

Jet Info

ИНФОРМАЦИОННЫЙ БЮЛЛЕТЕНЬ

№ 7 (86)/2000

XML на рубеже веков (стр. 2)

XML и связанные с ним стандарты

Краткий справочник для руководителя (стр. 8)



XML

КОРПОРАТИВНЫЕ
СИСТЕМЫ

XML на рубеже веков

Борис Тоботрас

Данный материал знакомит читателя с основными понятиями и терминологией, используемыми в области работы со структурированной информацией вообще, и XML, в частности. Введение в эту область дает выпуск JetInfo N 1(80) за 2000 год, а перечень часто встречающихся понятий приведен во второй статье текущего номера, "XML и связанные с ним стандарты. Краткий справочник для руководителя".

СОДЕРЖАНИЕ

1. Введение
2. Об участниках и докладах
3. Об электронных документах
4. Об интеграционном подходе с использованием XML
 - 4.1 Порталы и интеграция старых приложений
 - 4.2 NewsML и PRISM
 - 4.3 ebXML
5. О применении XML на WWW
 - 5.1 Связывание информации
 - 5.2 Пространства имен
 - 5.3 Метаданные и поиск
6. XML и беспроводные коммуникации
7. Графика и XML
8. О выставке

1. Введение

Поводом для написания этой статьи явились конференция и выставка «XML Europe 2000», проведенные в конгресс-центре Парижа Graphic Communications Association с 12 по 16 июня и собравшие более полутора тысяч заинтересованных представителей практически всех отраслей промышленности и образования со всего мира — как менеджеров и директоров компаний и их подразделений, так и инженеров и консультантов.

Кроме пленарных докладов, на конференции проходило несколько тематических секций, одновременно по пять-шесть в день. Среди них:

- XML: что это такое и зачем используется?
- стандарты
- электронная коммерция
- издательское дело
- XML-схема
- языки запросов
- использование технологии
- тематические карты
- WAP и WML
- электронная печать
- построение решений
- реализации продуктов
- графика
- трансформации
- XML в финансовой индустрии
- XML в здравоохранении и несколько других.

В процессе проведения конференции стало очевидно, что XML как технология достигла своей «зрелости». Если на конференции 1998 года данная технология только освещалась и намечались дальнейшие перспективы ее развития, в 1999 говорилось о первых промышленных внедрениях и реализации пилотных проектов, то в этом году — опыт промышленной эксплуатации и проекты уже второго поколения систем, в той или иной степени использующих технологию XML.

За эти годы несколько основополагающих стандартов в области XML (собственно XML, XSLT, SVG, XSchema, XLink и XPointer, RDF и другие) достигли относительно стабильного состояния, что безусловно способствовало быстрому развитию программных средств для работы с XML-данными. Выставочная часть конференции — яркое тому подтверждение: на подъеме технологии программная индустрия откликается огромным числом продуктов и услуг, направленных как на разработчиков и интеграторов решений, так и на конечных пользователей.

2. Об участниках и докладах

Интересен состав участников конференции, как докладчиков, так и слушателей, большинство из которых являются представителями индустрии. Доклады включали: обзоры стандартов и предложений, введения в конкретные технологии (WebCGM, SVG, XLink и другие), технологии программирования в приложении к XML-технологиям. Но большая часть докладчиков рассматривали конкретные реализации программ и систем, использующих тот или иной аспект XML-технологий. Многие из этих систем находятся в реальной эксплуатации и доступны пользователям через Internet. Миллионы людей уже пользуются XML-продуктами, зачастую даже не догадываясь об этом. Популярная программа RealAudio player, например, получает и интерпретирует мультимедийные данные, кодированные на XML-языке SMIL.

В бесконечном спектре применений XML-совместимой технологии можно попытаться выделить три области, в которых XML особенно популярен. Это электронная документация, интеграция приложений и WWW. Рассмотрим их более подробно.

3. Об электронных документах

Применение XML при подготовке электронной документации имеет давние корни, поскольку в этой области его предшественник SGML широко используется с конца 80-х годов.

