

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»

А. В. Быков, С. В. Пьянков

WEB-КАРТОГРАФИРОВАНИЕ

*Допущено методическим советом
Пермского государственного
национального исследовательского университета
в качестве учебного пособия для студентов,
обучающихся по направлению подготовки бакалавров
«Картография и геоинформатика»*



Пермь 2015

УДК 258.926:004
ББК 32.973.26-018.1
Б 95

Быков А. В.

Б 95 Web-картографирование: учеб. пособие / А. В. Быков, С. В. Пьянков; Перм. гос. нац. исслед. ун-т. – Пермь, 2015. – 110 с.

ISBN 978-5-7944-2565-9

В данном учебном пособии освещены базовые понятия компьютерных сетей и современных программных и информационных технологий для построения web-карт. Подробно рассмотрен ГИС-сервер ESRI ArcGIS for Server, современные картографические стандарты и механизмы для создания геопорталов.

Предназначено для студентов, обучающихся по направлению подготовки бакалавров «Картография и геоинформатика» и по направлению подготовки магистров «Математико-картографическое моделирование геосистем и комплексов». Может быть использовано студентами и преподавателями других географических специальностей.

УДК 258.926:004
ББК 32.973.26-018.1

*Печатается по решению редакционно-издательского совета
Пермского государственного национального исследовательского университета*

Рецензенты: декан факультета прикладной информатики, к. э. н., доцент **И. М. Глотина** (Перм. ГСХА); ООО «Региональный центр обслуживания информационно-коммуникационных технологий» (В. В. Юзеев)

ISBN 978-5-7944-2565-9 © Пермский государственный национальный
исследовательский университет, 2015
© Быков А. В., Пьянков С. В., 2015

ОГЛАВЛЕНИЕ

Введение.....	6
Глава 1. ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ И INTERNET	8
1.1. Общие понятия вычислительных сетей и средств коммуникации	8
1.2. Преимущества использования сетей.....	11
1.3. Локальная вычислительная сеть (ЛВС) и компоненты ЛВС.....	12
1.4. Рабочие станции	14
1.5. Сетевые адаптеры.....	15
1.6. Файловые серверы.....	16
1.7. Сетевые операционные системы	18
1.8. Сетевое программное обеспечение	18
1.9. Защита данных.....	19
1.10. Типовой состав оборудования локальной сети.....	20
Контрольные вопросы	21
Глава 2. WEB-ПРИЛОЖЕНИЯ.....	22
2.1. СУБД и серверы баз данных	22
2.2. Архитектура клиент-сервер	24
2.3. Web-серверы	25
2.4. Web-приложения	26
2.5. HTML.....	27
2.6. Дополнительные модули для браузеров, предназначенные для создания пользовательского интерфейса	28
Контрольные вопросы	30
Глава 3. ПРИНЦИПЫ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЙ.....	31
3.1. Навыки, необходимые для разработки web-приложений.....	31

3.2. HTML.....	32
3.3. Структура документа HTML.....	33
3.4. CSS.....	45
3.5. JavaScript	53
3.6. Средства создания web-приложений на стороне сервера.....	58
Контрольные вопросы	59
Глава 4. ГИС И INTERNET. КАРТОГРАФИЧЕСКИЕ СЕРВИСЫ И ГЕОПОРТАЛЫ.....	60
4.1. Определение и задачи web-картографии. Миграция ГИС в Internet	60
4.2. Основные виды приложений, имеющих отношение к web-картографии. 62	
4.3. Ключевые организации.....	63
4.4. Открытые картографические стандарты	64
4.5. Картографические сервисы и геопорталы.....	66
4.6. Примеры геопорталов.....	68
4.7. Примеры картографических сервисов	70
Контрольные вопросы	71
Глава 5. ГИС-СЕРВЕРЫ. ARCGIS FOR SERVER.....	72
5.1. Определение ГИС-сервера. Обзор известных ГИС-серверов	72
5.2. Составляющие ArcGIS for Server	74
5.3. Архитектура ArcGIS for Server	75
5.4. Web Adaptor	80
5.5. Развертывание ArcGIS for Server на нескольких компьютерах	81
5.6. Процесс публикации ГИС-ресурсов в качестве web-сервисов	83
5.7. Клиенты для ArcGIS for Server	88
5.8. Работа с ArcGIS Online и Portal for ArcGIS	88

Контрольные вопросы	90
Глава 6. СЕРВИСЫ ARCGIS	91
6.1. Типы сервисов ArcGIS.....	91
6.2. Функции сервисов ArcGIS	92
6.3. Сервисы объектов	94
6.4. Сервисы геокодирования.....	95
6.5. Сервисы геоданных.....	96
6.6. Сервис геометрии.....	97
6.7. Сервисы геообработки.....	97
6.8. Сервисы глобуса.....	98
6.9. Сервисы изображений	99
6.10. Сервисы KML	101
6.11. Картографические сервисы	101
6.12. Сервисы мобильных данных.....	104
6.13. Сервисы сетевого анализа	104
6.14. Сервисы OCG	105
6.15. Сервисы Schematics.....	105
6.16. Сервисы поиска	106
6.17. Кэширование картографических сервисов.....	106
Контрольные вопросы	108
СПИСОК ЛИТЕРАТУРЫ.....	109

Введение

В настоящее время геоинформационные технологии стремительно и успешно развиваются, сфера их применения постоянно расширяется. Кроме того, очень быстрыми темпами развиваются телекоммуникационные технологии (в том числе Internet), растет пропускная способность сетей и вычислительная мощность ЭВМ, растет количество пользователей сетей. Поэтому закономерным этапом развития геоинформационных систем становится миграция их в Internet, где они существуют в виде картографических сервисов, геопорталов и прочих картографических web-приложений.

В настоящее время существуют различные варианты публикации пространственных данных в сети Internet: от простой публикации share-файла на картографическом сервисе типа Яндекс-карты до создания сложных геопорталов, которые развертываются с применением специального программного обеспечения – ГИС-серверов. ГИС-серверы, кроме отображения геопространственных данных, предоставляют различные возможности для их обработки, анализа и защиты. Кроме того, ГИС-серверы имеют различные API для разработки программного обеспечения, взаимодействующего с ними (приложений для мобильных телефонов и смартфонов, настольных приложений для ПК, web-приложений); это взаимодействие означает, что по запросу клиентского приложения ГИС-сервер возвращает ему ответ, в котором передаются какие-либо данные (формат передачи данных стандартизирован).

В настоящее время монополистом в сегменте рынка ГИС-серверов выступает компания ESRI с продуктом ArcGIS for Server. Этот программный продукт является эталоном, ни один ГИС-сервер не обладает таким набором возможностей; кроме того, он обладает очень подробной документацией, которая позволяла бы его освоить (несмотря на то, что большая ее часть написана на английском языке).

Цели и задачи изучения дисциплины

Данное учебное пособие предназначено для преподавателей, читающих курс «Web-картографирование» и студентов направления «Картография и геоинформатика».

Целью учебного курса является обучение современным технологиям публикации пространственных данных в сети Internet. Курс ориентирован на формирование у студентов следующих теоретических знаний:

- Основных понятий компьютерных сетей и механизмов их работы.
- Современных технологий и архитектур для построения web-приложений.
- Современных технологий и принципов разработки web-приложений, таких как язык разметки HTML, язык стилей CSS и язык сценариев JavaScript, а также основных платформ для создания web-приложений.
- Принципов работы ГИС-серверов.
- Архитектуры ГИС-сервера ArcGIS for Server.
- Всех типов сервисов, поддерживаемых ArcGIS for Server.

Глава 1. ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ И INTERNET

Перед тем, как приступить к изучению механизмов публикации геопространственных данных в сети Internet, необходимо изучить основные понятия о вычислительных сетях, их архитектуру и механизмы их работы. [5; 6] Рассмотренные понятия будут использованы в последующих темах, в которых раскрываются принципы работы ГИС-серверов, серверов СУБД и т.д.

1.1. Общие понятия вычислительных сетей и средств коммуникации

Вычислительная сеть – это совокупность объектов, образуемых устройствами передачи и обработки данных. Международная организация по стандартизации определила вычислительную сеть как последовательную бит-ориентированную передачу информации между связанными друг с другом независимыми устройствами.

Сети обычно находятся в частном ведении пользователя и занимают некоторую территорию; по территориальному признаку они разделяются на:

- Локальные вычислительные сети (ЛВС) или Local Area Network (LAN), расположенные в одном или нескольких близко расположенных зданиях. ЛВС обычно размещаются в рамках какой-либо организации (корпорации, учреждения), поэтому их называют корпоративными.

- Распределенные компьютерные сети, глобальные или Wide Area Network (WAN), расположенные в разных зданиях, городах и странах, которые бывают территориальными, смешанными и глобальными. В зависимости от этого глобальные сети бывают четырех основных видов: городские, региональные, национальные и транснациональные. В качестве примеров распределенных сетей очень большого масштаба можно назвать: Internet, EUNET, Relcom, FIDO.

В состав сети в общем случае включаются следующие элементы:

- сетевые компьютеры (оснащенные сетевым адаптером);
- каналы связи (кабельные, спутниковые, телефонные, цифровые, волоконно-оптические, радиоканалы и др.);
- различного рода преобразователи сигналов;
- сетевое оборудование.

Различают два понятия сети: коммуникационная сеть и информационная сеть (см. рис. 1.1):

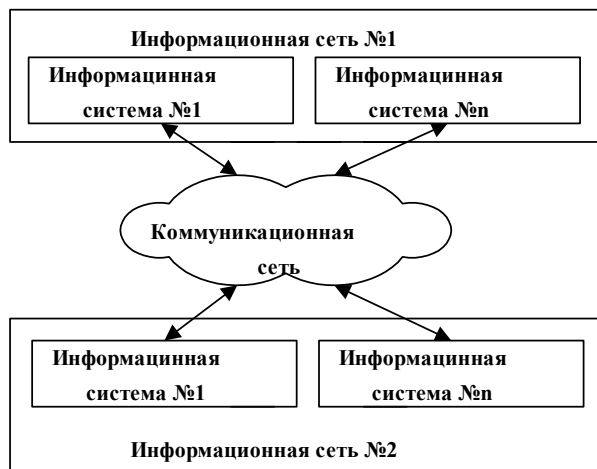


Рис. 1.1. Коммуникационная сеть

Коммуникационная сеть предназначена для передачи данных, также она выполняет задачи, связанные с преобразованием данных. Коммуникационные сети различаются по типу используемых физических средств соединения.

Информационная сеть предназначена для хранения информации и состоит из информационных систем. На базе коммуникационной сети может быть построена группа информационных сетей:

- Под информационной системой следует понимать систему, которая является поставщиком или потребителем информации.
- Компьютерная сеть состоит из информационных систем и каналов связи.

Под информационной системой следует понимать объект, способный осуществлять хранение, обработку или передачу информации. В состав информационной системы входят: компьютеры, программы, пользователи и другие составляющие, предназначенные для процесса обработки и передачи данных. В дальнейшем информационная система, предназначенная для решения задач пользователя, будет называться «рабочая станция» (client). Рабочая станция в сети отличается от обычного персонального компьютера (ПК) наличием сетевой карты (сетевое адаптера), канала для передачи данных и сетевого программного обеспечения.

Под каналом связи следует понимать путь или средство, по которому передаются сигналы. Средство передачи сигналов называют абонентским, или физическим, каналом.

Каналы связи (data link) создаются по линиям связи (передающей среде) при помощи сетевого оборудования и физических средств связи. Физические средства связи построены на основе витых пар, коаксиальных кабелей, оптических каналов или эфира. Между взаимодействующими информационными системами через физические каналы коммуникационной сети и узлы коммутации устанавливаются логические каналы.

Логический канал – это путь для передачи данных от одной системы к другой. Логический канал прокладывается по маршруту в одном или нескольких физических каналах. Логический канал можно охарактеризовать как маршрут, проложенный через физические каналы и узлы коммутации.

Информация в сети передается блоками данных по процедурам обмена между объектами. Эти процедуры называют протоколами передачи данных.

Протокол – это совокупность правил, устанавливающих формат и процедуры обмена информацией между двумя или несколькими устройствами.

Загрузка сети характеризуется параметром, называемым трафиком. Трафик (traffic) – это поток сообщений в сети передачи данных. Под ним понимают количественное измерение в выбранных точках сети числа проходящих блоков данных и их длины, выраженное в битах в секунду.

Существенное влияние на характеристику сети оказывает метод доступа. Метод доступа – это способ определения того, какая из рабочих станций сможет следующей использовать канал связи, и как управлять доступом к каналу связи (кабелю).

В сети все рабочие станции физически соединены между собой каналами связи по определенной структуре, называемой топологией. Топология – это описание физических соединений в сети, указывающее, какие рабочие станции могут связываться между собой. Тип топологии определяет производительность, работоспособность и надежность эксплуатации рабочих станций, а также время обращения к файловому серверу. В зависимости от топологии сети используется тот или иной метод доступа.

Состав основных элементов в сети зависит от ее архитектуры. Архитектура – это концепция, определяющая взаимосвязь, структуру и функции взаимодей-

ствия рабочих станций в сети. Она предусматривает логическую, функциональную и физическую организацию технических и программных средств сети. Архитектура определяет принципы построения и функционирования аппаратного и программного обеспечения элементов сети.

В основном выделяют три вида архитектур: архитектура терминал-главный компьютер, архитектура клиент-сервер и одноранговая архитектура.

Современные сети можно классифицировать по различным признакам: по удаленности компьютеров, топологии, назначению, перечню предоставляемых услуг, принципам управления (централизованным и децентрализованным), методам коммутации, методам доступа, видам среды передачи, скоростям передачи данных и т.д.

1.2. Преимущества использования сетей

Компьютерные сети представляют собой вариант сотрудничества людей и компьютеров, обеспечивающего ускорение доставки и обработки информации. Объединять компьютеры в сети начали более 30 лет назад. Когда возможности компьютеров выросли и ПК стали доступны каждому, развитие сетей значительно ускорилось.

Соединенные в сеть компьютеры обмениваются информацией и совместно используют периферийное оборудование и устройства хранения информации.

С помощью сетей можно разделять ресурсы и информацию. Ниже перечислены основные задачи, которые решаются с помощью рабочей станции в сети, и которые трудно решить с помощью отдельного компьютера:

Компьютерная сеть позволит совместно использовать периферийные устройства, включая:

- принтеры;
- плоттеры;
- дисковые накопители;
- приводы CD-ROM;
- дисководы;
- стримеры;
- сканеры;
- факс-модемы;

Информационная сеть позволяет совместно использовать информационные ресурсы:

- каталоги;
- файлы;
- прикладные программы;
- игры;
- базы данных;
- текстовые процессоры.

Компьютерная сеть позволяет работать с многопользовательскими программами, обеспечивающими одновременный доступ всех пользователей к общим базам данных с блокировкой файлов и записей, обеспечивающей целостность данных. Любые программы, разработанные для стандартных ЛВС, можно использовать в других сетях.

Совместное использование ресурсов обеспечит существенную экономию средств и времени. Например, можно коллективно использовать один лазерный принтер вместо покупки принтера каждому сотруднику или постоянных обращений со съемными запоминающими устройствами к единственному принтеру при отсутствии сети.

1.3. Локальная вычислительная сеть (ЛВС) и компоненты ЛВС

Компьютерная сеть состоит из трех основных аппаратных компонент и двух программных, которые должны работать согласованно. Для корректной работы устройств в сети их нужно правильно установить и установить рабочие параметры.

Основными аппаратными компонентами сети являются следующие:

1. Абонентские системы:

- компьютеры (рабочие станции или клиенты и серверы);
- принтеры;
- сканеры и др.

2. Сетевое оборудование:

- сетевые адаптеры;
- концентраторы (хабы);

- мосты;
 - маршрутизаторы и др.
3. Коммуникационные каналы:
- кабели;
 - разъемы;
 - устройства передачи и приема данных в беспроводных технологиях.

Основными программными компонентами сети являются следующие:

1. Сетевые операционные системы, где наиболее известные из них это:
 - Windows NT;
 - Windows for Workgroups;
 - LANtastic;
 - NetWare;
 - Unix;
 - Linux и т.д.
2. Сетевое программное обеспечение (сетевые службы):
 - клиент сети;
 - сетевая карта;
 - протокол;
 - служба удаленного доступа.

ЛВС (Локальная вычислительная сеть) – это совокупность компьютеров, каналов связи, сетевых адаптеров, работающих под управлением сетевой операционной системы и сетевого программного обеспечения.

В ЛВС каждый ПК называется рабочей станцией, за исключением одного или нескольких компьютеров, которые предназначены для выполнения серверных функций (файл-серверов). Каждая рабочая станция и файл-сервер имеют сетевые карты (адаптеры), которые посредством физических каналов соединяются между собой. В дополнение к локальной операционной системе на каждой рабочей станции активизируется сетевое программное обеспечение, позволяющее станции взаимодействовать с файловым сервером.

Компьютеры, входящие в ЛВС клиент-серверной архитектуры, делятся на два типа: рабочие станции, или клиенты, предназначенные для пользователей, и файловые серверы, которые, как правило, недоступны для обычных пользователей и предназначены для управления ресурсами сети.

Аналогично на файловом сервере запускается сетевое программное обеспечение, которое позволяет ему взаимодействовать с рабочей станцией и обеспечивать доступ к своим файлам.

1.4. Рабочие станции

Рабочая станция (workstation) – это абонентская система, специализирующаяся на решении определенных задач и использующая сетевые ресурсы. К сетевому программному обеспечению рабочей станции относятся следующие службы:

- сетевой клиент;
- служба доступа к файлам и принтерам;
- сетевые протоколы для данного типа сетей;
- сетевая плата;
- контроллер удаленного доступа.

Рабочая станция отличается от обычного автономного персонального компьютера следующим:

- наличием сетевой карты (сетевого адаптера) и канала связи;
- на экране во время загрузки ОС появляются дополнительные сообщения, которые информируют о том, что загружается сетевая операционная система;
- перед началом работы необходимо сообщить сетевому программному обеспечению имя пользователя и пароль. Это называется процедурой входа в сеть;
- после подключения к ЛВС появляются дополнительные сетевые дисковые накопители;
- появляется возможность использования сетевого оборудования, которое может находиться далеко от рабочего места.

1.5. Сетевые адаптеры

Для подключения ПК к сети требуется устройство сопряжения, которое называют сетевым адаптером, интерфейсом, модулем, или картой. Оно вставляется в гнездо материнской платы. Карты сетевых адаптеров устанавливаются на каждой рабочей станции и на файловом сервере. Рабочая станция отправляет запрос через сетевой адаптер к файловому серверу и получает ответ через сетевой адаптер, когда файловый сервер готов.

Сетевые адаптеры вместе с сетевым программным обеспечением способны распознавать и обрабатывать ошибки, которые могут возникнуть из-за электрических помех, коллизий или плохой работы оборудования.

Последние типы сетевых адаптеров поддерживают технологию Plug and Play («вставляй и работай»). Если сетевую карту установить в компьютер, то при первой загрузке система определит тип адаптера и запросит для него драйверы.

Различные типы сетевых адаптеров отличаются не только методами доступа к каналу связи и протоколами, но еще и следующими параметрами:

- скорость передачи;
- объем буфера для пакета;
- тип шины;
- быстродействие шины;
- совместимость с различными микропроцессорами;
- использование прямого доступа к памяти (DMA);
- адресация портов ввода/вывода и запросов прерывания;
- конструкция разъема.

IP-адрес (айпи-адрес, сокращение от англ. Internet Protocol Address) – это уникальный сетевой адрес узла в компьютерной сети, построенной по протоколу IP. В сети Интернет требуется глобальная уникальность адреса; в случае работы в локальной сети требуется уникальность адреса в пределах сети.

MAC-адрес (от англ. Media Access Control – управление доступом к среде, также Hardware Address) – это уникальный идентификатор, присваиваемый каждой единице активного оборудования компьютерных сетей. Адреса в каждом из пространств теоретически должны быть глобально уникальными. MAC-адрес «защит» в сетевой адаптер на аппаратном уровне.

1.6. Файловые серверы

Сервер – это компьютер, предоставляющий свои ресурсы (диски, принтеры, каталоги, файлы и т.п.) другим пользователям сети.

Файловый сервер обслуживает рабочие станции. В настоящее время это обычно быстродействующий компьютер (часто многопроцессорный). Чаще всего файловый сервер выполняет только эти функции. Но иногда в малых ЛВС файл-сервер используется еще и в качестве рабочей станции. На файловом сервере должна стоять сетевая операционная система, а также сетевое программное обеспечение. К сетевому программному обеспечению сервера относятся сетевые службы и протоколы, а также средства администрирования сервера.

Файловые серверы могут контролировать доступ пользователей к различным частям файловой системы. Это обычно осуществляется разрешением пользователю присоединить некоторую файловую систему (или каталог) к рабочей станции пользователя для дальнейшего использования в качестве локального диска.

По мере усложнения возлагаемых на серверы функций и увеличения числа обслуживаемых ими клиентов происходит все большая специализация серверов. Существует множество типов серверов. Рассмотрим основные из них:

- Первичный контроллер домена, сервер, на котором хранится база бюджетов пользователей и поддерживается политика защиты.
- Вторичный контроллер домена, сервер, на котором хранится резервная копия базы бюджетов пользователей и политики защиты.
- Универсальный сервер, предназначенный для выполнения несложного набора различных задач обработки данных в локальной сети.
- Сервер баз данных, выполняющий обработку запросов, направляемых базе данных.
 - Проху-сервер, подключающий локальную сеть к сети Internet.
 - Web-сервер, предназначенный для работы с web-информацией.
 - Файловый сервер, обеспечивающий функционирование распределенных ресурсов, включая файлы, программное обеспечение.
- Сервер приложений, предназначенный для выполнения прикладных процессов. С одной стороны, взаимодействует с клиентами, получая задания, а

с другой стороны, работает с базами данных, подбирая данные, необходимые для обработки.

- Сервер удаленного доступа, обеспечивающий сотрудникам, работающим дома торговым агентам, служащим филиалов, лицам, находящимся в командировках, возможность работы с данными сети.

