

Содержание:

Введение

Все в большей степени успешная практическая деятельность современного целеустремленного человека зависит от эффективной организации обмена информацией. Увеличились информационные потоки и повысились требования к скорости обработки, передачи данных, а так же своевременному владению информацией, ее усваиванию и скорости ее познания.

В связи с этим развитием информационных технологий, сетей, а также информационных систем получили широкое распространение гипертекстовая технология, языки гипертекстовой разметки, в частности 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.

Курсовая работа предназначена для обобщения накопленного отечественного и зарубежного опыта в разработке информационных систем связанная с интернет-технологиями, выявление общих положений и принципов их построения и развития.

Цель курсовой работы - изучить теоретический материал по тематике курсовой работы.

Для достижения поставленной цели были выделены следующие задачи:

Во-первых, проанализировать литературу по теме курсовой работы;

Во-вторых, рассмотреть и изучить понятия: «гипертекстовая технология»

В-третьих, охарактеризовать основные технологии Web, такие как: HTML, XML, XHTML;

Для написания данной работы использовалась следующая литература:

Основы языка гипертекстовой разметки HTML и CSS: Учебное пособие/Сост. С. М. Наместников. – Ульяновск: УлГТУ, 2014. – с.

Целью изучения данного учебного пособия является компактное и ясное понимание основных принципов языка гипертекстовой разметки HTML и CSS. При отборе материала предпочтение отдавалось тем конструкциям, которые наиболее часто используются на практике. Поэтому приведенное изложение HTML и CSS не претендует на полноту описания, но позволит понять достаточный минимум для создания несложных HTML-документов.

Брайан Пфаффенбергер, Стивен Шафер, Чак Уайт, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2006 г., 752 с.

В этой книге находится информация о современных технологиях и стандартах в мире HTML, XHTML и CSS. Начинающие пользователи HTML и Web могут использовать книгу в качестве превосходного учебника и сборника справочных материалов. Опытные пользователи найдут здесь описание новых стандартов и технологий, которые наиболее эффективны на сегодняшний день, а также справочные сведения для повседневной работы. Изучены принципы согласованного форматирования, выполняемого с помощью CSS, профессиональные методы HTML-кодирования, приемы разработки сценариев на языке JavaScript и многое другое. Большое количество практических примеров и иллюстраций позволит сделать процесс разработки Web-сайтов максимально наглядным и эффективным.

Дидре Хейз. Освой самостоятельно HTML и XHTML.10 минут на урок.3-е издание. 2002 г., 224 с

В книге последовательно, урок за уроком, рассматриваются основные и дополнительные возможности языка HTML. Каждый из этих уроков рассчитан на 10 минут и посвящен определенной особенности или группе связанных особенностей языка HTML. Теоретический материал этой книги сопровождается большим количеством примеров и рисунков. Благодаря этому, вы сможете не только попрактиковаться в программировании, но и увидеть результат выполнения кода HTML. Знаний, полученных после завершения этого самообразовательного курса, вполне достаточно для того, чтобы создавать собственные Web-страницы для World Wide Web. Более того, используя возможности HTML, вы сможете проводить уникальные профилактические и сервисные работы для вашей компании или же рассказать о себе пользователям "всемирной паутины".

. Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 752 с.

Шестое издание «HTML и XHTML» – самая полная и современная книга по языкам HTML и XHTML, разъясняющая работу и взаимодействие каждого их элемента. Она удачно сочетает в себе лучшие качества понятного учебного пособия, адресованного начинающим, и всеобъемлющего справочника, который всегда под рукой даже у опытных веб-программистов. Этот труд, ставший классическим, содержит все от базового описания синтаксиса и семантики до практических советов, поможет вам найти свой неповторимый стиль и в совершенстве овладеть языком веб-дизайна.

Описаны стандарты HTML 4.01, XHTML 1.0 и CSS2, приведен обзор еще не вступивших в силу стандартов XHTML 2 и CSS 3.

Уделено внимание и новейшим инициативам разработчиков XHTML (XForms, XFrames и модуляризации), а также основам XML. Рассмотрены: управление внешним видом документа с помощью таблиц стилей; работа с HTML-кодом, генерированным автоматически; работа с фреймами, интерактивными формами, динамическими документами; интеграция HTML-кода с мультимедийными данными, сценариями JavaScript и Java-апплетами.

Галактионов В.В. Расширяемый язык разметки XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру программных средств Интернет следующего поколения. Сообщение ОИЯИ, P10-2000-44, Дубна, 2000

Уделено внимание и новейшим инициативам разработчиков XHTML (XForms, XFrames и модуляризации), а также основам XML. Рассмотрены: управление внешним видом документа с помощью таблиц стилей; работа с HTML-кодом, генерированным автоматически; работа с фреймами, интерактивными формами, динамическими документами

С.В. Одиночкина Основы технологий XML - СПб: НИУ ИТМО, 2013. – 56 с.

В пособии представлено руководство по основным принципам использования языка разметки XML и связанным технологиям по дисциплине “Создание программного обеспечения инфокоммуникационных систем”.

Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>

Учебное пособие посвящено вопросам подготовки современных электронных изданий различного типа. Рассматриваются представление информации и форматы

данных, программные средства автоматизации проектирования электронных изданий и программные пакеты для их просмотра (браузеры). Освещаются различные аспекты создания и возможности гипертекстовых структур, гиперссылок и средств навигации, вопросы практической реализации электронных изданий в виде гипертекста и мультимедиа с помощью современных языковых и программных средств создания и редактирования HTML-документов; использование в электронных публикациях графики, анимации, оцифрованной аудио- и видеoinформации; синтез мультимедийных изданий. Анализируются современное состояние технологий подготовки электронных изданий и основополагающие тенденции их развития.

Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack , Питер , 2017 с.336

Рассматривается JavaScript - самый популярный язык написания клиентских сценариев. Это основополагающая технология для создания всевозможных анимаций и переходов. Без JavaScript практически невозможно обойтись, если требуется добиться современной функциональности на стороне клиента.

Николай Прохоренко - Разработка Web-сайтов с помощью Perl и MySQL, , Питер , 2015 с.336

Автор описывает в своей книге различные полезные приемы написания сценариев на Perl, а также рассказывает о том, как правильно работать с СУБД MySQL.

Прочитав эту книгу можно научиться:

Устанавливать необходимое программное обеспечение для создания сайтов на Perl и MySQL;

Основным понятиям и принципам работы с Perl;

Создавать формы;

Управлять файлами и каталогами;

Получать данные из Интернета при помощи Perl;

Создавать базы данных различной сложности;

Глава 1 Краткая характеристика языков гипертекстовой разметки

Основой современных технологий, используемых в Internet, являются языки разметки гипертекста и написания сценариев, апплетов и скриптов публикаций WWW (World Wide Web). Сегодня разработано достаточно много различных программных языков, либо специально ориентированных на использование в Web, либо включающих функции, предназначенные для Internet.

