

# Программирование в WEB - системах

## 1. Обоснование постановки вопроса

Анализируя современное состояние технологий Internet и Intranet, не сложно заметить, что наиболее яркой частью сети Internet и корпоративных Intranet-сетей являются средства World Wide Web (WWW, 3W, Web-система). Именно Web-система имеет массового пользователя, проводящего часы в путешествиях по Всемирной паутине, и именно Web-система является притягательным центром для многочисленных фирм, организующих свои "представительства" в сети Internet.

Даже в России функционирует около 2,5 тыс. общедоступных Web-серверов (в США их больше в 100 раз) и 50 тыс. пользователей России работают с Internet в режиме on-line (данные из <http://www.relcom.ru/Internet/Duma/InternetRussia>). Оставив в стороне оценку общего количества серверов и конечных пользователей "у нас и у них", обратим внимание на тот факт, что в мире существуют сотни тысяч организаций и фирм, поддерживающих функционирование Web-серверов как для своего представительства, так и с целью предоставить Web-страницы в аренду самым разным потребителям. Указанная поддержка функционирования Web-серверов предусматривает следующие виды деятельности:

1. Информационное наполнение и актуализация уже созданных разделов, т.е. подготовка материалов к Web-публикации, редактирование, дизайн, соблюдение единого стиля и единообразного оформления Web-страниц, поддержка связности Web-документов и т.д.

2. Расширение возможностей Web-сервера за счет разработки новых служб и средств (приложений). Например, создание интерактивных Web-страниц, организация разных видов поиска информации, учет посетителей (статистика), регистрация пользователей, предоставление регламентированного доступа к информации, организация доступа к базам данных и т.д.

3. Администрирование Web-сервера как компонента системы World Wide Web, функционирующей в глобальной сети Internet. Здесь кроме принципиально необходимой работы по первоначальной настройке Web-сервера и поддержанию его в работоспособном состоянии существуют проблемы кодировок (особенно русификации), организации доступа к другим ресурсам Internet (FTP, WAIS и т.д.), обеспечения конфиденциальности доступа к той или иной информации и т.д.

Как следует из перечисленных задач, специалисты, ведущие Web-сервер, должны иметь весьма высокую квалификацию в достаточно разных областях. На одной границе спектра профессий, связанных с ведением Web-сервера, находится специалист по конкретной предметной области, литератор или публицист - автор будущих Web-страниц, на противоположной границе - системный администратор, наблюдающий за функционированием Web-

сервера наряду с другими компонентами того узла Internet, где размещен Web-сервер.

Анализируя функциональные обязанности специалистов, необходимых для ведения Web-сервера, мы остановимся только на программистах. Именно их роль в решении трех перечисленных выше задач (информационное наполнение, расширение возможностей и администрирование Web-сервера) будет предметом нашего рассмотрения. Хотя вопросы дизайна, литературного содержания, маркетинга, менеджмента и другие особенности Web-сервера являются не менее важными, но мы будем рассматривать вопросы применения сетевых технологий и программирования в Web-системе. Точнее говоря, предметом нашего рассмотрения будут специфика программирования при решении задач ведения Web-сервера и вопросы подготовки специалистов по Web-программированию. Специфика Web-программирования, обусловленная особенностями Web-системы, и массовая потребность в специалистах, способных вести сервер и создавать необходимые приложения, заставляют обратить внимание на Web-программирование как на учебную дисциплину.

Исходя из этих целей, далее в работе:

1. Кратко описаны особенности архитектуры и динамики функционирования системы WWW;
2. Проанализирован состав программных средств WWW и подчеркнута их деление на системные и прикладные;
3. Выявлены основные направления работ по программированию для Web-системы;
4. Выполнен анализ круга задач подготовки кадров по программированию для WWW;
5. Даны рекомендации по тематике и практическому наполнению дисциплины "Программирование для WWW".

## **2. Краткая характеристика World Wide Web**

Не пытаясь дать строгого определения WWW, обратим внимание на генетическую связь этой системы с информационно-поисковыми системами (ИПС) и глобальными сетями. По существу, WWW представляет собой результат применения возможностей доступа к территориально распределенной информации для создания глобальных гипертекстовых и мультимедиа ИПС.

