, как online-магазин, действительно принципиальным является способность параллельной обработки очень большого числа обращений. Но для профессиональной GIS-системы (точнее говоря, для конкретного экземпляра системы) количество одновременных обращений не будет значительным; гораздо важнее сама возможность обращения к ее ресурсам через Internet. Еще одним фактором в пользу отказа от HTTP является потенциальная возможность передачи данных в более компактном формате.

Следующим важным свойством является потенциальная необходимость в кэшировании на сервере данных, к которым относятся формируемые изображения с картографической информацией, а так же выборки из таблиц с атрибутивными (называемых так же семантически) данными. Как говорилось выше, нагрузка на сервер с точки зрения числа параллельных подключений не планируется быть особенно значительной, но операция рендеринга (прорисовка изображения хранимых в базе данных объектов) является весьма трудоемкой, что хорошо заметно даже при работе с GIS-системой в однопользовательском режиме. Полученные же изображения сравнительно компактны, а так же стабильны с точки зрения своего содержания.

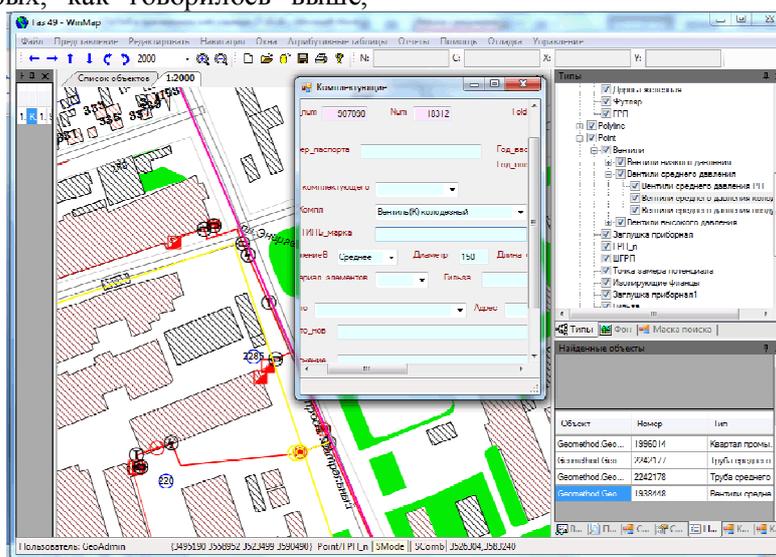


Рис. 1. Пример приложения для газовых служб на основе WinMAP

Разделение серверной части на две архитектурные компоненты, в свою очередь, обусловлено следующими факторами. Прежде всего,

поскольку речь идет в первую очередь о предоставлении доступа к существующей GIS-системе WinMAP (рис.1), важнейшим экономи-

ческим фактором является минимизация трудозатрат по ее переработке (т.е. изменению программного кода). Представляется нецелесообразным напрямую трансформировать ее в масштабируемый сервер, обслуживающий параллельные подключения. Оптимальным решением в данном случае является переработка системы WinMAP в приложение типа Windows-службы; детальное описание архитектуры описано ниже.

Кроме того, разделение серверной части на две компоненты обусловлено различными платформами – WinMAP является .NET-приложением, функционирующим в Windows, а сервер разрабатывается под Linux. Разнесение этих частей на два независимых узла потенциально будет способствовать распределению общей вычислительной нагрузки.

Наконец, решение использовать в качестве клиентской платформы мобильные Android-

устройства связано с их большей приспособленностью к работе в «полевых условиях».

Архитектура системы WebMAP.

Архитектура разрабатываемой системы (неформально называемой WebMAP) состоит из трех элементов – клиентским приложением, функционирующем под мобильными Android-устройствами, сервером, и GIS-компонентой (рис. 2).

Клиентская часть системы является Java-приложением. Его две основные задачи следующие:

1. создание и последующее управление соединения с сервером и обменом с ним данными;
2. визуализации получаемых картографических и атрибутивных данных и реализация интерфейса с пользователем.