В 1986 г. Международная организация по стандартизации (ISO) приняла первый стандарт ISO-8879, озаглавленный «Standard Generalized Markup Language (SGML)». Этот стандарт был посвящен описанию языка SGML - обобщенного метаязыка, позволяющего строить системы логической и структурной разметки любых разновидностей текстов. [Брайан Пфаффенбергер, Стаивен Шафер, Чак Уайта, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2014 г., 72 с.].

Он был разработан для стандартизации обмена данными между разнородными программами и платформами. Язык SGML являлся не только мощным, но и очень сложным средством. Кроме того, SGML-документа было трудно интерпретировать без определений языка разметки, который хранится в определении типа документа DTD (**document type definition**). DTD необходимо было посылать вместе с SGML-документом или включать в документ, для того чтобы можно было распознать теги, созданные пользователем. Под тегами понимались специальные управляющие коды, записываемые в виде текста, ограниченного угловыми скобками

Необходимо отметить, что основной проблемой того времени оставалась проблема корректного отображения ранее созданных страниц текста ввиду достаточно большого разнообразия программ создания текстовых документов и различных аппаратно-программных платформ.

В связи с этим разработчики SGML старались создать программно- и аппаратно-независимые языки, с тем чтобы размеченный текст мог без труда интерпретироваться любой программой, работающей с любым устаревшим выводом.

Фактически SGML представлял собой не готовую систему разметки текста, а лишь удобный метаязык, позволяющий создавать такие системы для конкретных обстоятельств. Стандарт SGML определял лишь синтаксис записи элементов разметки текста - тегов и их атрибутов, а также содержал правила определения новых тегов и указания структурных отношений между ними. Для практической же разметки документов использовалось приложение SGML: набор определенных в соответствии со стандартом тегов, являющийся, по сути, формальным описанием структуры документа. Теги представляли собой последовательность символов, обрамляющих текст [Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 752 с.].

Однако язык SGML не получил массового распространения до тех пор, пока он не был выбран в качестве основы для нового языка разметки гипертекстовых документов. Этот язык (самое известное из приложений SGML) был назван **HTML** (Hyper Text Markup Language, «язык разметки гипертекста»).

По сути, язык HTML - это набор предписаний SGML, сформулированных в виде DTD. В случае языка HTML DTD хранится в браузере и эта DTD во много раз меньше, чем DTD SGML. Для передачи информации по Internet HTML используется протокол передачи гипертекстов (Hypertext Transfer Protocol) [Брайан Пфаффенбергер, Стивен Шафер, Чак Уайта, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2014 г., 752 с.].

HTML - это технология представления данных, не несущая содержательной информации, кроме заключенной в тегах. Значимость тегов в этом языке не определена, поэтому нельзя представить иерархию данных. В качестве платформы для приложений используются браузеры.

HTML был попыткой разрешить уже достаточно очевидное к тому времени противоречие между идеологией структурной разметки и потребностями пользователей, заинтересованных в первую очередь в гибких и разнообразных возможностях визуального представления информации.

Чтобы разрешить это противоречие, не нарушая основные концепции разметки гипертекста, авторы версии 3.0 ввели новое средство: иерархические стилевые спецификации (Cascading Style Sheets - CSS). Система CSS формально независима от HTML, имеет совершенно иной синтаксис, не

наследуется никаких идеологических ограничений и позволяет уже совершенно в иных терминах задавать параметры визуального представления для любого тэга HTML. С помощью CSS автор может указать, каким шрифтом, какого кегля набирать заголовки такого-то уровня. Причем это было достигнуто не введением новых тэгов, а разработкой механизма, позволяющего влиять на интерпретацию уже существующих тэгов логической разметки, с учетом контекста его употребления [Брайан Пфаффенбергер, Стивен Шафер, Чак Уайта, Билл Кароу. *HTML и CSS. Библия пользователя* 3-е изд., 2014 г., 752 с.]. Браузеры с текстовым интерфейсом или синтезаторы речи могут проигнорировать эту дополнительную информацию и использовать для оформления свои обычные средства.

В поддержку языка HTML была создана целая серия программных языков, таких, как Java, JavaScript, Perl, PHP и т.д. Там самым открылась возможность для создания динамических Web-страниц, включающих формы, таблицы, фреймы и другие элементы.

Упростив SGML для использования в Web, группа предложила расширяемый язык разметки **XML** (Extensible Markup Language - рекомендация W3C по статусу на февраль 1998 г.). XML представляет собой подмножество SGML, причем любой действительный документ XML является действительным документом SGML [Оригинал статьи The future of HTML, Part 2: XHTML 2.0. (www.ibm.com/developerworks/web/library/x-futhtml2.html)]. Как и SGML, XML - это метаязык, определяющий другие языки разметки для специфических целей. Например, язык синхронизированной интеграции мультимедиа (Synchronized Multimedia Integration Language - SMIL) базируется на XML.

Чтобы сделать Web-страницу интерактивной, необходим язык, выполняемый в контексте браузера. В качестве таких языков выступают Java, JavaScript, PHP, Perl и др. [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Создаваемые программы, называемые сценариями или скриптами, включаются в состав Web-страниц и распознаются и обрабатываются браузером отдельно от основного HTML-кода.

Обычно скриптовые языки используются для:

отображения диалоговых панелей и сообщений в статической строке браузера;

создания динамического содержимого страницы во время ее загрузки или уже после того, как она полностью загружена;

изменения содержимого страницы, атрибутов графического изображения, содержимого других HTML-элементов и отклика на события;

проверки вводимой пользователем информации и управления отсылкой содержимого форм;

навигации по другим страницам [Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack, Питер, 2017 с.33];

управления встроенными в Web-страницу объектами типа Java-апплетов и элементов ActiveX;

обеспечения связи Web-страницы с базами данных.

Глава 2 Языки гипертекстовой разметки

Язык разметки гипертекста HTML

Электронные версии и печатные издания на базе WWW подготавливаются, хранятся и используются в среде World Wide Web. WEB-публикации представляют собой многосложные гипертекстовые структуры, для создания которых был разработан специальный язык, получивший название «язык разметки гипертекста» (HTML). [Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 2 с.].

Язык HTML приобрел популярность в середине 90-х годов благодаря экспоненциальному росту числа пользователей сети Internet. К этому времени возникла необходимость стандартизации языка. Работу по созданию спецификации HTML взяла на себя организация World Wide Web Consortium (сокращенно - W3C). В ее задачу входило составление спецификации, отражающей современный уровень развития языка с учетом разнообразных предложений разработчиков браузеров.

Схема утверждения спецификации состоит в следующем. Консорциум W3C готовит проект спецификации, после обсуждения которого выпускает так

называемый черновой, рабочий (draft) вариант спецификации и предлагает его к обсуждению на определенный период. После обсуждения рабочий вариант спецификации может стать рекомендацией, т.е. официально признанным вариантом спецификации HTML [Брайан Пфаффенбергер, Стивен Шафер, Чак Уайт, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2014 г., 52 с.]