Быстрое развитие XML-инструментария и появление пользовательских продуктов ускорило внедрение логической разметки при подготовке документации. В этой области XML позволяет разделить задачу на собственно подготовку текстов, иллюстраций и другого мультимедийного содержания электронных документов и их публикацию, то есть, предоставление потребителю в каком-либо формате на каком-либо носителе.

Такое разделение имеет далеко идущие последствия, в частности, оно позволяет:

- Выделять структуру документов. Автор может оперировать логическими частями документа в удобном для него виде, и при помощи программного обеспечения — проверять соответствие структуры документа требуемой.
- Готовить документы, не зависящие от платформы и используемого программного обеспечения. Часто этот аспект недооценивается, однако он весьма важен, если предполагается длительное хранение документов: только представьте себе, как легко будет прочесть архив файлов в формате Microsoft Word'97 лет через пятнадцать? Или как уже сегодня прочесть файлы редактора Sprint или ChiWriter, бывших популярными еще лет семь назад?
- Готовить один экземпляр документа для публикации на разнообразных носителях и с различным способом представления. Примерами могут служить:
 - файлы формата PDF для печати на бумаге;
 - файлы PDF для просмотра на экране (он обычно отличается от предыдущего варианта выбором шрифта, размером страницы, гиперссылками, интерактивностью);
 - файлы WML для доставки информации на мобильные устройства;
 - файлы HTML для размещения на WWW, возможно с дополнительной функциональностью, интерактивными возможностями, с различными вариантами оформления для разных WWW-навигаторов и пожеланий пользователей;
 - печатное издание в виде книги, буклета, листовки;
 - голосовой вывод для незрячих пользователей или для использования в автомобилях.
- Кодировать максимально доступное в момент подготовки документов количество информации. Различная метаинформация — библиография, статус различных частей документа, права доступа к ним, правила активизации мультимедийного содержания и многое другое — хранится в том же самом файле, что и основное содержимое, и доступна для обработки теми же программными средствами.

- Кодировать сколь угодно сложные взаимосвязи и гиперссылки между частями документа и между различными документами, не испытывая при этом ограничений вроде только однонаправленных связей между двумя точками, как это принято сейчас на WWW.
- Программно обрабатывать тексты документов, поскольку их содержимое размечено формальным образом.

Подготовка электронных документов уже давно является необходимым звеном технологической цепочки во многих отраслях промышленности. Пионерами в использовании SGML для этих целей были машиностроительные и авиакосмические фирмы. Объем документации, сопровождающей их продукцию, чрезвычайно велик. Кроме того, сама документация меняется довольно часто. В этих условиях выпуск и поддержка жизненного цикла обычной бумажной документации становится очень дорогостоящим и медленным процессом. На конференции выступали с докладами представители таких известных компаний, как Boeing, McDonnell-Douglas, Siemens, Ericsson, и множества других. Интересные цифры были названы в выступлении инженеров из фирмы Cessna, выпускающей большую часть современных коммерческих реактивных самолетов. Объем документации на их продукцию составляет примерно 230 тысяч страниц в год. Готовится она в электронном виде отделом из шестидесяти человек. Выходным форматом являются компакт-диски с технической (конструкторской, ремонтной и эксплуатационной) и полетной документацией.

4. Об интеграционном подходе с использованием XML

В программной индустрии существует давняя и с переменным успехом решаемая проблема взаимодействия различных программ в единой системе. Сложность заключается в том, что современные коммерческие программы разрабатываются различными компаниями и объединить их развитыми интерфейсами на сколь бы то ни было длительное время оказывается невозможным. Необходимость интеграции слабо связанных продуктов привела к развитию систем класса middleware и стандартов наподобие CORBA. С появлением XML родились средства интеграции нового поколения.