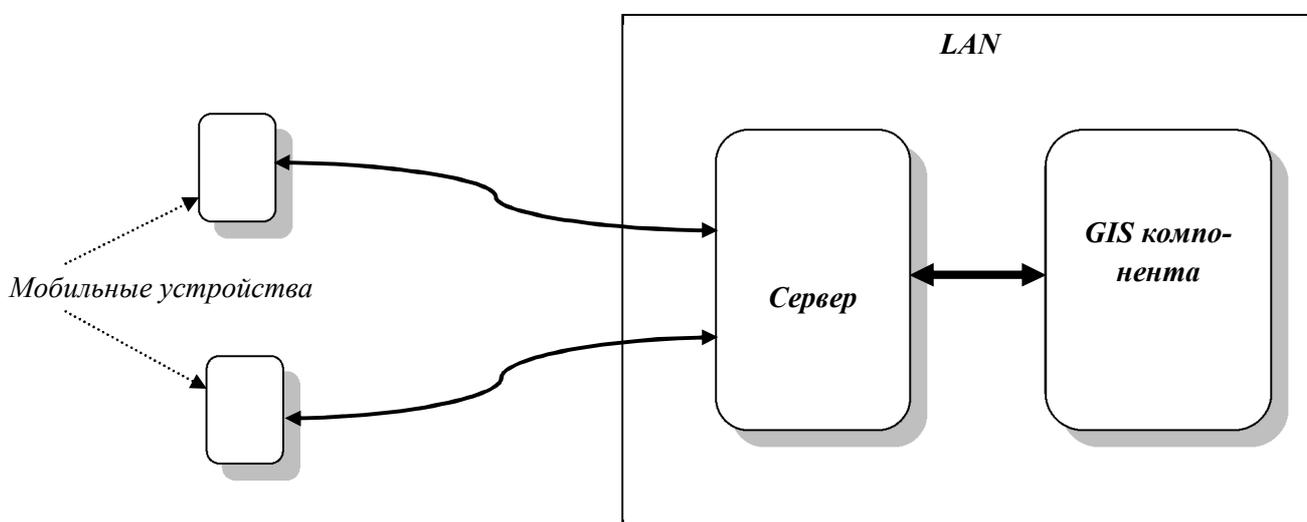


Рис. 2. Архитектура WebMAP

Функции сервера более сложны; он является промежуточным звеном между клиентскими устройствами, и собственно GIS-системой. Сервер реализуется при помощи языка Python на платформе Linux.

Сервер решает следующие основные задачи:

1. управление сеансами с клиентскими приложениями, включая авторизацию и управление учетными записями.
2. взаимодействие с GIS-компонентой.
3. управление кэшированием данных, полученных от GIS.

Взаимодействие между клиентами и сервером осуществляется на основе разработанного протокола, программная реализация которого используется традиционные средства сокетных соединений TCP/IP-сетей. Протокол определяет формат запросов к серверу и ответов.

Запросы бывают следующих категорий:

1. запрос на создание сеанса (сопровождается данными для авторизации);
2. запрос на прекращение сеанса;
3. запрос на получение картографических данных;
4. запросы на получение атрибутивных данных и данных о графических объектах;
5. запросы на выполнение поиска графических объектов и связанных с ними атрибутивными данными.

Подобная классификация запросов отражает традиционные для системы WinMAP два сценария работы. Согласно первому сценарию на экране воспроизводится картографическое изображение в соответствии с заданными координатами, на котором прорисованы попадающие в этот прямоугольник графические объекты и типами согласно используемой маске. Затем пользователь может выделить на экране группу объектов, и получить их детализированный список

и атрибутивную информацию в виде оконных бланков (этот сценарий проиллюстрирован на Рис.1). Другой сценарий предполагает предварительный просмотр список объектов, который можно формировать при помощи критериев селекции (типы, пространственные координаты, и др.), или списков записей таблиц с атрибутивными данными, а затем воспроизводить картографическое изображение, соответствующее найденным графическим объектам.