Июль 1997 г. ознаменовался выходом предлагаемой спецификации HTML 4.0, которая в декабре 1997 г. стала официальной рекомендацией. На сегодняшний день это последняя из принятых спецификаций. [Диэдре Хейз. Освой самостоятельно HTML и XHTML.10 минут на урок.3-е издание. 2012 г., 22 с.]

Язык разметки гипертекста (HTML) изначально был ориентирован на создание простых текстовых страниц для их последующей передачи адресату в системе электронной почты или других сервисах вычислительных сетей.

Сегодня этот язык превратился в средство, на котором базируются современные технологии создания и оформления всего многообразия электронных документов и их компонентов, использующих цвет, видео- и аудиоинформацию, различные растровые образы в качестве фона и даже встроенные прикладные программы. [Диэдре Хейз. Освой самостоятельно HTML и XHTML.10 минут на урок.3-е издание. 2012 г., 224 с.]

Язык HTML обеспечивает управление форматированием символов, заголовков, списков, связей и элементов изображений, командами для меню и интерактивных форм интерфейса пользователя. Пользователям Web предоставляется возможность контроля над компоновкой страниц с помощью таких конструктивных элементов, как таблицы, кадры, фреймы, средства для работы с таблицами, фоновыми изображениями, математическими уравнениями, мультимедийными элементами. Предусматривается возможность использования утилит, верхних и нижних индексных символов [Брайан Пфаффенбергер, Стивен Шафер, Чак Уайт, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2014 г., 22 с.]. Коды форматирования таблицы и работы с изображениями позволяют управлять отображением этих элементов, не дожидаясь загрузки всей таблицы или изображения целиком. Их сфера управления расширена и охватывает содержимое таблиц, кадров, фреймов и мультимедийных элементов.

Используемые коды языка HTML совместимы с современными браузерами.

Язык HTML применяет каскадные таблицы стилей (cascading style sheets - CSS1) . [Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 52 с.]. Эти таблицы стилей дают авторам Web возможность отделить информацию о компоновке и визуальной структуре страницы от ее содержания, делая большинство браузеров способными визуализировать содержательный материал Web-страницы.

Остановимся на некоторых характерных особенностях языка разметки гипертекста HTML.

Спецификация HTML определяет набор символов для документов HTML как набор однобайтовых символов ISO 8859/1 (латинский алфавит - Latin 1). Это набор из 256 символов, из которых первые 128 идентичны символам ASCII (ISO-646). Они должны быть совместимы с ISO 10646 и Unicode.[Галактионов В.В. Расширяемый язык разметки XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру программных средств Интернет следующего поколения. Сообщение ОИАЯИА, Р10-2000-44, Дубна, 2014.].

XHTML (англ. Extensible Hypertext Markup Language)

XHTML (англ. Extensible Hypertext Markup Language - Расширяемый язык разметки гипертекста) - язык разметки веб-страниц, по возможностям сопоставимый с HTML, однако является подмножеством XML. Как и HTML, XHTML соответствует спецификации SGML. Вариант XHTML 1.1 одобрен в качестве Рекомендации Консорциума Всемирной паутины (W3C) 31 мая 2001 года. [Справочник по спецификации XHTML 2.0 (<http://www.w3.org/TR/xhtml2>)].

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. Семейство типов документов XHTML базируется на XML и предназначено для работы с пользовательскими агентами на базе[XHTML 1.0: The Extensible HyperText Markup Language (Открытый язык разметки гипертекстов) Оригинал перевода:<http://www.stack.ru/~julia/XHTML/xhtml1.phtml>].

XHTML 1.0 (настоящая спецификация) представляет первый тип документов семейства XHTML. В ней три типа документов HTML 4 переформулируются в терминах XML 1.0. Она предназначена для использования в качестве языка содержимого, как соответствующего XML, так и, если соблюдены некоторые простые требования, работающего в конформных пользовательских агентах HTML 4. Разработчики, переносящие свои документы в XHTML 1.0, получают следующие преимущества:

Документы XHTML соответствуют XML. Как таковые они без труда просматриваются, редактируются и проверяются на корректность стандартными средствами XML[XHTML 1.0: The Extensible HyperText Markup Language (Открытый язык разметки гипертекстов) Оригинал перевода:<http://www.stack.ru/~julia/XHTML/xhtml1.phtml>].

Документы XHTML могут работать лучше, чем они работали в существующих пользовательских агентах, соответствующих HTML 4, а также в новых пользовательских агентах, соответствующих XHTML 1.0.

Документы XHTML могут использовать прикладные программы (например, скрипты и апплеты), базирующиеся на HTML Document Object Model или XML Document Object Model [DOM] .

По мере расширения семейства XHTML документы, соответствующие XHTML 1.0, будут с большей вероятностью совместимы с различными средами XHTML[Справочник по спецификации XHTML 2.0 (<http://www.w3.org/TR/xhtml2>)].

Семейство XHTML является следующим шагом в эволюции ИАнтернет. 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 XHTML.

Для XHTML можно применять множество технологий разработанных для XML. Например, XSLT и XPath.

Анализ XHTML проще и быстрее, чем HTML. Поскольку синтаксис XML строже, чем SGML, обработка XHTML возможна даже на мобильных телефонах с малыми ресурсами.

Разли́чия между XHTML и́ HTML.

Все элементы должны быть закрыты. Теги́, которые не и́меют закрывающего тега (напри́мер, `` и́ли `
`) должны и́меть на конце / (напри́мер, `
`).

Булевы атри́буты запи́сываются в развёрнутой форме. Напри́мер, следует пи́сать `<option selected="selected">` и́ли `<td nowrap="nowrap">`.

Все значе́ния атри́бутов обяза́тельно должны быть заклю́чены в двойные, ли́бо оди́нарные кавычки́.

И́мена тегов и́ атри́бутов должны быть запи́саны строчными́ буква́ми (напри́мер, `` вместо ``).

XHTML гораздо стро́же относи́тся к оши́бкам в коде; `<` и́ `&` везде, даже в URL, должны замещаться `<` и́ `&` соответственнo. По рекоменда́ции W3C браузеры, встрети́в оши́бку в XHTML, должны сообщи́ть о ней и́ не обрабаты́вать документ. Для HTML браузеры должны бы́ли попы́таться понять, что хотел сказа́ть автор [Мусси́ано, Кеннеди́.html и́ XHTML. Подробное руководство, 2002 г., 44 с.].

Коди́ровкой по умолча́нию является UTF-8 (в отли́чие от HTML, где коди́ровкой по умолча́нию является ISO 8859-1).

Отли́чия переходного (англ. transitional) XHTML от HTML незначи́тельны и́ предназна́чены ли́шь для при́ведения его в соответстви́е с XML. Остальные верси́и отли́чаются ли́шь набором тегов.