Обмен данными и событиями происходит в виде сообщений. Каждое сообщение является экземпляром XML документа. Интерфейсы обычно формализуются в виде так называемых схем, являющихся сочетанием описаний типов документов (DTD) с описанием типов данных и

накладываемых на элементы данных ограничений. В сочетании со стандартизированным протоколом передачи таких сообщений, в качестве которого часто используется популярный протокол HTTP, такие описания позволяют обмениваться данными между программами, причем, в отличие от существующих методов, требующих достаточно сложного низкоуровневого программирования, использование XML не накладывает вообще никаких ограничений ни на функционирование программ, ни на выбор языка или технологий программирования. Особенно это удобно, когда взаимодействовать приходится программным комплексам, принадлежащим различным организациям.

Одной из важных задач на этом пути становится выработка вышеупомянутых описаний, диктующих правила обмена данными между приложениями. Часто для этого формируется консорциум, формулирующий стандарты обмена по определенной тематике, обычно в рамках одной вертикальной индустрии. Уже выпущены или находятся в работе спецификации, охватывающие следующие области: бухгалтерию, рекламу, строительство и архитектуру, астрономию и исследования космоса, автомобильную и авиакосмическую промышленность, банковское дело, библиографию и каталогизацию, связь, компьютерную графику, content syndication, отношения с заказчиками, сети и распределенное управление, экономику, образование, электронную коммерцию, обмен электронными данными (EDI), энергетику, информационные порталы, управление предприятием, финансовые рынки, пищевую промышленность, географию, здравоохранение, управление кадрами (HR), автоматизацию производства, страхование, юриспруденцию, музыку, новости, издательское дело, недвижимость, научные дисциплины, программное обеспечение, управление производственными поставками, языковые переводы, отдых и путешествия, синтез голоса, прогнозы погоды, WWW-приложения, и многие, многие другие. Этот впечатляющий список охватывает тематику, по которой работают известные отраслевые комитеты. Задачей каждого из них является выработка словаря для обмена определенной информацией между ПО, обрабатывающим данные в рамках какой-либо отрасли.

Наличие такого словаря, поддерживаемого разработчиками ПО, позволяет очень динамично строить и развивать взаимодействие как между компонентами информационной системы организации, так и между различными сотрудничающими организациями. Кроме того, в отличие от классических систем на базе RPC, CORBA и т.п., данные оказываются не зависящими ни от приложений, их производящих и потребляющих, ни от транспортных протоколов, которыми

они передаются. Более того, они хранятся или транспортируются в текстовом формате, что дает возможность читать их человеку и дополнительно упрощает разработку и внесение модификаций во взаимодействующее ПО.

4.1. Порталы и интеграция старых приложений

Такой подход находит свое применение в построении различного рода открытых и, в особенности, корпоративных Интернет-порталов. Выработка корпоративного словаря для обмена данными (или принятие в качестве такового существующих или создаваемых индустриальных словарей) становится основой для построения интегрирующей платформы. Программное обеспечение такой платформы получает в свое распоряжение все возможности, связанные с XML-технологией: описание данных и метаданных в синтаксисе XML, связывание информации при помощи XPointer, XLink и XPath, описание метайнформации средствами RDF.

Принципиальное преимущество использования XML в качестве интегрирующей технологии становится заметным при рассмотрении затрат на подключение к portalу каждого нового приложения. Обычно для организации взаимодействия этого приложения с остальными участниками информационного обмена необходимо разрабатывать множество интерфейсов: «каждый с каждым». В случае использования единого словаря обмена данными необходимо лишь преобразовывать данные в формат такого словаря и обратно. Не преуменьшая сложности этой задачи, можно заметить, что сложность подключения приложений становится зависящей лишь от свойств приложений, а не от количества и свойств других участников обмена.

Далее рассмотрим некоторые инициативы и выработанные в их рамках стандарты из приведенного списка.

