

Содержание:

ВВЕДЕНИЕ

Совершенствование технических возможностей средств вычислительной техники, развитие коммуникационных средств и технологий управления информационными ресурсами в последние годы привели к появлению более крупных информационных систем. Речь идет о масштабах систем не только относительно объема поддерживаемых информационных ресурсов, но и числа их пользователей. Объем информационных ресурсов Web в настоящее время исчисляется многими миллионами страниц.

В связи с этим развитием информационных технологий, сетей, а также информационных систем получил широкое распространение язык гипертекстовой разметки HTML. Информационные системы при этом рассматриваются как инструмент моделирования реальности, реализующей различные подходы. В последние годы стали появляться инструментальные средства и крупные информационные системы, в которых совместно используются различные информационные технологии. Сейчас существует множество специализированных программ для разработки Web сайтов. Такие программы облегчают работу разработчикам в создании Web страниц со сложным дизайном, позволяют динамически генерировать страницы Web.

Для информационных технологий характерна деятельность по стандартизации различных аспектов. Такая деятельность направлена на обеспечение переносимости приложений и информационных ресурсов между различными программно – аппаратными платформами, повторное использование ресурсов, в частности это может быть использование программных компонентов приложений.

Информационные системы сегодня применяются во всех областях общественной жизни и научной деятельности. Курсовая работа предназначена для обобщения накопленного отечественного и зарубежного опыта в разработке информационных систем связанная с Web-технологиями, выявление общих положений и принципов их построения и развития. Представленная работа показывает значимость и эффективность использования информационных систем в первую очередь для поддержки человеческой деятельности в различных областях науки, образования и

культуры.

Глава 1. ОПРЕДЕЛЕНИЕ СОДЕРЖАНИЯ ОСНОВНЫХ ПОНЯТИЙ

1.1. Понятие «информационная система» в Web

Создание современных электронных вычислительных машин позволило автоматизировать обработку данных во многих сферах человеческой деятельности. Без современных систем обработки данных трудно представить сегодня передовые производственные технологии, управление экономикой на всех ее уровнях, научные исследования, образование, издательское дело, функционирование средств массовой информации, проведение крупных спортивных состязаний и т.д. Значительно расширило сферу применения систем обработки данных появление персональных компьютеров.

Одним из наиболее распространенных классов систем обработки данных являются информационные системы (ИС). Любой вид деятельности основывается на информации о свойствах состояния и поведения той части реального мира, с которой связана эта деятельность. Для получения такой информации во многих случаях необходимо регулярно через некоторые интервалы времени проводить измерения или наблюдения, позволяющие определить характеристики состояния сущностей реального мира и протекающих процессов, соответствующие моментам времени, когда эти изменения производятся. Именно для этого существует специальный класс систем обработки данных – автоматизированные информационные системы (АИС).

Автоматизированной информационной системой называется комплекс, включающий вычислительное и коммуникационное оборудование, программное обеспечение, лингвистические средства информационные ресурсы, а также системный персонал и обеспечивающий поддержку динамической информационной модели некоторые части реального мира для удовлетворения информационных потребностей пользователей [3, С. 13].

Под динамической моделью в данном понятии изменяемость модели во времени. Это «живая», действующая модель, в которой отображаются изменения,

происходящие в предметной области. Такая система должна обладать памятью, позволяющей ей сохранять не только сведения о текущем состоянии предметной области, но и в некоторых случаях предысторию. Поскольку эта модель, поддерживаемая ИС, материализуется в форме организованных необходимым образом информационных ресурсов, она называется информационной моделью.

АИС не всегда функционирует самостоятельно. Она может входить в качестве компонента (подсистемы) в более сложную систему, такую, например, как система управления торговой компанией, САПР или система управления производством, учреждением и т. д.

Информационные системы уже многие десятки и даже сотни лет существуют и используются на практике в фактографических системах, которые основаны на технологиях баз данных и оперируют структурированными данными, системы текстового поиска, оперирующие документами на естественных языках, глобальную гипермедийную информационную систему Web и др. По этой причине в определении используется обобщенный термин информационные ресурсы. Частными его случаями являются данные для систем баз данных, документы для систем текстового поиска, HTML-страницы или XML-документы для Web и т.д.

Важный факт состоит в том, что единого устоявшегося и общепринятого определения понятия «информационная система» в настоящее время не существует, да и вряд ли оно может существовать. Дело в том, что в зависимости от необходимости в разных случаях используются разные точки зрения на такой сложный продукт высоких технологий, каким являются современные информационные системы.

Специалисты по системному проектированию трактуют понятие ИС более широко, чем комплекс, о котором идет речь в приведенном выше определении. При этом в состав ИС включаются, например, организационно-методические и технологические документы.

Приведем определение информационной системы, заимствованное в одном из наиболее авторитетных международных научных журналов в рассматриваемой области - «Information Systems», выпускаемом с 1975 года крупным английским издательством Pergamon Press. Информационная система определяется как «аппаратно-программные системы, которые поддерживают приложения с интенсивной обработкой данных (Data-Intensive Applications)». В этом определении акцентируется внимание на весьма важном, но лишь единственном аспекте

информационных систем. Заметим, что приложение информационной системы понимается здесь как надстройка над информационной системой, обеспечивающая решение некоторого комплекса задач в интересах какой-либо сферы деятельности.

Большинство опубликованных определений информационной системы трактует это понятие с функциональной точки зрения, а именно как «систему, предназначенную для сбора, передачи, обработки, хранения и выдачи информации потребителям и состоящую из следующих основных компонентов: программное обеспечение; информационное обеспечение; технические средства; обслуживающий персонал». При этом остается в стороне направленность этих функций, цель, для достижения которой они осуществляются.

В отличие от многих других публикаций, в приведенном определении делается акцент на главном назначении информационных систем, а не на их функциях и ресурсах, которые они не используют. Поддержка динамической информационной модели — это то общее, что свойственно любой информационной системе независимо от характера информационных ресурсов, которыми она оперирует, и, следовательно, от информационных технологий, на которых она основана.

Именно такой подход является наиболее продуктивным в данной работе, поскольку хотелось бы с единых позиции рассмотреть здесь базовые направления технологий современных информационных систем, а именно технологии Web.

ИС используют ресурсы нескольких категорий — средства вычислительной техники, системное и прикладное программное обеспечение, информационные, лингвистические и человеческие ресурсы. Кроме того, хотя об этом не говорится в известных определениях автоматизированных информационных систем, но подразумевается, как само собой разумеющееся, для функционирования системы необходимы и другие ресурсы — помещения, их техническое оснащение, всевозможная оргтехника, электроснабжение и т.д.

Пользовательские информационные ресурсы в Web — это страницы Web-сайтов, ресурсы «скрытого» Web — базы данных, а также различные доступные пользователям Web-документы, представленные в форматах, отличных от HTML. В Web нового поколения к информационным ресурсам, кроме того, относятся не только представленные на Web-сайтах XML-документы, но и различные метаданные. Они описывают схемы XML-документов, их семантику, онтологии.

Во многих публикациях употребляется словосочетание специализированная информационная система. Из выше приведенного определения информационной

системы следует, что универсальных информационных систем не бывает. Каждая из них существует в единственном числе, ее тиражирование бессмысленно, поскольку такая система моделирует конкретную предметную область, поддерживает характеризующие ее свойства информационные ресурсы, которые ассоциированы с конкретными моментами или периодами времени. Поэтому специализированной является каждая информационная система.

Усиливается тенденция глобализации ИС. Глобализация информационных систем имеет две стороны – обеспечение глобального доступа пользователей к системе и интеграция информационных ресурсов, распределенных в глобальной сети. Уникальной глобальной ИС является Web. В нем воплощаются обе указанные стороны глобализации ИС. Он обеспечивает глобальный доступ к явно представленным на Web-сайтах информационным ресурсам, а также к ресурсам «скрытого» Web. Вместе с тем на платформе Web создаются разработанные приложения, обеспечивающие интеграцию распределенных в Web информационных ресурсов. Многочисленные глобальные системы создаются в настоящее время как приложения Web для электронного бизнеса, для поддержки научной кооперации различных коллективов ученых во многих областях знаний в международном и национальном масштабе, в библиотечном деле и в других сферах. Среда Web предоставляет для поддержки таких систем идеальные условия.

1.2. Понятие «технология Web»

. Не стоит путать понятия Интернет и всемирная паутина (WWW). Интернет это глобальная сеть сетей, а WWW это один из сервисов Интернет. Но эта путаница скорее говорит о глобальной популярности всемирной паутины (WWW) и подчеркивает ее значимость для пользователей.



Основные понятия веб-технологий

Основные понятия веб-технологий: веб-страница и веб-сайт. Их не стоит путать.

Веб-страница – это минимальная единица сервиса WWW. По своей сути это документ, который уникализирован в WWW своим URL адресом.