Возможности доступа к территориально-распределенной информации обеспечивает для WWW всемирная сеть Internet или та локальная компьютерная сеть, где Web-средства используются (в режиме Intranet). Наследуя базовые черты ИПС, Web-система в основном развивается как хранилище слабоструктурированной, разноплановой и часто несогласованной информации и тем отличается от баз данных, где информация структурирована и взаимосвязана.

Широкое применение Web-система получила из-за простоты опубликования информационных материалов и удобства доступа к информационным ресурсам.

Указанные достоинства привели к широкому использованию Web-технологии не только в Internet, но и в глобальных и локальных корпоративных сетях, обозначаемых в этом случае термином Intranet.

И в Internet и в Intranet WWW является прикладной службой компьютерной сети (всемирной, глобальной, локальной). Сеть обеспечивает передачу данных между компьютерами. Правила передачи регламентируются протоколами и системой адресации. В Internet и Intranet принята система протоколов TCP/IP, обеспечивающая для WWW транспортный уровень передач. Как и ряд других прикладных служб сети Internet, Web-система использует схему клиент-сервер, т.е. в каждом сеансе программные средства клиента и сервера совместно решают конкретную прикладную задачу, выполняя информационное обслуживание пользователя. Схематично взаимодействие Web-клиента и Web-сервера укладывается в общую схему клиент-сервер (рис. 1).

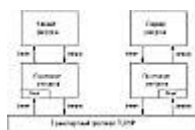


Рис. 1. Схема клиент-сервер в Internet

Как уже сказано, WWW является гипермедиа системой, т.е. представляет собой сеть Web-узлов, содержащих гипермедиа-документы и связи, позволяющие из одного документа ссылаться на другие, размещенные как на том же узле, так и на других. Представление Web-системы с точки зрения пользователя, иллюстрирует схема на рис. 2.

Основу Web-системы составляют следующие компоненты:

1. Универсальная адресация (идентификация) ресурсов, обеспеченная применением универсальных идентификаторов ресурсов (URI);
2. Механизмы обменов и доступов к ресурсам, регламентируемые протоколом HTTP (протокол ресурса на рис. 2);
3. Гипертекстовая технология, обеспечивающая навигацию и перемещение по мультимедиа-документам - реализована с помощью языка HTML (Hyper Text Markup Language - язык гипертекстовой разметки), затем расширена средствами VRML (VRML - Virtual Reality Modeling Language - язык программирования виртуальной реальности);
4. CGI - Common Gateway Interface - стандарт, определяющий форматы данных для обмена информацией между сервером, другими программными средствами Web-узла и браузером.

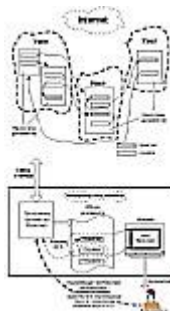


Рис 2. Взаимодействие пользователя с Web-системой

Именно эти компоненты на основе общих принципов построения сети Internet и в особенности на базе системы протоколов TCP/IP сделали возможным функционирование WWW.

Обратим внимание на тот факт, что, общаясь с WWW, пользователь в каждый конкретный момент времени устанавливает связь только с одним Web-узлом, т.е. взаимодействие пользователя с WWW всегда укладывается в схему клиент-сервер (рис. 1), несмотря на то, что серверы, т.е. Web-узлы, могут сменяться даже во время одного сеанса, а управляет этой сменой узлов пользователь (клиент) с помощью активации ссылок в изображении просматриваемого документа.

Рассмотрим отдельно задачи Web-узла, т.е. вычислительной машины с Web-сервером, и Web-клиента - вычислительной машины, на которой выполняется программа просмотра Web-документов (Web-браузер). Для наших целей нелишне отметить, что Web-сервер и Web-браузер могут быть размещены на одной автономной вычислительной машине, могут быть связаны через локальную сеть, могут находиться как угодно далеко и быть связаны средствами Internet.