Соответственно, в запросах (в зависимости от их категорий) передаются координаты пририсовываемого прямоугольника, список типов графических объектов, критерии селекции для таблиц объектов и таблиц атрибутивных данных, и управляющая информация.

Получив запрос (типы 3,4 и 5), сервер обращается к GIS-компоненте и передает ему полученные в запросе данные. GIS-компонента выполняет либо рендеринг запрошенного картографического фрагмента, либо поиск в таблицах внутренней базы данных объектов и записей атрибутивных данных в соответствии с условиями.

Поскольку сервер предназначен для поддержки многих сеансов, для взаимодействия с GIS-компонентой формирует очередь заданий, диспетчеризируемой по принципу FIFO.

Важной функцией сервера является кеширование растровых изображений, получаемых от GIS-компоненты. Как показывает практика, при работе с GIS-системой является весьма типичной ситуация, когда один и тот же участок электронной карты просматривается несколько раз, чередуя различные масштабы и «сдвигая» изображение. Естественно, в данной ситуации кеширование предыдущих изображений является крайне целесообразным и технически несложным. Хранимые к кеше сервера изображения не

ассоциируются с конкретным сеансом, и могут быть при необходимости переданы другому сеансу при аналогичном запросе. В случае изменения графических объектов кеш очищается, однако такая ситуация весьма редкая.

Третья составляющая WebMAP является графическим ядром системы WinMAP, модифицированной в Windows-службу. Эта служба поддерживает постоянное сетевое соединение с сервером, получает от него запросы указанного выше формата, и выполняет рендеринг изображения и поисковые операции, взаимодействуя с внутренней базой данных.

Выводы. В работе представлены результаты по трансформации GIS-системы WinMAP в Web-образное приложений с целью обеспечения доступа к функциям этой системы при помощи мобильных устройств. Для этого были разработаны трехуровневая архитектура распределенного приложения, протокол взаимодействия компонент этой системы. Проведено критическое сравнение предложенного подхода с традиционной схемой реализации GIS-систем в виде Web-приложений.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шаши Шекхар, Санжей Чаула. Основы пространственных баз данных. М., Кудриц-Образ., 2004. 428 с.
2. Официальный сайт КБ «Панорама». <http://www.gisinfo.ru/>
3. Дино Эспозито. Программирование с использованием Microsoft ASP.NET 3.5. СПб. Питер. 2009 г. 1008 с.
4. Заставной Д.А. Встроенный язык скриптов для GIS-системы WinMap // Известия ЮФУ. Технические науки. 2011. №1. С. 144–150.

Zastavnoy D.A.

DEVELOPING A THREE-TIE WEB-APPLICATION FOR THE WINMAP GIS

GeoInformation Systems are widespread among modern information applications both professional ones and free online internet resources. But developing a professional GIS system accessible through wireless communications is still a welcomed approach. Most modern adaptations of such systems are based on a convenient http-style web application. Our approach is focused on developing a more sophisticated communication protocol to fulfill the complexity of the modern GIS applications. The three-tie architecture is involved to distribute communication functions of the server from the specific tasks of the GIS, such as raster image rendering, data fetching etc. This approach may increase the overall performance of the application as well as provides new possibilities such as implementing user authorization and intermediate image caching.

Key words: *Web-applications, GIS systems, tree-tie architecture*

Заставной Дмитрий Александрович, кандидат технических наук, доцент кафедры информатики и вычислительного эксперимента института математики, механики и компьютерных наук им. И.И. Воровича.

Южный Федеральный Университет.

Адрес: Россия, 344090, Ростов-на-Дону, ул. Мильчакова, д. 8а.

E-mail: dzast@sfedu.ru