Веб-сайт – это набор веб-страниц связанных общей тематикой. Веб-сайт находится на одном сервере (хостинге) и принадлежит одному владельцу. Как вариант, веб-сайт может состоять из одной веб-страницы (сайт – визитка). Совокупностью всех веб-сайтов и образуют всемирную паутину, WWW – сервиса Интернет, созданного для поиска и обмена нужной информации.

WWW (World Wide Web) или Web – поддерживаемая в Internet глобальная открытая бесконечно масштабируемая распределенная гипермедийная информационная система с архитектурой “клиент-сервер” распределение и неоднородность ресурсов которой прозрачны для пользователей. Система обладает огромным интенсивно наращиваемым информационным ресурсом, большинство из которых предоставляется для свободного доступа в любой момент времени. Среда WWW способна интегрировать ресурсы других информационных сервисов Internet – Gopher, FTP, Archie, WAIS, Telnet, электронной почты. Она обеспечивает также телекоммуникационный доступ к базам данных. Наиболее активно используемыми сервисами Internet являются WWW, электронная почта, сервисы передачи файлов, поддержки телеконференций, удаленного доступа к вычислительным ресурсам.

Все информационные сервисы Internet строятся на основе архитектуры «клиент-сервер». Некоторые из них, например, WWW, поддерживают распределенные информационные ресурсы.

Для информационных сервисов Internet создано разнообразное свободно распространяемое и коммерческое программное обеспечение, функциональные возможности которого не зависят от специфических особенностей конкретных аппаратно-программных платформ, на которых оно используется. Это достигается благодаря стандартизации технологий, на которых эти сервисы базируются, и поддержке этих стандартов в указанном программном обеспечении. Благодаря тому, что сеть Internet построена на основе стандарта эталонной модели сетевого взаимодействия открытых систем (Open System Interconnection — OSI), это программное обеспечение не зависит также от особенностей сетей, входящих в состав Internet. Его место в эталонной модели — прикладной уровень. Таким образом, неоднородность используемых в Internet аппаратно-программных платформ и сетевых возможностей является прозрачной для пользователя рассматриваемых сервисов.

В глобальной коммуникационно – вычислительной сети Internet функционирует ряд информационных и других сервисов, услуги которых свободно доступны для любого пользователя или предоставляются при условии, если пользователь обладает необходимыми полномочиями. В последнем случае обычно используется механизм, предусматривающий предъявление пользователем своего идентификатора (имени) и пароля для подтверждения его полномочий доступа.

Непрерывно эволюционирующее, глобальное информационное пространство, неразрывно связано с введением новых информационных технологий.

Под понятием “информационная технология” понимается, как комплекс методов, подходов, стандартов и инструментальных средств, используемых для создания, поддержки и применения компьютерных систем какого-либо класса в некоторой среде функционирования.

Термин “технологии Web” или “Web – технологии” объединяет в себе два выше рассмотренных понятия “Web” и “Информационная технология”. В большинстве энциклопедий и словарей под Web - технологией понимается “технология построения Всемирной паутины, представление разного рода документов, находящихся в Интернете в виде связанных между собой системой ссылок”

Глава 2. ТЕХНОЛОГИИ WEB

Всемирная паутина стала столь популярной, что для большинства пользователей понятия Интернет и WWW являются синонимами, хотя такое мнение, разумеется, является ошибочным. Но это лишний раз подчеркивает значимость Всемирной паутины и веб-технологий, используемых в ней.

Для начала необходимо разобраться с основными понятиями веб-технологий: веб-сайт и веб-страница. Часто неопытные пользователи их неправомерно смешивают. Веб-страница – это минимальная логическая единица Всемирной паутины, которая представляет собой документ, однозначно идентифицируемый уникальным URL. Веб-сайт – это набор тематически связанных веб-страниц, находящихся на одном сервере и принадлежащий одному владельцу. В частном случае веб-сайт может быть представлен одной единственной веб-страницей. Всемирная паутина является совокупностью всех веб-сайтов.

Основой всей Всемирной паутины является язык разметки гипертекста HTML – Hyper Text Markup Language (рис. 14). Он служит для логической (смысловой) разметки документа (веб-страницы). Иногда его неправомерно используют для управления способом отображения содержимого веб-страниц на экране монитора или при выводе на принтер, что в корне противоречит идеологии, принятой во всемирной паутине.

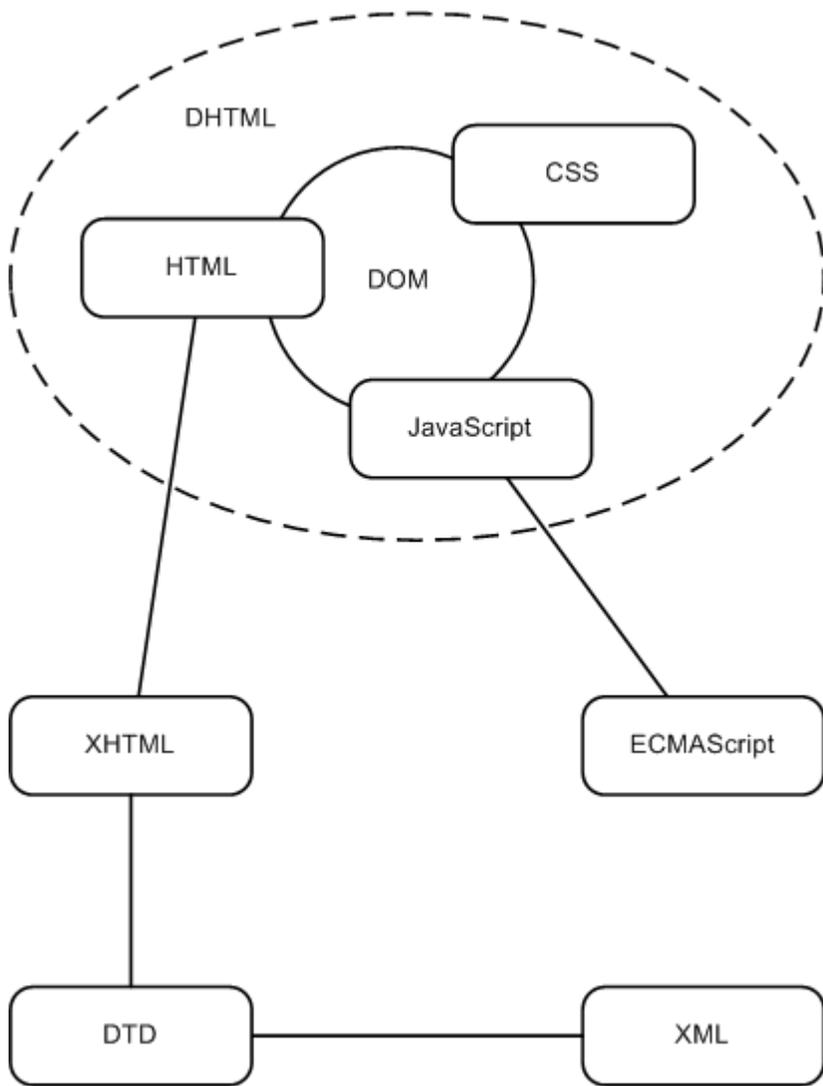


Рис. 14. Веб-технологии

Для целей управления отображением содержимого веб-страниц предназначены каскадные таблицы стилей (CSS). CSS во многом сходны со стилями, применяемыми в популярном текстовом процессоре Word.

Идеология, подразумевающая использование логической разметки и стилей, является очень удобной, так как позволяет изменить оформление всего сайта путем изменения соответствующего стиля. В противном случае пришлось бы менять все теги, отображение содержимого которых требуется изменить.

Для придания веб-страницам динамизма (выпадающие меню, анимация) используются языки написания скриптов. Стандартным скриптовым языком во всемирной паутине является JavaScript. Ядром языка JavaScript является ECMAScript.

HTML, CSS, JavaScript – являются языками, с помощью которых можно создавать сколь угодно сложные веб-сайты. Но это всего лишь лингвистическое обеспечение, в то время как в браузерах документы представляются в виде набора объектов, множество типов которых является объектной моделью браузера (BOM). Объектная модель браузера уникальна для каждой модели и таким образом возникают проблемы при создании межбраузерных приложений. Поэтому Веб-консорциум предложил объектную модель документа (DOM), являющуюся стандартным способом представления веб-страниц с помощью набора объектов.

В отличие от объектной модели браузера DOM содержит набор объектов лишь для содержимого документа и не имеет объектов, позволяющих управлять окнами и рамками окон. При написании приложений в целях поддержки межбраузерной переносимости необходимо придерживаться стандартов DOM, а к объектной модели браузера прибегать лишь при крайней необходимости. Такая необходимость может возникнуть, например, при управлении окнами и строкой состояния.