4.2. NewsML и PRISM

Инициатива по разработке языка передачи новостей NewsML была вызвана к жизни проблемой, стоящей перед поставщиками и потребителями новостей. Перед поставщиками, например, информационным агентством, стоит задача постоянного вещания на множество потребителей, использующих разные носители: одним новости необходимы в текстовом формате, другим — в звуковом или видео, третьим — в каких-то их сочетаниях. Кроме собственно информации, каждая единица новостей должна сопровождаться метайнформацией: заголовок, источ-

ник, время, место, разнообразные ограничения на распространение, тема, степень важности и многое другое. Перед получателями, например, средством массовой информации, стоит смежная задача: объединение потоков новостей от различных источников, автоматизированная их сортировка, хранение, поиск и обработка, выборка нужной информации в нужном формате.

Такие возможности и предоставляет разработанный язык NewsML. Являясь приложением XML, он позволяет максимально полно и точно описать передаваемые новостные данные. Разработанный и поддерживаемый ведущими информационными агентствами мира и средствами массовой информации, он уже активно используется для вещания. Например, агентство Reuters с декабря 1999 года передает свои новости в виде NewsML файлов.

4.3. ebXML

Первые попытки использовать XML для электронной коммерции натолкнулись на необходимость выработки механизмов и прикладных протоколов обмена транзакциями между компаниями (фактически — их программным обеспечением). Комитетом по электронной торговле ООН был инициирован проект ebXML, целью которого является разработка открытой инфраструктуры, базирующейся на XML и обеспечивающей глобальное использование информации, циркулирующей в рамках приложений электронной коммерции, совместимым, надежным и непротиворечивым образом. В рамках проекта выпущена черновая версия спецификаций, общедоступная для использования в создаваемых программных продуктах. Проект поддержан несколькими сотнями компаний, заинтересованных в развитии электронной коммерции.

Существует в какой-то степени конкурирующий с проектом ebXML консорциум BizTalk, организованный компанией Microsoft и поддерживаемый многими промышленными фирмами. В его рамках Microsoft выпустила BizTalk Framework, обозначающий предварительную инфраструктуру реализации электронной коммерции.

5. О применении XML на WWW

Как правило, когда речь заходит о XML в контексте WWW, имеется в виду передача страниц на том или ином языке с синтаксисом XML, вместо используемого ныне HTML. Это довольно очевидное применение XML, для которого он, в известной степени, и был задуман. С появлени-

ем Microsoft IE 5 и бета-версий Netscape 6 стало возможно доставлять XML-страницы на десктоп пользователю. В сочетании с механизмами стилей (стандартов CSS1 и XSL) это позволяет, с одной стороны, выражать содержимое страниц в «естественных» терминах вместо жестких ограничений HTML, а с другой — легко варьировать стилистику отображения материала без изменений собственно WWW-страниц. Но возможности XML далеко не ограничиваются расширениями HTML. XML-инфраструктура включает в себя и другие возможности, которых так сильно не хватает сегодняшним системам, построенным вокруг WWW.

5.1. Связывание информации

В инфраструктуру XML входят две спецификации, относящиеся к формализации связей между информационными фрагментами: XPointer и XLink. XPointer задает синтаксис задания сложных адресов внутри XML-документов, тогда как XLink описывает семантику отношений между участвующими в связи документами. В отличие от общеизвестных гиперссылок в HTML, связи, задаваемые при помощи XLink/XPointer, отличаются, в частности:

- возможностью задавать двунаправленные связи;
- возможностью помещать спецификации связей вне документов, которые они связывают. Это особенно важно, когда необходимо формировать ссылки на документы, запись в которые запрещена (чужие WWW-страницы) или невозможна (данные на CD-ROM);
- возможностью специфицировать точки отправления и назначения связи в произвольных терминах структуры документов;
- возможностью задавать в качестве концов связи не конкретных точек в документах, а диапазонов структурных элементов (например, «см. разделы 1-3»).

5.2. Пространства имен

Рекомендация по пространствам имен позволяет дизайнеру использовать на одной странице элементы XML, носящие одинаковые имена, но относящиеся к различным прикладным словарям. Классическим примером является элемент «<set>», который в MathML означает математическое множество, тогда как в SVG — операцию присваивания. В том случае, когда в одном документе встречаются и математика (MathML), и графика (SVG), возникает конфликт имен, который и разрешается при помощи разнесения имен по разным адресным пространствам. Каждое пространство специфицируется при помо-

щи глобально уникального адреса (URL).