- Телефонный сервер, предназначенный для организации в локальной сети службы телефонии. Этот сервер выполняет функции речевой почты, автоматического распределения вызовов, учет стоимости телефонных разговоров, интерфейса с внешней телефонной сетью. Наряду с телефонией сервер может также передавать изображения и сообщения факсимильной связи.

- Почтовый сервер, предоставляющий сервис в ответ на запросы, присланные по электронной почте.

- Сервер доступа, дающий возможность коллективного использования ресурсов пользователями, оказавшимися вне своих сетей (например, пользователями, которые находятся в командировках и хотят работать со своими сетями). Для этого пользователи через коммуникационные сети соединяются с сервером доступа и последний предоставляет необходимые ресурсы, имеющиеся в сети.

- Терминальный сервер, объединяющий группу терминалов, упрощающий переключения при их перемещении.

- Коммуникационный сервер, выполняющий функции терминального сервера, но осуществляющий также маршрутизацию данных.

- Видеосервер, который в наибольшей степени приспособлен к обработке изображений, снабжает пользователей видеоматериалами, обучающими программами, видеоиграми, обеспечивает электронный маркетинг. Имеет высокую производительность и большую память.

- Факс-сервер, обеспечивающий передачу и прием сообщений в стандартах факсимильной связи.

- Сервер защиты данных, оснащенный широким набором средств обеспечения безопасности данных и, в первую очередь, идентификации паролей.

1.7. Сетевые операционные системы

Сетевые операционные системы (Network Operating System – NOS) – это комплекс программ, обеспечивающих в сети обработку, хранение и передачу данных.

Для организации сети кроме аппаратных средств необходима также сетевая операционная система. Операционные системы сами по себе не могут поддерживать сеть. Для дополнения какой-нибудь ОС сетевыми средствами необходима процедура инсталляции сети.

Сетевая операционная система необходима для управления потоками сообщений между рабочими станциями и файловым сервером. Она является прикладной платформой, предоставляет разнообразные виды сетевых служб и поддерживает работу прикладных процессов, реализуемых в сетях. NOS используют архитектуру клиент-сервер или одноранговую архитектуру.

NOS определяет группу протоколов, обеспечивающих основные функции сети. К ним относятся:

- адресация объектов сети;
- функционирование сетевых служб;
- обеспечение безопасности данных;
- управление сетью.

1.8. Сетевое программное обеспечение

Клиент для сетей обеспечивает связь с другими компьютерами и серверами, а также доступ к файлам и принтерам.

Сетевая карта является устройством, физически соединяющим компьютер с сетью. Для каждой сетевой карты устанавливаются свои драйверы, значение IRQ (требования к прерыванию) и адреса ввода/вывода.

Протоколы используются для установления правил обмена информацией в сетях.

Служба удаленного доступа позволяет делать файлы и принтеры доступными для компьютеров в сети.

Применение многопользовательских версий прикладных программ резко увеличивает производительность. Многие системы управления базами данных позволяют нескольким рабочим станциям работать с общей базой данных. Большинство деловых прикладных программ также являются многопользовательскими.

1.9. Защита данных

Защита данных от несанкционированного доступа при работе в ЛВС необходима по следующим причинам:

- Необходимость обеспечения гарантии от разрушений (При работе в сети неопытных пользователей возможно уничтожение файлов и каталогов).
- Необходимость защиты конфиденциальности.
- Необходимость защиты от мошенничества (Некоторые расчетные ведомости несут в себе большие денежные суммы, и бывает, пользователи поддаются искушению выписать чек на свое имя).
- Необходимость защиты от преднамеренных разрушений.

Использование паролей и ограничение доступа

Первый шаг к безопасности – это введение пароля. Каждому пользователю ЛВС присваивается пароль – секретное слово, известное только этому пользователю. При вводе пароля высвечиваются звездочки. Сетевая операционная система хранит информацию по всем именам и паролям (в закодированной форме), а также о правах доступа к директориям и другие атрибуты пользователей.

Еще одна возможность защиты данных заключается в ограничении доступа к определенным директориям или определенным серверам. Доступ к дискам рабочих станций выбирается посредством вкладки Управление доступом в программе Сетевое окружение. Доступ между серверами организуется посредством установки доверительных отношений между серверами.

1.10. Типовой состав оборудования локальной сети

Фрагмент вычислительной сети включает основные типы коммуникационного оборудования, применяемого сегодня для образования локальных сетей и соединения их через глобальные связи друг с другом.

Для построения локальных связей между компьютерами используются различные виды кабельных систем, сетевые адаптеры, концентраторы, повторители. Для связей между сегментами локальной вычислительной сети используются концентраторы, мосты, коммутаторы, маршрутизаторы и шлюзы.

Для подключения локальных сетей к глобальным связям используются:

- специальные выходы (WAN-порты) мостов и маршрутизаторов;
- аппаратура передачи данных по длинным линиям – модемы (при работе по аналоговым линиям);
- устройства подключения к цифровым каналам (ТА – терминальные адаптеры сетей ISDN, устройства обслуживания цифровых выделенных каналов типа CSU/DSU и т.п.).

На рисунке 1.10.1 приведен фрагмент вычислительной сети.

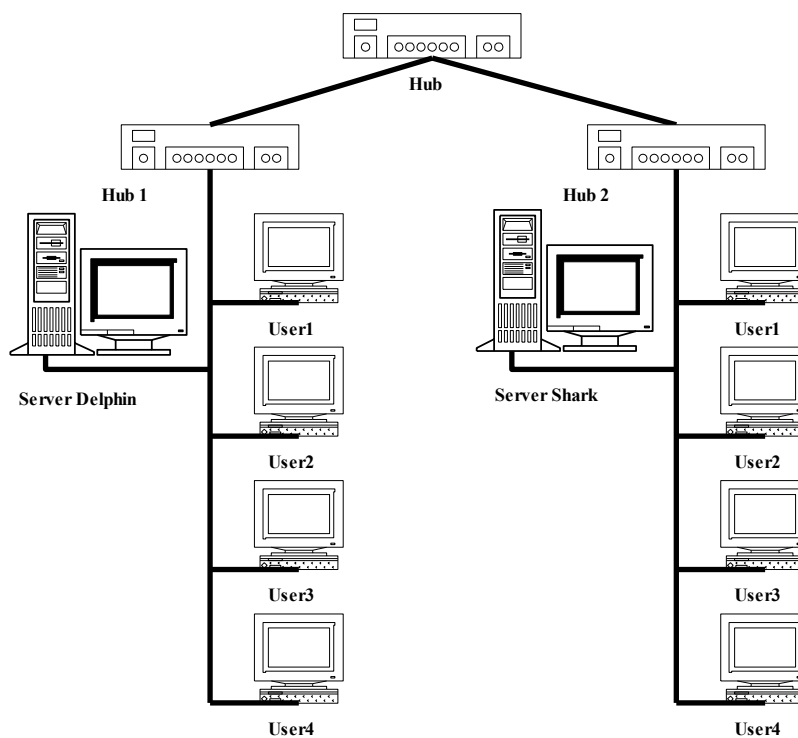


Рис. 1.10.1. Пример фрагмента вычислительной сети

Контрольные вопросы

1. Что такое вычислительная сеть? Какие основные элементы входят в состав вычислительной сети?
2. Что такое локальная вычислительная сеть (ЛВС)? Какие элементы входят в состав ЛВС?
3. Что такое рабочая станция?
4. Что такое файловый сервер? Какие существуют основные типы серверов?
5. Для чего необходима защита данных в компьютерных сетях?

Глава 2. WEB-ПРИЛОЖЕНИЯ

2.1. СУБД и серверы баз данных

Для хранения и обработки данных используется специальное программное обеспечение – системы управления базами данных. СУБД предназначены для организации хранения данных, обеспечения их целостности и организации доступа и управления ими [14].

Для модификации, организации, и управления данными в реляционных СУБД используется специальный язык – SQL (Structured Query Language – язык структурированных запросов), который является набором операторов, инструкций и вычисляемых функций. Этот язык независим от конкретной СУБД и соответствует единому стандарту, который постоянно совершенствуется. В разных СУБД существуют различные процедурные расширения функционала SQL, реализующие дополнительные функции для работы с данными (например, математические функции, не включенные в стандарт или функции для работы с геопространственными данными).

Назначение SQL:

- создание БД и таблиц с полным описанием их структуры;
- выполнение основных операций манипулирования данными (таких, как вставка, модификация и удаление данных из таблиц);
- выполнение простых и сложных запросов.

Одна из ключевых особенностей языка SQL заключается в том, что с его помощью формируются запросы, описывающие, какую информацию из базы данных необходимо получить, а пути решения этой задачи СУБД определяет сама.

Большинство современных СУБД являются многопользовательскими, то есть они могут предоставлять одновременный доступ к информации, хранящейся в базах данных сразу для нескольких пользователей. При этом у каждого пользователя есть свой набор прав и ограничений при работе с данными. Например, одному пользователю разрешается только просматривать информацию из базы данных, а другому – и просматривать и редактировать. Можно

также настроить права доступа не только для всей информации в базе данных, но и для каждой отдельной таблицы.

Для публикации базы данных в сети Internet используется сервер баз данных – специальное программное обеспечение, обеспечивающее безопасный доступ к базе данных средствами SQL с использованием аутентификации в ней. Сервер СУБД также обеспечивает операции ввода-вывода при доступе к информации, содержащейся в базах данных.

Для реализации подключения к серверу баз данных используется строка подключения, которая содержит IP-адрес сервера, имя базы данных, имя пользователя базы данных и пароль к ней. Пример подключения к базе данных:

```
"Server=212.192.85.220;Database=gts;User  
ID=sde;Password=*****"
```

Примеры наиболее известных серверов баз данных:

- Microsoft SQL Server. Это проприетарное программное обеспечение, используемое для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка. В качестве языка запросов использует Transact-SQL (SQL со значительно расширенной функциональностью). MSSQL может работать под управлением операционных систем семейства Windows (имеется неполная поддержка Unix и Linux).

- Oracle Database Center. Самая распространенная корпоративная СУБД. Очень мощный программный продукт, самые крупные предприятия и корпорации используют именно его. Отличается очень высокой ценой и высокой надежностью; имеет графический интерфейс для настройки СУБД и управления базами данных. Еще одно преимущество ODC заключается в возможности работы под управлением различных операционных систем (Windows Server, Solaris, Unix, Linux, Mac OS Server).

- PostgreSQL. Это свободная СУБД с открытым исходным кодом. Достоинства: поддержка многочисленных стандартов SQL и форматов данных, возможность расширения функциональности посредством написания скриптов на различных языках программирования. Недостатки: сложность в освоении и настройке.

- MySQL. Это свободная СУБД с открытым кодом. Достоинства: возможность работать под управлением множества операционных систем и поддержка большого количества форматов данных.

2.2. Архитектура клиент-сервер

Клиент-сервер – вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределена между поставщиками услуг (серверами) и потребителями (заказчиками) услуг (клиентами) [4]. Чаще всего клиенты и серверы взаимодействуют через компьютерную сеть и зачастую являются как различными физическими устройствами, так и программным обеспечением. Например, сервером является вычислительный кластер, работающий под управлением ОС Linux, а клиентом является web-браузер смартфона с ОС Android.

Главным преимуществом архитектуры клиент-сервер является резкое уменьшения объема данных, передаваемых между устройствами. Кроме того, данная архитектура имеет еще ряд преимуществ:

- 1) Программный код, выполняемый на стороне сервера, не дублируется программой-клиентом;
- 2) Все вычисления выполняются на стороне сервера, поэтому требования к аппаратному обеспечению на стороне клиента снижаются;
- 3) Сервер, как правило, защищен лучше большинства клиентов, поэтому на нем можно обеспечить соответствующий контроль прав доступа для программ-клиентов.

Но архитектура клиент-сервер имеет и недостатки:

- 1) В случае неработоспособности сервера прекращает работать вся сеть. Сервер может стать неработоспособным из-за нехватки производительности вследствие подключения большого количества клиентов или наличия неисправностей в работе оборудования.
- 2) Для поддержания сервера в работоспособном состоянии возникает необходимость найма сотрудников – администраторов сервера.
- 3) Высокая стоимость оборудования.

Кроме двухуровневой архитектуры клиент-сервер, существуют архитектуры с тремя и более уровнями. Самой распространенной архитектурой является архитектура «сервер приложений – сервер баз данных – клиент». (Желательно, чтобы web-сервер и сервер баз данных физически располагались на разных вычислительных машинах, так как в этом случае будет обеспечена максимальная производительность системы.) Для геопортала (геопорталы будут подробно рассмотрены в следующей теме) свойственна четырехуровневая архитектура: «web-сервер – ГИС-сервер – сервер баз данных – клиент». Существуют и другие, более сложные вычислительные архитектуры.

2.3. Web-серверы

Web-сервером принято называть сервер, принимающий HTTP-запросы от клиентов (обычно web-браузеров) и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, файлом, медиа-поток или другими данными. Обмен данными между web-сервером и web-браузером проходит по протоколу HTTP или по протоколу HTTPS, поддерживающему шифрование данных. Клиент, которым обычно является веб-браузер, передает веб-серверу запросы на получение ресурсов, обозначенных URL-адресами. Ресурсы – это HTML-страницы, изображения, файлы, медиа-поток или другие данные, которые необходимы клиенту. В ответ веб-сервер передает клиенту запрошенные данные.

Web-сервером называется как физический сервер, так и программное обеспечение, выполняющее функции web-сервера. Рассмотрим кратко самые распространенные web-серверы.

- Apache. Это кроссплатформенный свободный web-сервер, самый популярный в мире, под его управлением работает более половины всех web-сайтов в мире. Отличается сложностью в настройке, так как она производится через текстовые файлы;
- NGINX. Это второй по популярности web-сервер, также кроссплатформенный, свободный и с открытым исходным кодом. Разрабатывается российскими программистами;
- IIS (Internet Information Services) – проприетарный Web-server от компании Microsoft, поставляемый вместе с операционной системой Windows Server.

Отличается легкостью в освоении и графическим интерфейсом для управления web-сервером;

- Прочие, с очень низкой популярностью: Google Web Server, Resin, Cherokee, и так далее.

Веб-серверы могут иметь различные дополнительные функции, например:

- автоматизация работы web-страниц;
- ведение журнала обращений пользователей к ресурсам;
- аутентификация и авторизация пользователей;
- поддержка динамически генерируемых страниц;
- поддержка HTTPS для защищенных соединений с клиентами.

Часто на компьютере вместе с web-сервером устанавливается также и почтовый сервер.

Зачастую web-серверы располагают не на физическом, а на виртуальном оборудовании (на виртуальных машинах). Это означает, что на одной большой многопроцессорной машине (суперкомпьютере) с большими объемами оперативной и дисковой памяти эмулируется аппаратное обеспечение для сразу нескольких вычислительных машин. Преимущество такого подхода состоит в более рациональном распределении ресурсов, то есть в каждый момент времени каждая виртуальная машина получает ровно столько ресурсов, сколько ей необходимо. Кроме того, при таком подходе повышается отказоустойчивость (надежность суперкомпьютера выше, чем у других машин) и уменьшается вероятность потери данных. Существует множество программного обеспечения (как свободного, так и проприетарного) для создания виртуальных машин (VMware, VirtualBox и т.д.).

2.4. Web-приложения

Web-приложением называется приложение, в котором в роли сервера выступает HTTP-сервер, а в роли клиента – web-браузер. Выполнение программного кода происходит на сервере по запросу клиента, результатом выполнения является сгенерированная сервером web-страница с мультимедиа-контентом (аудио-, видеопотоками, изображениями и другими данными). Web-

приложение выполняется независимо от того, какая операционная система установлена на компьютере клиента.

Web-приложение состоит из клиентской и серверной частей, тем самым реализуя технологию «клиент-сервер».

Клиентская часть реализует пользовательский интерфейс, формирует запросы к серверу и обрабатывает ответы от него. Серверная часть получает запрос от клиента, выполняет вычисления, после этого формирует web-страницу и отправляет ее клиенту по сети с использованием протокола HTTP.

Само web-приложение может выступать в качестве клиента других служб, например базы данных или другого web-приложения, расположенного на другом сервере. Ярким примером web-приложения является система управления содержимым статей Википедии: множество ее участников могут принимать участие в создании сетевой энциклопедии, используя для этого браузеры своих операционных систем (будь то Microsoft Windows, GNU/Linux или любая другая операционная система) и не загружая дополнительных исполняемых модулей для работы с базой данных статей.

В настоящее время набирает популярность новый подход к разработке web-приложений, называемый Ajax. При использовании Ajax страницы web-приложения не перезагружаются целиком, а лишь догружают необходимые данные с сервера, что делает их более интерактивными и производительными.

Для создания веб-приложений на стороне сервера используются разнообразные технологии и любые языки программирования, способные осуществлять вывод в стандартную консоль.

Для создания пользовательского интерфейса используются различные технологии: HTML+CSS+JavaScript (язык разметки+язык стилей+язык сценариев web-страниц) или различные дополнительные модули для браузера (Flash, Flex, SilverLight и т.д.). Рассмотрим все эти технологии для создания пользовательского интерфейса.

2.5. HTML

HTML – стандартный язык разметки документов, который легко интерпретируется браузерами для отображения в привычной для человеческого восприятия форме [7]. HTML является теговым языком разметки, то есть каждый до-

кумент, описанный этим языком, представляет собой набор элементов, начало и конец которых обозначаются тегами. Для одной части элементов используются как открывающие, так и закрывающие теги (<head></head>), в то же время для другой части закрывающие теги не нужны (например, для перевода строки
)

Для формального описания внешнего вида HTML-документа используется CSS – язык стилей. Правила CSS пишутся в виде так называемых таблиц стилей, которые могут быть включенными в HTML-документ с помощью ссылки на файл с таблицей разметки стилей либо находиться в самом документе HTML. Каждое правило CSS настраивает внешний вид и поведение как отдельных элементов HTML страницы, так и совокупностей элементов HTML. Например, описание внешнего вида и поведения элемента может выглядеть так:

```
html {
    height: 100%;
    width: 100%;
    margin: 0;
    padding: 0;
}
```

Для придания интерактивности web-приложению используется язык сценариев – JavaScript.

2.6. Дополнительные модули для браузеров, предназначенные для создания пользовательского интерфейса

Далее будут рассмотрены дополнительные модули (плагины для браузеров), которые устанавливаются отдельно. Они предназначены для воспроизведения мультимедийного контента на web-странице и зачастую используются для создания интерактивного пользовательского интерфейса для web-приложений.

- Microsoft Silverlight – это программная платформа, включающая в себя модуль для браузера, который позволяет запускать приложения, содержащие анимацию, векторную графику и аудио- видеоролики. Данная платформа несовместима со многими браузерами и операционными системами.

- Adobe Flash – мультимедийная платформа компании Adobe Systems для создания web-приложений или мультимедийных презентаций. Широко используется для создания рекламных баннеров, анимации, игр, а также воспроизведения на web-страницах видео- и аудиозаписей. Платформа включает в себя ряд средств разработки, а также программу для воспроизведения flash-контента – Adobe Flash Player, хотя flash-контент умеют воспроизводить и многие плееры сторонних производителей. Adobe Flash позволяет работать с векторной, растровой и с трехмерной графикой, используя при этом графический процессор, а также поддерживает двунаправленную потоковую трансляцию аудио и видео.

- Adobe Flex – это платформа, расширяющая возможности Adobe Flex; Flex позволяет описывать интерфейс приложения на XML, ускоряя и упрощая процесс разработки насыщенных web-приложений. Логика приложения пишется на языке программирования ActionScript 3.

- CGI – это стандарт интерфейса, используемого для связи внешней программы с web-сервером. Программу, которая работает по такому интерфейсу совместно с web-сервером, принято называть шлюзом, хотя многие предпочитают названия «скрипт» (сценарий) или «CGI-программа». Сам интерфейс разработан таким образом, чтобы можно было использовать любой язык программирования, который может работать со стандартными устройствами ввода-вывода.

Стоит отметить, что из вышперечисленных платформ корректно работать в большинстве операционных систем и браузеров будет только платформа CGI, но она достаточно сложна в освоении, поэтому используется редко.

В настоящее время данные дополнительные модули вытесняются новой технологией HTML5, в которой поддержка мультимедиа-технологий значительно улучшена, к тому же она не требует установки дополнительных модулей для браузеров и поддерживается большим количеством браузеров и операционных систем.

Контрольные вопросы

1. Перечислите наиболее известные серверы СУБД.
2. Для чего предназначен язык SQL?
3. Что такое web-сервер? Перечислите наиболее известные web-серверы.
4. Что такое web-приложение? Приведите примеры web-приложений.
5. Перечислите основные технологии для создания интерфейса web-приложений.

Глава 3. ПРИНЦИПЫ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЙ

3.1. Навыки, необходимые для разработки web-приложений

Технология разработки web-приложений является обособленным разделом разработки программного обеспечения. Для разработки web-приложений необходимы следующие навыки:

1. Знание языка разметки страниц HTML.

2. Знание языка стилей страниц CSS. При этом разработчику не обязательно иметь дизайнерские и творческие способности для оформления web-страниц. В данном случае CSS используется для настройки элементов интерфейса и поведения приложения.

3. Знание языка сценариев JavaScript для придания интерактивности интерфейсу приложения.

4. Знание современных мультимедиа-технологий (также для придания интерактивности и динамичности интерфейсу приложения).

5. Примерно в половине случаев требуются навыки настройки и обслуживания серверов (навыки серверного и системного администрирования) и понимание архитектуры web-приложений.

6. Также примерно в половине случаев требуются навыки разработки серверной части web-приложения. Чаще всего это либо технология ASP.NET, либо PHP. Есть и другие, менее распространенные технологии (например, Carpusino или CGI). Для разработки серверной части необходимо знать языки программирования C#.NET, Visual Basic .NET или PHP соответственно (для CGI можно использовать широкий спектр языков программирования).