Верси́и XHTML.

XHTML 1.0 Переходный (Transitional): предназна́чен для лёгкой ми́грации и́з HTML 3.2 и́ для тех, кто и́спользует и́нлайн-фрэймы.

XHTML 1.0 Строгий (Strict): полностью отделя́ет содержи́е документа от оформле́ния (которое тепе́рь задаётся только через CSS), многи́е атри́буты (таки́е как, напри́мер, `bgcolor` и́ `align`) более не поддержи́ваются, и́х поведе́ние можно зада́вать только через табли́цу сти́лей.

XHTML 1.0 Фрэймовый (Frameset): и́спользуется, если́ необходи́мо раздели́ть окно браузера на несколько фрэймов.

XHTML 1.1 Модульный (Module-based): авторы могут импортировать дополнительные свойства в их разметку.

XHTML Основной (Basic): специальная облегчённая версия XHTML для устройств, которые не могут использовать полный набор элементов XHTML - в основном используется в миниатюрных устройствах, таких как мобильные телефоны. Подразумевается, что он заменит WML и C-HTML.

XHTML мобильного профиля (Mobile Profile): основанный на XHTML Basic, добавляет специфические элементы для мобильных телефонов.

XHTML 2.0. Пока в разработке. Синтаксис еще больше приближен к синтаксису XML. Также является модульным языком. [Справочник по спецификации XHTML 2.0 (<http://www.w3.org/TR/xhtml2>)].

Задачей XHTML 1.0 был переход HTML на свойственный XML словарь. Эта спецификация ввела ограничения синтаксиса XML в HTML: регистрозависимость, принудительное заключение в двойные кавычки значений атрибутов и симметричные теги. Таким образом XHTML 2.0 пытается решить проблемы HTML как языка для разметки web-страниц. [Диэдре Хейз. Освой самостоятельно HTML и XHTML.10 минут на урок.3-е издание. 2012 г., 24 с.].

Язык Java

Язык Java ориентирован на самые передовые технологии, связанные с сетями Internet. К сети Internet подключены компьютеры самых разных типов. Создавая Web-сервер в сети Internet, разработчик стремится к тому, чтобы им могло пользоваться как можно большее число пользователей, работающих на различных аппаратных платформах (Pentium PC, Macintosh, рабочие станции Sun и т.д) и в различных операционных системах[Итан Браун Web Development with Node and Express: Leveraging the Java Stack ,Питер , 2017 с.336]

. Созданное на базе Java программное обеспечение отвечает таким требованиям.

Программы, составленные на языке программирования Java, по своему назначению можно разделить на две группы:

К первой группе относятся приложения Java, предназначенные для автономной работы под управлением специальной интерпретирующей «машины Java». Реализации этой машины созданы для всех основных компьютерных платформ.

Вторая группа - это так называемые апплеты (applets). Апплеты представляют собой разновидность приложений Java, которые интерпретируются виртуальной машиной Java, встроенной практически во все современные браузеры.

Приложения, относящиеся к первой группе (просто приложения Java), - это обычные автономные программы. Так как они не содержат машинного кода и работают под управлением специального интерпретатора, их производительность заметно ниже, чем у обычных программ, составленных, например, на языке программирования C++. Однако не следует забывать, что программы Java без перетрансляции способны работать на любой платформе, что само по себе имеет большое значение в плане разработок для Internet Язык Java.

Апплеты Java встраиваются в документы HTML, хранящиеся на Web-сервере. С помощью апплетов можно сделать страницы Web-сервера динамичными и интерактивными. Из соображений безопасности апплеты (в отличие от обычных приложений Java) не имеют доступа к файловой системе локального компьютера. Все данные для обработки они могут получить только от Web-сервера. Более сложную обработку данных можно выполнять, организовав взаимодействие между апплетами и расширениями Web-сервера: приложениями CGI и ISAPI [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Язык Java является объектно-ориентированным и поставляется с большой библиотекой классов. Эти библиотеки классов Java значительно упрощают разработку приложений, предоставляя в распоряжение программиста средства решения наиболее распространенных задач.

Внутренняя реализация библиотек классов зависит от платформы. Однако все загрузочные модули, реализующие возможности этих библиотек, поставляются в готовом виде вместе с виртуальной машиной Java. Например, для операционной системы Windows поставляются библиотеки

динамической загрузки DLL, внутри которых заложены все функциональные возможности стандартных классов Java [Итан Браун Web Development with Node and Express: Leveraging the Java Stack, Питер, 2017 с.30].

Абстрагируясь от аппаратуры на уровне библиотек классов, программисты могут больше не заботиться о различиях в реализации программного интерфейса конкретных операционных систем. Это позволяет создавать по-настоящему мобильные приложения, не требующие при переносе на различные платформы перетрансляции и изменения исходного текста. К тому же в языке Java все базовые типы данных имеют фиксированную разрядность, которая не зависит от платформы. Поэтому программисты всегда знают размеры переменных в своей программе.

Функциональные возможности языка Java

Кратко перечислим основные возможности языка Java.

Компоненты в окне апплета. В окне апплета можно размещать следующие элементы управления Windows:

кнопки,

переключатели с независимой и зависимой фиксацией,

статические текстовые поля,

однострочные и многострочные поля редактирования текста,

списки,

полосы прокрутки [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Особенностью работы с элементами управления в Java является то, что их размещением в окне апплета занимается не программист, а специальная настраиваемая система Layout Manager, для которой достаточно задать один из пяти режимов работы.

Язык JavaScript

JavaScript разработан фирмой Netscape и ориентирован на создание и обработку скриптов Web-страниц, а также используется для написания сценариев интерактивного поведения Web-страниц Всемирной Паутины (WWW - World Wide Web). В этом плане он является вложенным в язык HTML, но не дублирует его функциональные возможности. В результате все методы оформления Web-страниц осуществляются языком HTML, а JavaScript предоставляет возможность «динамизации» Web-публикаций и способствует снижению трафика в сети [Дидре Хейз. Освой самостоятельно HTML и XHTML. 10 минут на урок. 3-е издание. 2012 г., 45 с.].

JavaScript напоминает язык Java. Различие заключается в том, что Java разрабатывался как универсальный объектно-ориентированный язык, а JavaScript рассматривается как быстрое и простое в использовании средство расширения возможностей Web-страниц и Web-серверов [Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack, Питер, 2017 с.89].

Особенностью обработки кода скриптов на JavaScript является то, что они обрабатываются интерпретатором браузера по мере загрузки документа. При этом формат кода преобразуется в более эффективный внутренний формат, который выполняется позднее. Скрипты можно располагать в любом месте HTML-страницы. При этом они обрамляются тегами `<SCRIPT></SCRIPT>`.