Следует отметить, что не все браузеры в полной мере поддерживают DOM, но, тем не менее, их последние версии обеспечивают такую поддержку в объеме, достаточном для практического использования DOM. DOM поддерживается в браузерах IE 5.5+ и NN 7.1+ (знак «+» означает версию не ниже указанной).

Совокупность HTML, CSS, JavaScript и DOM часто называют динамическим HTML – Dynamic HTML или DHTML.

Синтаксис современного HTML описан с помощью расширяемого языка разметки XML – Extensible Markup Language. XML позволяют создавать собственные языки разметки, аналогичные HTML в виде DTD. Существует множество таких языков: для представления математических и химических формул, знаний и т. д.

Как видно из вышесказанного, все веб-технологии тесно взаимосвязаны. Понимание этого факта позволит легче осознать назначение того или иного механизма, применяемого при создании веб-приложений.

2.1. Язык гипертекстовой разметки HTML

Основу WWW составляет гипертекстовый документ, создаваемый с помощью языка разметки гипертекстовых документов. **Гипертекст** – это расширенный текст,

элементы которого могут храниться на различных ресурсах в сети, связь с которыми осуществляется с помощью гиперссылок.

HTML(Hyper Text Markup Language) - стандартный язык разметки документов во Всемирной паутине. Язык HTML интерпретируется браузерами и отображается в виде документа, в удобной для человека форме. HTML является описательным языком. В состав языка входят развитые средства для создания различных уровней заголовков, шрифтовых стилей, различные списки, таблицы, иллюстраций, аудио- и видеофрагментов и др. Современная версия языка 5.0.

Основной единицей языка HTML является тег. Теговая модель описывает документ как совокупность контейнеров, каждый из которых начинается и заканчивается тегами. Общая структура контейнера:

<"имя тега" "список атрибутов">

содержание контейнера

</"имя тега">

Большинство тегов спарены т.е. за открывающим тегом следует соответствующий закрывающий тег, а между ними содержится текст или другие теги. В таких случаях два тега и часть документа, отделенная ими, образуют блок, называемый HTML элементом. Некоторые теги являются элементами HTML сами по себе, и в рамках спецификации HTML для них соответствующий конечный тег необязателен, однако для спецификации XHTML закрывать тег обязательно для валидности WEB документа.

Пример:

<I>Текст курсивом</I>

<p>обычный текст Текст жирный обычный текст </p>

**
 разрыв строки
 разрыв строки**

Для большинства тегов определяется множество возможных атрибутов, однако атрибутов может и не быть. Конечные теги никогда не содержат атрибутов. Общий вид задания атрибута:

имени атрибута = значения атрибута

Пример: Задание таблицы шириной 570 пикселей, с выравниванием по центру, ширина бордюра 5 пикселей.

<TABLE WIDTH=570 ALIGN=center BORDER=5>

Кроме тегов, элементами HTML являются специальные символы CER (Character Entity Reference). Спецсимволы могут задаваться трехзначным кодом – **&#nnn** или именем элемента – **&имя**.

Таблица 2.1 – Специальные символы

Числовой код	Имя символа	Символ	Описание
"	"	"	Кавычка
&	&	&	Амперсанд
<	<	<	Меньше
>	>	>	Больше
			Неразрывный пробел
¢	¢	¢	Цент
£	£	£	Фунт
¤	¤	¤	Валюта
¥	¥	¥	Йена

..	Умляют
©	©	©	Копирайт
«	«	«	Левая угловая кавычка
®	®	®	Торговая марка
±	±	±	Плюс или минус
»	»	»	Правая угловая кавычка

Все теги HTML по их назначению и области действия можно разделить на следующие основные группы:

- определяющие структуру документа;
- задающие оформление элементов документа;
- гипертекстовые ссылки и закладки;
- формы для организации диалога;
- вызов программ.

Динамический и статический HTML-документы

Различают два вида html-документов – статические и динамические. Статические документы хранятся в файлах той файловой системы, которая используется web-сервером или браузером при просмотре локальных файлов. При размещении информации на web-сервере можно использовать динамические документы - такие, которые не существуют постоянно в виде файлов, а генерируются в момент запроса клиента. При чем для конечного пользователя не имеет значения динамический или статический способ представления документов.

Для генерирования динамического документа HTML требуется специально написанная программа по правилам, определяемым web-сервером. При планировании размещения информации на web-сервере, для правильного определения использования, какого либо вида документов, необходимо учитывать

степень обновленных данных, их объем и частоту обращения.

Динамический способ определяет хранение данных в формализованном виде, например в базе данных.

Если же данные хранятся в формализованном виде, то, используя шаблоны документов, в которых были произведены изменения, генерируются статические документы. Для генерирования статических документов можно использовать любые средства отчетов, имеющихся в той системе управления баз данных (СУБД), которой обработаны и формализованы данные.

Перспективы HTML

Новых версий языка HTML не будет, однако существует дальнейшее развитие HTML под названием XHTML (англ. Extensible Hypertext Markup Language — расширяемый язык разметки гипертекста). Пока XHTML по своим возможностям сопоставим с HTML, однако предъявляет более строгие требования к синтаксису. Как и HTML, XHTML является подмножеством языка SGML, однако XHTML, в отличие от предшественника, соответствует спецификации XML. Вариант XHTML 1.0 был одобрен в качестве Рекомендации Консорциума Всемирной паутины (W3C) 26 января 2000 года. Необходимо, однако, учесть одну серьезную деталь — в этом формате создано большое количество информационных ресурсов, что они долго еще будут "пониматься" web-браузерами и использоваться в своем первоначальном виде. Кроме того, все новые форматы будут разрабатываться (и уже разрабатываются — например XML) с поддержкой технологий HTML.

Стиль работы меняется, меняются и средства доступа к содержимому. Язык HTML уже изначально создавался как платформо-независимый язык. Новые технологии применяются практически везде и довольно скоро пространство World Wide Web перестанет быть достоянием лишь пользователей настольных персональных компьютеров, уже сейчас некоторые пользователи активно пользуются голосовыми браузерами для незрячих или браузерами, использующими азбуку Брайля, зачастую содержимое выводится не на монитор компьютера, а в телевизор, когда применяются приставки с выходом в сеть или на телетайп, или на монохромные дисплеи различных организаторов-пейджеров и прочие.

Статические сайты

Основа любого сайта — это конечно же HTML - язык гиперразметки текста, к нему добавляется оформление внешнего вида сайта. Чаще всего всё оформление выносят в отдельный файл CSS – каскадных таблиц стилей. Сайт созданный с использованием HTML и CSS, как раз и есть Статический сайт. И никакие дополнительные баннеры, и скрипты никогда не переведут этот сайт в разряд Динамических.

Как правило все Статические сайты имеют не очень большое количество страниц (примерно до 50 отдельных web-страниц). Это могут быть сайты небольших компаний или частных лиц не очень часто обновляемые, как правило это так называемые Сайты-Визитки. Почему?

Если у Вас уже есть свой сайт, то Вы наверняка уже знаете, что управлять сайтом с большим количеством страниц, разделов и рубрик довольно проблематично и долго. Представьте себе сколько уйдет времени, чтобы обновить всего одну ссылку в меню сайта, например, на 20 – 30 страницах... А если больше? Ведь нужно отредактировать каждую из web-страниц сайта, затем все это обновить на сервере, а это ВРЕМЯ. Очень много времени!

Вот и получается, что тем больше сайт - тем сложнее, дольше, а главное рутиннее становится обычное сопровождение и обновление сайта. В такой ситуации творчество превращается в обычную рутину. Но если сайт обновляется не часто, то вполне достаточно иметь обычный Сайт-Визитку, созданный на HTML, пусть и с большим количеством страниц. Я и сейчас знаю множество успешных сайтов в несколько сотен страниц каждый!

Динамические сайты

Но человеческая мысль не стоит на месте, и для того, чтобы избежать рутины по обслуживанию и обновлению сайтов были написаны различные CMS (системы управления контентом) на различных языках программирования. Одним из самых удачных языков программирования для «сайтостроения» оказался язык PHP.

При написании кода web-страниц, в HTML код с помощью специальных операторов, подгружаются вставки кода PHP. Поэтому HTML всегда является основой, на которую как бы наращивают элементы программирования. Любая HTML-страничка с легкостью может стать PHP страницей, достаточно просто переименовать файл, при этом изменив расширение с .html на. php.

Все вставки PHP кода предварительно обрабатываются на сервере, а уже затем готовая web-страница отдается браузеру. Поэтому PHP еще называют серверным языком.

С появлением PHP стало возможным разбивать web-страницу на отдельные блоки и элементы. Например, однотипные блоки сайта, такие как: Шапка сайта (Header), Подвал сайта (Footer), Меню (а иногда и несколько меню) и другие блоки можно вынести в отдельные файлы.