По запросу клиента (браузера) Web-сервер (см. нумерацию на рис 3):

1. Отправляет ASCII-коды HTML-страниц (или VRML-документа), включающие при необходимости команды языка JavaScript;
2. Отсылает двоичный код запрошенного ресурса (изображения, аудиопотока, видеопотока...);
3. Отсылает байт-код Java-аплета;
4. Принимает конкретную информацию от пользователя (когда браузер запрашивает эту информацию, предлагая пользователю заполнить HTML-форму) и помещает ее в базу данных на сервере. Так, например, регистрируются новые пользователи;
5. Принимает сообщения от пользователя и регламентирует доступ к ресурсам Web-узла на основе анализа принятой информации (проверка паролей, контроль оплаты за доступ и т.д.), т.е. обеспечивает авторизованный доступ;
6. Принимает информацию от пользователя и в зависимости от нее динамически формирует HTML-страницы, либо VRML-документы, обращаясь, при необходимости, к базам данных и существующим на Web-узле HTML-страницам и VRML-документам.

Пункты 1-3 являются типичными для современных Web-серверов.

Пункты 4-6 могли бы исполняться Web-сервером, но программу-сервер в этом случае необходимо было бы специальным образом перепрограммировать каждый раз, когда требуется учитывать новые типы запросов от пользователей. Например, при подключении к серверу новой базы данных, информацию из которой нужно оформлять в виде динамических HTML-страниц пришлось бы изменять программы Web-сервера.

Чтобы устранить необходимость таких модификаций программы Web-сервера, на Web-узле используются CGI-скрипты или CGI-модули - специальные программы, учитывающая требования CGI-стандарта.

Основные требования к CGI-скрипту - соответствие соглашениям той операционной системы, под управлением которой работает Web-узел. Другими словами, CGI-скрипты, как и Web-сервер - это прикладные программы, работающие в операционной системе Web-узла. Исходный текст CGI-скрипта может быть написан на любом языке программирования, доступном в операционной системе Web-узла. Часто для написания CGI-скриптов используются языки Си, Си++, Perl, Tcl и другие.

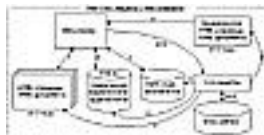


Рис. 3. Информационные взаимосвязи компонентов Web-узла

Получив ответ Web-сервера, клиент (см. рис 4):

1. Визуализирует HTML-страницу либо VRML-документ в окне браузера;
2. Интерпретируя команды JavaScript, модифицирует (обогащает) образ HTML-страницы;
3. Интерпретируя байт-код Java-апплета, дополняет новыми возможностями образ HTML-страницы;
4. Ведет диалог с пользователем, заполняющим формы, и создает новые запросы к Web-серверу;
5. С помощью утилит воспроизводит коды аудиопотоков и видеопотоков, т.е. обеспечивает мультимедийный режим;
6. Моделирует виртуальную реальность, просматривая VRML-документы.



Рис.4. Компоненты Web-клиента

Перечисленные задачи Web-клиента обеспечиваются возможностями Web-браузера и вспомогательными программами (утилитами), размещенными на машине клиента. Здесь следует отметить, что стандартов на построение Web-клиента и Web-браузера пока нет, и компонентный состав Web-клиента может отличаться от показанного на рис. 4. Например, программы для просмотра трехмерных сцен могут входить в состав Web-браузера, а могут быть оформлены в виде VRML-браузера, отдельно работающего на машине Web-клиента.

### 3. Web-программирование

Программирование для Web-узла включает:

1. Разработку и модификации программ Web-сервера;
2. Разработку сложных HTML-страниц, например, с командами на языке JavaScript;
3. Разработку CGI-скриптов;
4. Разработку Java-апплетов.

На стороне (на машине) клиента Web-системы программирование может потребоваться:

1. Для разработки и модификации браузера;
2. Для расширения возможностей браузера за счет создания специализированных приложений (анимация, мультимедийные возможности и пр.).

Рассматривая вопросы Web-программирования, необходимо обратить внимание на стандартную для компьютерной индустрии ситуацию: наличие в Web-системе общесистемной и прикладной частей. Программирование общесистемной части включает, например, разработку новых и модификацию уже существующих Web-серверов и Web-браузеров. Такими разработками занимаются обычно крупные специализированные фирмы, имеющие возможность вкладывать большие средства, как в создание собственно программных продуктов, так и в их продвижение на рынок программных средств.