Поскольку скрипты обрабатываются по мере загрузки Web-страницы, возникает необходимость определения функций, используемых на HTML-странице до момента их вызова. Обычно функции, которые вызываются несколько раз, располагают внутри элемента HEAD страницы, так как расположенные там функции анализируются раньше, чем загрузится страница, а значит, на момент их вызова не возникнет ошибок. Однако существуют варианты функций, которые, наоборот, необходимо помещать в конце документа, так как код таких функций не должен анализироваться и выполняться раньше, чем произойдет полная загрузка страницы (например, функция подсчета всех URL-ссылок на странице).

Различные программы-браузеры поддерживают незначительно отличающиеся друг от друга версии JavaScript. Тем не менее в языке JavaScript можно выделить некоторую общую часть, определяющую его функциональные особенности [Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack, Питер, 2017 с.67].

Язык JavaScript позволяет вводить в Web-страницы следующие элементы: текст, рисунки, фреймы, слои, окна, формы для заполнения, а также некоторые другие элементы. Он также способен оперировать перечисленными элементами с помощью специальных, ассоциированных с определенными событиями скриптов.

Так как JavaScript обеспечивает взаимодействие пользователя с Web-страницей после её загрузки, разработчики обычно используют его для решения следующих задач:

динамическое добавление, редактирование и удаление HTML-элементов и их значений;

проверка содержимого web-форм перед отправкой на сервер;

создание на компьютере клиента cookie-файлов для сохранения и получения данных при последующих визитах.

Перед началом изучения языка следует познакомиться с его основными принципами:

чтобы добавить JavaScript-код в HTML-файл, его необходимо поместить внутрь тегов **script** и добавить в файл атрибут `text/javascript` (см. листинг 1);

все выражения в JavaScript оканчиваются точкой с запятой;

язык чувствителен к регистру символов;

имена всех переменных должны начинаться с буквы или символа подчеркивания;

можно использовать комментарии, чтобы выделить определенные строки в сценарии; комментарии должны начинаться с двойного прямого слеша (`//`), за которым следует текст комментария;

комментарии также можно использовать для отключения фрагментов сценария; для отключения нескольких строк можно использовать конструкцию `/*` фрагмент кода `*/`; любой код внутри `/**/` не будет запускаться во время выполнения [Итан Браун *Web Development with Node and Express: Leveraging the JavaScript Stack*, Питер, 2017 с.67].

PHP - язык написания сценариев

PHP - это язык написания сценариев, внедряемых в HTML-документ. Многие из синтаксиса данного языка заимствовано из C, Java и Perl с добавлением специфичных для PHP возможностей. Задача языка состоит в том, чтобы предоставить возможность Web-разработчикам создавать динамично изменяемые HTML-страницы. Язык PHP - это скрипт-язык (scripting language), встраиваемый в HTML, который интерпретируется и выполняется на сервере [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Первая используемая версия PHP стала доступна в начале 1995 г. и была известна как Personal Home Page Tools. К середине 1995 г. появилась вторая версия, которая называлась PHP/FI Version 2. Приставка FI присоединилась из другого пакета Rasmus, который ориентировался на обработку форм (Form Interpretator). PHP/FI компилировался внутри сервера Apache и использовал стандартный API Apache. PHP-скрипты оказались быстрее аналогичных CGI-скриптов, так как серверу не было необходимости порождать новый процесс. Язык PHP по возможностям приблизился к Perl, самому популярному языку для написания CGI-программ. Была добавлена поддержка множества известных баз данных (например, MySQL и Oracle). Интерфейс к GD-библиотеке позволял генерировать изображения «на лету». С этого момента началось широкое распространение PHP/FI. К 1996 г. PHP/FI был использован по крайней мере на 15000 Web-серверах Internet. К 1998 г. число Web-серверов, использующих PHP, возросло до 150000 сайтов.

В июне 1998 г. вышла новая версия, которая была названа PHP 3. К лету 1999 г. PHP 3 был включен в несколько коммерческих продуктов. По данным NetCraft на ноябрь 1999 г., PHP использовался более чем в 1 млн. доменов.

В декабре 1999 была выпущена новая версия PHP 4, в которой внутренний «движок» (Zend) был переписан. Производительность новой версии в десятки раз выше, чем у предыдущих.

Традиционно динамические WEB-страницы реализовались с использованием CGI-скриптов, написанных на Perl. Но CGI-скрипты очень плохо масштабируемы. Каждый новый вызов CGI требует от ядра порождения нового процесса, а это занимает процессорное время и тратит оперативную память. PHP предлагает другой вариант - он работает как часть Web-сервера и этим похож на ASP (Microsoft) [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Синтаксис PHP аналогичен синтаксису C или Perl. В этом языке нет строгой типизации данных и нет необходимости в действиях по выделению/освобождению памяти. Программы, написанные на PHP, достаточно легко читаемы. Написанный PHP-код легко зрительно прочитать и понять.

Рассмотрим некоторые функциональные особенности языка PHP.

Работа с базами данных. Одной из отличительных особенностей языка PHP является интеграция Web-страниц с базами данных. В настоящее время поддерживается достаточно широкий перечень баз данных в режиме работы в собственной системе команд и через ODBC: Oracle, Adabas D, Sybase, FilePro, mSQL, Velocis, MySQL, Informix, Solid, dBase, ODBC, Unix dbm, PostgreSQL.

Аутентификация пользователя средствами PHP. HTTP-аутентификация пользователя (клиента) в языке PHP доступна только при использовании модуля Apache, в который включен PHP-скрипт, использующий специальную функцию Header(). С ее помощью осуществляется отправка сообщения "Authentication Required" браузеру пользователя, открывающая диалоговое окно Username/Password, в котором пользователю предлагается заполнить поля username и password. После их заполнения сервером осуществляется вызов URL-адреса пользователя с дополнительными переменными, содержащими введенную информацию (\$PHP_AUTH_USER, \$PHP_AUTH_PW и \$PHP_AUTH_TYPE). В данном случае может быть обеспечена только «основная» аутентификация [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Язык Perl

Язык Perl был создан в 1986 г. как инструмент для администрирования и конфигурирования системных ресурсов в сети, состоящей из Unix-компьютеров. Аббревиатура Perl расшифровывается как практический язык для извлечения текстов и генерации отчетов (Practical Extraction and Report Language) [Николай Прохоренок - Разработка Web-сайтов с помощью Perl и MySQL, , Питер , 2015 с.36]. Этот язык первоначально создан для автоматического формирования отчетов из иерархии файлов системы оповещения об ошибках. Perl является интерпретируемым языком для сканирования текстовых файлов, извлечения из них информации и вывода на основе полученных данных текстовых отчетов. То есть программы Perl используются для выполнения сценариев Perl. Язык Perl является

межплатформенным и обеспечивает удобную работу с текстом. Он используется в основном для выполнения рутинных задач, которые, однако, могут быть достаточно трудны или плохо переносимы, а также сложны для кодирования в С или ином используемом в ОС UNIX языке.