Также при разработке web-приложений необходимо учитывать то, что пользователи запускают web-приложение из разных браузеров. То есть приложение необходимо проектировать так, чтобы оно корректно отображалось и работало в подавляющем большинстве современных web-браузеров (Opera, Chrome, Firefox).

Теперь рассмотрим подробно язык разметки HTML, язык стилей CSS и язык сценариев JavaScript.

3.2. HTML

Язык HTML был разработан британским ученым Тимом Бернерсом-Ли приблизительно в 1986–1991 годах в стенах Европейского Центра ядерных исследований в Женеве (Швейцария) [7]. HTML создавался как язык для обмена научной и технической документацией, пригодный для использования людьми, не являющимися специалистами в области верстки. HTML успешно справлялся с проблемой сложности SGML (языка, предшествующего HTML) путем определения небольшого набора структурных и семантических элементов – дескрипторов. Дескрипторы также часто называют «тегами». С помощью HTML можно легко создать относительно простой, но красиво оформленный документ. Помимо упрощения структуры документа, в HTML внесена поддержка гипертекста. Мультимедийные возможности были добавлены позже.

Изначально язык HTML был задуман и создан как средство структурирования и форматирования документов без их привязки к средствам воспроизведения (отображения). В идеале текст с разметкой HTML должен был без стилистических и структурных искажений воспроизводиться на оборудовании с различной технической оснащенностью (цветной экран современного компьютера, монохромный экран органайзера, ограниченный по размерам экран мобильного телефона или устройства и программы голосового воспроизведения текстов). Однако современное применение HTML очень далеко от его изначальной задачи. Например, тег <TABLE> предназначен для создания в документах таблиц, но часто используется и для оформления размещения элементов на странице. С течением времени основная идея платформонезависимости языка HTML была принесена в жертву современным потребностям в мультимедийном и графическом оформлении.

Текстовые документы, содержащие разметку на языке HTML (такие документы традиционно имеют расширение .html или .htm), обрабатываются специальными приложениями, которые отображают документ в его форматированном виде. Такие приложения, называемые «браузерами» или «интернет-обозревателями», обычно предоставляют пользователю удобный интерфейс для запроса web-страниц, их просмотра (и вывода на иные внешние устройства) и, при необходимости, отправки введенных пользователем данных на сервер. Наиболее популярными на сегодняшний день (октябрь 2014 года) браузерами

являются Google Chrome (45,70%), Mozilla Firefox (17,48%), Opera (1,42%), Internet Explorer (20,72%) и Safari (10,87%).

3.3. Структура документа HTML

HTML – это теговый язык разметки документов, то есть любой документ на языке HTML представляет собой набор элементов, причем начало и конец каждого элемента обозначается специальными пометками, называемыми тегами [7]. Регистр, в котором набрано имя тега, в HTML значения не имеет. Элементы могут быть пустыми, то есть не содержащими никакого текста и других данных (например, тег перевода строки
). В этом случае обычно не указывается закрывающий тег. Кроме того, элементы могут иметь атрибуты, определяющие какие-либо их свойства (например, размер шрифта для тега). Атрибуты указываются в открывающем теге. Вот пример части разметки HTML-документа:

```
<p>Текст между двумя тегами - открывающим и закрывающим.</p>
  <a href="http://www.example.com">Здесь элемент содержит атри-
бут href.</a>
  А вот пример пустого элемента: <br>
```

Каждый HTML-документ, отвечающий спецификации HTML какой-либо версии, обязан начинаться со строки декларации версии HTML <!DOCTYPE>, которая обычно выглядит примерно так:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

Если эта строка не указана, то добиться корректного отображения документа в браузере становится труднее.

В зависимости от стандарта HTML строка DOCTYPE может выглядеть по-разному. В стандарте HTML версии 5 строка будет выглядеть так:

```
<!DOCTYPE html>
```

Далее обозначается начало и конец документа тегам `<html>` и `</html>` соответственно. Внутри этих тегов должны находиться теги заголовка (`<head></head>`) и тела (`<body></body>`) документа. Пример простейшей HTML страницы:

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
  </head>
  <body>
    <p>
      <b>
        Этот текст будет полужирным,
        <i>а этот - ещё и курсивным</i>
      </b>
    </p>
  </body>
</html>
```

Теги и их параметры нечувствительны к регистру. То есть `` и `` означают одно и то же.

В последних версиях HTML практически у каждого тега огромное число необязательных параметров — обычно не меньше 15. Самые важные из них будут приведены.

Гиперссылки

```
<A HREF="filename" target="_self">название ссылки</A>
```

Параметры:

- Атрибут **HREF** задает значение адреса документа, на который указывает ссылка.
- **filename** — имя файла или адрес Internet, на который необходимо сослаться.

- название ссылки – название гипертекстовой ссылки, которое будет отображаться в браузере, то есть показываться тем, кто зашел на страницу.

- **TARGET** – задает значение окна или фрейма, в котором будет открыт документ, на который указывает ссылка. Возможные значения атрибута:

- **_top** – открытие документа в текущем окне;
- **_blank** – открытие документа в новом окне;
- **_self** – открытие документа в текущем фрейме;
- **_parent** – открытие документа в родительском фрейме.

Значение по умолчанию: **_self**.

Текстовые блоки

<h1> ... </h1>, **<h2> ... </h2>**, ... , **<h6> ... </h6>** – заголовки 1, 2, ... 6 уровня. Используются для выделения частей текста (заголовков 1 уровня – самый крупный, 6 уровня – самый мелкий).

<p> – новый абзац. Можно в конце абзаца поставить **</p>**, но это не обязательно.

**
** – новая строка. Этот тег не закрывается (то есть не существует тега **</br>**)

<hr> – горизонтальная линия

<blockquote> ... </blockquote> – цитата. Обычно текст сдвигается вправо.

<pre> ... </pre> – режим preview (preformatted text). В этом режиме текст заключается в рамку и никак не форматируется (то есть теги, кроме **</pre>**, игнорируются, и переводы строки ставятся там, и только там, где они есть в оригинальном документе).

<div> ... </div> – блок

(обычно используется для применения стилей CSS)

** ... ** – строка

(обычно используется для применения стилей CSS)

Форматирование текста

** ... ** – логическое ударение (обычно отображается курсивным шрифтом)

** ... ** – усиленное логическое ударение (обычно отображается полужирным шрифтом)

<I> ... </I> – выделение текста курсивом

** ... ** – выделение текста жирным шрифтом

<U> ... </U> – подчеркивание текста

<S> ... </S> (или **<STRIKE> ... </STRIKE>**) – зачеркивание текста

<BIG> ... </BIG> — увеличение шрифта

<SMALL> ... </SMALL> – уменьшение шрифта

<BLINK> ... </BLINK> – мигающий текст. Этот тег не работает в браузере Internet Explorer версий 5 и ниже без применения JavaScript

<MARQUEE> ... </MARQUEE> – сдвигающийся по экрану текст.

_{...} – подстрочный текст. Например, н**₂**о создаст текст

H₂O.

^{...} – надстрочный текст. Например, E=mc**²** создаст текст

E=mc².

** ... ** – задание параметров шрифта. У этого тега есть следующие параметры:

- **COLOR=color** – задание цвета. Цвет может быть задан в шестнадцатеричной форме как #rrggbb (первые 2 шестнадцатеричные цифры задают красную компоненту, следующие 2 – зеленую, последние 2 – синюю) или названием.

- **FACE=шрифт** задание гарнитуры шрифта

- **SIZE=размер** задание размера шрифта. Размер от 1 до 7: стандартный по умолчанию 3. Есть много способов изменить стандартный размер.

• **SIZE=+изменение** или **SIZE=-изменение** – изменение размера шрифта от стандартного. Например, +2 означает размер на 2 больше стандартного.

Так, например,

Сигналом к началу атаки являются

```
<U>три</U> <FONT SIZE="+2">больших</FONT> <FONT  
COLOR="green">зелёных</FONT> свистка.
```

создаст текст

Сигналом к началу атаки являются три **БОЛЬШИХ** **зелёных** свистка.

Списки

```
<UL>  
  <LI> первый элемент </LI>  
  <LI> второй элемент </LI>  
  <LI> третий элемент </LI>  
</UL>
```

создаёт список

- **первый элемент**
- **второй элемент**
- **третий элемент**

Если вместо **** (Unordered List – нумерованный список) поставить **** (Ordered List – нумерованный список), список получится нумерованным:

1. **первый элемент**
2. **второй элемент**
3. **третий элемент**

У этих тегов есть параметры:

```
type = "тип"
```

где тип – форма: в **** – символов

- **square** – квадрат
- **round** – окружность
- **disk** – круг: по умолчанию

а в **** – цифр или букв

- **A** или **a** (латинскими буквами) – буквенный список: соответственно заглавными или строчными буквами
- **I** или **i** – римские цифры: соответственно заглавными или строчными буквами
- **1** – арабские цифры: по умолчанию

Например:

```
<ol type="i">  
<li> Первое </li>  
<li> Второе </li>  
<li> И т.д. </li>  
</ol>
```

В браузере будет выведено следующее

```
i.Первое  
ii.Второе  
iii.И т. д.
```

Параметр **start="начало"** (только для ****), определяющий начало нового отсчета: например, если нужно не 1, 2, 3, а 24, 25, 26

Например:

```
<ol start="24">
<li> Двадцать четыре </li>
<li> Двадцать пять </li>
<li> И т.д. </li>
```

В браузере будет выведено следующее:

```
24.Двадцать четыре
25.Двадцать пять
26.И т. д.
```

Далее, для тега `` параметр `value="следующий"` – если необходимо перескочить с одной цифры на другую: например, не 1, 2, 3, 4, а 1, 2, 22, 23.

Например,

```
<ol>
<li> Один </li>
<li> Два </li>
<li> value="22"> Двадцать два </li>
<li> Двадцать три </li>
```

В браузере будет выглядеть так:

```
1.Один
2.Два
22.Двадцать два
23.Двадцать три
```

Наконец, третьим, и последним, списком является список определений:

Например:

```
<DL>
<DT> География </DT> <DD> наука о Земле </DD>
<DT> Астрономия </DT> <DD> наука о Вселенной </DD>
```

```
<DT> Космология </DT> <DD> наука об эволюции Вселенной </DD>
</DL>
```

В браузере будет выглядеть так:

```
География
  наука о Земле
Астрономия
  наука о Вселенной
Космология
  наука об эволюции Вселенной
```

Теги ``, `<DT>`, `<DD>` можно не закрывать.

Объекты

- **EMBED** – вставка различных объектов: не-HTML документов и медиа-файлов
- **APPLET** – вставка Java-апплетов;
- **SCRIPT** – вставка скриптов.

Изображения

IMG – вставка изображения. Этот тег не закрывается. Параметры тега:

- **SRC** – имя или URL;
- **ALT** – альтернативное имя (отобразится, если в браузере запретить отображать картинки);
- **TITLE** – краткое описание изображения (отобразится при наведении курсора на картинку);
- **WIDTH, HEIGHT** – размеры (если не совпадают с истинными размерами картинки, то изображение «растянется» или «сожмется»);
- **ALIGN** – задает параметры обтекания текстом (top, middle, bottom, left, right);
- **VSPACE, HSPACE** – задают размеры вертикального и горизонтального пространства вокруг изображения.

Пример:

```
<IMG SRC=url ALT="текст" TITLE="текст" WIDTH="размер (пикс,  
%) " HEIGHT="размер (пикс, %) ">
```

Изображение можно сделать ссылкой:

```
<A HREF=url ><IMG SRC=url></A>
```

Карта изображений

<MAP>...</MAP> – создание карты изображений позволяющей хранить в одном изображении несколько ссылок.

Пример:

```
<IMG width="500" height="200" usemap="#somemap" src="url">  
<MAP name="somemap">  
    <AREA shape="rect" coords="6, 7, 140, 196" href="url1">  
    <AREA shape="circle" coords="239, 98, 92" href="url2">  
    <AREA shape="polygon" coords="386,16, 344,56, 350,189,  
385,132, 489,190, 496,74" href="url3">  
</MAP>
```

Таблицы

- **TABLE** – создание таблицы. Параметры тега:
 - **BORDER** – толщина разделительных линий в таблице;
 - **CELLSPACING** – расстояние между клетками.
- **CAPTION** – заголовок таблицы (этот тег необязателен);
- **TR** – строка таблицы;
- **TH** – заголовок столбца таблицы (этот тег необязателен);
- **TD** – ячейка таблицы;
- **height** – высота таблицы;
- **width** – ширина таблицы.

Так, например, следующий html-код

```
<TABLE BORDER="1" CELLSPACING="0">
  <CAPTION> Улов форели в пруду </CAPTION>
  <TH> Год </TH>
  <TH> Улов </TH>
  <TR>
    <TD> 1997 </TD>
    <TD> 214 </TD>
  </TR>
  <TR>
    <TD> 1998 </TD>
    <TD> 216 </TD>
  </TR>
  <TR>
    <TD> 1999 </TD>
    <TD> 207 </TD>
  </TR>
</TABLE>
```

создаст таблицу:

Улов форели в пруду

Год	Улов
1997	214
1998	216
1999	207

У тега **TABLE** есть ещё параметр **CELLPADDING**. Он определяет расстояние в пикселях между рамкой ячейки и её содержимым. Например:

```
<TABLE BORDER="1" CELLSPACING="0" CELLPADDING="5">
```

Другой параметр тегов **TABLE**, **TR**, **TH**, **TD** – **ALIGN**. Он определяет выравнивание. Возможные значения – **center** (по центру), **left** (по левому краю), **right** (по правому краю).

Параметр **ALIGN** в **TD** или **TH** определяет выравнивание для содержимого внутри данной ячейки, в **TR** – для содержимого всех ячеек строки, **TABLE** – для самой таблицы на страничке. Для каждой ячейки берется выравнивание из **TD** или **TH**, если оно не задано – из **TR**, если и оно не задано – по центру для **TH** или по левому краю для **TD**.

Например:

```
<TABLE BORDER="1" CELLSPACING="0" ALIGN="center">
<CAPTION> Улов форели в пруду </CAPTION>
<TH> Год </TH>
<TH> Улов </TH>
<TR>
<TD> 1997 </TD> <TD ALIGN="CENTER"> 214 </TD>
```

Незакрытые теги **TD**, **TR** и **TH** приводят к некорректному отображению, особенно при работе со вложенными таблицами.

Формы

Создание форм в HTML достаточно трудоемко. Здесь приводятся только названия основных тегов.

- **FORM** – создание формы;
- **INPUT** – элемент ввода (может иметь разные функции – от ввода текста до отправки формы);
- **TEXTAREA** – текстовая область (многострочное поле для ввода текста);
- **SELECT** – список (обычно в виде выпадающего меню);
- **OPTION** – пункт списка.

Символы

Чтобы создать символ, необязательно искать его на клавиатуре. Можно просто набрать тег (например, чтобы получить ¢, надо набрать следующее: **¢**). Пробел – ** **; , ударение — **́** и т. д.

Основные символы приведены в табл. 3.3.1:

Таблица 3.3.1

Коды основных символов

Код	Символ	Смысл
<code>&lt;</code>	<	меньше
<code>&gt;</code>	>	больше
<code>&amp;</code>	&	амперсанд
<code>&nbsp;</code>		неразрывный пробел (на этом пробеле не переводится строка)
<code>&sect;</code>	§	параграф
<code>&#8470;</code>	№	номер
<code>&copy;</code>	©	копирайт (copyright)
<code>&reg;</code>	®	зарегистрированный товарный знак (registered trademark)
<code>&#153;</code>	™	товарный знак (trademark)
<code>&deg;</code>	°	градусы
<code>&laquo;</code>	«	открывающая кавычка в русском языке
<code>&raquo;</code>	»	закрывающая кавычка в русском языке
<code>&hellip;</code>	...	многоточие
<code>&mdash;</code>	—	тире
<code>&#149;</code>	•	жирная точка
<code>&plusmn;</code>	±	плюс-минус
<code>&minus;</code>	–	минус

3.4. CSS

CSS (англ. Cascading Style Sheets – каскадные таблицы стилей) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки [7].

Преимущественно используется как средство описания, оформления внешнего вида web-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например к SVG или XUL.

CSS используется создателями web-страниц для задания цветов, шрифтов, расположения отдельных блоков и других аспектов представления внешнего вида этих web-страниц. Основной целью разработки CSS являлось разделение описания логической структуры web-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой web-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом. Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана), или при выводе устройствами, использующими шрифт Брайля.

Правила CSS пишутся на формальном языке CSS и располагаются в таблицах стилей, то есть таблицы стилей содержат в себе правила CSS. Эти таблицы стилей могут располагаться как в самом web-документе, внешний вид которого они описывают, так и в отдельных файлах, имеющих формат CSS. (По сути, формат CSS – это обычный текстовый файл. В файле .css не содержится ничего, кроме перечня правил CSS и комментариев к ним.)

Таким образом, таблицы стилей могут быть подключены, внедрены в описываемый ими web-документ четырьмя различными способами.

1 способ – когда таблица стилей находится в отдельном файле, она может быть подключена к web-документу посредством тега `<link>`, располагающегося в этом документе между тегами `<head>` и `</head>`. (Тег `<link>` будет иметь

атрибут `href`, имеющий значением адрес этой таблицы стилей). Все правила этой таблицы действуют на протяжении всего документа:

```
<!DOCTYPE html>
<html>
  <head>
    .....
    <link rel="stylesheet" href="Style.css">
  </head>
  <body>
    .....
  </body>
</html>
```

2 способ – когда таблица стилей находится в отдельном файле, она может быть подключена к web-документу посредством директивы `@import`, располагающейся в этом документе между тегами `<style>` и `</style>` (которые, в свою очередь, располагаются в этом документе между тегами `<head>` и `</head>`) сразу после тега `<style>`, которая также указывает (в своих скобках, после слова `url`) на адрес этой таблицы стилей. Все правила этой таблицы действуют на протяжении всего документа:

```
<!DOCTYPE html>
<html>
  <head>
    .....
    <style media="all">
      @import url(style.css);
    </style>
  </head>
</html>
```

3 способ – когда таблица стилей описана в самом документе, она может располагаться в нем между тегами `<style>` и `</style>` (которые, в свою очередь, располагаются в этом документе между тегами `<head>` и `</head>`). Все правила этой таблицы действуют на протяжении всего документа:

```

<!DOCTYPE html>
<html>
  <head>
    .....
    <style>
      body {
        color: red;
      }
    </style>
  </head>
  <body>
    .....
  </body>
</html>

```

4 способ – когда таблица стилей описана в самом документе, она может располагаться в нем в теле какого-то отдельного тега (посредством его атрибута **style**) этого документа. Все правила этой таблицы действуют только на содержимое этого тега:

```

<!DOCTYPE>
<html>
  <head>
    .....
  </head>
  <body>
    <p style="font-size: 20px; color: green; font-family:
arial, helvetica, sans-serif">
    .....
  </p>
  </body>
</html>

```

Принято считать, что в первых двух случаях говорят к документу применены внешние таблицы стилей, а во вторых двух случаях – внутренние таблицы стилей.

HTML-документы строятся на основании иерархии элементов, которая может быть наглядно представлена в древовидной форме. Элементы HTML

друг для друга могут быть родительскими, дочерними, элементами-предками, элементами-потомками, сестринскими.

Элемент является родителем другого элемента, если в иерархической структуре документа он находится сразу, непосредственно над этим элементом. Элемент является предком другого элемента, если в иерархической структуре документа он находится где-то выше этого элемента.

Например, в документе присутствуют два абзаца **p**, включающие в себя шрифт с полужирным начертанием **b**. Тогда элементы **b** будут дочерними элементами своих родительских элементов **p**, и потомками своих предков **body**. В свою очередь, для элементов **p** элемент **body** будет являться только родителем. И кроме того, эти два элемента **p** будут являться сестринскими элементами, как имеющими одного и того же родителя – **body**.

В CSS могут задаваться при помощи селекторов не только одиночные элементы, но и элементы, являющиеся потомками, дочерними или сестринскими элементами других элементов (см. подраздел «виды селекторов»).

В первых трех случаях подключения таблицы CSS к документу каждое правило CSS из таблицы стилей имеет две основные части – селектор и блок объявлений. Селектор, расположенный в левой части правила, определяет, на какие части документа распространяется правило. Блок объявлений располагается в правой части правила. Он помещается в фигурные скобки, и, в свою очередь, состоит из одного или более объявлений, разделенных знаком «;». Каждое объявление представляет собой сочетание свойства CSS и значения, разделенных знаком «: ». Селекторы могут группироваться в одной строке через запятую. В таком случае свойство применяется к каждому из них:

```
селектор, селектор {  
    свойство: значение;  
    свойство: значение;  
    свойство: значение;  
}
```

В четвёртом случае подключения таблицы CSS к документу (см. список) правило CSS (являющееся значением атрибута **style** тега, на который оно действует) представляет собой перечень объявлений («свойство CSS : значение»), разделенных знаком «;».

Виды селекторов

Универсальный селектор

```
* {  
  margin: 0;  
  padding: 0;  
}
```

Селектор элементов

```
p {  
  font-family: arial, helvetica, sans-serif;  
}
```

Селектор классов

```
.note {  
  color: red;  
  background-color: yellow;  
  font-weight: bold;  
}
```

Селектор идентификаторов

```
#paragraph1 {  
  margin: 0px;  
}
```

Селектор атрибутов

```
a[href="http://www.somesite.com"] {  
  font-weight: bold;  
}
```

Селектор потомков (контекстный селектор)

```
div#paragraph1 p.note {
    color: red;
}
```

Селектор дочерних элементов

```
p.note > b {
    color: green
}
```

Селектор сестринских элементов

```
h1 + p {
    font-size: 24pt;
}
```

Селектор псевдоклассов

```
a:active {
```

Селектор псевдоэлементов

```
p::first-letter {
    font-size: 32px;
}
```

Класс или идентификатор может быть присвоен какому-нибудь элементу (тегу) HTML посредством атрибутов **class** или **id** этого элемента (тега):

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
    <title>
      Селекторы классов и идентификаторов
    <style>
      p.Big {
```

```

        font-family: arial, helvetica, sans-serif;
        color: maroon;
    }
    div#First {
        background-color: silver;
    }
</style>
</head>
<body>
    .....
    <div id="First">
        .....
    </div>
    <p class="Big">
        .....
    </p>
</body>
</html>

```

Основное отличие между классами элементов и идентификаторами элементов в том, что в документе какой-нибудь класс может быть присвоен сразу нескольким элементам, а идентификатор – только одному. Еще одно отличие в том, что могут существовать множественные классы (когда класс элемента состоит из нескольких слов, разделенных пробелами). Для идентификаторов такое невозможно.