Распределенные по многочисленным Web-узлам массивы HTML-страниц, VRML-документы, Java-апплеты, CGI-скрипты, файлы ресурсов и прикладные мультимедийные программы в совокупности образуют прикладную часть Web-системы. Именно разработка прикладной части Web-системы является массовой деятельностью, требующей привлечения большого количества программистов, дизайнеров, авторов Web-страниц, редакторов и других специалистов, областью деятельности которых раньше были (и остаются) издательское дело, кино-фото промышленность, телевидение и реклама.

Программирование, относящееся к прикладной части Web-системы, предусматривает:

1. оформление стандартных (типичных) HTML-страниц (хотя эту работу можно не относить к программированию, т.к. авторами HTML-страниц могут быть люди вовсе не знакомые с классическим программированием);
2. разработку сложных HTML-страниц, включающих коды JavaScript и диалоговые формы;
3. создание VRML-страниц для воспроизведения трехмерных объектов и моделирования виртуальной реальности;
4. создание специализированных программ для обработки (на стороне клиента) файлов, получаемых от узлов сети Internet (например, воспроизведение мультимедийных файлов).
5. разработку CGI-скриптов для обработки (на стороне сервера) специфических запросов от Web-клиентов;

- б. разработку Java-апплетов для решения (на стороне клиента) нетиповых задач воспроизведения HTML-страниц (анимация, нестандартный диалог, графические построения и т.д.)

В рамках вопросов, относящихся к Web-программированию, наиболее важными для широкомасштабной подготовки специалистов являются темы, относящиеся к разработке прикладных систем на базе промышленных аппаратно-программных средств, входящих в Web-систему. Другими словами, массовому автору Web-публикаций нужно иметь квалификацию прикладного Web-программиста, причем особое внимание следует уделять:

- разработке HTML-страниц, включая кодирование на JavaScript;
- разработке VRML-документов;
- программированию CGI-скриптов;
- программированию Java-апплетов.

Обратите внимание, что результаты деятельности прикладного Web-программиста (HTML-страницы, VRML-документы, скрипты, апплеты) должны размещаться на Web-узле, но только CGI-скрипты выполняются на машине Web-узла. HTML-страницы, VRML-документы и Java-апплеты всегда интерпретируются программными компонентами Web-клиента после их получения от Web-узла.

#### **4. Методические аспекты преподавания Web-программирования**

Практически в каждой учебной дисциплине, относящейся к программированию, должны быть отражены:

- базовые принципы (теория предмета);
- особенности современной аппаратно-программной реализации базовых принципов;
- приемы практической работы (ремесленные навыки).

Процентное соотношение между тремя перечисленными частями существенно зависит от целей обучения и предварительной подготовки обучаемых. Однако, в любом случае трудность постановки и последующего преподавания практически любого курса в области программирования связана с фантастической скоростью изменения аппаратно-программных средств, реализующих базовые принципы. Именно эта изменчивость среды (аппаратно-программной реализации) заставляет преподавателей вновь и вновь пересматривать как набор, так и содержание учебных дисциплин, обеспечивающих подготовку программистов и специалистов по информационным технологиям.

В работе по модификации и обновлению содержания учебных курсов приходится постоянно сохранять "равновесие" между глубиной проработки теоретических вопросов, ясностью описания современного состояния среды реализации и практической направленностью обучения специалиста. Другими

словами, процентное соотношение между указанными частями, при котором должно соблюдаться "равновесие", существенно зависит от целей обучения.

При постановке курса программирования для World Wide Web обучаемым может быть:

- ученик колледжа, специализирующийся на подготовке электронных публикаций (секретарь-машинистка со знанием информационных технологий);
- учащийся техникума, получающий квалификацию, которую мы условно назовем "информатика";
- студент любого высшего учебного заведения, не специализирующийся в области сетевых информационных технологий и программирования ("предметник");
- студент вуза, обучающийся по специальностям, непосредственно связанным с сетевыми информационными технологиями и программированием (обобщенно назовем его программистом).