Данный язык в определенном плане является универсальным и подобен языку С и рассмотренным выше. Синтаксис и грамматика языка Perl и рассмотренных ранее объектно-ориентированных языков имеют сходство. Основная идея состоит в инкапсулировании данных и программ (называемых методами) в виде объектов. В результате каждый объект становится полуавтономным, приватные (внутренние) данные и методы обособливаются так, чтобы они не создавали беспорядок в области данных. Объект взаимодействует с остальной частью программы с помощью публичных (доступных для вызова извне) методов [Николай Прохоренок - Разработка Web-сайтов с помощью Perl и MySQL, , Питер , 2015 с.109]. В Perl используются:

Классы - пакеты, которые могут обеспечивать методы.

Метод - подпрограмма, встроенная в класс или объект. Метод использует ссылку на объект или имя класса, передаваемое ему в качестве первого аргумента.

Объект - ссылка на элемент данных. В отличие от других ссылок этот элемент знает, к какому классу он относится.

Наследование - процесс порождения одного класса (дочернего или порожденного). В результате этого процесса дочерний класс может использовать методы родительского класса. [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Эти конструкции являются важной частью языков объектно-ориентированного программирования, и Perl в частности.

Определенные функции языка Perl получили широкое распространение в связи с развитием компьютерной сети Internet. Поэтому при дальнейшем рассмотрении данного языка основное внимание уделим использованию функций Perl для программирования в области WWW [Николай Прохоренок - Разработка Web-сайтов с помощью Perl и MySQL, , Питер , 2015 с.336].

Для удобства программистов множество регулярно требуемых функций в среде Perl реализовано в виде так называемых Perl-модулей. Эти модули "подключаются" к реализации языка и позволяют вместо разработки собственных функций для той или иной задачи вызывать библиотечные функции, созданные авторами модуля. Например, существуют модули для работы с графикой (сам по себе Perl не имеет никаких графических инструментов), модули для автоматизации обработки web-форм и взаимодействия с CGI. Общее количество Perl-модулей давно измеряется сотнями, но среди них есть несколько десятков, ставших более-менее стандартными.

Набор установленных (и доступных для пользовательских программ) на хостинг-площадке Perl-модулей часто определяется хостинг-провайдером. Впрочем, не так уж редко хостинг-провайдер позволяет клиенту устанавливать и свои дополнительные модули. Набор доступных Perl-модулей весьма актуален в тех случаях, когда при создании web-сайта планируется использовать готовое программное обеспечение на языке Perl. Дело в том, что для успешной работы этого программного обеспечения могут потребоваться Perl-модули, не установленные у хостинг-провайдера - и, как результат, программы не будут работать. Поэтому, прежде чем выбирать программное обеспечение для "запуска" на web-сайте, следует уточнить, что хостинг-площадка поддерживает требуемые модули. [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Важным преимуществом Perl является то, что вокруг этого языка уже много лет назад сложилось сообщество программистов, разрабатывающих публично доступные, бесплатные программы для решения различных задач. В том числе и программы для web-сайтов. Готовые решения оформляются как в виде отдельных Perl-скриптов (программ), так и в виде Perl-модулей. Например, подробнейший архив Perl-модулей, упорядоченный по типам задач, ими решаемых, можно найти на сайте CPAN.

Как известно, в основе Web-страниц лежит текст. Web-серверы общаются с браузерами при помощи текстового протокола HTTP (Hypertext Transfer Protocol). Именно текстовая ориентация во многом обеспечила гибкость, широту возможностей и успех WWW. Взаимодействие приложений в Web строится по принципу клиент/сервер. Клиентские браузеры запрашивают документы у Web-серверов - таких, как Apache. Диалог браузера с сервером определяется протоколом HTTP. В основном сервер просто пересылает содержимое некоторого файла, однако иногда Web-сервер запускает другую программу для отправки

документа, которая может представлять собой HTML-текст, графическое изображение или иной тип данных. Диалог сервера с программой определяется протоколом CGI (Common Gateway Interface), а запускаемая сервером программа называется CGI-программой или CGI-сценарием.

Сервер сообщает программе CGI, какая Web-страница была затребована, какие значения были переданы в HTML-формах, откуда поступил запрос, какие данные использовались при аутентификации и многое другое. [Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 234 с.]

. CGI-программа может быть написана на любом из следующих языков: AppleScript, C++, Fortran, PHP, Perl, TCL, Visual Basic, языке командного интерпретатора UNIX (shell) и т.д. Поскольку при создании Web-страниц разработчики имеют дело с текстом, файлами, сетевыми коммуникациями и двоичными файлами, Perl подходит для Web-программирования.

Язык VRML

Язык VRML был задуман весной 1994 г. на первой ежегодной конференции World Wide Web в Женеве. Здесь в числе прочих обсуждались проблемы создания VR-интерфейсов для доступа к WWW и необходимость создания единого общего стандарта описания трехмерной сцены, включающей гиперсвязи по аналогии с HTML. Основными требованиями к языку являлись независимость от платформы, расширяемость и возможность работы по низкоскоростным каналам связи. Там же было предложено название VRML (Virtual Reality Markup Language). Впоследствии слово Markup было заменено словом Modeling, что отражает графическую природу языка. В качестве базы языка VRML был выбран урезанный формат текстового файла SGI Open Inventor, дополненный сетевыми возможностями. Формат Open Inventor позволял описать трехмерную сцену с графическими объектами, источниками света и текстурами. [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

В язык была добавлена анимация. Ряд объектов, называемых Interpolator, позволял описывать заранее определенное движение, например полет птицы, движение робота или маятника часов. Стало возможным описывать изменение цвета объекта или создавать объекты, изменяющие свою форму. Языку была добавлена интерактивность.

Особо важное значение имела возможность задания поведения объекта с помощью языков программирования, в результате чего графические объекты стали «интеллектуальными», поскольку их поведение могло описываться сколь угодно сложными программами.

Спецификация языка позволила разработчику использовать различные языки программирования (Java, JavaScript и т.д.) [Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack , Питер , 2017 с.209]. Открытость и расширяемость в сочетании с возможностью сетевого взаимодействия языка VRML 2.0 подготовили почву для создания интерактивных VRML-миров в Internet.

Язык VRML создавался как независимый от HTML, и для передачи файлов по Internet в нем предусмотрены те же самые протоколы. Поэтому концепции Internet и Web применимы и к VRML. Просмотр VRML-документов возможен с помощью VRML-браузера или HTML-браузера с дополнительным VRML-модулем. Унифицированный указатель ресурсов URL для VRML-документа выглядит так же, как указатель для Web-публикации, за исключением расширения. Для Web-документа используются расширения *.htm или *.html, а для VRML-документа - *.wrl, *.wrz или *.Flr.