Важно отметить следующее отличие идентификатора от класса: идентификаторы широко используются в JavaScript для нахождения уникального элемента в документе.

Имена классов и идентификаторов, в отличие от названий тегов и их атрибутов, чувствительны к регистру ввода букв.

Свойства классов и идентификаторов задаются с помощью соответствующих селекторов. Причем может быть задано как свойство класса в целом (в таком случае селектор начинается со знака «.»), или свойство идентификатора самого по себе (в таком случае селектор начинается со знака «#»), так и свойство какого-нибудь элемента этого класса или с этим идентификатором.

В CSS помимо классов, задаваемых автором страницы, существует также ограниченный набор так называемых псевдоклассов, описывающих вид гиперс-

сылок с определенным состоянием в документе, вид элемента, на котором находится фокус ввода, а также вид элементов, являющихся первыми дочерними элементами других элементов. Также в CSS существует четыре так называемых псевдоэлемента: первая буква, первая строка, применение специальных стилей до и после элемента.

Пример таблицы стилей (в таком виде она может быть либо размещена в отдельном файле .css, либо же обрамлена тегами <style> и размещена в «шапке» той самой веб-страницы, на которую она действует):

```
p {
    font-family: arial, helvetica, sans-serif;
}
h2 {
    font-size: 20pt;
    color: red;
    background: white;
}
.note {
    color: red;
    background-color: yellow;
    font-weight: bold;
}
p#paragraph1 {
    padding-left: 10px;
}
a:hover {
    text-decoration: none;
}
#news p {
    color: blue;
}
[type="button"] {
    background-color: green;
}
```

Здесь приведено семь правил CSS с селекторами `p`, `h2`, `.note`, `p#paragraph1`, `a:hover`, `#news p` и `[type="button"]`.

1. Первое правило присвоено HTML-элементу **p** (абзацу) – назначен стиль. Абзацы будут отображаться шрифтом **Arial**, или, если такой шрифт недоступен, тогда **Helvetica** или **Sans-serif**, иначе – другим шрифтом этого семейства.

2. Второе правило присвоено HTML-элементу **h2** (заголовку второго уровня). Заголовок второго уровня будет отображаться красным на белом фоне с увеличенным размером.

3. Третье правило будет применено к любому элементу, атрибут **class** которого равен «**note**». Например, к параграфу: `<p class="note">` Этот абзац будет выведен полужирным шрифтом красного цвета на желтом фоне. `</p>`

4. Четвертое правило будет применяться только к элементу **p**, атрибут **id** которого равен **paragraph1**. Такой элемент будет иметь внешний отступ в 10 пикселей (**padding**).

5. Пятое правило определяет стиль **hover** для элементов **a** – гиперссылок. По умолчанию, в большинстве браузеров текст элементов **a** подчеркивается. Это правило уберёт подчеркивание, когда указатель мыши находится над этими элементами.

6. Шестое правило, применяется для элементов **p**, которые находятся внутри какого-либо элемента с атрибутом **id**, равным «**news**» (**#news p** – это типичный случай селектора потомков, см. 5-й пункт списка выше).

7. Седьмое правило применяется для всех элементов, у которых атрибут **type** равен **button**. Например, это правило будет применено к элементу `<input type="button">` (обычная кнопка), изменив его цвет фона на зеленый.

3.5. JavaScript

JavaScript – прототипно-ориентированный сценарный язык программирования. Является диалектом языка ECMAScript [11].

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности web-страницам.

Для добавления JavaScript-кода на страницу, можно использовать теги `<script></script>`, которые рекомендуется, но не обязательно, помещать внутри контейнера `<head>`. Контейнеров `<script>` в одном документе может

быть сколько угодно. Атрибут «`type='text/javascript'`» указывать необязательно, так как по умолчанию стоит `javascript`.

Скрипт, выводящий модальное окно с классической надписью «Hello, World!» внутри браузера:

```
<script type="text/javascript">
    alert('Hello, World!');
</script>
```

Расположение скрипта внутри тега

Спецификация HTML описывает набор атрибутов, используемых для задания обработчиков событий. Пример использования:

```
<a href="delete.php" onclick="return confirm('Вы уверены? '); ">
    Удалить
</a>
```

В приведённом примере при нажатии на ссылку функция `confirm('Вы уверены? ');` вызывает модальное окно с надписью «Вы уверены?», а `return false;` блокирует переход по ссылке. Разумеется, этот код будет работать только в том случае, если в браузере есть и включена поддержка JavaScript, иначе переход по ссылке произойдет без предупреждения.

Использование кода JavaScript в контексте разметки страницы является плохим стилем программирования.

Вынесение в отдельный файл

Есть и третья возможность подключения JavaScript – написать скрипт в отдельном файле, а потом подключить его с помощью конструкции

```
<head>
    <script type="text/javascript"
src="http://Путь_к_файлу_со_скриптом">
    </script>
</head>
```

Элемент **script**, широко используемый для подключения к странице JavaScript, имеет несколько атрибутов.

- Обязательный атрибут **type** для указания MIME-типа содержимого.

• Известно, что использование «text» в качестве типа верхнего уровня данного типа содержимого проблематично; **text/javascript** и **text/ecmascript** считаются «устаревшими». Использование экспериментальных и незарегистрированных медиатипов, таких как перечисленные в части выше, не приветствуется.

- Медиатипы

- **application/javascript**,
- **application/ecmascript**,

которые также определяются в этом документе, предназначены для практического использования, им следует отдавать предпочтение.

• необязательный атрибут **src**, принимающий в качестве значения адрес к файлу со скриптом.

• необязательный атрибут **charset**, используемый вместе с **src** для указания используемой кодировки внешнего файла.

• необязательный атрибут **defer** указывает, что получение скрипта происходит асинхронно, но выполнение следует отложить до тех пор, пока страница не будет загружена целиком.

• необязательный атрибут **async** указывает, что получение скрипта происходит асинхронно, а выполнение будет произведено сразу по завершению скачивания. Очередность выполнения скриптов не гарантируется.

В JavaScript поддерживаются пять примитивных типов данных:

- числовой (англ. Number);
- строковый (англ. String);
- логический (англ. Boolean);
- нулевой (англ. Null);
- неопределенный (англ. Undefined).

Числовой тип данных в ECMAScript соответствует 64-битному формату чисел с плавающей запятой.

Также в языке имеется «составной» тип данных:

- объектный (англ. Object).

Инструкции JavaScript

Блок операторов

```
{ [<инструкции>] }
```

Объявление переменной

```
var <список объявления переменных>
```

Выражение

```
[строка до {, function}] инструкция
```

Условие

```
if (<инструкция>) <выражение>[ else <выражение>]
```

Цикл

```
do <выражение> while (<инструкция>)
```

```
while (<инструкция>) <выражение>
```

```
for ([<инструкция до начала>]; [<инструкция>]; [<инструкция>])
```

```
<выражение>
```

```
for (<объявление переменных>; [<инструкция>]; [<инструкция>])
```

```
<выражение>
```

```
for (<lvalue-выражение> in <инструкция>) <выражение>
```

```
for (<объявление переменных> in <инструкция>) <выражение>
```

Обработка исключений

```
try <блок> catch (<идентификатор>) <блок>
```

```
try <блок> finally <блок>
```

```
try <блок> catch (<идентификатор>) <блок> finally <блок>
```


Объявления переменных

Переменные определяются с помощью ключевого слова **var**. При объявлении переменная помещается в область видимости, соответствующую функции, в которой она объявляется. Если переменная объявляется вне функций, она помещается в глобальную область видимости. Создание переменной происходит при получении управления функцией с ее объявлением. Или программой, если переменная глобальна. При создании переменной в JavaScript она приобретает значение **undefined**. Если переменная объявлена с инициализацией, инициализация происходит не в момент создания переменной, а при выполнении строки с инструкцией **var**.

Операторы

Список операторов JavaScript в порядке убывания их приоритета:

- **.** (доступ к свойству), **[]** (доступ к свойству), **()** (вызов функции), **new** (создание нового объекта);
- **++** (инкремент), **--** (декремент), **-** (унарный минус), **+** (унарный плюс), **~** (поразрядное дополнение), **!** (логическое дополнение), **delete** (удаление свойства), **typeof** (определение примитивного типа данных), **void** (возврат неопределенного значения);
- ***** (умножение), **/** (деление), **%** (остаток от деления);
- **+** (сложение), **-** (вычитание), **+** (конкатенация строк);
- **<<** (сдвиг влево), **>>** (сдвиг вправо с расширением знакового разряда), **>>>** (сдвиг вправо с дополнением нулями);
- **<** (меньше), **<=** (меньше или равно), **>** (больше), **>=** (больше или равно), **instanceof** (проверка типа объекта), **in** (проверка наличия свойства);
- **==** (проверка на равенство), **!=** (проверка на неравенство), **===** (проверка на идентичность), **!==** (проверка на неидентичность);
- **=** (присваивание), ***=**, **/=**, **+=**, **-=**, **<<=**, **>>=**, **>>>=**, **&=**, **^=**, **|=** (присваивание с операцией).

В тексте программы именованную функцию в JavaScript можно определить одним из следующих способов:

```

// объявление функции
function sum(arg1, arg2) {
    return arg1 + arg2;
}

// задание функции с помощью инструкции
var sum2 = function(arg1, arg2) {
    return arg1 + arg2;
};

// задание функции с использованием объектной формы записи
var sum3 = new Function("arg1", "arg2", "return arg1 +
arg2;");

```

3.6. Средства создания web-приложений на стороне сервера

Как уже говорилось ранее, основными технологиями создания web-приложений на стороне сервера являются ASP.NET и PHP. Далее будет рассмотрена технология ASP.NET.

ASP.NET(Active Server Pages) – технология создания web-приложений и web-сервисов от компании Майкрософт. Она является составной частью платформы Microsoft .NET и развитием более старой технологии Microsoft ASP.

ASP.NET внешне во многом сохраняет схожесть с более старой технологией ASP, что позволяет разработчикам относительно легко перейти на ASP.NET. В то же время внутреннее устройство ASP.NET существенно отличается от ASP, поскольку она основана на платформе .NET и, следовательно, использует все новые возможности, предоставляемые этой платформой.

Хотя ASP.NET берет свое название от старой технологии Microsoft ASP, она значительно от нее отличается. Microsoft полностью перестроила ASP.NET, основываясь на Common Language Runtime (CLR), которая является основой всех приложений Microsoft .NET. Разработчики могут писать код для ASP.NET, используя практически любые языки программирования, входящие в комплект .NET Framework (C#.NET, Visual Basic.NET, Delphi Prism). ASP.NET имеет преимущество в скорости по сравнению со скриптовыми технологиями, так как при первом обращении код компилируется и помещается в специальный кэш, и

впоследствии только исполняется, не требуя временных затрат на синтаксический анализ, оптимизацию, и т.д.

Создание серверной части происходит в среде разработки Visual Studio. Использование этой среды разработки упрощает создание web-приложений. Например, структура web-приложения создается автоматически, выводятся всплывающие подсказки названий тегов HTML, свойств CSS, операторов и переменных JavaScript и языков программирования серверной части web-приложения. Также эта среда позволяет вести разработку на языках программирования (C#.NET, VB.NET, Delphi Prism) с грамотной организацией компоновки кода и широким спектром инструментов для отладки кода и оптимизации работы приложения.

Контрольные вопросы

1. Какие навыки необходимы для создания web-приложений?
2. Что такое HTML?
3. С какой декларации должен начинаться любой документ HTML?
4. Какие тэги предназначены для форматирования текста?
5. Как создавать различные виды списков при помощи HTML?
6. Перечислите основные виды селекторов CSS.
7. Как включить скрипт JavaScript в документ HTML?
8. Как организовать цикл в JavaScript?
9. Какие существуют платформы для создания web-приложений?

Глава 4. ГИС И INTERNET. КАРТОГРАФИЧЕСКИЕ СЕРВИСЫ И ГЕОПОРТАЛЫ

4.1. Определение и задачи web-картографии. Миграция ГИС в Internet

В последние годы развитие Internet-технологий все в большей степени стало позволять пользователям Сети использовать новые средства доставки информации, в том числе и пространственной [12].

Web-картография – это область компьютерных технологий, связанная с доставкой пространственных данных конечному пользователю. Безусловно, web-картография является одним из направлений геоинформационных технологий в целом. Основными задачами web-картографии являются:

- Визуализация существующей информации – пространственное представление информации.
- Облегчение работы с пространственной информацией в сети Internet, поиск, прокладка маршрутов и другие услуги, основанные на местоположении объектов (LBS – location based services).

Несмотря на то, что для большинства пользователей Интернет «рождение» web-картографии (да и в целом, широкое осознание возможностей пространственных технологий) напрямую ассоциируется с выходом на рынок продуктов компании Google в 2005 году, в реальности картографические web-приложения появились значительно раньше. Фактически, датой рождения web-картографии можно считать 1993 год, когда впервые был запущен web-сервис Xerox PARC Map Viewer, позволявший пользователям в интерактивном режиме отправлять запросы из браузера к серверу и получать фрагменты карт в формате GIF. Именно это приложение и его функциональная концепция стала родоначальником большинства более поздних версий web-ГИС.

На ранних этапах становления web-картографии (до 1998 года) отличительной особенностью большинства сервисов была их локальность и узкая тематическая направленность, что серьезно ограничивало круг потенциальных пользователей таких сервисов. Один из первых решительных шагов по популяризации web-ГИС был сделан в 1998 году в Великобритании, когда был запущен сайт (успешно работающий и до сих пор) www.streetmap.co.uk. Этот сервис, в отличие от своих предшественников, не был ориентирован на визуализа-

цию локального участка земной поверхности и насыщение ее узкотематической информацией. Напротив, создатели сервиса пошли иным путем – они выложили простейшую топографическую информацию, но покрыли всю территорию Великобритании. Именно этот подход и предопределил бешеную популярность сервиса: тысячи людей могли без особого труда определить месторасположение торгового центра, дома и любого другого объекта, зная всего лишь его почтовый индекс, а затем послать готовую схему проезда на печать. 1998 год также ознаменовывается появлением специального «out of the box» программного обеспечения, позволяющего любому пользователю сети создавать собственные web-ГИС – Mapserver. Примерно в это же время четкое понимание перспектив веб приводит к тому, что крупные компании производители ПО ГИС (ESRI, Intergraph) принимают решение о разработке специальных коммерческих приложений для создания специального программного обеспечения для web-ГИС. Однако, в силу ограниченной пропускной способности, мало кто мог представить до каких масштабов может разрастись web-картография.

Переломным моментом стал 2005 год, когда компания Google практически одновременно запустила два глобальных картографических сервиса – Google-Maps и Google-Earth. Ключевым словом в предыдущем предложении является слово «глобальный» – так как ни один из запущенных ранее сервисов не мог похвастаться столь «широкой» географией. Кроме того, был использован принципиально новый подход в организации самого сервиса: вместо классического подхода, в котором пользователь посылает запрос на сервер, ждет обработки и получает обратно сгенерированную «на лету» картинку, все данные были подготовлены и обработаны заранее, что в сочетании с технологиями AJAX, позволило добиться необычно быстрой работы с картами и «бесшовности» данных при навигации.

Последующие годы характеризуются поистине колоссальным интересом к web-картографии и ее возможностям в целом, а также значительным ростом числа сервисов в той или иной форме использующих картографические web-технологии. Наиболее важными тенденциями, наблюдаемыми в последние полтора-два года, являются появление большого числа бесплатных проектов, реализующих концепцию предоставления предобработанных данных; увеличение возможностей персонализации сервисов; возможности по интеграции соб-

ственных данных с существующими сервисами; глобальность сервисов; также наблюдается все большая интеграция таких служб в повседневную жизнь.

4.2. Основные виды приложений, имеющих отношение к web-картографии

Разнообразие современных механизмов для создание веб-картографических приложений велико [12]:

- Виртуальные глобусы (Google Maps, Google Earth, Virtual Earth, ArcGIS Explorer) – простое и эффективное средство быстрого создания и публикации данных в Internet. Эта категория инструментов характеризуется массовым распространением и быстрой доставкой данных пользователям. Они могут использоваться в качестве клиента как web-браузер, так и отдельное приложение. Как правило, они включают доступ по умолчанию к некой «подложке» – базе данных, что является одновременно их большим плюсом и не меньшим минусом, так как сменить эту подложку в большинстве случаев нельзя. Также, как правило, этим инструментам свойственны проблемы при работе с большими объемами пользовательских данных, настраиваемостью, элементарным анализом (обрезка, пересечение слоев данных).

- Пользовательские ГИС (ArcGIS, Mapinfo, QGIS, gvSIG) – большая и сложная категория, тесно связанная с web-картографией. Как правило, пользовательские ГИС, с одной стороны, играют роль клиентов, работающих с данными поставляемыми картографическими web-серверами, а с другой, в них осуществляется массовая подготовка и анализ данных перед публикацией их в web.

- Картографические web-серверы (MapServer, GeoServer, OpenLayers и др.) – целое семейство продуктов свободного и проприетарного характера, предназначенных для быстрой публикация пользовательских данных в web. Эти инструменты позволяют создать интерфейс нужной сложности, интегрировать сервис с базой данных, поддерживающей классы пространственных данных (PostgreSQL, SQL Server, MySQL, ArcSDE). Главное отличие подобных систем от Google Maps – полный контроль над программным обеспечением и самими данными, однако взамен приходится расплачиваться большей сложно-

стью установки и настройки, часто требующей хотя бы начальных знаний языков программирования (JavaScript, PHP) и основ администрирования.

Особый тип инструментов, появившихся совсем недавно – пользовательские ГИС, интегрированные с виртуальными глобусами, которые играют роль одного из способов представления данных. Примером подобного приложения является расширение для ArcGIS, разработанное Brian Flood и позволяющее интегрировать его с Virtual Earth.

4.3. Ключевые организации

Скорость внедрения ГИС в Интернет, как и развитие компьютерных технологий в целом, является достаточно бесконтрольным процессом. В то же время, в геоинформационном сообществе существует целый ряд ключевых организаций, которые различными методами регулируют деятельность разработчиков. Наиболее удобным способом такого «контроля» в современном сообществе становится внедрение и продвижение определенных стандартов, протоколов и RFC разработки. На рынке web-ГИС присутствуют различные виды организаций, наиболее активных представителей которых можно классифицировать следующим образом [12]:

- Ассоциации и регулирующие организации: OGC (Open Geospatial Consortium) – некоммерческая организация, занимающаяся поддержкой и продвижением стандартов и архитектур, связанных с пространственными данными. Членами консорциума являются наиболее значительные компании, чья деятельность связана с пространственными данными. В стратегическую категорию членов организации входят USGS, NASA, NGA, главными членами являются ESRI, Google, Microsoft и другие.

- Opensource группы: OSGeo – также некоммерческая организация, созданная специально для поддержки проектов с открытым кодом, как правило, поддерживаемых открытыми сообществами специалистов. Проекты, проходящие инкубацию в OSGeo, получают и место в совете организации. Президентом организации является Frank Warmerdam, создатель и один из основных авторов GDAL\OGR.

- Профессиональные ГИС: ESRI – корпорация, специализирующаяся на ГИС, и до недавних пор не имевшая особенных конкурентов. Последнее время ESRI активно пытается усилить свои пошатнувшиеся позиции на рынке web-картографии, развивая ArcGIS Server. Несмотря на наличие на рынке и других игроков (Mapinfo, Autodesk), фактически является стандартом де-факто.

- Интернет-гиганты: Google и ее компаньоны – группа компаний (включая также Microsoft, Yahoo и Yandex), рассматривающая web-картографические проекты как один из способов размещения рекламы и активно развивающих онлайн присутствие. В основном популярность достигается за счет предоставления широкому кругу пользователей доступа к ранее недоступным базам данных космической съемки высокого разрешения и сопутствующих технологий маршрутизации и поиска.

- Генераторы данных: поставщики пространственных данных, как правило, коммерческих, например цифровой картографической информации (Navteq/Teleatlas), спутниковых данных (GeoEye, DigitalGlobe). В последнее время в этом секторе появляются и некоммерческие участники (OpenStreetMap).

4.4. Открытые картографические стандарты

В настоящее время общие принципы и стандарты в области разработки программного обеспечения, предоставляющего картографические веб-сервисы, разрабатываются и декларируются международной некоммерческой организацией Open GIS consortium (OGC) [8; 12]. OGC была основана 25 сентября 1994 года и на момент создания включала только 8 членов. С 1992 по 2004 годы их число возросло с 8 до 250, и на сегодняшний день в OGS представлены наиболее крупные коммерческие, академические и государственные организации, занимающиеся разработкой или исследованиями в области развития и разработки геоинформационного или IT ПО (в том числе такие крупнейшие корпорации как Boeing, Oracle, ESRI, MapInfo, Intergraph, Google (членство с весны 2006 года) и многие другие).

Во многом деятельность OGC в области геоинформационных систем можно сравнить с деятельностью W3C по стандартизации процессов и технологий во всемирной сети. Так, одной из первых разработок OGC были стандарты по

созданию GML – Geography Markup Language – языка группы XML, предназначенного для описания географически привязанных объектов. Стоит отметить, что GML может быть использован и как язык моделирования, и как язык передачи пространственной информации в сети.

Спецификации OGC предлагают следующие типы картографических web-сервисов.

Web Map Service

- определяет параметры запроса и предоставления картографической (пространственной) информации в виде графического изображения или набора объектов;
- описывает условия получения и предоставления информации о содержимом карты (например, свойства объекта в определенном месте на карте);
- характеризует условия получения и предоставления информации о возможностях сервера по представлению различных типов картографической информации.