Действительно, если на сайте есть однотипные и постоянные блоки (Шапка, Подвал, Меню и т.д.), которые очень редко изменяются, есть смысл вынести в отдельный файл и просто подгружать ко всем страницам сайта. При необходимости что-то исправить, изменения вносятся в один единственный файл. А затем этот файл подгружается ко всем страницам сайта.

Управление сайтом очень сильно упростилось, и теперь стало неважно сколько страниц на сайте (10 или 1000), добавил в Меню новую ссылку, и эта ссылка появилась в Меню всех страниц сайта.

Но программисты пошли еще дальше. Были написаны множество CMS (система управления контентом), которые позволили не только управлять сайтами: добавлять, удалять, редактировать страницы и целые разделы, но еще и вообще отказаться от получения элементарных знаний по «сайто-строению».

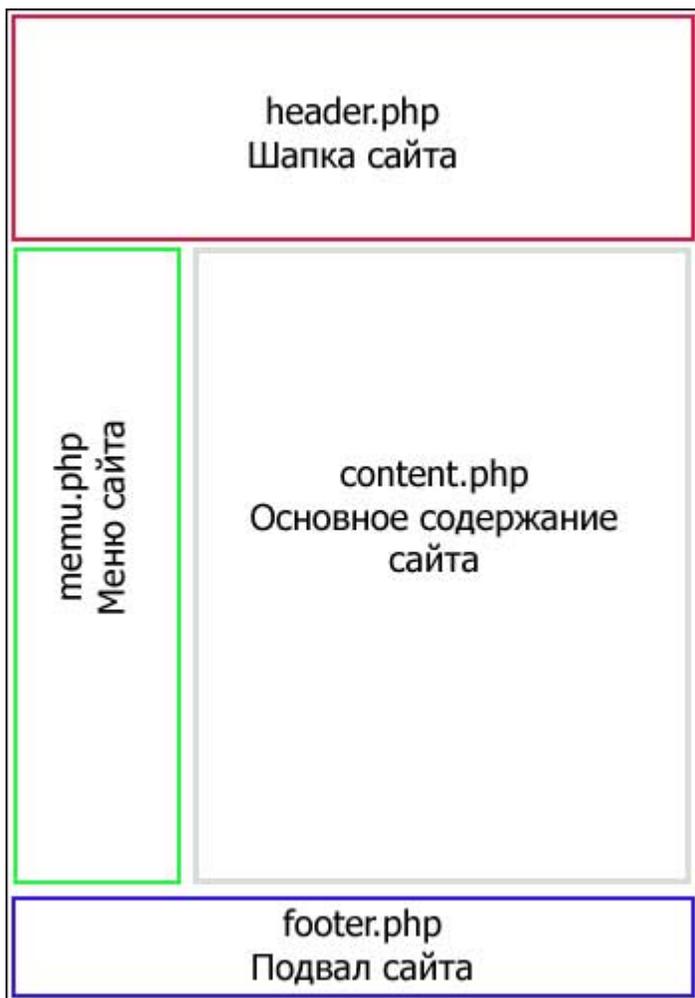
Большинство CMS предоставляют администратору сайта визуальный графический редактор, с помощью которого можно как в WORDe редактировать свои материалы на сайте. Так появилась разновидность сайта: блоги.

Хотя на мой взгляд этого такого сильного упрощения не нужно было делать. Чтобы научиться читать, для начала нужно выучить буквы.... Ну да ладно, будем исходить из того, что есть....

Теперь сайт стало возможным собирать как бы по кусочкам из отдельных файлов.

Типовые названия таких

файлов: header.php (шапка), footer.php (подвал), menu.php (меню), content.php (основное содержание web-страницы). Название это условные, и файлам Вы можете дать любые имена. Схематически это выглядит примерно так:



Скелетная схема php-страницы.

Конечно же это самая упрощенная модель web-страницы, на реальном сайте подобных блоков (кусочков) гораздо больше. Но общий принцип создания php-страниц (php-сайтов) именно такой. Возвращаясь немного выше, напоминаю Вам, что PHP язык серверный, т.е. установлен на сервере хостинговой компании и обработка PHP кода и сборка страницы происходит именно на сервере, а в браузер отдается уже готовая страница. Вот именно такая модель и получила название Динамический сайт.

Кроме того, сайты написанные на PHP получили множество дополнительных возможностей: полезную информацию выносят в отдельный файл, например content.php (статьи, уроки, заметки, описания товаров и услуг), эту часть информации стало возможным хранить в базе данных, что дает нам множество дополнительных возможностей: уменьшается общий вес страницы, появилась возможность вести подсчет загрузок страницы, добавлять комментарии, персонализация страниц, авторизация, доступ к закрытым разделам и т.д. и т.п.

Появилась возможность использовать множество готовых PHP Скриптов.

Но главное помните, что вся обработка php-кода происходит на сервере, иногда говорят: «страница обрабатывается на лету». Каждая web-страница собирается как конструктор, из отдельных кусочков (файликов). Причем этот кусочек может и еще дополнительно отдельно обрабатываться. Вот это и есть основные признаки всех Динамических сайтов.

В тоже время никто не запрещает Вам использовать на Динамических сайтах анимированные баннеры и традиционные скрипты.

2.2 Расширяемый язык разметки XML

Строго говоря, HTML является

одной из реализаций SGML (стандартного обобщенного языка разметки – стандарт ISO 8879), причем реализацией добровольно примитивной – одно только описание стандарта SGML представляет собой 40-мегабайтный PDF-документ. И первое, что пришло в голову разработчикам консорциума W3C, – представить язык разметки

10

веб-страниц в более упорядоченном и структурированном виде, приводя его к другому производному от SGML стандарту, получившему к тому времени широкое распространение расширяемому языку разметки XML. В результате на свет появился стандарт XHTML (Extensible Hypertext Markup Language – расширяемый язык разметки гипертекста). Основные отличия его от HTML можно перечислить в нескольких пунктах:

□ все теги (основные элементы HTML/XHTML) должны быть

закрываются. Даже не имеющие закрывающего тега изначально.

В XHTML, например, элемент `` станет таким: ``;

☐ все имена тегов и атрибутов должны быть записаны строчными буквами (никаких `<BODY><HEAD></HEAD>`, только так: `<body>`);

☐ все атрибуты обязательно заключаются в кавычки;

☐ булевы атрибуты записываются в развернутой форме. Например:

```
<input type = "checkbox" checked = "checked" />;
```

☐ все служебные символы, не относящиеся к разметке, должны быть заменены HTML-сущностями. Например: `<` на `<`; `&` на `&`.

Кроме того, XHTML-документ должен подчиняться правилам валидации обычного XML: допустимо существование только одного корневого элемента, не принимается нарушение вложенности тегов (например, конструкции вида `<a><i> Text </i>`, вполне позволенные в HTML).

Впрочем, самое главное отличие заключалось не в синтаксисе а в отображении XHTML-документа браузером. При встрече браузером значения поля `content-type` в заголовке `http` пакета, равного `application/xhtml+xml`, документ обрабатывается `xhtml`-парсером, аналогично обработке XML-документа. При этом ошибки в документе не исправляются. Согласно рекомендациям W3C, браузеры, встретив ошибку в XHTML, должны прекратить обрабатывать до-

кумент, выдав соответствующее сообщение.

Спецификация XHTML 1.0 была одобрена в качестве рекомендации консорциума Всемирной паутины в январе 2000 года. В августе 2002 года была опубликована вторая редакция спецификации – XHTML 1.1. Параллельно полным ходом началась разработка XHTML 2.0, призванного стать новым уровнем представления документов во Всемирной сети. Разработчики пошли на довольно смелый шаг – нарушение обратной совместимости, но нововведения, которые они собирались внести, стоили того. XHTML 2.0 содержит спецификации Xforms, Xframes, призванные заменить стандартные HTML-формы и фреймы соответственно, ML Events – API для управления DOM-структурой документа, встроенную поддержку модулей Ruby character и многое другое. Работа шла полным ходом, но было несколько обстоятельств, совершенно не радующих авторов спецификаций. Если коротко, XHTML просто не получил должного распространения.

Во-первых, огорчали веб-разработчики, которые после вольницы HTML никак не хотели принимать новые правила в полном объеме. Расставлять в нужном месте кавычки и сущности оказалось просто непосильной задачей. Что там говорить про XHTML, если и со стандартами HTML4 веб-верстальщики обходились достаточно вольно. И что самое возмутительное – производители браузеров активно им в этом потакали!