Язык VRML позволяет создавать при помощи текстовых команд сложные трехмерные сцены на экране ПК. Эти команды описывают многоугольные объекты и специальные эффекты для имитации освещения, окружающей обстановки и для придания реалистичности изображению [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Для сценариев VRML подходит язык JavaScript, но для узла сценария допускается применение и обычных языков программирования. Сценарии для миров VRML чаще всего составляются на языке Java. Разработчики VRML предпочитают язык Java, потому что этот язык программирования ориентирован на Internet. Большинство поставщиков VRML-продуктов включают в свои комплекты браузеры для работы с Java.

VRMLScript - подмножество языка JavaScript, которое предусматривает непосредственно спецификация VRML 2.0. Хотя язык VRMLScript не пользуется такой широкой популярностью, как JavaScript, он достаточно полезен, в особенности потому, что предназначен специально для миров VRML.

В основе языка VRML лежат три основных понятия: форма, сцена и объект.

Формы представляют собой объемные выпуклые геометрические фигуры, для которых определены операции добавления к уже присутствующим в виртуальном мире объемным телам и вычитания из них. Операция добавления формы создает в пустом пространстве сплошное тело, соответствующее внесенной форме. Части добавляемой формы, попадающие в уже занятый объем пространства, не создают в нем новых граней или границ. При вычитании, наоборот, из тела удаляются части, соответствующие вычитаемой форме. Таким образом, формы играют роль элементов конструкции при построении виртуального мира.

Сценой называется создаваемое трехмерное пространство. Любая сцена может включать две различные части: «статическую» и «мобильную». Статическая часть сцены образуется добавлением объемных фигур из галереи форм или удалением частей объема уже имеющихся в сцене фигур при операциях вычитания форм.

Мобильная часть сцены образована объектами. Они также могут содержать статическую часть, взаимное положение фрагментов которой нельзя изменить. Однако положение, ориентация, пропорции и размеры объекта в сцене могут быть изменены.

Объект представляет собой как бы сцену в сцене, построенную в своей собственной системе координат, с другим, изменяемым пользователем началом отсчета, направлением и масштабом осей. Объекты могут включать в себя объекты-потомки, положение которых по отношению к статической части родительского объекта может изменяться. Операции изменения ориентации, размеров, положения, вырезания и удаления родительского объекта выполняются одновременно и для его потомков.

Совокупность объектов и их взаимную подчиненность в сцене можно наглядно отобразить в виде иерархической древовидной структуры. Все узлы дерева, за исключением узла сцены - корня, образуют ее мобильную часть, размеры и структура которой могут изменяться независимо от геометрических фигур статической части сцены [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Каждая грань объекта в виртуальном мире имеет набор свойств, описывающих ее материал:

Цвет. Пользователь может задать цвет любой грани. При этом вся грань будет окрашена в выбранный цвет не равномерно, а с учетом ее ориентации по

отношению к источнику света в сцене.

Текстура. На грань можно наложить изображение. Текстура будет наложена на всю грань, с автоматическим повтором рисунка, если размеры грани больше размеров элемента изображения текстуры. При наложении текстуры цвет грани не виден.

Прозрачность. Можно установить коэффициент прозрачности материала грани. Например, если прозрачность материала 50%, то грань станет полупрозрачной и в окне будут видны как ее материал, так и предметы, расположенные позади нее. Установка параметра прозрачности в 100% сделает грань невидимой. Подбором прозрачности грани можно имитировать стеклянные поверхности и другие элементы сцены.

Объекты, описываемые VRML-программой, называются **узлами** (nodes). При создании трехмерных объектов можно использовать встроенные узлы, такие, как конусы, цилиндры, кубы и сферы. Описав форму объекта, задают текстуру, освещенность, положение камеры и трансформацию объектов.

Особенности VRML

Язык VRML (Virtual Reality Modeling Language) предназначен для описания трехмерных изображений и оперирует с формами, сценами и объектами, описывающими геометрические фигуры и их расположение в пространстве [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

VRML-файл представляет собой обычный текстовый файл, интерпретируемый браузером. Поскольку большинство браузеров не имеет встроенных средств поддержки VrmI, для просмотра VrmI-документов может использоваться вспомогательная программа - VrmI-браузер, например Live3D или Cosmo Player.

Язык XML

XML - это расширяемый язык разметки (Extensible Markup Language), разработанный специально для размещения информации в World Wide Web, наряду с HTML, который давно стал стандартным языком создания Web-страниц. [Галактионов В.В. Расширяемый язык разметки XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру программных средств Интернет следующего

поколения. Сообщение ОИЯИ, P10-2000-44, Дубна, 2014.].

В отличие от HTML, вместо использования ограниченного набора определённых элементов вы имеете возможность создавать ваши собственные элементы и присваивать им любые имена по вашему выбору.

XML решает ряд проблем, которые не решает HTML, например:

Представление документов любого (не только текстового) типа, например, музыки, математических уравнений и т.д.[Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Сортировка, фильтрация и поиск информации.

Представление информации в структурированном (иерархическом) виде.

В зависимости от уровня соответствия стандартам документ может быть "верно сформированным" ("well-formed"), либо "валидным" ("valid"). Вот несколько основных правил создания верно сформированного документа:

Каждый элемент XML должен содержать начальный и конечный тэг (либо пустой тэг типа <TAG />, который может нести информацию посредством своих атрибутов).

Любой вложенный элемент должен быть полностью определён внутри элемента, в состав которого он входит.

Документ должен иметь только один элемент верхнего уровня.

Имена элементов чувствительны к регистру.

Есть три основных способа сообщить браузеру, как отображать каждый из созданных вами XML-элементов:

Каскадная таблица стилей (Cascading Style Sheet - CSS) или расширяемая таблица в формате языка стиливых таблиц (Extensible Stylesheet Language - XSL).

Связывание данных. Этот метод требует создания HTML-страницы, связывания с ней XML-документа и установления взаимодействий HTML-элементов с элементами XML. В дальнейшем HTML-элементы автоматически отображают информацию из связанных с ними XML-элементов.[Галактионов В.В. Расширяемый язык разметки

XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру программных средств Интернет следующего поколения. Сообщение ОИЯИ, P10-2000-44, Дубна, 2014.].

Написание сценария. Этот метод требует создания HTML-страницы, связывания с ней XML-документа и получение доступа к XML-элементам с помощью кода сценария JavaScript или VBScript.

Индивидуальный пользователь, компания или комитет по стандартам может определить необходимый набор элементов XML и структуру документа, которые будут применяться для особого класса документов. Подобный набор элементов и описание структуры документа называют XML-приложением или XML-словарём [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

XML-приложение обычно определяется созданием описателя типа документа (DTD), который является допустимым компонентом XML-документа. DTD устанавливает и определяет имена элементов, которые могут быть использованы в документе, порядок, в котором элементы могут появляться, и доступные к применению атрибуты элементов. DTD обычно включается в XML-документ и ограничивает круг элементов и структур, которые будут использоваться. Примечание: приложение XML Schema позволяет разрабатывать подробные схемы для ваших XML-документов с использованием стандартного синтаксиса XML и является альтернативой DTD.