Web Feature Service

- определяет условия получения и обновления пространственно привязанной информации клиентской частью приложения с использованием Geography Markup Language (GML);
- описывает стандартный интерфейс доступа и манипуляции с географическими объектами с помощью HTTP-протокола.

Web Coverage Service

- расширяет возможности WMS для предоставления растровой географической информации;
- в отличие от WMS, coverage service разрабатывается для представления свойств и значений в каждой конкретной точке географического пространства (а не для создания готовых изображений), а также позволяет проводить интерпретацию данных не на сервере, а на клиентской части приложения.

Однако рост популярности картографических web-сервисов порождает все большее число различных модификаций существующих языков и стандартов

передачи пространственных данных. По-видимому, уже в ближайшем будущем OGC придется включить в сферу своих интересов рассмотрение специализированных языков программирования, форматов передачи данных и стандартов, их описывающих.

Web Processing Service

Эта спецификация описывает правила для входящих и исходящих данных (запросов и ответов на них) для сервисов геопроецессинга (геообработки), таких, как пересечение полигонов и др. WPS определяет веб-интерфейсы доступа к сервису геопроецессинга. Геопроецессинг может включать любой алгоритм, расчет или модель, оперирующие векторными или растровыми данными с пространственной привязкой.

Web Map Tile Service

Эта спецификация расширяет возможности WMS для создания кэшированных картографических сервисов (будут рассмотрены в главе 6).

Кроме перечисленных открытых картографических стандартов, существуют еще и другие, редко используемые. Например, Grid Coverage Service (расширение WCS на сеточные данные), Sensor Planning Service, Web Coverage Processing Service (сервис геообработки растровых данных), Sensor Observation Service (сервис данных с датчиков – например, данных о температуре) и другие.

Отдельно стоит отметить GeoSPARQL – язык запросов, разработанный OGC для сервисов, созданных при помощи открытых картографических стандартов. Иными словами, он служит для создания запросов к ГИС-серверам точно так же, как язык SQL служит для написания запросов к серверам баз данных.

4.5. Картографические сервисы и геопорталы

В настоящее время, в связи с бурным ростом телекоммуникационных технологий, ГИС мигрировали в Internet и компьютерные сети, где они существуют в виде картографических сервисов и геопорталов.

Картографический сервис (см. рис. 4.5.1) предоставляет пользователю возможность лишь просмотреть геоданные через сеть и, в некоторых случаях, про-

вести несложные манипуляции с ними. Примерами картографических сервисов являются Google Maps, Yandex-карты, сервисы, создаваемые ArcGIS for Server, Mapserver, и другими вариантами серверных продуктов для работы с ГИС.

Функции геопортала значительно шире, чем у картографического сервиса. Геопорталом считается web-портал, предназначенный для поиска и доступа к геопространственной информации и связанные с этим сервисы (визуализации, редактирования, анализа и так далее). Также геопорталом называется web-сайт или его эквивалент, перечень функций которого, реализованных в виде геосервисов (web-сервисов) включает в себя поиск наборов данных, их визуализацию, загрузку и трансформирование, а также вызов других сервисов. Иными словами, геопортал является своеобразной надстройкой над картографическими сервисами, значительно расширяющей их возможности. Тип геопортала определяется набором функций, которые он способен поддерживать.

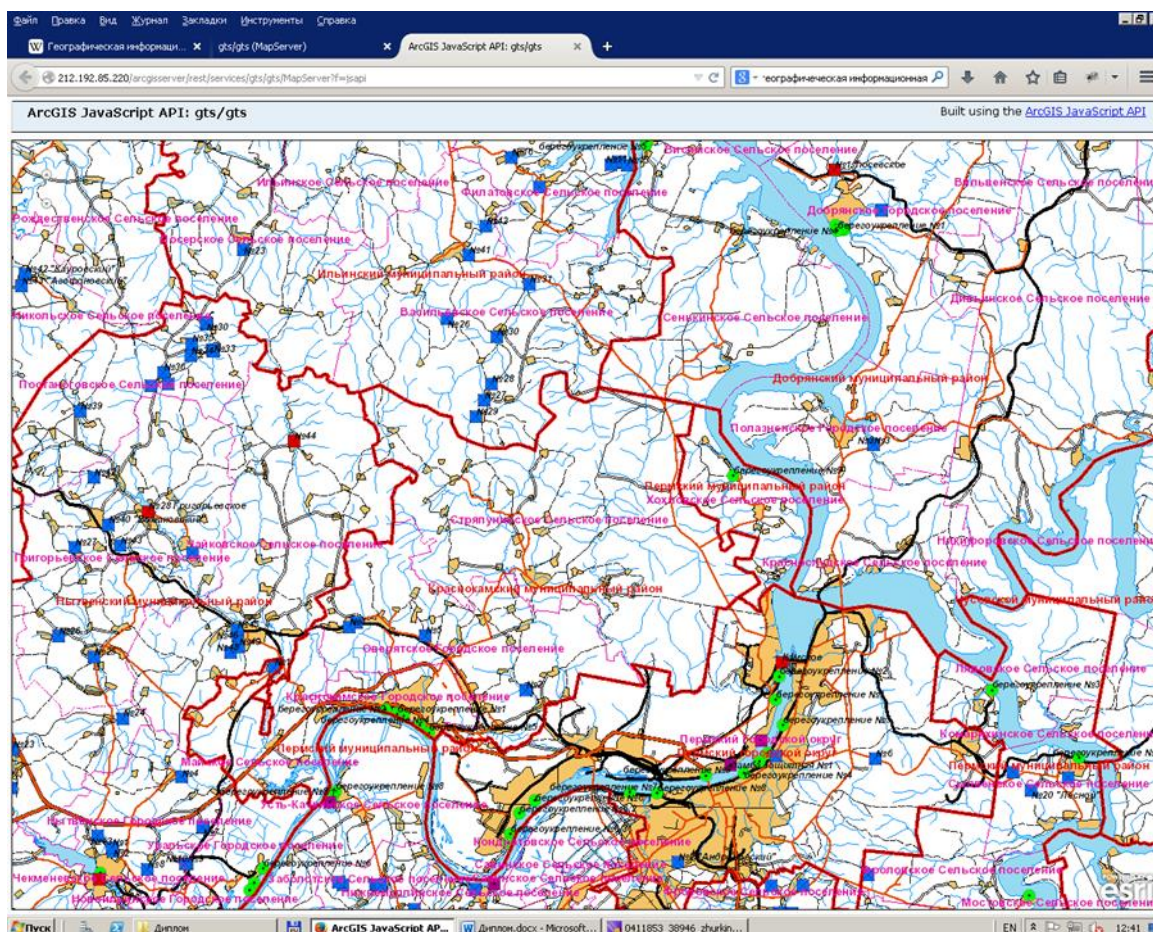


Рис. 4.5.1. Пример картографического сервиса

Картографические сервисы хотя и имеют в наличии функции ГИС, обеспечивающие поиск объектов и маршрутов, но все же ведется поиск не информационных объектов из баз геоданных, а объектов реального мира по их цифровым моделям.

Картографические сервисы в большинстве своем являются планами городов, картами-схемами территорий (в том числе и всего земного шара). Они основаны на данных дистанционного зондирования Земли. Во многих из них доступно создание, импорт и экспорт векторных пространственных данных. В большинстве картографических сервисов поддерживаются API (application programming interface – набор готовых классов, процедур, функций, предоставляемых сервисом, библиотекой или приложением для использования его во внешних программных продуктах).

Геопорталы делятся на национальные, региональные и локальные. Зачастую первые два типа геопорталов в последние годы больше напоминают национальные и региональные атласы, причем спектр их возможностей значительно шире, чем у их бумажных аналогов. Далее будут рассмотрены несколько примеров геопорталов и картографических сервисов.

4.6. Примеры геопорталов

Геопортал Республики Коми

Геопортал Республики Коми [2] представляет собой многоцелевой геопортал регионального охвата. Разработкой геопортала руководят органы местного самоуправления Республики Коми. Необходимость разработки геопортала возникла в связи с высокой потребностью органов государственной власти и местного самоуправления, заинтересованных предприятий и граждан в оперативном получении и использовании достоверной и актуальной информации о пространственных объектах и процессах на территории Республики Коми. Геопортал ориентирован на решение широкого комплекса отраслевых задач министерств и ведомств, информационное обеспечение основных отраслей хозяйства региона, мониторинг качества жизни и состояния окружающей среды. На геопортале опубликованы следующие тематические ГИС:

1) «Карты Республики Коми» (общедоступная ГИС, выполняет роль атласа Республики Коми);

2) «Новая Республика» (ГИС для мониторинга строительства промышленных, жилых и социально-культурных объектов в Республике Коми; общедоступная);

3) «Природно-ресурсный потенциал» (общедоступная ГИС для анализа природно-ресурсного потенциала районов республики, имеется возможность формирования паспорта ПРП для выбранного региона);

4) «Особо охраняемые природные территории» (ГИС с ограниченным доступом для получения данных об ООПТ Республики Коми);

5) «Лесные пожары» (ГИС с ограниченным доступом для мониторинга лесных пожаров);

6) «Зоны операторов связи» (ГИС с ограниченным доступом, содержащая информацию о зонах покрытия операторов мобильной связи).

Кроме того, планируется разработать и включить в геопортал ГИС «Разливы нефти».

Геопортал Роскосмоса

Данный геопортал [3] был создан для организации доступа к единому банку космических снимков Федерального Космического агентства. На геопортале можно посмотреть все снимки из каталога (присутствует возможность их заказа), а также заказать съемку спутником определенного объекта. Также на геопортале можно просмотреть фактические данные со спутников Земли (температуру, влажность, облачность, осадки, действующие лесные пожары). Доступны снимки с множества различных спутников с разной детализацией.

Экологический геопортал Сканэкс

Данный геопортал [9] создан некоммерческим партнерством «Прозрачный мир», он включает в себя несколько различных тематических ГИС для экологического мониторинга территории России:

1) ГИС «Разливы нефти» используется для мониторинга в режиме реального времени мест добычи нефти, передвижения нефтеналивных судов. В случае возникновения пятен от нефтепродуктов ведется мониторинг их перемещения и степени угрозы экосистемам морей. ГИС реализована для мест добычи нефти на море (Каспийское море, Охотское море);

2) ГИС «Лесной сектор и ценные природные территории». Она предназначена для мониторинга лесов на территории нашей страны. В нее входит достаточно подробная карта лесов с указанием преобладающих лесных пород, карта заповедников. Для некоторых областей (например, для Карелии) составлена очень подробная карта лесов. Для мониторинга состояния леса используется мозаика из спутниковых снимков, которая постоянно обновляется.

4.7. Примеры картографических сервисов

Yandex-карты

Этот ресурс в последние годы обогнал по функционалу Google. Каждый зарегистрированный пользователь Yandex может редактировать объекты (улицы, строения, дорожную сеть, топографию), а также атрибутивные данные (например, список организаций в здании). Посетители могут находить оптимальный маршрут до какой-либо точки (с использованием различных транспортных средств, в том числе и маршрутных, рельсовых), искать организации и населенные пункты, просматривать погоду и информацию о транспортных заторах, маршрутах общественного транспорта (в том числе и междугороднего), смотреть фотографии различных зданий, а также совершать виртуальные прогулки по улицам (сервис «Панорамы»). На большинство крупных населенных пунктов имеются актуальные спутниковые снимки сверхвысокого разрешения. Любой человек, знакомый с web-программированием, способен создавать свои карты на основе картографического сервиса Yandex-карт (для этих целей создана API для браузерного языка JavaScript). Присутствует возможность импорта векторных данных в различных форматах [10].

Google Maps

Этот картографический сервис представляет собой карту-схему и спутниковые снимки всего мира. Присутствует возможность редактирования, доступна возможность создания точечных и полигональных объектов, а также возможность экспорта и импорта векторных данных в формате KML/KMZ (формат Google).

Имеется Google Maps API, созданный с целью интеграции карт Google в сторонние сайты. Языком программирования для API, как и для API Yandex-карт, является браузерный язык JavaScript [13].

2ГИС

2ГИС – это картографический сервис, поставляющий детальные планы крупных российских городов (а также некоторых зарубежных). Включает в себя гораздо больше возможностей, чем у Yandex-карт, для поиска информации в пределах города и его окрестностей. В частности, имеется возможность поиска маршрутов (как добраться из одной точки города в другую), с возможностью воспользоваться как городским общественным транспортом (в том числе и рельсовым), так и автомобилем. Все городские организации объединены по виду деятельности, и, вообще, возможностей по их поиску значительно больше, чем в картографических сервисах глобального охвата.

Для встраивания карт в сайты предоставляется 2GIS API. Язык программирования – JavaScript. В настоящее время API только начинает свое развитие [1].

Контрольные вопросы

1. Что такое web-картография? Для чего она предназначена?
2. Перечислите основные картографические стандарты.
3. Чем отличаются картографические сервисы от геопорталов?
4. Приведите примеры картографических сервисов.
5. Какие виды приложений имеют отношение к web-картографии?

Глава 5. ГИС-СЕРВЕРЫ. ARCGIS FOR SERVER

Эта глава была написана на основе справочной системы ArcGIS for Server [8]. Большая ее часть доступна в сети Internet на русском языке.

5.1. Определение ГИС-сервера. Обзор известных ГИС-серверов

ГИС-сервером называется комплекс программного обеспечения, предназначенного для публикации геопространственных данных в локальных или глобальных сетях.

Существуют различные варианты программного обеспечения для публикации пространственных данных в Internet (для создания ГИС-серверов), как одно-, так и кроссплатформенные, как бесплатные со свободной лицензией, так и закрытые с коммерческой лицензией. Рассмотрим некоторые наиболее известные из них.

MapServer

Данное программное обеспечение является одним из самых популярных; оно бесплатное, со свободной лицензией и открытым исходным кодом. MapServer является кроссплатформенным продуктом, то есть он может работать под управлением различных операционных систем (Windows Server, Linux, BSD). Также MapServer имеет возможность интеграции с большинством современных СУБД. Кроме того, MapServer имеет очень большой спектр возможностей для разработки ГИС-приложений (имеются API для всех популярных языков программирования). Несмотря на широчайшие возможности, MapServer имеет существенный недостаток – сложность настройки и установки, так как настройка MapServer производится через текстовые файлы, что создает большую преграду для неопытных пользователей. Несмотря на такой недостаток, по гибкости и функциональности превосходит проприетарное программное обеспечение.

GeoServer

GeoServer – это кроссплатформенный бесплатный ГИС-сервер с открытым исходным кодом. Отличие от MapServer состоит в том, что полученные с по-

мощью него данные можно не только использовать для просмотра и анализа, но и редактировать их, причем после редактирования исходная информация на сервере обновляется. GeoServer поддерживает огромное количество форматов данных и спецификаций картографических серверов. Еще одной важной особенностью GeoServer является легкость его настройки, так как имеется визуальный интерфейс для управления ГИС-сервером. С другой стороны, GeoServer имеет меньший набор возможностей для разработки ГИС-приложений.

ArcGIS for Server

ArcGIS Server – это коммерческий программный продукт, работающий под управлением операционной системы Windows Server, предназначенный для публикации пространственных данных в сети, созданный компанией ESRI. Он имеет большие возможности работы с пространственными данными, которые можно редактировать. Имеется возможность создания web-приложений и различные API с хорошей документацией. ГИС-сервер интегрируется с системой управления web-сайтами IIS (Internet Information Services) и позволяет использовать графические средства для настройки как через web-интерфейс, так и традиционным способом – через оконный интерфейс операционной системы. Недостатками данного программного обеспечения являются высокая цена, отсутствие кроссплатформенности, а также необоснованное потребление ресурсов сервера.

Отдельно стоит упомянуть ArcSDE – интерфейс для хранения и управления пространственными данными в большинстве СУБД (Microsoft SQL Server, Oracle, PostgreSQL), реализованный в серверных и настольных продуктах компании ESRI. ArcSDE реализует возможность совместного доступа к пространственным данным, позволяет эффективно использовать встроенные в СУБД средства идентификации пользователей, создания резервных копий и менеджмента БГД.

В настольном ПО (программном обеспечении) ArcGIS for Desktop создаются, редактируются, анализируются и управляются ГИС-ресурсы, такие как пространственные данные, карты и инструменты геообработки. Серверное ПО ArcGIS for Server позволяет публиковать эти ГИС-ресурсы в сети Internet, работать с ними и обмениваться ими посредством web-сервисов. Web-сервисы, в свою очередь, доступны через Internet и могут использоваться разными клиен-

тами, такими как смартфоны, планшеты, настольные приложения и web-приложения.

Можно также зарегистрировать web-сервисы и открыть к ним доступ с помощью облачной инфраструктуры ArcGIS Online, тогда другие пользователи смогут легко найти и использовать их, что значительно расширит число людей, которые смогут взаимодействовать с ГИС-ресурсами. Размещение ГИС-ресурсов на основе ArcGIS for Server в сети Internet позволяет принимать более взвешенные, в том числе коллективные, решения, поскольку каждый может получить доступ и работать с одной и той же географической информацией.

ArcGIS for Server является ключевым компонентом платформы ArcGIS. Он позволяет совместно использовать ГИС-ресурсы в качестве web-сервисов, которые легко доступны через Internet и могут быть использованы самыми разными клиентами. Геопортал, например ArcGIS Online или Portal for ArcGIS, может обеспечить дружелюбный интерфейс для сервера, который позволяет пользователям с минимальным уровнем подготовки в сфере ГИС создавать и совместно использовать web-карты. Такие инструменты расширяют возможности всей организации, ее подразделений и сотрудников для принятия лучших решений, поскольку каждый может получить доступ и/или совместно с другими работать с одной и той же географической информацией.

5.2. Составляющие ArcGIS for Server

ArcGIS for Server включает несколько составных элементов: ГИС-сервер, Web Adaptor и технологию ArcSDE.

GIS Server (ГИС-сервер) – это ядро программного обеспечения, которое устанавливается на сервере и позволяет создать сайт ArcGIS Server. Сайт ArcGIS Server является экземпляром (службой) ArcGIS for Server, с помощью которого можно совместно использовать ГИС-ресурсы в качестве web-сервисов.

Web Adaptor (веб-адаптор) – это отдельное программное обеспечение, которое можно установить на внешнем web-сервере (Microsoft IIS, IBM WebSphere, Oracle WebLogic и др.). Web Adaptor представляет собой простое приложение, которое позволяет интегрировать ArcGIS for Server с существующим в организации web-сервером.

ArcSDE – это технология хранения пространственных данных в многопользовательской базе данных, администрирования и предоставления доступа к ним.

5.3. Архитектура ArcGIS for Server

При установке ArcGIS for Server пользователи могут получить доступ к нему немедленно, выполняя запросы web-сервисов через порт 6080 (по умолчанию). При установке ArcGIS for Server создается сайт ArcGIS Server (рис. 5.3.1). Сайт ArcGIS Server может состоять из одного компьютера или из нескольких компьютеров. ArcGIS for Server доступен для операционных систем Windows и Linux и может быть установлен либо локально, либо быть частью облачного решения. Независимо от того, как сайт развернут физически, его возможности одинаковы.

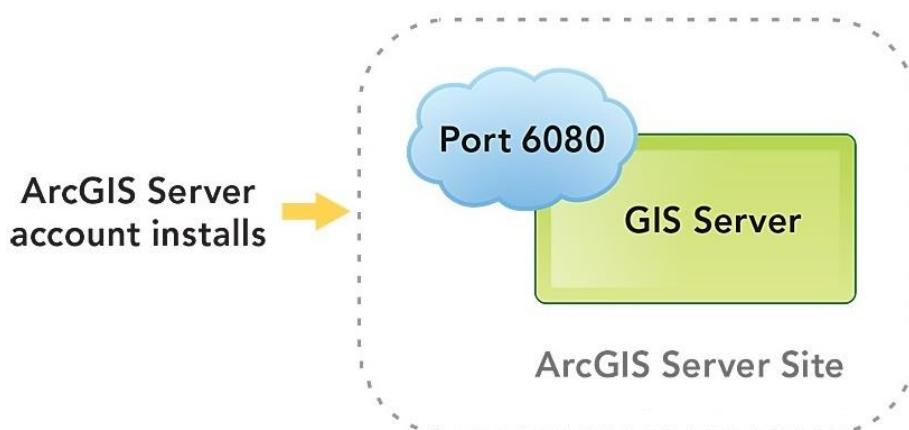


Рис. 5.3.1. Архитектура ArcGIS for Server

При установке программного обеспечения GIS Server на компьютере необходимо создать или назначить учетную запись ArcGIS Server на уровне операционной системы. По умолчанию эта учетная запись называется arcgis. Она используется ArcGIS for Server для выполнения своих операций в фоновом режиме. Поэтому необходимо дать этой учетной записи права доступа и чтения из папки с данными. После установки можно создать новый ГИС-сайт или присоединиться к существующему ГИС-сайту.

Создание нового сайта ArcGIS Server требует основной учетной записи администратора сайта. Эта учетная запись администратора предназначена для управления сайтом и настройки нового сайта. Эта учетная запись является отдельной и отличной от учетной записи ArcGIS Server, которая была создана при установке программного обеспечения ГИС-сервера. Основная учетная запись администратора сайта существует только для сайта ArcGIS Server, а не является учетной записью операционной системы, как учетная запись ArcGIS Server. В табл. 5.3.1 приведены различия учетных записей ArcGIS for Server.

Таблица 5.3.1

**Учетная запись ArcGIS for Server
в сравнении с основной учетной записью администратора сайта**

Учетная запись	Иерархический уровень	Функция	Разрешения доступа	Использование
ArcGIS Server	Операционная система	Установка GIS Server и выполнение операций сервера	Чтение и запись папок данных и файлов папки установки ArcGIS for Server	GIS Server в фоновом режиме
Основного администратора сайта	Сайт ArcGIS Server	Настройка и управление сайтом	Роль администратора сайта	Администратор сайта

Сайт ArcGIS Server имеет три основных точки доступа: приложение ArcGIS Server Manager (рис. 5.3.2, А), каталог сервисов (Services Directory, рис. 5.3.2, Б) и web-каталог администрирования (Administrator Directory, рис. 5.3.3).