Остановимся в нашей статье только на потребностях вузов, причем, как говорилось выше, затронем только требования к подготовке специалистов в области Web-программирования, а не в области Web-публикаций.

Студент любого вуза, будущий автор Web-страниц, содержащих материалы по его специальности ("предметник"), может быть знаком с Web-программированием в объеме языка HTML и навыками работы с современными программными средствами для автоматизации подготовки HTML-страниц. В некоторых прикладных областях будет полезным изучение VRML, но степень освоения технологий VRML зависит от потребностей основной специальности. Например, VRML будет весьма полезен будущему архитектору жилых районов или проектировщику садово-парковых ансамблей. При таком сугубо потребительском изучении азов Web-программирования необходимость в освоении специализированных средств для разработки динамических Web-страниц, включающих средства взаимодействия с пользователем (язык JavaScript, механизм входных форм HTML, CGI-скрипты, Java-апплеты и т.д.) возникает, обычно не на этапе обучения, а в процессе будущей работы специалиста (не профессионального программиста). Поэтому включение в учебную программу указанных вопросов зависит от возможностей, интересов и особенностей конкретной кафедры или факультета.

В ряде случаев (и достаточно часто) наряду с началами Web-программирования "предметнику" нужно осваивать не более сложные средства Web-программирования, а уделить внимание задачам комплексного системного проектирования своих Web-страниц (дизайн, графика, стилистика, связность документов, редактирование и т.д. - см. выше, начало этой статьи).

Что касается студентов, обучающихся по специальностям, относящимся к информационным системам, сетевым технологиям и программированию, то Web-программирование для них должно учитывать существование двух групп программных средств Web-системы - прикладные программы и общесистемная часть. (Не следует путать с компонентами "клиент" и "сервер".) Как мы уже говорили выше, количество специалистов по прикладному программированию



ванию для Web-системы существенно превышает число программистов, занятых разработкой общесистемных средств WWW.

Подготовка специалистов по прикладному Web-программированию требует изучения следующих вопросов:

1. Базовые понятия об Internet, включая принципы адресации и систему протоколов;
2. Особенности реализаций технологии "клиент-сервер" для разных ресурсов сети Internet;
3. Гипертекстовая технология и ее расширения за счет мультимедийных средств;
4. SGML - обобщенный стандарт языков разметки;
5. Язык Web-публикаций HTML - как приложение SGML;
6. Язык трехмерного моделирования VRML - как приложение SGML;
7. URI, HTTP, HTML, CGI, VRML - как базовые компоненты Web-технологии;
8. Архитектура WWW и динамика ее функционирования (включая принципиальные особенности Web-серверов, Web-браузеров и обменов между ними);
9. Правила и рекомендации по разработке HTML-документов;
10. Язык JavaScript как средство расширения возможностей HTML-страниц;
11. Соглашения CGI, разработка CGI-скриптов, соответствующие языки программирования (например, Perl, Си++, С, Tcl...);
12. Язык Java, Java-апплеты, виртуальные Java-машины, способы, методы и средства для создания динамических документов по запросам пользователей;
13. Механизмы и средства связи Web-сервера с базами данных.

Подготовка специалистов по системному Web-программированию дополнительно к перечисленным вопросам должна включать следующие темы:

1. Проектирование и программирование Web-серверов и Web-браузеров;
2. Разработка программ автоматического поиска информации на узлах Web-системы (Web-роботы, Web-агенты и т.д.);
3. Администрирование Web-узлов;
4. Разработка и применение утилит для защиты Web-узлов от несанкционированных посещений, для сбора статистики, для организации авторизованного доступа, для шифрования информации и т.д.

Для усвоения материала курсов по Web-программированию у студентов должна быть соответствующая предварительная подготовка.

"Предметники" - будущие авторы Web-публикаций - должны обладать компьютерной грамотностью и общей культурой в области информационных технологий. Требуемый набор знаний и навыков для них (подготовка данных в текстовых и графических редакторах, навыки работы в конкретной ОС и т.д.) обеспечивает современный курс информатики.