И именно в этом заключалась вторая проблема. На самом деле все довольно понятно – те, кто делали браузеры, просто не могли допустить, чтобы какой-либо значимый контент в них был не доступен пользователю из-за каких-то неясных принципиальных соображений, и, надо сказать, они были по-своему правы (ну в самом деле, веб нам нужен для общения с миром, а не для того, чтобы все атрибуты были снабжены кавычками!). В результате наиболее популярные браузеры имели два режима отображения XHTML-документов, причем по умолчанию обычно работал «нестрогий» режим, при котором огрехи в разметке милосердно прощались. Хуже того, безусловно, самый распространенный на тот момент браузер Internet Explorer вообще не реагировал на MIME-тип application/xhtml+xml и не имел в своем составе парсера обработки XHTML-документов вплоть до восьмой версии.

Главная причина неудачи повсеместного внедрения XHTML довольно проста. Строгие правила вариации, атрибуты, взятые в кавычки, закрытые одиночные теги... все это, может, и хорошо, но нужны эти тонкости в основном самим разработчикам и блюстителям стандарта, но никак не пользователям, которым, строго говоря, дел нет до всех этих тонкостей. И никак не создателям веб-контента, которым эти правила попросту ничего не дают, кроме несильной головной боли. В общем, сложилось что-то вроде революционной ситуации – создателям стандарт не нужен, а потребителям он не

нужен тем более. А что всем им было нужно? Живой интерактивный веб-контент, воплощающий социальные потребности современного XHTML – стандарта.

Плоский мир HTML с этим справлялся плохо, и к концу

90-х на веб-странице появились не относящиеся к языку разметки компоненты.

Перспективы XML

XML — отнюдь не модное направление, а естественный результат развития Web-технологий, следствие стремления к более эффективному использованию уникальных возможностей открытой глобальной информационной среды, которую они поддерживают. Создание платформы XML — это новая эпоха в развитии Всемирной паутины, это — начало нового, более наукоемкого и технологически более совершенного этапа в ее истории. Сегодня XML, несомненно, стал стандартом де-факто. Все ведущие поставщики программного обеспечения не только Web, но и систем баз данных, включают в свои программные продукты поддержку языка XML или даже создают специализированные системы.

Большую работу по продвижению стандартов платформы XML в практику ведет крупный Международный, некоммерческий консорциум OASIS (Organization for Structured Information Standards, Организация по стандартизации структурированной информации) (Приложение 2.

Международная организация OASIS), в составе, которого более 600 корпоративных и индивидуальных членов из различных стран мира. Эта деятельность является основной задачей консорциума. OASIS разрабатывает, координирует разработки и распространяет информацию о методологиях применения, технологиях и реализациях этих стандартов. В его задачу входит также создание приложений для «вертикальной» индустрии (например, разработки описания типов документов (Document Type Definition, DTD), схем XML и пространств имен XML), спецификаций интероперабельности (в частности, создание спецификаций профилей, включающих стандарты рассматриваемой категории), тестов на соответствие рассматриваемым стандартам.

Распространению стандартов XML-платформы существенным образом способствует политика W3C, направленная на обеспечение доступности их спецификаций,

создание ряда свободно распространяемых синтаксических анализаторов для языка, то большое внимание, которые создатели стандартов XML уделяют обеспечению преемственности для существующей HTML-платформы и накопленных на ее основе ресурсов.

Хотя язык XML и базирующиеся на нем стандарты получают все более широкое распространение, имеются вместе с тем факторы, которые сдерживают массовое распространение XML в среде Web.

Во-первых, существует связанная с экономическими и иными причинами естественная инерционность столь масштабной среды, какой является сегодняшний Web. Эта инерция может преодолеваться только постепенно.

Во-вторых, пока еще не завершена работа над двумя важнейшими стандартами платформы XML, которые позволяют строить из отдельных XML-документов и их компонентов гипермедийную среду. Речь идет о стандартах XPointer (XMLPointerLanguage, язык указателей XML) и XLink (XMLLinkingLanguage, язык ссылок XML). Эти стандарты решают задачу определения гиперссылок в языке XML. Возможности стандартов XPointer и XLink предусматривают существенно более богаты возможности работы с гиперссылками, чем у имеющихся в HTML.

Технологии XML начинают распространяться и в нашей стране. В этой связи приобретает важное значение русскоязычная терминология в этой области.

Платформа XML имеет благоприятные перспективы для широкого практического применения. В пользу этого свидетельствуют не только богатые функциональные возможности рассмотренного семейства стандартов, но и высокая активность в области разработки и развития стандартов, а также производства программного обеспечения, на них основанного.

2.2. Расширяемый язык разметки гипертекста XHTML

XHTML (EXtensible HyperText Markup Language, расширяемый язык разметки гипертекста) - язык разметки веб-страниц, по возможностям сопоставимый с HTML, но с более строгим синтаксисом.

В XHTML сохранены все особенности HTML, однако привнесены более строгие правила создания страниц, что позволяет делать сайты независимыми от устройства отображения и браузера. Это значит, что сайт будет корректно отображаться во всех современных браузерах и платформах вроде компьютеров, смартфонов, КПК, и др.

XHTML поддерживается во всех современных браузерах.

Отличие HTML и XHTML

При написании кода XHTML используется характерный для HTML синтаксис. Разница между HTML и XHTML состоит в наборе некоторых обязательных правил.

Структура документа

- Объявление типа документа (DTD) `<!DOCTYPE>` является обязательным
- Наличие атрибута `xmlns` (указывает на пространство имен `xml`) является обязательным
- Теги `<html>`, `<head>`, `<title>` и `<body>` являются обязательными.

Пример

XHTML элементы

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<title>Заголовок документа</title>
```