ArcGIS Server Manager представляет собой web-приложение, которое позволяет администрировать сайт, а именно:

- управлять web-сервисами;
- настраивать свойства сайта (например, управлять директориями, настраивать безопасность);
- выполнять запросы;
- просматривать журналы.

Для использования всех возможностей ArcGIS Server Manager необходима основная учетная запись администратора сайта или другое подключение администратора. Пользователи, обладающие правами издателя, могут войти в систему с ограниченной функциональностью.

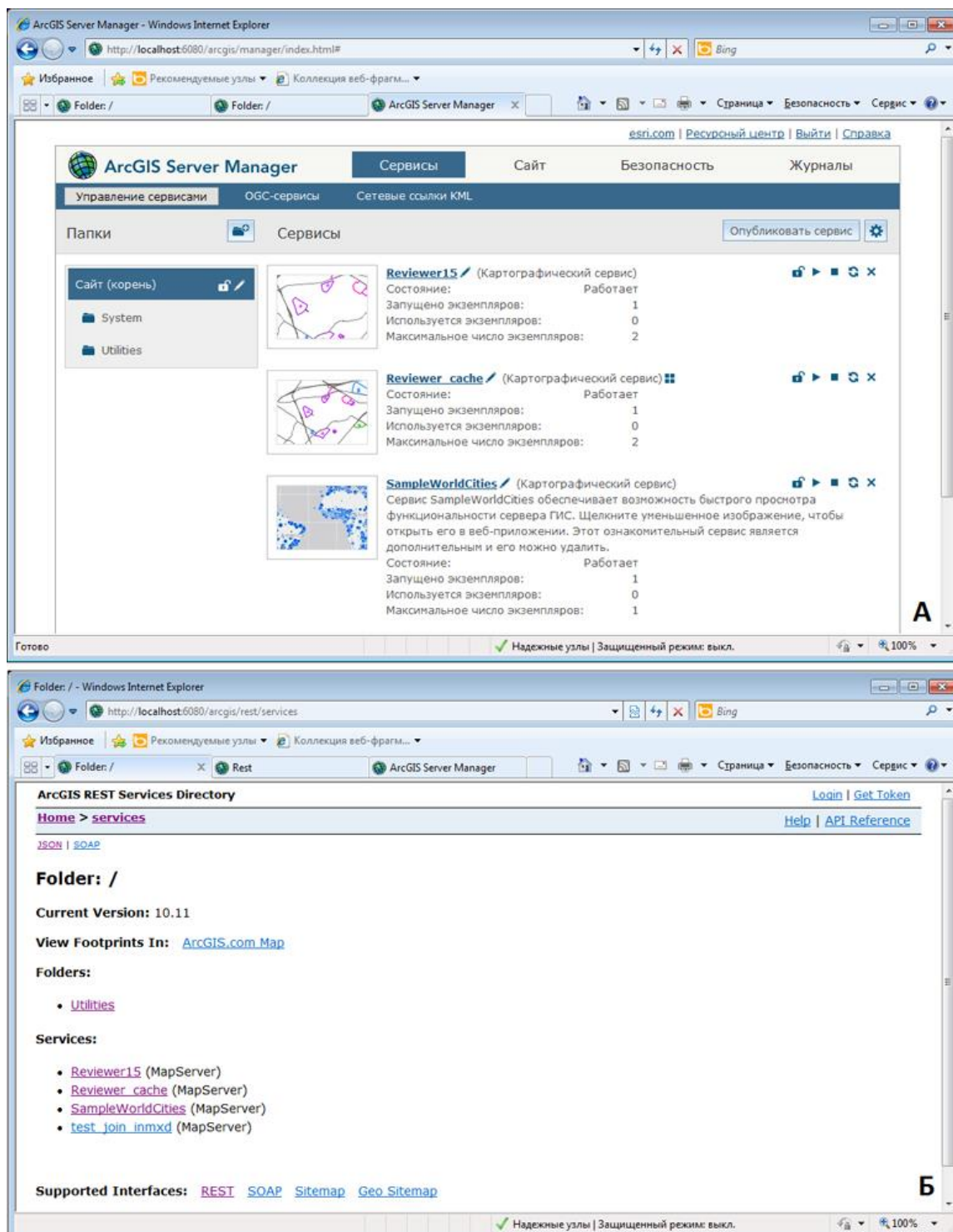


Рис. 5.3.3. ArcGIS for Server, основные точки доступа:
 А) ArcGIS Server Manager; Б) каталог сервисов (Services Directory)

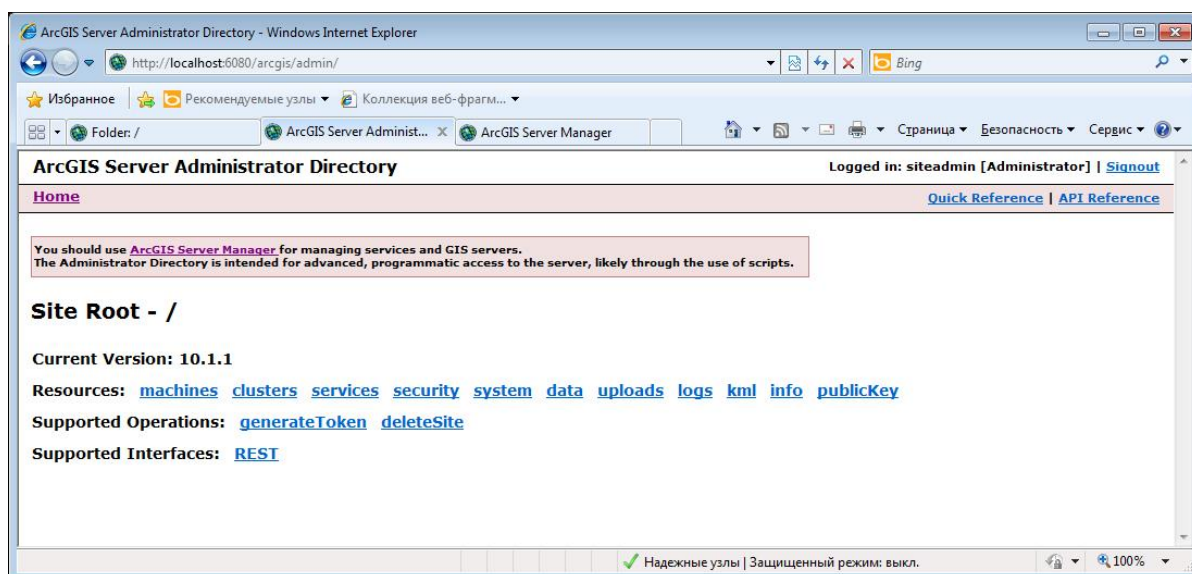


Рис. 5.3.2. Каталог администрирования

Services Directory представляет собой web-приложение со списком всех ГИС-ресурсов, которые доступны на сайте. ГИС-ресурсы, такие как карты, модели геообработки или локаторы, могут быть доступными на сайте ArcGIS Server как web-сервисы. Эти web-сервисы взаимодействуют через архитектуру, основанную на Representational State Transfer (REST). Можно получить информацию и совершить операции на сервере с использованием URL-запросов согласно заданному формату. При использовании Services Directory, по адресной строке браузера можно отследить, как разные URL-запросы используются для получения информации с помощью REST.

ArcGIS Server Administrator Directory обычно называют REST Admin API. Это представление на основе web-браузера установок конфигурации сайта ArcGIS Server. Оно представляет собой программный вход для администрирования сайта. Есть возможность написания сценариев (скриптов) для автоматизации различных задач администрирования (например, на языке программирования Python). Для того чтобы войти в систему и работать с этой точкой доступа, необходимо использовать основную учетную запись администратора сайта или другое подключение администратора. Подключения с правами издателя могут войти в систему с ограниченной функциональностью.

Помимо GIS Server существуют два других важных компонента сайта ArcGIS Server: хранилище конфигурации и директории сервера. Хранилище конфигурации – это папка, которая содержит все основные свойства сайта

(например, информацию о web-сервисах, пользователях, ролях, данных и настройках безопасности). Наглядно все компоненты сайта изображены на рис. 5.3.4. Общая архитектура сервера ArcGIS for Server, развернутого на нескольких компьютерах, показана на рис. 5.3.5.

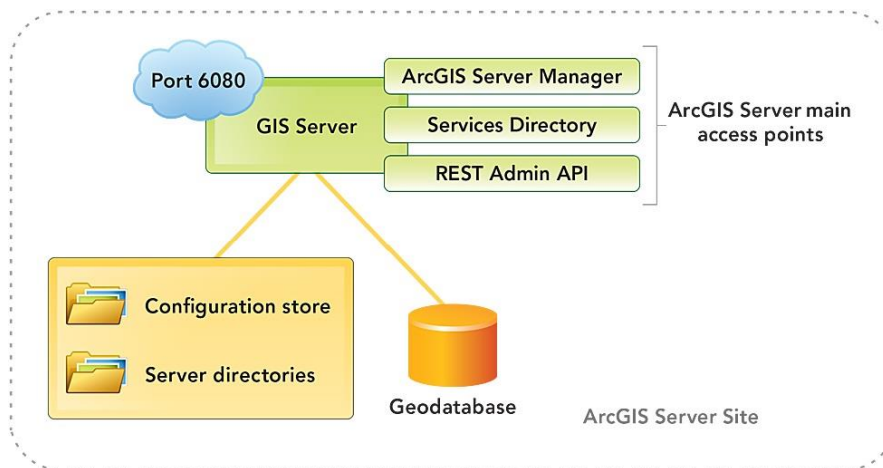


Рис. 5.3.4. Компоненты сайта ArcGIS for Server

Директория сервера включает четыре поддиректории: кэш (cache), рабочая (job), выходная (output) и системная (system).

- Директория кэша хранит временные изображения предварительно сохраненных листов карты, которые картографические сервисы или сервисы глобуса могут использовать для более быстрого отображения. Эта директория обычно занимает много дискового пространства, объем которого зависит от уровня детализации кэшированных сервисов.

- Рабочая директория хранит файлы, необходимые сервисам геообработки, например временные файлы, информацию о текущих процессах и результатах.

- Выходная директория предназначена для хранения временных файлов, необходимых серверу.

- Системная директория используется для хранения информации для сайта, например состояние сервисов, машин и подключений к базам данных. Файлы в этой директории не должны изменяться или удаляться вручную.

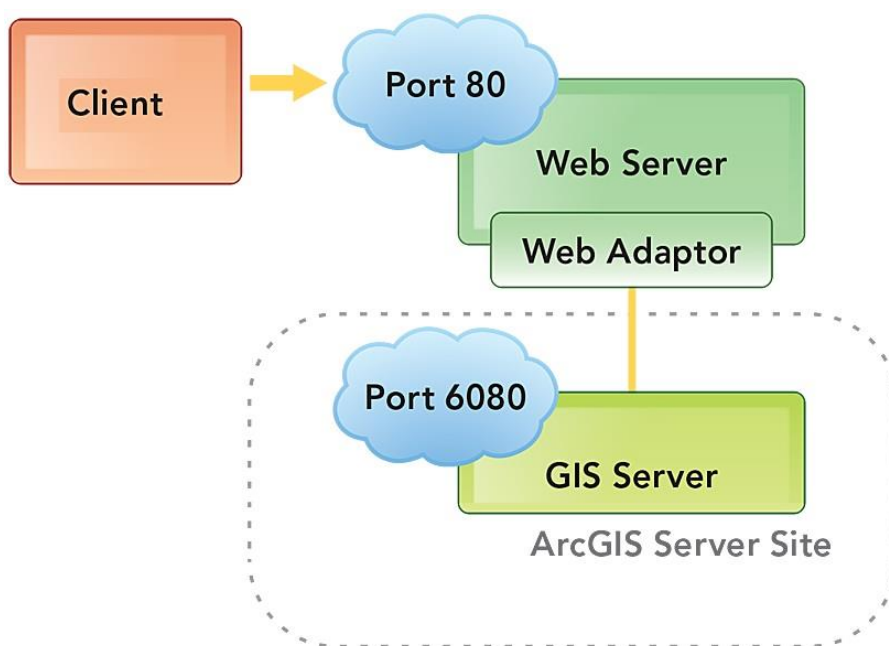


Рис. 5.3.5. Архитектура ArcGIS for Server (развертывание на одном компьютере)

Оба эти компонента (хранилище конфигурации и директории сервера) имеют важное значение для сайта ArcGIS Server. Настоятельно рекомендуется, чтобы создавались резервные копии этих директорий, если сайт поддерживает критически важные приложения. При создании сайта ArcGIS Server с несколькими компьютерами, хранилище конфигурации сервера и серверные директории должны находиться в сетевом ресурсе, доступном для всех ГИС-серверов.

Другим общим компонентом сайта ArcGIS Server является многопользовательская база геоданных – центральное хранилище данных для платформы ArcGIS. Хранение пространственных данных в базе геоданных позволяет применять настраиваемые бизнес-правила и отношения, определять геопространственные реляционные модели (например, топологические отношения, геометрические сети, сетевые наборы данных) и поддерживать множество многопользовательских рабочих процессов. Многопользовательская база геоданных создается внутри СУБД с помощью инструментов геообработки ArcGIS.

5.4. Web Adaptor

Web Adaptor связывает ArcGIS Server с другими web-серверами. Эта конфигурация предоставляет дополнительные преимущества:

- Web Adaptor обеспечивает единую точку доступа на сайт;
- Есть возможность предоставить доступ к сайту ArcGIS Server через стандартный для организации web-сайт и порт вместо ArcGIS for Server порта по умолчанию 6080;
- Web Adaptor обеспечивает более высокий уровень безопасности, предоставляя возможность блокировки доступа к ArcGIS Server Manager и ArcGIS Server Administrator Directory для внешних пользователей; это позволяет увеличить безопасность и использовать функции протоколирования web-сервера.

Web Adaptor облегчает обмен данными между web-сервером и сайтом ArcGIS Server. Клиенты ArcGIS for Server получают доступ к web-сервисам ГИС, посылая запросы web-серверу; запрос передается через Web Adaptor на ГИС-сервер. Один Web Adaptor может быть настроен только на один сайт ArcGIS Server, но один сайт ArcGIS Server может поддерживать несколько Web Adaptor'ов. Иными словами, сайт ArcGIS Server может быть настроен для многих конечных точек доступа с использованием нескольких Web Adaptor.

5.5. Развертывание ArcGIS for Server на нескольких компьютерах

ArcGIS for Server полностью поддерживает интеграцию с корпоративными ИТ-системами предприятия. Развертывание ArcGIS for Server на нескольких компьютерах может быть использовано в организациях, которые имеют бизнес-процессы с высокими требованиями к объемам обрабатываемой информации и большое количество пользователей. При данном сценарии развертывания сайта на каждый ГИС-сервер, входящий в состав сайта ArcGIS for Server, устанавливается ArcGIS for Server (см. рис. 5.5.1). Например, на сайте, имеющем три ГИС-сервера, в случае, если один из компьютеров, на котором развернут ГИС сервер, становится недоступным, сайт по-прежнему будет работать должным образом, потому что есть два резервных ГИС сервера. При этом Web Adaptor выступает в качестве единой точки входа на сайт и обеспечивает взаимодействие между ГИС-серверами. Каждый ГИС-сервер должен иметь доступ к хранилищу конфигурации ArcGIS Server, директориям сервера и источникам данных для веб-сервисов (например, многопользовательской базы геоданных). Такая архитектура расширяема, и вы можете добавлять необходимое число ГИС-серверов для сайта для обеспечения ваших рабочих процессов.

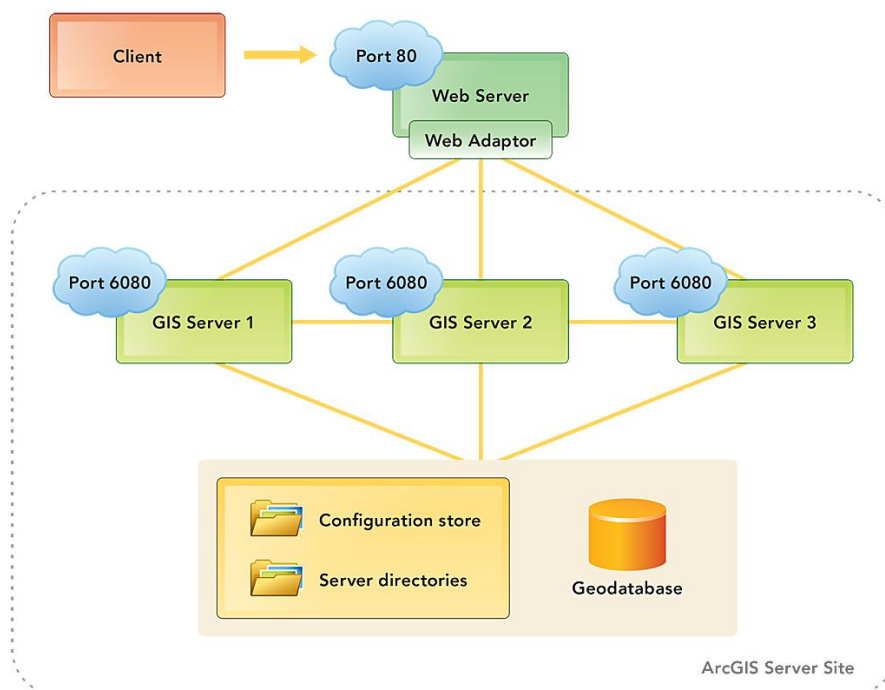


Рис. 5.5.1. Архитектура ArcGIS for Server (развертывание на нескольких компьютерах)

На сайтах с несколькими компьютерами ГИС-серверы могут быть объединены в кластер. Кластер – это логическое объединение компьютеров с одинаковой спецификацией оборудования (см. рис. 5.5.1). Каждый кластер может быть настроен для запуска выделенного набора web-сервисов. Например, можно создать один кластер для выполнения картографических сервисов (рис. 5.5.2 показывает ГИС-серверы 1 и 2 в кластере А), а другой кластер использовать для запуска сервисов геообработки (рис. 5.5.2 показывает ГИС-сервер 3 в кластере В).

Каждый ГИС-сервер можно перемещать из одного кластера в другой в любое время, что позволяет по мере необходимости перераспределять ваши ГИС-серверные ресурсы. По умолчанию каждый сайт ArcGIS Server (как для одного компьютера, так и на нескольких компьютерах) состоит из одного кластера.

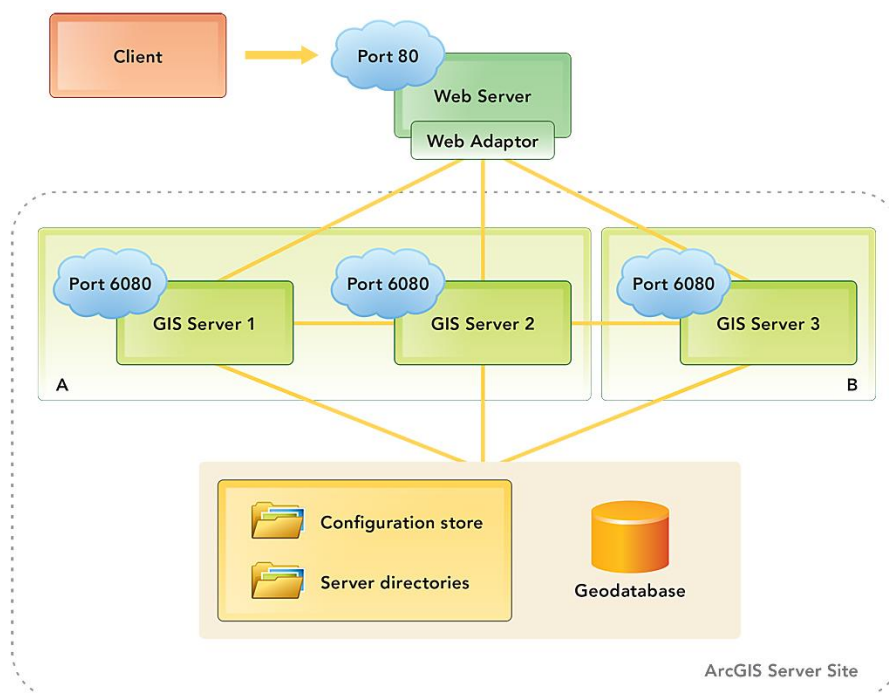


Рис.5.5.2. ArcGIS for Server:
развертывание на нескольких компьютерах с использованием кластеров

5.6. Процесс публикации ГИС-ресурсов в качестве web-сервисов

Процесс отбора ГИС-ресурсов и превращение их в web-сервисы ГИС начинается с ArcGIS for Desktop. В приложении ArcMap создаются, редактируются, анализируются ГИС-ресурсы (например, пространственные данные, карты и инструменты геообработки). К большинству ГИС-ресурсов ArcGIS for Server может предоставить общий доступ в сети Internet с помощью Мастера публикации «Совместно использовать как сервис» (Share As Service), к которому можно обратиться, если выбрать команду «Файл (File) > Совместно использовать как (Share As) > Сервис (Service)». С помощью этого мастера можно создать новый web-сервис, перезаписать существующий web-сервис или опубликовать файл определения сервиса.

Возможно также создание файла определения сервиса. В нем определяются свойства web-сервиса и сохраняются настройки в виде файла на диск, из которого сервис может быть опубликован позднее. Можно также зарегистрировать web-сервис в ArcGIS Online или Portal for ArcGIS, тогда его смогут легко использовать другие.

С помощью Мастера публикации можно опубликовать сервис на сайте ArcGIS Server, а также можно определить дополнительные свойства сервиса например, использование кэширования, или описать метаданные.

ArcGIS for Desktop создает подключения к сайту ArcGIS Server в качестве администратора, для публикации, а также в качестве пользователя. Каждому из этих подключений соответствует тип роли в сайте (таблица 5.6.1).