Базовая предварительная подготовка для курса по прикладному Web-программированию должна предусматривать изучение дисциплин:

- Программирование и алгоритмические языки (язык Си в обязательном порядке);
- Базы данных, включая язык SQL;
- Основы операционных систем (MS DOS, UNIX, Windows 95/NT, Apple Macintosh System и др.);
- Объектно-ориентированное программирование (желательны языки Си++ и Java).

Полезно знакомство с курсами по машинной графике и основам сетевых технологий.

Для будущих специалистов по системному Web-программированию дополнительно к названным выше дисциплинам будут полезны курсы:

Локальные и глобальные сети ЭВМ;

Администрирование операционных систем и компьютерных сетей;

Системное программирование с ориентацией на мультизадачные операционные системы.

Преподавание и изучение предлагаемой дисциплины "Программирование для World Wide Web" невозможно без организации полноценного практикума, включающего курсовую работу и лабораторные занятия по большинству из названных выше разделов. Здесь вуз, факультет или кафедра могут находиться в разных условиях и поэтому добиваться единообразия в организации практикума нецелесообразно и невозможно. Web-сервер, Web-браузер, инструментальные средства для разработки Web-документов и вспомогательных программ Web-системы могут находиться на разных аппаратно-программных платформах. Наибольшее распространение в России имеют вычислительные комплексы, функционирующие под операционными системами Windows 95/NT, Unix (разные модификации), Apple Macintosh System и уж совсем редко IBM OS/2. Здесь можно было бы выполнить сравнительный анализ возможностей разных платформ (Unix, Apple Macintosh, IBM OS/2, Windows) по отношению к средствам WWW, но это тема для другой работы.

Важно отметить, что для подготовки специалистов по Web-программированию подключение к сети Internet только желательно. Обучать нужно Web-программированию, а не навигации и путешествиям в глобальной сети Internet.

Следовательно, практикум можно поставить и на отдельно стоящих ПЭВМ, и в локальной вычислительной сети или на мощной вычислительной машине с операционной системой, поддерживающей многозадачный режим работы (обычно это разновидности системы UNIX). Для организации нового учебного класса в современных условиях наиболее доступны IBM-совместимые ПЭВМ с операционной системой Windows 95/NT либо ПЭВМ Apple Macintosh с операционной системой System 7.5. Первый вариант, т.е. класс IBM-совместимых машин, значительно дешевле. Именно поэтому остановимся на программных средствах для практикума под Windows 95/NT.

Для студентов, не специализирующихся в области программирования, практикум можно проводить на ПЭВМ 486 и старше с 8 Мбайт (или более)

оперативной памяти, работающих под операционной системой Microsoft Windows 95.

Из программных средств нужны:

- Браузер Netscape Navigator (Communicator) или Microsoft Internet Explorer (желательно наиболее свежая версия);
- Текстовый редактор - любой под Windows 95;
- Графический редактор - любой под Windows 95;
- Любой пакет, позволяющий автоматизировать процесс подготовки HTML-страниц и "соглашающийся" работать при 8 Мбайтах оперативной памяти.

Для практикума по прикладному Web-программированию потребуется увеличение оперативной памяти ПЭВМ по крайней мере до 16 Мбайт. Из программных средств дополнительно нужны:

- JDK - Java Developer's Kit - инструментальные средства фирмы Sun Microsystems для разработки Java-программ;
- Компилятор с языков Си и/или Си++ для подготовки программ, выполняемых в качестве CGI-скриптов;
- Web-сервер, функционирующий под Windows 95;
- Виртуальная Java-машина (входит в состав Netscape Navigator, в Microsoft Internet Explorer);
- Программный инструментальный для организации интерфейса Web-сервера с базами данных, размещенных на машине с Web-сервером.