Термин "примитив" в широком смысле относится к любому из следующих типов единиц хранения информации для XML-документов:

Собственно XML-документ как единое целое (файл).

Внешнее подмножество DTD (файл).

Внешний файл, определённый как внешний примитив в DTD и допускающий использование посредством ссылки (файл).

Строка в кавычках, определённая как внутренний примитив в DTD и допускающая использование посредством ссылки (не файл).

Однако чаще термин "примитив" используется в узком смысле, а именно - в двух последних смыслах.[Галактионов В.В. Расширяемый язык разметки XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру

программных средств Интернет следующего поколения. Сообщение ОИЯИ, P10-2000-44, Дубна, 2014.].

Механизм примитивов может оказаться удобным для сокращения времени набора документа, а также может облегчить внесение изменений в документ. Кроме того, механизм примитивов необходим при включении не XML-данных в XML-документ. В принципе, механизм примитивов сходен с определением констант в различных языках программирования.

Примитивы классифицируются по трём признакам:

Общие и параметрические. Общий примитив включает текстовые или нетекстовые данные, которые вы можете использовать внутри корневого элемента.

Параметрический примитив содержит XML-текст, который может быть помещён в DTD.

Внутренние и внешние. Внутренний примитив содержится внутри строки в кавычках. Внешний примитив содержится в отдельном файле [Гасов В.М. Цыганенко А.М. Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>].

Разбираемые и неразбираемые. Разбираемый примитив содержит XML-текст (символьные данные, разметка или и то, и другое). Когда вы вставляете ссылку на разбираемый примитив в документ, ссылка замещается содержимым примитива. Синтаксический анализатор разбирает содержимое примитива точно так же, как он сканирует текст, непосредственно введённый в документ. Неразбираемый примитив может содержать как XML-данные, так и не XML-данные. Его содержимое нельзя непосредственно вставить в документ посредством ссылки.

Реально в XML поддерживаются пять типов примитивов:

Общие внутренние разбираемые.

Общие внешние разбираемые.

Общие внешние неразбираемые.

Параметрические внутренние разбираемые.

Параметрические внешние разбираемые.

Примитив создаётся путём объявления его в DTD документа [С.В. Одиночкина Основы технологий XML - СПб: НИУ ИТМО, 2013. - 56 с.].

Заключение

Глобальная информатизация общества приводит к тому, что потребность в информации, растет с каждым новым пользователем сети. При этом задачей специалистов в области информационных технологий обеспечить пользователей полной и достоверной информацией путем простого и удобного для пользователей доступа к накопленным массивам данных.

Гипертекстовая технология позволяет быстро и точно осуществить поиск необходимой информации не только в рамках отдельных документов на компьютера и web-страниц, созданных с их использованием, но и в локальных вычислительных сетях и всемирной сети Internet.

Идея гипертекстовой информационной системы состоит в том, что пользователь имеет возможность просматривать документы (страницы текста) в том порядке, в котором ему это больше нравится, а не последовательно, как это принято при чтении книг.

Простой на первый взгляд механизм построения ссылок оказывается довольно сложной задачей, т. к. можно построить статические ссылки, динамические ссылки, ассоциированные с документом в целом или только с отдельными его частями.

Расширение понятия гипертекста происходит за счет других информационных ресурсов, включая графику, аудио- и видео-информацию, до понятия гипермедиа.

Безусловно у электронных гипертекстовых систем есть множество преимуществ, по сравнению с обычными печатными текстами. Например:

- двунаправленный характер ссылок, по сравнению с направленными только вперед ссылками обычных текстов;
- комментарии к тексту могут быть сделаны в самом тексте, не нарушая в то же время его целостность;

- однажды пройденный маршрут легко запоминается и может быть автоматически пройден многократно; это равносильно тому, что читатель формирует свой личный текст;

- электронная форма представления текста позволяет автоматизировать процесс формирования массива ключевых слов путем автоматического компьютерного анализа текста, и прочие преимущества.

Но при всем многообразии преимуществ прослеживается и ряд недостатков:

- у пользователя теряется ощущение собственного положения и движения в многомерном документе и появляется дополнительная умственная нагрузка для выбора оптимального пути изучения материала;

- чтобы электронная гипертекстовая система была эффективной для пользователя, электронное информационное пространство должно быть достаточно обширным, а применяемые технические средства достаточно мощными;

- создание электронных гипертекстовых систем требует новых подходов к проблеме защиты авторского права: во-первых, это совместный труд большого числа специалистов (в том числе программистов); во-вторых, развитый аппарат электронных гипертекстовых систем обеспечивает возможность непосредственного включения в данную систему текстов других авторов.

При подготовке данной курсовой работы был изучен материал из нескольких книг, в которых поднимались вопросы гипертекстовой технологии, так же были использованы материалы глобальной сети Интернет.

Материал данной курсовой работы дает необходимые сведения о языках гипертекстовой разметки и пищу для размышлений, о нем можно рассказать еще много интересного и познавательного, для этого следует обратиться к специальным статьям, книгам и прочей информации, посвященной различным аспектам вопросов, посвященным гипертекстовым технологиям.

Список использованной литературы

1. Брайан Пфаффенбергер, Стивен Шафер, Чак Уайт, Билл Кароу.html, XHTML и CSS. Библия пользователя 3-е изд., 2006 г., 752 с.
2. Дидре Хейз. Освой самостоятельно HTML и XHTML.10 минут на урок.3-е издание. 2002 г., 224 с.

3. Муссиано, Кеннеди.html и XHTML. Подробное руководство, 2002 г., 752 с.
4. Галактионов В.В. Расширяемый язык разметки XML (Extensible Mark-up Language): промышленный стандарт, определяющий архитектуру программных средств Интернет следующего поколения. Сообщение ОИЯИ, P10-2000-44, Дубна, 2000.
5. Оригинал статьи The future of HTML, Part 2: XHTML 2.0. (www.ibm.com/developerworks/web/library/x-futhtml2.html)
6. Справочник по спецификации XHTML 2.0 (<http://www.w3.org/TR/xhtml2/>)
7. Гасов В.М. Цыганенко **А.М.** Методы и средства подготовки электронных изданий Учебное пособие <http://www.hi-edu.ru/e-books/xbook081/01/part-008.htm>
8. XHTML 1.0: The Extensible HyperText Markup Language (Открытый язык разметки гипертекстов) Оригинал перевода: <http://www.stack.ru/~julia/XHTML/xhtml1.phtml>
9. Николай Прохоренок - Разработка Web-сайтов с помощью Perl и MySQL, ,Питер , 2015 с.336
10. Итан Браун Web Development with Node and Express: Leveraging the JavaScript Stack ,Питер , 2017 с.336