Таблица 5.6.1

Типы подключений ArcGIS for Server

Роль	Функция	Точки подключения к сайту ArcGIS Server
Администратор	Администрирует сайт ArcGIS Server	<ul style="list-style-type: none"> • ArcGIS Server Manager – Полная функциональность • Services Directory • ArcGIS Server Administrator Directory – Полная функциональность
Издатель	Публикует web-сервисы ГИС на сайте ArcGIS Server	<ul style="list-style-type: none"> • ArcGIS Server Manager – Ограниченная функциональность • Services Directory • ArcGIS Server Administrator Directory – Ограниченная функциональность
Пользователь	Использует web-сервисы ГИС	<ul style="list-style-type: none"> • Только Services Directory (если есть права на доступ к ней)

При публикации сервиса автоматически запускается ArcMap для анализа ГИС-ресурсов, чтобы гарантировать оптимизацию web-сервиса и сообщать о любых проблемах отображения или наличия неподдерживаемых объектов в созданной карте, которые должны быть решены до создания web-сервиса. После публикации web-сервиса, сервис будет внесен в ArcGIS Server Manager и Services Directory сайта. По умолчанию web-сервисы являются открытыми, но доступ к ним можно ограничить. ArcGIS for Server поддерживает различные стандарты безопасности web-сервисов. Список картографических сервисов, поддерживаемых ArcGIS for Server, указан в табл. 5.6.2.

Типы картографических сервисов ArcGIS for Server

Web-сервис	Функциональность	Примечания
<p>Картографический сервис</p> <ul style="list-style-type: none"> • Кэшированный • Динамический • KML • OGC 	<p>Делает карту доступной в виде web-сервиса</p> <p>Содержит полученные заранее изображения данных и контента для более быстрого отображения</p> <p>Данные генерируются по запросу на сервер</p> <p>Делает доступными данные в формате Keyhole Markup Language, поддерживающем Google Earth и многие другие геонавигаторы</p> <p>Делает доступными данные в открытом, признанном на международном уровне, стандартном формате (стандарты Open Geospatial Consortium, Inc.)</p>	<p style="text-align: center;">—</p> <p>Рекомендуется для статических и редко изменяющихся данных</p> <p>Рекомендуется для большинства пространственных данных</p> <p style="text-align: center;">—</p> <p>Поддерживает форматы, включая WCS, WFS, WMS, и WMTS.</p>
Сервисы объектов	<p>Получение геометрии векторных данных и атрибутов из карты или слоя. Объекты отображаются клиентским приложением, а не на сервере. Обычно используются для web-редактирования.</p>	<p>Обычно используется совместно с картографическим сервисом</p>

Веб-сервис	Функциональность	Примечания
<ul style="list-style-type: none"> • OGC 	<p>Делает доступными данные в открытом, признанном на международном уровне, стандартном формате (стандарты Open Geospatial Consortium, Inc.)</p>	<p>Поддерживает форматы, включая WCS, WFS, WMS и WMTS</p>
<p>Сервисы геообработки</p> <ul style="list-style-type: none"> • OGC 	<p>Позволяет выполнять геообработку через web-сервис (напр., печать в web)</p> <p>Делает доступными данные в открытом, признанном на международном уровне, стандартном формате (стандарты Open Geospatial Consortium, Inc.)</p>	<p>Могут быть опубликованы только после выполнения операции геообработки в ArcMap</p> <p>Поддерживает форматы, включая WPS.</p>
<p>Сервис изображения</p> <ul style="list-style-type: none"> • Кэшированный • OGC 	<p>Позволяет быстро получить изображение и растры в качестве web-сервиса</p> <p>Содержит полученные заранее изображения данных для более быстрого отображения</p> <p>Делает доступными данные в открытом, признанном на международном уровне, стандартном формате (стандарты Open Geospatial Consortium, Inc.)</p>	<p style="text-align: center;">—</p> <p>Рекомендуется для статических и редко изменяющихся данных</p> <p>Поддерживает форматы, включая WCS, WMS и WMTS.</p>
<p>Сервис геокодирования</p>	<p>Позволяет получить сопоставление адресов в виде web-сервиса</p>	<p style="text-align: center;">—</p>

Веб-сервис	Функциональность	Примечания
<p>Сервис геоданных</p> <ul style="list-style-type: none"> • OGC 	<p>Предоставляет возможность для выполнения репликации базы геоданных, создания копий данных и выполнения запросов в базе геоданных Делает доступными данные в открытом, признанном на международном уровне, стандартном формате (стандарты Open Geospatial Consortium, Inc.)</p>	<p>Используется с файловыми и многопользовательскими базами геоданных</p> <p>Поддерживает форматы, включая WCS и WFS.</p>
<p>Сервис сетевого анализа</p>	<p>Выполняет анализ транспортных сетей, например, поиск ближайшего пункта обслуживания, выбор оптимального маршрута для транспорта, поиск объектов с помощью размещения-распределения, расчет матрицы стоимости источник-назначение и создание областей обслуживания</p>	<p>Требуется модуль Network Analyst for Server. Работает с сетевыми наборами данных</p>
<p>Сервис геометрии</p>	<p>Помогает клиентским приложениям делать геометрические расчеты, такие как буферизация, упрощение, расчет площадей и длин и проецирование</p>	<p>Используется в программных web-интерфейсах приложений ArcGIS для создания и изменения геометрии пространственных объектов во время web-редактирования</p>
<p>Сервисы Schematics</p>	<p>Предоставляет web-приложению доступ к схематическим представлениям через web-сервис</p>	<p>Требуется модуль Schematics for Server</p>

Веб-сервис	Функциональность	Примечания
Сервис поиска	Делает доступным в локальной сети индекс ГИС-ресурсов организации, по которому можно выполнять поиск	Используется в крупных организациях, где ГИС-данные распределены по нескольким базам данных и файловым ресурсам
Сервис глобуса	Предоставляет возможность работы с 3D-сервисами (на уровне глобуса)	Используются 3D-данные
Сервис мобильных данных	Позволяет приложению ArcGIS Mobile получить доступ к исходным данным документа карты с помощью web-сервиса	—
Сервис Workflow Manager	Предоставляет доступ к возможностям управления рабочими процессами в качестве web-сервиса	—

5.7. Клиенты для ArcGIS for Server

Web-сервисы, ГИС-приложения ArcGIS for Server доступны для различных платформ ArcGIS-клиентов и сторонних клиентов, включая ArcGIS for Desktop, ArcGIS for mobile (iOS, Android, Windows Phone), ArcPad, Esri Location Analytics (Esri Maps for Office, IBM Cognos и Microsoft SharePoint), Business Analyst, Community Analyst, ArcGIS Viewers for Flex и Silverlight, ArcGIS Online, Portal for ArcGIS Map Viewer.

5.8. Работа с ArcGIS Online и Portal for ArcGIS

Ресурс ArcGIS Online и ПО Portal for ArcGIS обеспечивают возможность конечным пользователям просматривать и открывать web-сервисы ГИС и создавать свои собственные web-карты и web-приложения. Первый разворачива-

ется на мощностях Esri, второе может разворачиваться в Вашей организации. Оба способа дополняют и развивают широкие возможности ArcGIS for Server, поскольку они могут концептуально рассматриваться как предоставление привлекательного способа использования web-сервисов Вашего сайта ArcGIS Server (см. рис. 5.8.1).

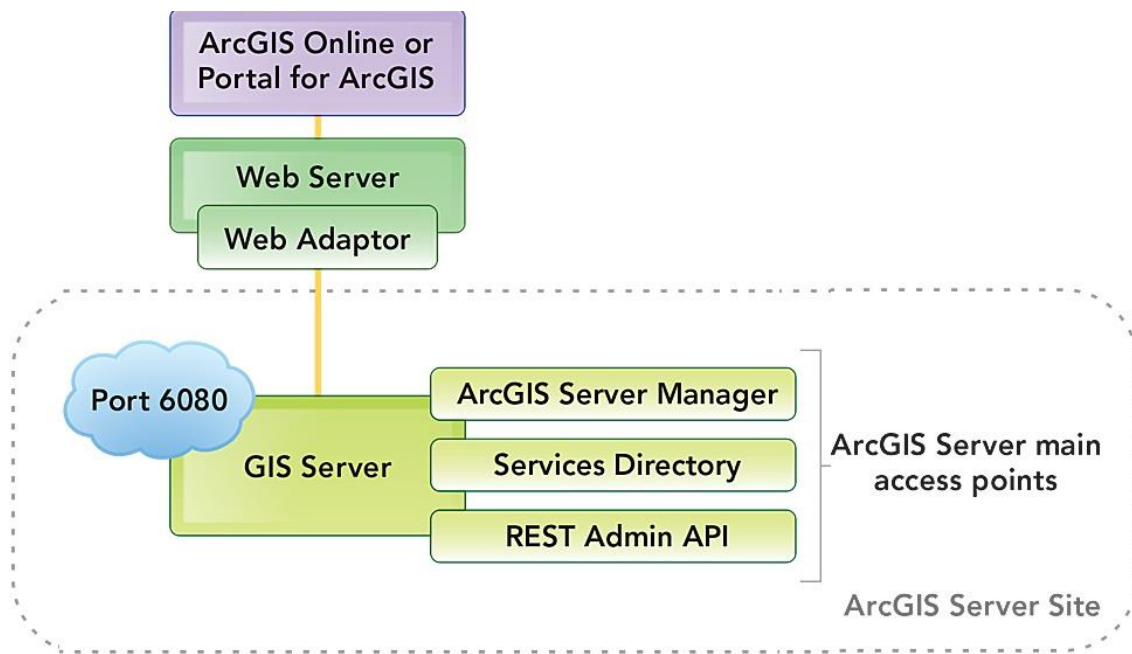


Рис. 5.8.1. ArcGIS Online/Portal for ArcGIS с ArcGIS for Server

Если web-сервисы ArcGIS for Server являются зарегистрированными на ArcGIS Online или Portal for ArcGIS, то конечные пользователи могут легко их найти и использовать. С помощью ArcGIS Online или Portal for ArcGIS Map Viewer web-сервисы ГИС могут быть скомбинированы с базовыми картами онлайн-ресурса и портала, которые могут использоваться в онлайн web-картах. Это может еще больше расширить распространение и использование web-сервисов ГИС в рамках Вашей организации. Вместо того чтобы искать web-сервисы ГИС через Services Directory, можно использовать ArcGIS Online и Portal for ArcGIS как четвертую точку доступа к сайту ArcGIS Server – более удобную для быстрого поиска и использования web-сервисов ГИС.

Контрольные вопросы

1. Что такое ГИС-сервер?
2. Перечислите наиболее известные ГИС-серверы. Какими достоинствами и недостатками они обладают?
3. Из каких частей состоит ArcGIS for Server? Каково предназначение этих частей?
4. Каким образом ArcGIS for Server развертывается на нескольких компьютерах?
5. Какие типы сервисов поддерживает ArcGIS for Server?
6. Какие поддиректории включает в себя директория ArcGIS for Server?
7. Какие типы пользователей могут подключаться к ArcGIS for Server? Какие роли им соответствуют?

Глава 6. СЕРВИСЫ ARCGIS

Эта тема была написана на основе справочной системы ArcGIS for Server [8]. Большая ее часть доступна в сети Internet на русском языке.

6.1. Типы сервисов ArcGIS

Основные типы сервисов, которые поддерживает ArcGIS for Server, перечислены в табл. 6.1.1.

Таблица 6.1.1

Типы сервисов, поддерживаемых ArcGIS for Server

Тип сервиса	Обязательный ГИС-ресурс
Картографический сервис	Документ карты (.mxd)
Сервис геокодирования	Локаатор адресов (.loc, .mxs, пакетный локаатор SDE)
Сервис геоданных	Файл подключения к базам данных (.sde) или файловая база геоданных
Сервис геообработки	Результат геообработки из окна Результаты (Results) в ArcGIS for Desktop
3D-сервис	Документ глобуса (.3dd)
Сервис изображений	Набор растровых данных, набор данных мозаики или файл слоя, ссылающийся на набор растровых данных или мозаику
Сервис поиска (Search)	Папки и базы геоданных содержания ГИС, которые вы хотите найти
Служба Менеджер рабочих процессов (Workflow Manager)	Репозиторий Менеджер рабочих процессов (ArcGIS Workflow Manager)

6.2. Функции сервисов ArcGIS

При публикации ресурса ГИС на сервере можно включить функции, которые определяют различные способы использования сервиса, доступные клиентам. Например, можно активизировать функцию доступа к объектам картографического сервиса, чтобы разрешить редактирование объектов через Интернет. Включая функции, можно предоставить возможности ГИС, необходимые пользователям создаваемых приложений.

Администратор Сервера ArcGIS выбирает ресурсы ГИС для сервера и создает на их основе сервисы. А клиентское представление сервера больше концентрируется на функциях, включенных Вами при публикации ресурса, которые они используют. Администратор может опубликовать один ГИС-ресурс и включить множество функций для этого ресурса. Пользователь этих сервисов увидит данные функции как отдельные сервисы.

В табл. 6.2.1 представлены функции, которые можно включить на сервере ArcGIS и необходимые для каждой из них ГИС-ресурсы.

Таблица 6.2.1

Функции сервера ArcGIS for Server

Возможность	Функции	Сервисы, которые предоставляют эту функцию
Доступ к объекту (Feature Access)	Предоставляет доступ к векторным объектам на карте	Картографические сервисы
Геокодирование	Предоставляет доступ к локатору адресов. Эта функция всегда активна при публикации сервиса геокодирования	Сервисы геокодирования
Геоданные	Предоставляет доступ к содержанию базы геоданных для запросов, извлечения и репликации данных. Эта функция всегда активна при публикации сервиса геоданных	Сервисы геоданных

Возможность	Функции	Сервисы, которые предоставляют эту функцию
Geoprocessing	Предоставляет доступ к моделям геообработки. Эта функция всегда активна при публикации сервиса геообработки	Сервисы геообработки
Globe	Предоставляет доступ к содержанию документа глобуса. Эта функция всегда активна при публикации сервиса глобуса	Сервисы глобусов
Изображения	Предоставляет доступ к содержимому набора растровых данных или набора данных мозаики, включая значения пикселей, метаданные и каналы. Эта функция всегда активна при публикации сервиса изображений	Сервисы изображений
JPIP	Предоставляет функцию потоковой передачи JPIP при использовании JPEG 2000 или NITF (с типом сжатия JPEG 2000) и JPIP Server от ITT VIS	Сервисы изображений
KML	Использует документ карты для создания пространственных объектов Keyhole Markup Language (KML)	Картографические сервисы
Картографирование	Предоставляет доступ к содержимому карты, например слоям и их атрибутам. Эта функция всегда активна при публикации картографического сервиса	Картографические сервисы
Мобильный доступ к данным	Позволяет извлекать данные из карты на мобильное устройство	Картографические сервисы
Сетевой анализ	Решает проблемы сетевого анализа с помощью дополнительного модуля ArcGIS Network Analyst	Картографические сервисы

Возможность	Функции	Сервисы, которые предоставляют эту функцию
Данные схем (Schematics)	Позволяет выполнять просмотр, создание, обновление и редактирование схематических представлений	Картографические сервисы
WCS	Создает сервис, совместимый с web-сервисом покрытий (WCS), разработанным организацией Open Geospatial Consortium, Inc. (OGC)	Картографические сервисы, сервисы изображений и сервисы геоданных
WFS	Создает сервис, совместимый со спецификацией OGC Web Feature Service (WFS)	Картографические сервисы, сервисы геоданных
WMS	Создает сервис, совместимый со спецификацией OGC Web Map Service (WMS)	Картографические сервисы, сервисы изображений
WMTS	Создает сервис, совместимый со спецификацией OGC Web Map Tile Service (WMTS)	Картографические сервисы, сервисы изображений
WPS	Создает сервис, совместимый со спецификацией OGC Web Processing Service (WPS)	Сервисы геообработки

6.3. Сервисы объектов

Сервисы объектов позволяют отображать объекты через Интернет и предоставляют символы, используемые при отображении объектов. Клиенты могут выполнять запросы для получения объектов и выполнять операции редактирования, разрешенные на сервере. Сервисы объектов предоставляют шаблоны, которые можно использовать для расширенного редактирования на стороне клиента. С помощью сервисов объектов также можно выполнять запросы в классах отношений и непространственных таблицах и редактировать эти классы и таблицы (рис. 6.3.1).

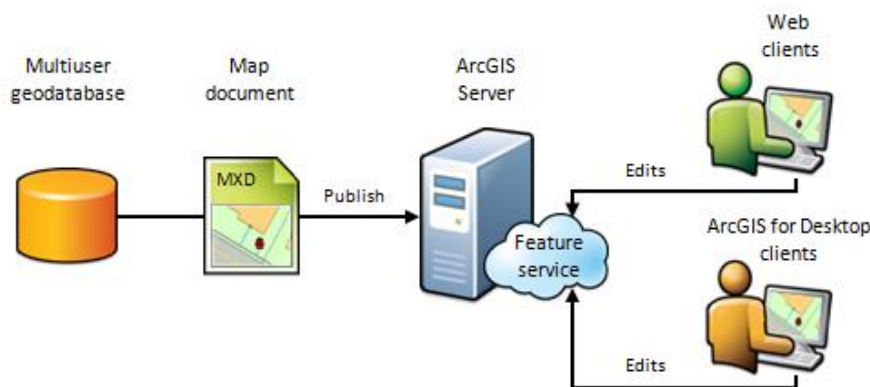


Рис. 6.3.1. Схема сервиса объектов

Сервисом объектов отображаются слои и таблицы, добавленные к документу карты. Требования к данным сервиса объектов следующие:

- Все данные должны принадлежать одной базе геоданных ArcSDE.
- Все данные должны быть зарегистрированы в этой базе.
- Чтобы было возможно редактирование данных, необходимы права на запись данных ArcSDE. При использовании авторизации ОС эти права должны быть предоставлены учетной записи ArcGIS Server.

Поддерживаются версионные и неверсионные данные; если планируется редактировать с помощью сервиса непростые типы (например, ребра сети), поддержка версий обязательна.

В сервисах пространственных объектов не поддерживаются следующие типы данных:

- аннотации;
- объекты-размеры;
- наборы данных Terrain;
- наборы растровых данных;
- слои и таблицы на основе представлений SDE.

6.4. Сервисы геокодирования

Сервисы геокодирования поддерживают широкий диапазон приложений: от управления бизнесом и клиентами до доставки и распространения (для предоставления путевых листов до места назначения). Геокодирование позво-

ляет находить и определять адреса на карте и просматривать, как они относятся к окружающим объектам. Иногда можно увидеть отношения, просто взглянув на карту; в прочих случаях для просмотра неочевидной информации следует использовать инструменты пространственного анализа.

Несмотря на наличие большого количества коммерчески доступных сервисов геокодирования, они могут не соответствовать потребностям организации вследствие нескольких причин: сведения об адресах не обновлены, форматирование адресов отличается или необходимо организовать поиск по местному или распространенному названию объекта (например, Белый дом). Во всех этих случаях необходимо использовать специальное решение геокодирования. Вложив средства в разработку собственного сервиса геокодирования, можно убедиться в соответствии потребностям в геокодировании.

Чтобы сделать сервис геокодирования доступным в сети, необходимо создать локатор адресов в ArcGIS for Desktop, затем опубликовать его в качестве сервиса геокодирования в ArcGIS Server. После публикации сервиса можно использовать ArcGIS Viewer for Flex, ArcGIS Viewer for Silverlight или веб-интерфейсы API для ArcGIS для создания клиентского приложения, которое использует сервис геокодирования для отображения адресов на карте.

6.5. Сервисы геоданных

Сервис геоданных позволяет получать доступ к базе геоданных через локальную сеть или Интернет, используя ArcGIS for Server. Сервис предоставляет возможность для выполнения операций по репликации базы геоданных, создания копий с помощью извлечения данных и выполнения запросов в базе геоданных. Сервис геоданных может быть добавлен в базы геоданных ArcSDE и файловые базы геоданных.

Сервисы геоданных полезны в тех случаях, когда необходимо получить доступ к базам геоданных в удаленных местах. Например, компания желает настроить базы геоданных ArcSDE для управления данными в филиалах Лос-Анджелеса и Нью-Йорка. После создания каждый филиал сможет публиковать свою базу геоданных ArcSDE в Интернете с помощью сервиса геоданных. Сервисы геоданных также могут использоваться для создания реплик баз геоданных ArcSDE. Благодаря возможности репликации базы геоданных сервисы гео-

данных можно использовать для периодической синхронизации изменений в каждой базе геоданных через Интернет.

6.6. Сервис геометрии

Сервис геометрии позволяет выполнять в приложениях геометрические расчеты, например построение буферных объектов, упрощение, расчет площадей и длин, а также проецирование. Кроме того, в программных веб-интерфейсах приложений ArcGIS для создания и изменения геометрии пространственных объектов во время веб-редактирования используется сервис геометрии. Сервис геометрии является альтернативой платформе разработки мелкозадачных модулей ArcObjects или сервису геообработки.

Сервис геометрии создается автоматически. На сервере ArcGIS может быть опубликован только один сервис геометрии.

Простые пользователи, подключающиеся к серверу, не видят сервиса геометрии. Однако на них будут влиять расширенные функциональные возможности, которые предоставляет сервис геометрии через клиентские приложения, разработанные с помощью ArcGIS Viewer for Flex (Flex Viewer for ArcGIS), ArcGIS Viewer for Silverlight (Silverlight Viewer for ArcGIS) и веб API ArcGIS.

6.7. Сервисы геообработки

Сервисы геообработки – это способ предоставления мощных аналитических функций ArcGIS в Интернете. Сервисы геообработки содержат задачи геообработки, каждая задача использует простые данные из веб-приложений, обрабатывает их и возвращает значимые выходные данные в форме объектов, карт, отчетов и файлов. С помощью задачи можно вычислить возможную зону эвакуации при аварии на химическом предприятии, спрогнозировать направление, силу и скорость урагана, создать отчет по землепользованию и составу почвы в определенной пользователем водосборной области, составить карту участков с историческими сведениями о владении или разработать контроль системы стоков. Возможности использования этих сервисов безграничны.