```
</head>
```

```
<body>
```

Содержание страницы

```
</body>
```

```
</html>
```

- Вложенность тегов должна быть правильной.

XHTML критично относится к некорректной вложенности одного тега в другой и расположению тега в не соответствующем контейнере. К примеру, в разметке тексте `<p> Lorem <i> ipsum dolor sit </i> amet </p>` есть ошибка, тег `` должен быть расположен после тега `</i>`, а не предшествовать ему.

- Все теги должны быть закрыты.

В отличие от HTML, где допускается в некоторых случаях опускать закрывающий тег, в XHTML закрывающий тег требуется всегда и везде.

`
` - break

`<hr />` - horizontal line

`` - image

- Все теги должны быть набраны в нижнем регистре

В XHTML теги, записанные в верхнем и нижнем регистре, различаются. Чтобы не возникало путаницы, все теги должны, а также их атрибуты должны быть указаны в нижнем регистре.

- В XHTML документе должен быть как минимум один корневой элемент.

XHTML атрибуты

- Все атрибуты должны быть набраны в нижнем регистре (строчными символами).
- Значения любых атрибутов необходимо заключать в кавычки.

Это правило есть и в HTML, однако отсутствие кавычек не влияет на корректность кода, а в XHTML заключать в кавычки значения атрибутов надо обязательно.

`<table width="200" border="1" cell padding="5" cells pacing="0">`

- Запрещается использование сокращенных атрибутов тегов.

Сокращенным называют атрибут, которому не присвоено значение. Атрибуту можно присвоить атрибуту значение, совпадающее с названием. К примеру, вместо `checked` (HTML) используется `checked="checked"` (XHTML).

- Вместо атрибута name следует указывать id.

В XHTML атрибут name частично вышел из употребления, поэтому вместо него рекомендуется использовать id.

Глава 3. СПЕЦИФИКАЦИИ ТЕХНОЛОГИЙ WEB

Расширяемый Язык Разметки (XML) является поднабором SGML и полностью описан в спецификации. Он создан с целью обеспечения обслуживания, передачи и обработки в WEB исходного SGML теми же способами, которые в данный момент имеются в HTML. XML был разработан для облегчения создания конкретных реализаций и для взаимодействия с SGML и HTML.

Роль W3C в составлении Рекомендаций заключается в том, чтобы привлечь внимание к данной спецификации и способствовать её широкому распространению. Это расширит функциональность и возможности Web.

Этот документ специфицирует синтаксис, создаваемый путём подразделения существующих широко распространённых международных стандартов обработки текста для использования в World Wide Web.

Extensible Markup Language, сокращённо XML, описывает класс объектов данных, называемых XML-документы, и частично описывает поведение обрабатывающих их компьютерных программ. XML является профилем приложения или ограниченным вариантом SGML - The Standard Generalized Markup Language. По структуре документы XML являются "соответствующими" документами SGML.

Документы XML состоят из единиц хранения, называемых экземпляры, которые содержат разбираемые или неразбираемые данные.

Разбираемые данные состоят из символов, некоторые из которых образуют символьные данные, а другие - разметку. Разметка кодирует описание схемы и логической структуры единиц хранения документа. XML предоставляет механизм наложения ограничений на схему и логическую структуру единиц хранения.

XML был разработан XML Working Group (ранее известной как SGML Editorial Review Board), сформированной под руководством World Wide Web Consortium (W3C) в 1996 году.

Её возглавил JonBosak из SunMicrosystems при активном участии XMLSpecialInterestGroup (ранее известной как SGMLWorkingGroup), также организованной W3C. Члены XML Working Group указаны в Приложении. Dan Connolly является контактом рабочей Группы с W3C.

Цели создания XML:

1. XML будет широко распространён в Internet.
2. XML будет поддерживать большой диапазон приложений.
3. XML будет совместим с SGML.
4. Он будет лёгким для написания программ, обрабатывающих документы XML.
5. Количество свойств по выбору (optional) в XML будет сведено к абсолютному минимуму, в идеале - к нулю.
6. Документы XML должны быть разборчивыми и ясными по смыслу.
7. Дизайн XML должен выполняться быстро.
8. Дизайн XML должен быть формальным и кратким.
9. Документы XML должны легко создаваться.
10. Краткость в разметке XML имеет минимальное значение.

Эта спецификация, вместе с ассоциированными стандартами, предоставляет всю информацию, необходимую для понимания XML и создания компьютерных программ его обработки.

Символы – это разбираемый экземпляр содержит текст, последовательность символов, которая может представлять символьные данные или разметку. Текст состоит из смеси символьных данных и разметки.

Комментарии могут появляться в любом месте документа вне прочей разметки; кроме того, они могут появляться внутри объявления типа документа в тех местах, которые допускаются грамматикой. Они не являются частью символьных данных документа: процессор XML может, но не должен, давать приложению возможность запрашивать текст комментариев.

Инструкции процесса (ИП) позволяют вводить в текст документа инструкции для приложений. Разделы CDATA могут появляться там же, где и символные данные; они используются для escape-блоков текста, содержащего символы, которые иначе будут распознаваться как разметка.

Документы XML должны начинаться объявлением XML, которое специфицирует используемую версию XML.

Поскольку будущие версии ещё не сформированы, эта конструкция даётся как средство предоставления возможности автоматического распознавания версии и должна, следовательно, быть включена обязательно. Процессоры могут сигнализировать об ошибке, если получают документ, помеченный неподдерживаемой версией.

Функцией разметки в документе XML является обязанность описывать структуру хранения данных и логическую структуру и ассоциировать пары атрибут-значение с их логическими структурами. XML предоставляет механизм объявления типа документа для определения ограничений в логической структуре и для поддержки использования predetermined единиц хранения.

Документ XML является правильным/valid, если он имеет ассоциированное объявление типа документа и если документ выполняет ограничения, выраженные в нём.

Объявление типа документа XML содержит или указывает на объявления разметки, предоставляющие грамматику для класса документов. Эта грамматика известна как определение типа документа или DTD. Объявление типа документа может указывать на внешний поднабор (особый вид внешнего экземпляра), содержащий объявления разметки, или может непосредственно содержать объявления разметки во внутреннем поднаборе, или может иметь и то, и другое. DTD документа состоит из обоих соединённых поднаборов. Объявление разметки — это объявление типа элемента, объявление списка атрибутов и объявление экземпляра, или объявление нотации. Эти объявления могут полностью или частично содержаться внутри экземпляров параметров.

Каждый документ XML содержит один или более элементов, ограниченных либо начальными и конечными тэгами, либо - для пустых элементов - тэгами пустых элементов. Каждый элемент имеет тип, идентифицируется по имени, которое иногда называется "generic identifier" (GI) - родовой идентификатор, и может иметь набор спецификаций атрибутов. Каждая спецификация атрибутов имеет имя и

значение.

Начало каждого непустого элемента XML обозначается начальным тэгом. Окончание каждого элемента, начатого начальным тэгом, обязано быть отмечено конечным тэгом, содержащим имя, отражающее тип элемента, как это было дано в начальном тэге. Текст между начальным и конечным тэгами называется содержимым элемента.

Элемент без содержимого называется пустым. Пустой элемент представлен либо начальным тэгом, после которого непосредственно следует конечный тэг, либо тэгом пустого элемента. Тэг пустого элемента имеет особую форму.

Структура элемента документа XML может, для целей проверки, быть ограничена путём использования объявлений типа элемента и списка атрибутов. Объявление типа элемента ограничивает содержимое элемента.

Объявление типа элемента часто ограничивают типы элементов, которые могут появляться в качестве потомков элемента.

Тип элемента имеет содержимое элемента, если элементы данного типа обязаны содержать только дочерние элементы (а не символьные данные), которые могут быть, по усмотрению, разделены пробелами.

В этом случае ограничение включает модель содержимого, простую грамматику, управляющую разрешёнными типами дочерних элементов и порядком, в котором они могут появляться.

Тип элемента имеет смешанное содержимое, если элементы этого типа могут содержать символьные данные, перемежаемые дочерними (необязательными) элементами.

Атрибуты используются для ассоциирования пар имя-значение с элементами. Спецификации атрибутов могут появляться только в начальных тэгах и тэгах пустых элементов; поэтому продукции, используемые для их распознавания, появляются в разделе.

Прежде чем значение атрибута передаётся приложению или проверяется на правильность, процессор XML обязан нормализовать значение атрибута путём применения к нему нижеприведённого алгоритма или путём использования некоторых других методов так, чтобы значение, передаваемое приложению, было тем же, что и произведённое алгоритмом.

Документ XML может состоять из одной или более единиц хранения. Они называются экземплярами; они имеют содержимое и все (исключая экземпляр документа и внешний поднабор ОТД) идентифицируются по name\имени экземпляра. Содержимое разбираемого экземпляра называется его замещающим текстом; этот текст считается неотъемлемой частью документа.

Не разбираемый экземпляр — это ресурс, чье содержимое может, или может не быть, текстом, и, если это текст, может не быть XML. Каждый не разбираемый экземпляр имеет ассоциированную нотацию, идентифицируемую по имени. Помимо требования к процессору XML сделать идентификаторы экземпляра и нотации доступными приложению, XML не накладывает никаких ограничений на содержимое не разбираемых экземпляров.

Общие экземпляры — это экземпляры для использования внутри содержимого документа. В этой спецификации ОЭ иногда называются неквалифицированным термином экземпляр, если это не приводит к неоднозначности.