Здесь следует сделать некоторые пояснения. В настоящее время широко используются интегрированные инструментальные среды разработки приложений на языке Java (Latte фирмы Borland, Semantec Cafe, Visual J++ и т.д.). Однако пионером в области разработки инструментальных средств языка Java является фирма Sun Microsystems с ее продуктом JDK. Кроме того в связи с "молодостью" и постоянным развитием Java-технологии постоянно появляются новые структурные, лингвистические, технологические и программные решения. Фирма Sun Microsystems очень оперативно вносит соответствующие изменения в JDK и бесплатно предлагает новые версии JDK на своем Web-узле (<http://www.sun.com>). Интегрированные среды "вторичны". Они строятся на основе JDK и обычно включают не последнюю версию JDK. Именно поэтому автор считает целесообразным использовать JDK, а не интегрированные среды других фирм.

Для расширения практикума за счет включения вопросов системного Web-программирования дополнительно потребуются средства программирования той операционной системы, под управлением которой функционирует Web-сервер. В случае применения Windows 95 или Windows NT можно использовать Microsoft Visual C++ и интерфейс прикладного программирования Windows Sockets (Winsock API). Напомним, что Winsock API представляет со-

бой переработанную для Windows 95/NT библиотеку функций системы Unix, реализующих интерфейс Беркли для сетей TCP/IP.

Заканчивая краткое описание возможного варианта аппаратно-программных средств для организации практикума по Web-программированию, следует еще раз обратить внимание на высокий динамизм развития Web-технологии и поддерживающих ее лингвистических и программных средств. То, что можно рекомендовать сегодня (май 1998 года) может безнадежно устареть или по крайней мере претерпеть существенные метаморфозы даже в течение одного года. В качестве примера отмечу, что в марте 1997 года была разработана и предложена для использования версия 3.2 языка HTML. В декабре 1997 года W3 консорциум предложил новую версию HTML 4.0, которая существенно отличается от версии 3.2. (см <http://www.w3.org/TR/REC-html40>). Второй пример - эволюция JDK. Все многочисленные бумажные публикации на русском языке по языку Java ориентированы на JDK 1.0.... Но в ноябре 1997 года фирма Sun "положила" на свой сервер версию JDK 1.1.5, где многие методы классов из 1.0 считаются не одобренными (deprecated). К моменту редактирования этой работы уже появилась бета версия следующего варианта JDK 1.2 со своими особенностями и возможностями. К таким изменениям нужно постоянно быть готовым. Лучший способ готовности - изучить базовые принципы, понятия и методы, а затем постоянно следить за происходящими изменениями, что теперь можно делать оперативно с помощью Internet.

## 5. Заключение

Данная работа анализирует тенденции развития Web-системы в сети Internet и отражает опыт автора как в разработке информационно-справочных систем на основе Web-технологии, так и в постановке и преподавании курса "Программирование для World Wide Web". Не претендуя на истину в последней инстанции, автор считает, что существующее состояние Web-системы и перспективы развития и применения Web-технологии делают целесообразным изучение дисциплины "Программирование для WWW" в разных учебных заведениях, начиная от лицеев и кончая университетами. В первом случае это может быть знакомством с разметкой HTML-страниц в курсе информатики, а возможности для студентов университетов подробно изложены выше. В настоящей статье нет детально разработанной учебной программы, однако названы практически все основные составляющие, которые должны быть включены в содержание учебного курса. Полнота предлагаемого набора тем и обоснование каждой из них возможно потребуют общественного обсуждения и конструктивной критики. Только после этого может быть разработан достаточно долговечный вариант учебной программы дисциплины.

Не во всех случаях может оказаться удачным деление обучаемых (студентов) на три категории. Например, подготовка специалистов по Web-программированию может вестись без деления на системное и прикладное программирование. С другой стороны авторам Web-публикаций может быть весьма полезен материал, отнесенный в данной статье к прикладному программированию.

Несмотря на то, что автор постарался учесть основные тенденции развития Web-технологии, но скорость ее развития и конкурентная борьба фирм-производителей программных продуктов требуют постоянной методической работы, что возможно только на основе чтения Internet-публикаций. (Можно начать чтение с Web-сервера <http://www.w3.org>, который ведет W3 консорциум). Именно поэтому в данной статье нет рекомендаций по литературным источникам, хотя в настоящее время статей, монографий и пособий по Web-системе опубликовано очень много, но некоторая часть из них уже частично устарела.