Сервис геообработки содержит одну или несколько задач геообработки. Задача геообработки – это инструмент геообработки, выполняющийся на сервере, его работа и выходные данные контролируются сервером. При предоставлении результата геообработки для совместного пользования в виде сервиса геообработки создается соответствующая задача геообработки из инструмента, которым был создан этот результат. Задача – это термин, который используется в web-API (таких как JavaScript, SilverLight и Flex) для процедур, работающих на сервере и возвращающих результаты.

Работа с сервисами представляет собой комплекс четырех самостоятельных видов деятельности: проектирование, создание, публикация (или совместное пользование) и использование (или потребление). Проектирование сервиса геообработки может вестись при помощи Model Builder или скриптов Python.

6.8. Сервисы глобуса

Сервис глобуса – это тип web-сервиса в ArcGIS for Server, создаваемый из документа ArcGlobe (.3dd). Это один из способов предоставления общего доступа к 3D-содержимому в Internet, используя ArcGIS. Сервис глобуса предоставляет доступ к 3D-содержимому, созданному в ArcGlobe.

ArcGIS for Server и дополнительный модуль Дополнительный модуль ArcGIS 3D Analyst работают совместно, чтобы предоставить 3D-сервисы через Интернет. Сервис глобуса поддерживает те же способы отображения слоя, которые доступны для локальных слоев. Например, можно создать сервис глобуса для общего доступа к 3D-объектам, например к виртуальному городу с объектами зданий и уличным оборудованием, или к поверхности высот высокого разрешения, например ЦМР с разрешением 1 метр, построенной по данным лидара.

Опубликованный сервис глобуса может предоставлять общий доступ практически ко всем поддерживаемым типам слоев в ArcGlobe.

Сходство между сервисами глобуса и картографическими сервисами состоит в том, что сервисы глобусов делают данные ГИС доступными для различных типов клиентских приложений. Тем не менее, порождающие приложения для сервисов различны: картографические сервисы создаются в ArcMap, а

сервисы глобуса создаются в ArcGlobe. Два типа сервиса также имеют различные возможности.

И наконец, хотя оба типа сервисов используют кэш, кэш глобуса хранится в другом формате и в другой папке, чем кэш карты.

Кэш глобуса хранит predetermined листы изображений на диске. При использовании клиента сервиса глобуса, например ArcGIS Explorer Desktop или ArcGlobe, для просмотра области сервиса, сервер может легко извлечь лист из кэша вместо того, чтобы тратить время на перерисовку вида. Администратор ГИС-сервера может самостоятельно определить размер исходного кэша (частичный или полный), но необходимо создать хотя бы частичный кэш, чтобы оптимизировать производительность сервисов.

Объем создаваемого кэша зависит от наличия ресурсов для его построения и хранения. Например, растровые данные, служащие в качестве высот, обязательно должны иметь полный кэш, построенный до начала их использования в сервисе глобуса.

6.9. Сервисы изображений

Данные изображений и растров можно опубликовать в качестве сервиса изображений, используя ArcGIS Server. Сервис изображений предоставляет доступ к растровым данным через web-сервис. Можно также обслуживать данные как часть документа, например документа карты или глобуса, или части других сервисов, например сервиса геоданных.

Исходными данными для сервиса изображений могут быть: набор растровых данных (из базы геоданных или файла на диске), набор данных мозаики или же файл слоя, содержащий ссылку на набор растровых данных или набор данных мозаики. В функции сервиса изображений входит публикация наборов растровых данных или растровых слоев, которые включают обработку данных «на лету» (например, условные обозначения или функции растра), и для этого не требуется дополнительного модуля. Существует дополнительный модуль ArcGIS Image, он требуется для публикации набора данных мозаики или растрового слоя, содержащего функцию мозаики. Этот модуль работает не только с сервисом изображений. Например, если есть документ карты, содержащий набор данных мозаики, то необходим дополнительный модуль ArcGIS Image.

В качестве сервиса изображений могут быть также опубликованы данные лазерного сканирования. Наборы данных Terrain и LAS либо файлы формата LAS можно добавить в набор данных мозаики, а затем опубликовать полученный набор данных как сервис изображений. В этом случае данные лазерного сканирования будут представлены клиентам как растровые, а не как точечные данные, но их исходные файлы будут доступны для загрузки, если такая функция активирована на сервисе изображений.

При публикации сервиса изображений определение сервиса и все сопутствующие данные перемещаются на сервер. Перемещению и дублированию данных на сервере есть следующие альтернативы:

- Размещение данных на диске общего доступа, зарегистрированном на сервере. Например, при публикации набора данных мозаики, необходимо разместить его в подобном местоположении с общим доступом на основе данных, размещенных там же.
- Дублирование (репликация) данных на сервере. Например, данные, используемые в наборе данных мозаики, имеют в местоположении, зарегистрированном на сервере, ту же структуру папок, что и на локальном компьютере.

Эти два варианта размещения данных устанавливаются в свойствах сервера в ходе подключения к серверу. Они не входят в число функций, настраиваемых при публикации сервиса изображений.

Параметры сервиса изображений отвечают за то, каким образом растровые данные публикуются в качестве сервиса изображений и какие возможности имеются у клиентов для взаимодействия с сервисом. К примеру, можно задать параметры по умолчанию, которые будут действительны для всех сервисов изображений, вне зависимости от их исходных данных (например, метод пересчета по умолчанию или допустимые методы сжатия). Для каждого сервиса изображений можно дополнительно настроить обработку «на лету», связав с сервисом некоторые функции. Некоторые параметры применяются только в том случае, если в качестве исходных данных используется набор данных мозаики. К ним относится управление каталогом, в частности, настройка видимости полей в атрибутивной таблице и разрешение загрузки или редактирования (например, добавления) данных.

При публикации сервиса изображений следует учитывать способ подключения клиентов к сервису. Сервис изображений всегда публикуется с возмож-

ностями работы с изображениями, позволяя клиентам подключать их с помощью соединения с ArcGIS for Server или через REST. Однако кроме этого, можно опубликовать сервис изображений с возможностями подключения по стандартам организации Open Geospatial Consortium, Inc. (OGC) – Web Map Service (WMS) или Web Coverage Service (WCS). Эти дополнительные возможности откроют доступ к сервису изображений для широкого круга клиентских приложений и устройств.

6.10. Сервисы KML

Keyhole Markup Language (KML) – это основанный на XML формат файлов, использующийся для представления географических пространственных объектов в таких приложениях, как ArcGIS Explorer и Google Earth. KML позволяет помещать точки, линии и полигоны на карты и глобусы и делать их доступными для других пользователей. Используя KML, можно также назначать текст, изображения, видеоролики или ссылки на другие ГИС-сервисы, которые появятся при нажатии на объект. Большинство клиентских приложений являются бесплатными и имеют простой интерфейс и несложную навигацию.

ArcGIS Server позволяет совместно использовать карты и данные в виде KML несколькими способами:

- Картографические сервисы и сервисы изображений предоставляют ссылку на сеть KML через Representational State Transfer (REST).
- Можно создать собственную ссылку на сеть KML, используя Manager или Services Directory.
- При построении запросов к слоям карты или выполнении операции геообработки или геокодирования через REST, можно получить итоговые данные в виде KML.

6.11. Картографические сервисы

Картографический сервис – это способ предоставления web-доступа к картам с помощью ArcGIS. Карта создается в ArcMap, а затем публикуется в качестве сервиса на сайте ArcGIS Server. Пользователи Интернет и Интранет могут

использовать картографические сервисы в web-приложениях, ArcGIS for Desktop, ArcGIS Online и других приложениях.

Картографический сервис создает карты, объекты и данные атрибутов внутри многих типов клиентских приложений.

Картографические сервисы предоставляют другим пользователям созданную кем-либо карту, находящуюся на сервере. Картографические сервисы спроектированы таким образом, что могут работать с большим количеством сценариев в Internet и Intranet. Одним и тем же картографическим сервисом в ArcMap может пользоваться один пользователь, web-приложением – другой, ArcGIS Online – третий пользователь, а мобильным приложением – четвертый. Далее приведены общие основания для использования картографического сервиса.

Предоставление динамических карт через сервер

Прорисовка динамических карт осуществляется в тот момент, когда на них поступает запрос от пользователя. У них низкое быстродействие, но они подходят для высокопроизводительных интранет-приложений и приложений, для которых необходимо отображение данных «в реальном времени».

Предоставление динамических слоев через сервер

У картографических сервисов есть встроенные функции, позволяющие клиентам, например программным web-интерфейсам приложения ArcGIS, динамически изменять поведение каждого слоя и его оформление. С помощью динамических слоев можно повысить уровень интерактивности карт, с которыми пользователи будут работать в сети Internet.

Предоставление кэшированных карт через сервер

Кэшированные картографические сервисы (в которых используется набор предварительно созданных изображений) являются самым быстрым способом публикации карт в Internet. Этим способом публикации карт пользуются такие сервисы, как ArcGIS Online, Карты Google (Google Maps), Карты Bing (Bing Maps) и многие другие сервисы. Хотя пользователи будут видеть статические изображения карты, можно дополнительно предоставить кэшированные картографические сервисы, которые будут запускаться при выполнении задач поис-

ка, идентификации и выполнения запросов, для доступа к лежащим в основе базовым данным. Также можно регулярно обновлять кэш по мере изменения данных.

Картографические сервисы не кэшируются автоматически; необходимо самостоятельно создать кэш после публикации сервиса.

Предоставление пространственных объектов через сервер

С помощью картографических сервисов отображаются не только изображения. Можно настроить картографический сервис таким образом, чтобы он возвращал набор пространственных объектов, которые будут работать в рамках создаваемого приложения. Извлечение этих пространственных объектов осуществляется с помощью задач, которые будут добавлены в приложение.

Например, может возникнуть необходимость сделать запрос к картографическому сервису и отобразить полученные в результате пространственные объекты в виде графики на карте. Запрос может выглядеть следующим образом: «Выдайте мне все объекты, в имени которых есть 'Delaware'» или «Выдайте мне все пространственные объекты, длина которых превосходит 100 миль».

Предоставление функций сетевого анализа через сервер

Картографический сервис можно настроить для выполнения сетевого анализа на сервере. Если документ карты содержит слой сетевого анализа, то будет доступна функция сетевого анализа, которую можно активизировать при публикации сервиса. После этого, во время разработки приложений, работающих с ArcGIS for Server, станут доступны операции сетевого анализа. В документе карты можно задать свойства, установленные для анализа по умолчанию в документе карты, например для строений и ограждений.

Предоставление карт или пространственных объектов с помощью KML

KML – это способ представления географических данных и карт посредством открытой структуры на базе XML. KML широко используется в геонавигационных приложениях, таких как ArcGIS Explorer и Google Earth для отображения наложений карты или пространственных объектов с атрибутами, иногда доступными во всплывающих информационных окнах. Все картографические

сервисы по умолчанию обладают функциональной способностью возвращать KML.

Предоставление изображений, пространственных объектов или растров с помощью спецификаций OGC

Организация Open Geospatial Consortium издала спецификации для отображения в Интернете изображений карты (WMS), векторных пространственных объектов (WFS), растровых наборов данных (WCS), и листов web-карты (WMTS). Некоторым организациям необходимо, чтобы их географические данные и карты соответствовали этим спецификациям. Картографические сервисы можно конфигурировать таким образом, что они будут возвращать изображения и данные, соответствующие спецификациям OGC. Для получения более подробных сведений см. раздел Сервисы OGC.

Предоставление карт для мобильных устройств

Можно конфигурировать картографический сервис таким образом, чтобы полевые работники могли получать данные с карты на свои мобильные устройства. При публикации карты можно включить функцию Доступ к мобильным данным (Mobile Data Access), которая обеспечивает доступ к карте с мобильных устройств через web-сервис.

6.12. Сервисы мобильных данных

Сервис мобильных данных позволяет приложению ArcGIS Mobile получить доступ к исходным данным документа карты с помощью веб-сервиса. Прежде чем создать сервис мобильных данных, необходимо создать документ карты с данными, которые нужно сделать доступными. Затем следует опубликовать карту как сервис с включенной функцией Доступ к мобильным данным (Mobile Data Access).

6.13. Сервисы сетевого анализа

Существуют web-приложения, которые прокладывает маршрут от одной точки до другой или указывают объекты, например больницы, в определенном

районе на основе введенного адреса. Эти приложения используют функции сетевого анализа на сервере для поиска нужных маршрутов и пунктов обслуживания на основе предоставленных сведений. Нет необходимости в специальном программного обеспечении для сетевого анализа на компьютере для получения результатов, так как сервер выполняет всю работу и возвращает результаты через web-сервис.

С помощью ArcGIS for Server можно размещать сервисы, которые выполняют операции сетевого анализа, например, поиск ближайшего пункта обслуживания, поиск оптимального маршрута для одного автомобиля или парка автомобилей, поиск пунктов обслуживания с использованием механизма расчета размещение-распределение, расчет матрицы Источник-Назначение и создание областей обслуживания. Эти сервисы используют дополнительный модуль Network Analyst для выполнения анализа на основе набора сетевых данных.

6.14. Сервисы OGC

Применение web-сервисов по стандарту организации Open Geospatial Consortium, Inc. (OGC) дает возможность предоставлять карты и данные в открытом формате, признаваемом в Сети на международном уровне. Спецификации стандартов OGC позволяют просматривать карты и данные всем пользователям с поддерживаемым клиентским приложением. Эти приложения могут быть созданы любыми разработчиками с использованием спецификаций OGC. Клиентское приложение может быть настолько простым, что карту можно просматривать в веб-браузере. Или оно может иметь богатые функциональные возможности, как, например, ArcMap.

6.15. Сервисы Schematics

Сервис Schematics предоставляет web-приложению доступ к схематическим представлениям через web-сервис. Этот сервис использует возможности дополнительного модуля Schematics, чтобы получать доступ, создавать, обновлять и редактировать схематические представления.

6.16. Сервисы поиска

Сервис поиска делает доступным в локальной сети индекс ГИС-ресурсов организации, по которому можно выполнять поиск. Сервисы поиска наиболее полезны в крупных организациях, где ГИС-данные распределены по нескольким базам данных и файловым ресурсам. Вместо поиска этих источников данных на диске в операционной системе ГИС-аналитики могут ввести URL-адрес поискового сервиса и ввести поисковый запрос для нахождения нужных данных. ArcMap предоставляет интерфейс пользователя для поиска данных и переноса на карту полученных наборов данных.

Сервисы поиска работают на основе индекса ГИС-ресурсов, который создается и поддерживается сервером. При выполнении поискового запроса сервис производит поиск по индексу для нахождения соответствий, а затем возвращает результаты.

На первое построение индекса уходит некоторое время. После построения индекса новые элементы периодически добавляются в индекс (по умолчанию один раз в час). Чтобы найти удаленные элементы или изменения существующих элементов, индекс должен быть повторно создан «с нуля». По умолчанию это происходит раз в день.

Индекс поддерживается с помощью сервиса геообработки, который создается при создании поискового сервиса. Индекс сохраняется в директории на сервере.

6.17. Кэширование картографических сервисов

Кэширование карт – очень эффективный способ заставить карты и сервисы изображений работать быстрее. При создании кэша карты сервер отрисовывает всю карту на нескольких уровнях масштабирования и сохраняет копии этих изображений (рис. 6.17.1). После этого, когда кто-нибудь запрашивает карту, сервер может предоставить эти изображения. Всякий раз, когда сервер получает запрос на карту, он гораздо быстрее возвратит кэшированное изображение, чем заново нарисует карту. Другое преимущество кэширования состоит в том, что высокая детализация изображения не сказывается на времени предоставления информации.

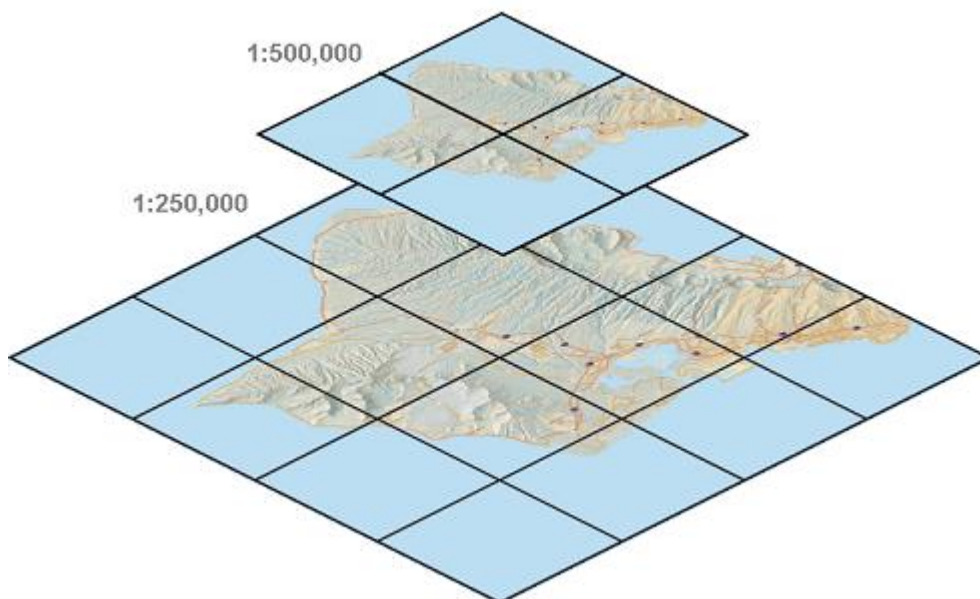


Рис. 6.17.1. Иллюстрация механизма кэширования

Кэширование не происходит автоматически. Чтобы создать кэш, сначала необходимо разработать карту, а затем предоставить ее для совместного использования. После этого следует задать некоторые параметры кэша и начать создание листов. Можно создать все листы сразу или разрешить формирование некоторых листов по необходимости, т.е. при первом обращении к ним.

При кэшировании карты отрисовывают ее на нескольких заданных уровнях масштабирования, чтобы затем пользователи смогли приближать и отдалять карту. При выборе параметров кэша необходимо определить, какие масштабы требуется кэшировать. Если это просто первая проба инструментов кэширования, можно разрешить компьютеру выбрать какие-либо масштабы. Однако обычно масштабы заранее подбирают так, чтобы обеспечить качественное отображение карты в каждом из этих масштабов. Следует записать масштабы и использовать их при разработке карт в ArcMap. Когда придет время создавать кэш, достаточно указать эти масштабы в настройках инструмента создания кэша.

Кэш хранится в директории кэша сервера. При установке ArcGIS for Server директория кэша сервера создается в локальной папке. При добавлении дополнительных ГИС-серверов для поддержки сайта следует настроить общий доступ к директории кэша сервера (используя опции операционной системы, та-

кие как общий доступ к файлам Windows), для того чтобы она была доступна с других задействованных компьютеров.

Выбираемые масштабы и устанавливаемые значения параметров кэша составляют схему листов. В каждом кэше есть файл схемы листов, который можно импортировать при создании новых кэшей, чтобы все кэши использовали одни и те же размеры листов и масштабы. Это, как правило, помогает поднять производительность веб-приложений, содержащих более одного кэшированного сервиса. Дополнительно можно выбрать именно ту схему листов, которая используется в ArcGIS Online, Bing Maps и Google Maps. Это позволит накладывать листы кэша на листы этих картографических онлайн-сервисов.

Контрольные вопросы

1. Для чего необходим сервис объектов?
2. Что такое сервис геометрии?
3. Какие типы сервисов публикуются только в единичном экземпляре?
4. В чем состоит отличие кэшированных картографических сервисов от динамических? В чем заключается механизм кэширования?
5. Поддерживает ли ArcGIS for Server сервисы OCG?

СПИСОК ЛИТЕРАТУРЫ

1. 2ГИС. URL: <http://2gis.ru/> (дата обращения: 20.02.2015).
2. Геопортал Республики Коми. URL: <http://gis.rkomi.ru/> (дата обращения: 20.02.2015).
3. Геопортал Роскосмоса. URL: <http://gptl.ru/> (дата обращения: 20.02.2015).
4. Многоуровневые системы клиент-сервер - № 06, 1997 | Журнал «Сети» / Network World. Издательство «Открытые системы». URL: <http://www.osp.ru/nets/1997/06/142618>. (дата обращения: 20.02.2015).
5. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2014. С. 25–136.
6. Таненбаум Э., Уэзеролл Д. Компьютерные сети. СПб.:Питер, 2012. С. 16–100.
7. Справочник по HTML. URL: <http://htmlbook.ru/>. (дата обращения: 20.02.2015).
8. Справка. ArcGIS Resources. URL: <http://resources.arcgis.com/ru/help/> (дата обращения: 20.02.2015).
9. Экологический геопортал Сканэкс. URL: <http://gis.transparentworld.ru/en/russiaks/>. (дата обращения: 20.02.2015).
10. Яндекс карты. URL: <http://maps.yandex.ru>. (дата обращения: 20.02.2015).
11. JavaScript.ru. URL: <http://javascript.ru> (дата обращения: 20.02.2015).
12. GIS-Lab: Веб-ГИС. URL: <http://gis-lab.info/qa/webgis.html>. (дата обращения: 20.02.2015).
13. Google карты. URL: <http://maps.google.com>. (дата обращения: 20.02.2015).
14. Sql.ru. – Все про SQL, базы данных, программирование и разработку информационных систем. URL: <http://sql.ru>. (дата обращения: 20.02.2015).

Учебное издание

**Пьянков Сергей Васильевич
Быков Алексей Васильевич**

WEB-КАРТОГРАФИРОВАНИЕ

Учебное пособие

Редактор *Е. В. Шумилова*
Корректор *В. Е. Пирожкова*
Компьютерная верстка *А. В. Быкова*

Подписано в печать 31.07.2015. Формат 60x84 1/16.
Усл. печ. л. 6,39. Тираж 50 экз. Заказ _____

Издательский центр
Пермского государственного
национального исследовательского университета.
614990, г. Пермь, ул. А.И. Букирева, 15