Экземпляры параметров — это разбираемые экземпляры для использования внутри ОТД. Эти два типа экземпляров используют разные формы ссылок и распознаются в различных контекстах. Следовательно, они занимают разные пространства имён; экземпляр параметра и общий экземпляр с одним именем - это два разных экземпляра.

Ссылка символа ссылается на специфический символ в наборе символов ISO/IEC 10646, например, ссылка на символ, не доступный напрямую из устройства ввода. Ссылка экземпляра ссылается на содержимое именованного экземпляра.

Если процессор XML обнаруживает ссылку на разбираемый экземпляр, то, для того чтобы проверить документ, процессор обязан включить его (экземпляра) замещающий текст. Если экземпляр является внешним, а процессор не пытается проверить документ XML, то процессор может, но это не является необходимым, включить замещающий текст экземпляра. Если не проверяющий процессор не включает замещающий текст, он обязан информировать приложение, что он обнаружил, но не прочитал, экземпляр.

Это правило базируется на том, что автоматическое распознавание, предоставляемое механизмом экземпляров SGML и XML, первоначально созданным для поддержки модульности в авторизации, не обязательно подходит для других приложений, особенно для просмотра документов. Браузеры, например, при обнаружении ссылки на внешний разбираемый экземпляр, могут избрать

визуальное предупреждение о том, что экземпляр существует, и запрашивать его для показа только по требованию.

Литеральное значение экземпляра — это заковыченная строка, реально представленная в объявлении экземпляра, соответствующая нетерминальному EntityValue. Определение: Замещающий текст это содержимое экземпляра после замещения мнемоник символов и ссылок экземпляров параметров.

Нотации идентифицируют по имени формат не разбираемых экземпляров, формат элементов, которые породили атрибут нотации, или приложение, которому адресуется инструкция процесса. Объявления нотации предоставляют имя нотации для использования в объявлениях экземпляра и списка атрибутов и в спецификациях атрибутов, а также внешний идентификатор для нотации, который может позволить процессору XML или его клиентскому приложению локализовать вспомогательное приложение, способное обработать данные в данной нотации.

Соответствующие процессоры XML делятся на два класса: проверяющие и не проверяющие. Проверяющие и не проверяющие процессоры оба обязаны выводить сообщения о нарушениях ограничений правильно сформированности данной спецификации в содержимом экземпляра документа и любых других разбираемых экземплярах, которые они читают.

Проверяющие процессоры обязаны, по выбору пользователя, сообщать о нарушениях ограничений, выраженных объявлениями в ОТД, и невозможности выполнения ограничений правильности, данных в этой спецификации. Чтобы выполнить это, проверяющие процессоры XML обязаны читать и обрабатывать все ОТД и все внешние разбираемые экземпляры, на которые имеются ссылки в документе.

От непроверяющих процессоров требуется лишь проверить экземпляр документа, включая весь внутренний поднабор ОТД, на правильное формирование.

Поскольку не требуется проверять документ на правильность/верность, необходимо обработать все объявления, прочитанные во внутреннем поднаборе ОТД и во всех экземплярах параметров, которые прочитаны, до первой ссылки на экземпляр параметра, который не прочитан; то есть информация в этих объявлениях обязана использоваться для нормализации значений атрибутов, включения замещающего текста внутренних экземпляров поддержки значений по умолчанию в атрибутах.

Формальная грамматика XML даётся в данной спецификации с использованием нотации Extended Backus-Naur Form (EBNF). Каждое правило грамматики определяет один символ.

3.3 О спецификации XHTML

В настоящей спецификации определяется XHTML 1.0, переформулирован HTML 4 в виде приложения XML 1.0, и три DTD, соответствующих типам, определяемым HTML 4. Семантика элементов и их атрибутов определена в рекомендации W3C HTML 4. Данная семантика представляет собой основу для будущего расширения языка XHTML.

XHTML представляет собой семейство имеющихся на данный момент и могущих появиться в будущем типов документов и модулей, являющихся копиями, подмножествами или расширениями языка HTML 4 [HTML]. Семейство типов документов XHTML базируется на XML и предназначено для работы с пользовательскими агентами на базе. Более подробную информацию об этом семействе и его эволюции можно найти в разделе "Направления развития".

Семейство XHTML является следующим шагом в эволюции Интернет. Переходя сегодня на XHTML, разработчики содержимого (контента) могут вступить в мир XML со всеми его преимуществами, сохраняя при этом совместимость содержимого с более старыми и более новыми версиями.

Преимущества перехода на XHTML 1.0 описаны выше. Вот несколько основных преимуществ:

Разработчики документов и создатели пользовательских агентов постоянно открывают новые способы выражения своих идей в новой разметке. В XML ввод новых элементов или атрибутов достаточно прост. Семейство XHTML разработано так, чтобы принимать расширения путем модулей и технологий XHTML для разработки новых соответствующих XHTML модулей (описанных в готовящейся спецификации Модуляризации XHTML). Модули позволят комбинировать существующие и новые наборы функций при разработке содержимого и создании новых пользовательских агентов.

Постоянно вводятся альтернативные методы доступа в Интернет. По некоторым оценкам, в 2002 году 75% обращений к документам в Интернет будет выполняться

с альтернативных платформ. Семейство XHTML создавалось с учетом общей совместимости пользовательских агентов. С помощью нового механизма профилирования пользовательских агентов и документов серверы, прокси и пользовательские агенты смогут преобразовывать содержимое наилучшим образом. В конечном счете станет возможной разработка соответствующего XHTML содержимого, пригодного для любого соответствующего XHTML пользовательского агента.

В настоящей спецификации используются следующие термины, которые расширяют определения, данные в [RFC2119] аналогично определениям ISO/IEC 9945-1:1990 [POSIX.1]:

Описываются общие термины XHTML: атрибут, DTD, возможности, документ, пользовательский агент, правильно построенный, представление (генерация), проверка корректности, реализация, синтаксический разбор, элемент.

В настоящей версии XHTML предоставляется определение строго конформных документов XHTML.

Строго конформный документ XHTML - это документ, которому необходимы только возможности, описанные в настоящей спецификации как обязательные.

Пространство имен XHTML может использоваться с другими пространствами XML в соответствии с [XMLNAMES], хотя такие документы не являются строго конформными XHTML 1.0 в соответствии с приведенным выше определением. В будущих работах W3C будут определены способы указания документов, в которых используется несколько пространств имен.

Конформный пользовательский агент должен соответствовать всем определенным в спецификации критериям.

Говорится о различиях которые присутствуют в языке XHTML. Поскольку XHTML является приложением XML, некоторые приемы, допустимые в языке HTML, основанном на SGML, должны быть изменены.

К документам XHTML 1.0 не предъявляется требование совместимости с существующими пользовательскими агентами, но на практике оно достаточно легко реализуемо.

Спецификация XHTML 1.0 закладывает основу семейства типов документов, которые будут расширениями и подмножествами XHTML, для поддержания

широкого диапазона новых устройств и приложений путем определения модулей и механизма объединения этих модулей. Такой механизм позволит унифицировать способы расширения XHTML 1.0 и использования его подмножеств путем определения новых модулей.

По мере перемещения XHTML с традиционных пользовательских агентов на рабочем столе на другие платформы становится ясно, что не все элементы XHTML будут необходимы на всех платформах.

Процесс модуляризации разбивает XHTML на ряд более мелких подмножеств элементов. Модуляризация дает определенные преимущества.

В профиле документа определяется синтаксис и семантика набора документов. Соответствие профилю документа обеспечивает основу гарантии совместимости. В профиле документа определяются возможности, необходимые для обработки документа этого типа.

Для авторов профили устраняют необходимость написания нескольких различных версий документов для различных клиентов.

Глава 4. РЕАЛИЗАЦИЯ ТЕХНОЛОГИЙ WEB В ИНФОРМАЦИОННОЙ СИСТЕМЕ «УЧЕБНО-МЕТОДИЧЕСКИЙ РЕСУРС»

Разработка информационной системы «Учебно-методический ресурс» включает в себя создание двух подсистем: главной страницы (страницы навигации) и шаблона исходного представления документа.

Для создания этих двух частей выбран табличный дизайн. Он организуется с помощью таблиц. Таблицы используются для организации расположения на страницах текстовой и числовой информации в табличном формате, а также это инструмент для точного позиционирования объектов на Web-странице. Истинная ценность таблиц заключается в возможности их применения для компоновки страниц. HTML-таблицы позволяют компоновать такие Web-страницы, создание которых было достаточно сложным, если не вовсе невозможным, до введения таблиц.

В таблицы могут использоваться различные тэги, возможно размещение тэгов графики и текста на Web-странице. С помощью таблиц можно также группировать навигационные кнопки вверху, внизу или по бокам страницы. Использование таблицы для организации навигационных кнопок в упорядоченную структуру с

одинаковым относительным месторасположением на каждой странице существенно упрощает навигацию по сайту. Например,

Для описания таблиц используется тег <TABLE>. Тег <TABLE>, как и многие другие, автоматически переводит строку до и после таблицы.

Создание строки таблицы — тег <TR>. Тег <TR> создает строку таблицы. Весь текст, другие теги и атрибуты, которые требуется поместить в одну строку, должны размещаться между тегами <TR> </TR>.

Определение ячеек таблицы - тег <TD>. Внутри строки таблицы обычно размещаются ячейки с данными. Каждая ячейка, содержащая текст или изображение, должна быть окружена тегами <TD> </TD>. Число тегов <TD> </TD> в строке определяет число ячеек (открыть). Например,

```
<TABLE>
```

```
<TR>
```

```
<TD COLSPAN=3> В таблице два тега TR значит в ней две строки. </TD>
```

```
</TR>
```

```
<TR>
```

```
<TD>В строке три тега TD</TD>
```

```
<TD>значит</TD>
```

```
<TD>три столбца. </TD>
```

```
</TR>
```

```
</TABLE>
```

Для эффективного использования таблиц необходимо применять специальные тэги и атрибуты. Например,

- Тег <TH> — заголовки столбцов таблицы.
- Тег <CAPTION> — использование заголовков таблицы.
- Атрибут NOWRAP

- Атрибут COLSPAN
- Атрибут ROWSPAN
- Атрибут WIDTH
- Атрибут CELLPADDING
- Атрибут CELLSPACING и другие.

При создании любого HTML документа можно говорить, что текст – это единственный объект Web-страницы, который не требует специального определения. Иными словами, произвольные символы интерпретируются по умолчанию как текстовые данные. Но для форматирования текста существует большое количество элементов.

<p> </ p> - Элемент абзаца (paragraph).

**
** - Элемент, обеспечивающий принудительный переход на новую строку.

<pre> </ pre> - Элемент для обозначения текста, отформатированного заранее (preformatted).

<blockquote> </blockquote> - Обозначение цитаты

< center ></ center > - Элемент используется для центрирования текста, а точнее, любого содержимого.

<div> </ div> - Элемент, похожий на предыдущий. Он позволяет выравнивать содержимое по левому краю, по центру или по правому краю. Для этого стартовый тег должен содержать соответствующий атрибут:

align=" left"

align=" center"

align=" right"

** </ b>** - Выделение текста полужирным шрифтом.

** </ font>**

Определение типа, размера и цвета шрифта. Все эти характеристики определяются при помощи соответствующих атрибутов. Например, абсолютный размер шрифта задается при помощи size (размер):

size =Абсолютный размер шрифта

Размер шрифта может задаваться относительно базового:

size =+Число

size =-Число

При назначении величины для size необходимо учитывать величину базового размера. Обе они в сумме должны соответствовать одному из абсолютных размеров. Так для базового размера, равного 3, относительный размер может находиться в пределах от -2 до +4. Если величина выходит за допустимый предел, то используется или шрифт размера 7, или шрифт размера 1.

Для элемента FONT можно использовать атрибут цвета:

color =" цвет"

Атрибут face (вид) также можно использовать для элемента font. Он позволяет задавать тип шрифта:

face="Название шрифта"

Правда, есть одна проблема. Web-страницы просматривают множество людей, и нет гарантии, что у каждого из них окажется нужный шрифт. Если в системе не установлен шрифт точно с таким же названием, то браузер использует свой стандартный шрифт. И другие.

Для создания и форматирования HTML-страниц используется множество тэгов, а также множество различных атрибутов, к этим тэгам.

ЗАКЛЮЧЕНИЕ

Создание Web по праву можно считать одним из крупнейших научно - технических достижений последнего десятилетия XX века. Благодаря реализации этого проекта рождается целый ряд новых информационных технологий, имеющих весьма значимые социально-экономические последствия.

Одним из наиболее распространенных классов систем обработки данных являются информационные системы. В настоящее время усиливается тенденция глобализации ИС.

Современные информационные Web-технологии быстро изменяют наш мир и непосредственно влияют на развитие Web-технологий. Эта технологическая революция сильно повлияла на все сферы человеческой деятельности. Внутренняя сложность и предельная простота применения современных информационных Web-технологий делает их доступными каждому, кто ежедневно сталкивается с применением их в своей профессиональной деятельности.

Главное преимущество Web-технологий в современных условиях заключается в их простоте и как следствие в повышении эффективности их применения.

В этой курсовой работе я попытался показать обширную проблематику технологий современных информационных систем, технологий Web, появления новых тенденций в развитии технологий Web. В данной работе проанализированы необходимые для данной курсовой работы спецификации и разработан фрагмент информационной системы «Учебно – методический ресурс».

Список использованных источников и литературы.

1. Баранов, Д.В. Современные информационные технологии. / Д.В. Баранов. – Томск: ИДО (ТУСУР), 2005. – 130 с.
2. Ваулина, Ч.Ю. Информатика: толковый словарь / Ч.Ю. Ваулина. – М.: Изд-во Эксмо, 2005. – 480 с.
3. Когаловский, М.Р. Перспективные технологии информационных систем / М.Р. Когаловский. – М.: Компания АйТи, 2003. – 288 с.
4. Когаловский, М.Р. Энциклопедия технологий баз данных / М.Р. Когаловский. – М.: Финансы и статистика, 2005. – 800 с.
5. Крис, Д. Креативный Web-дизайн. HTML, XHTML, CSS, JavaScript, PHP, ASP, ActiveX. Текст, графика, звук и анимация. Учебник Пер с англ. / Д. Крис, К. Кинг, Э. Андерсон. – М.: ООО «ДиаСофтЮП», 2005. 672 с.
6. Мишенин, А.И. Теория экономических информационных систем / А.И. Мишенин. – М.: Финансы и статистика, 2002. – 240 с.

7. Непейвода, Н.Н. Основания программирования / Н.Н. Непейвода, Скопин И.Н. – Москва-Ижевск: Институт компьютерных исследований, 2003. – 868 с.
8. Основы Web – технологий: учеб. пособие / П.Б. Храмцов [и др.]. – М.: Изд-во Интуит.ру “Интернет-Университет Информационных Технологий”, 2003. – 512 с.
9. Пауэлл Томас, А. Справочник программиста / Томас А Пауэлл, Д. Уитворт. – М.: АСТ, Мн.: Харвест, 2005. – 384 с.
10. Петров, В.Н. Информационные системы: учеб. пособие / В.Н. Петров. – СПб.: Питер, 2002. – 588 с.
11. Экономическая информатика: Введение в экономический анализ информационных систем: учебник. – М.: ИНФРА-М, 2005. – 958 с. – (Учебники экономического факультета МГУ им. М.В. Ломоносова).
12. Когаловский М.Р. XML: возможности и перспективы [Электронный ресурс] / сост. и ред. М.Р. Когаловский – OSP.RU: Издательство "Открытые системы", [2001]. <<http://www.osp.ru/cio/2001/02/016.htm>> (15.02.2001).
13. Когаловский М.Р. XML: сферы применений [Электронный ресурс] / сост. и ред. М.Р. Когаловский – OSP.RU: Издательство "Открытые системы", [2001]. <<http://www.osp.ru/cio/2001/04/010.htm>> (17.04.2001).
14. Когаловский М.Р. Развитие стандартов XML: новые возможности и применения [Электронный ресурс]: (Материалы Второй всероссийской конференции «Стандарты в проектах современных информационных систем») [тез. докл.] / М.Р. Когаловский Институт проблем рынка РАН <<http://www.cemi.rssi.ru/mei/articles/conf02st.htm>> (27-28.03.2002).
15. Международная организация OASIS: сообщество рабочих групп Журнал: Intersoft Lab [Электронный ресурс] Новая деловая газета: CitCity [Web-сайт]. 24.02.1997// <<http://www.citforum.ru/internet/xml/oasis/>> (09.04.2003).
16. Международный консорциум W3C: от Рабочего проекта до Рекомендации Журнал: Intersoft Lab [Электронный ресурс] Новая деловая газета: CitCity [Web-сайт]. 24.02.1997// <<http://www.citforum.ru/internet/xml/w3c/>> (08.04.2003).
17. Российские Электронные Библиотеки Extensible Hypertext Markup Language (XHTML) [Электронный ресурс] / Аннотированный указатель стандартов платформы XML <<http://www.elbib.ru/index.phtml?page=>

elbib/rus/methodology/xmlbase/standarts_2002/XHTML> (13.01.2006).

18. Российские Электронные Библиотеки Hypertext Markup Language (HTML) [Электронный ресурс] / Аннотированный указатель стандартов платформы XML) / <http://www.elbib.ru/index.phtml?page=elbib/rus/methodology/xmlbase/standarts_2002/HTML > (23.12.2003).

19. Extensible Markup Language (XML) [Электронный ресурс] / Спецификация <<http://www.w3.org/XML/> >.

20. OASIS Advancing E-Business Standart Since [Электронный ресурс] / <<http://www.oasis-open.org/>> (1993).

21. SimpleSolutions - разработка и сопровождение Web-сайтов. Введение в учебник HTML, немного истории [Электронный ресурс] / Web-студия Ретюхина Александра <<http://www.sasha.by/doc2.php?page=html&theme=0>> (2000-2006).

22. SimpleSolutions - разработка и сопровождение Web-сайтов. PHP и XML. [Электронный ресурс] / Web-студия Ретюхина Александра <<http://www.sasha.by/doc2.php?page=html&theme=1>> (2000-2006).

23. SimpleSolutions - разработка и сопровождение Web-сайтов. Основные понятия [Электронный ресурс] / Web-студия Ретюхина Александра <<http://www.sasha.by/doc2.php?page=html&theme=1>> (2000-2006).

24. World Wide Web Consortium W3C, World Wide Web, Web, WWW, Consortium, computer, access, accessibility, semantic, worldwide, W3, HTML, XML, standard, language, technology, link, CSS, RDF, XSL, Berners-Lee, Berners, Lee, style sheet, cascading, schema, XHTML, mobile, SVG, PNG, PICS, DOM, SMIL, MathML, markup, Amaya, Jigsaw, free, open source, software [Электронныйресурс] / <<http://www.w3.org/>> (03.1989).

25. XHTML 1.0: The Extensible HyperText Markup Language (Second Edition) [Электронныйресурс] / Спецификация <<http://www.w3.org/TR/xhtml1>> (26.01.2000, 01.08.2002).

26. HTML 4.01 Specification [Электронный ресурс] / Спецификация <<http://www.w3.org/TR/html4>> (24.12.1999).

27. Сухов К.С89 HTML5 - путеводитель по технологии. - М.: ДМК Пресс,

2013. – 352 с.: ил.

28. <https://studfile.net/preview/3106069/page:4/>

29. <http://citforum.ru/programming/khramtsov/html.shtml>

30. <http://citforum.ru/programming/khramtsov/html.shtml>

31. <http://www.mini-soft.ru/document/yazyk-gipertekstovoy-razmetki-html>

32. <http://www.mini-soft.ru/document/yazyk-gipertekstovoy-razmetki-html>

33. <http://www.structuralist.narod.ru/it/internet/webintroduction.htm>

34. http://www.luksweb.ru/view_post.php?id=312

35. <https://msiter.ru/tutorials/uchebnik-xml-dlya-nachinayushchih/chto-takoe-xml>

36. <https://ru.w3docs.com/uchebnik-html/xhtmll.html>