Таким образом, в рамках одного XML-документа возможно произвольно смешивать словари, необходимые для формирования конкретного предметного языка.

5.3. Метаданные и поиск

Неоценимым следствием от внедрения XML в кодирование прикладных данных или использование его для формирования текстовых документов является появление технологии для фиксации любого необходимого количества метainформации. Это могут быть не только известные по HTML ключевые слова, но и разнообразные «данные о данных»: произвольные, зависящие только от прикладной области, свойства как документов, так и их компонентов, вплоть до фрагментов текста. Формальное кодирование такой информации открывает возможность более «интеллектуальной» обработки массивов данных.

6. XML и беспроводные коммуникации

Достаточно было обвести взглядом слушающих доклады конференции, чтобы убедиться, что практически каждый второй делал записи не на бумаге, а на различных блокнотных компьютерах. На фоне многочисленных карманных Palm, Psion, Cassiopea обычные ноутбуки на столах выглядели неприлично громоздкими, тяжелыми и неудобными. Широкое распространение карманных компьютеров предвещает, по мнению многих, настоящую революцию в информационных технологиях. Заметим, однако, что WWW-навигаторы на этих устройствах довольно ограничены карманными ресурсами и относительно мало совместимы с «классическими» NN и IE.

Одно из заметных отличий Европы от США — почти повсеместное доминирование стандарта мобильной связи GSM и все более широкое распространение технологии WAP. С его развитием создается громадная сеть подключенных к Интернету и WWW устройств, также мало совместимых со старыми навигаторами. Эта тенденция заметна уже и в нашей стране.

По прогнозам Gartner Group, к 2005 году более четырех пятых подключенных к Интернету клиентских систем будут не персональными компьютерами, а совершенно другими устройствами — телефонами, карманными и автомобильными компьютерами, и даже холодильниками и микроволновыми печками.

Особенностью использования языков логи-

ческой разметки является то, что, создавая электронные документы в XML, их издатель оказывается в большой степени защищен от проблем, связанных с появлением новых форматов и сред публикации информации. Это особенно актуально ввиду быстрого появления множества разнообразных устройств, подключенных к WWW. В этой связи интересно проследить, как изменяется со временем пользовательский интерфейс WWW.

К тому моменту, когда WWW получил широкую популярность, большинство пользователей работало с Netscape Navigator. Позже его догнал Internet Explorer. В стремлении максимально полно использовать возможности клиентов, разработчикам WWW-серверов пришлось зачастую делать по два варианта страниц для каждого из популярных навигаторов. Сейчас IE занимает около 80% рынка, и есть тенденция появления серверов, полноценно работающих только с IE. Ситуация, однако, развивается довольно быстро, и скоро разработчикам придется адаптироваться к появлению множества других средств отображения их серверов: уже довольно часто можно встретить варианты страниц, сделанных для системы AvantGo, обеспечивающей удобный доступ к информации в режиме off-line, или для WAP-навигаторов.

Использование XML позволяет поставщику информации не бояться того, что с развитием технологии его документы устареют, или их потребуется постоянно преобразовывать в очередной «формат дня».

7. Графика и XML

Современное состояние передачи графики средствами WWW оставляет удручающее впечатление для всякого пользователя, сколько-нибудь знакомого с состоянием компьютерной графики вообще. Многочисленные ограничения включают:

- отсутствие общеупотребительного векторного формата и, как следствие, неудовлетворительное качество изображения;
- ограниченные возможности масштабирования;
- отсутствие управления цветом, шрифтами, преобразованием координат;
- невозможность работы с текстом в графике (поиск и индексирование, копирование и замена);
- отсутствие возможности связывания графики с другими данными (например, гиперссылок внутри графического изображения).

Одно из интересных применений XML в области передачи графики — спецификация SVG (Scalable Vector Graphics), предназначенная для разрешения множества существующих проблем

с графической информацией. Ее характерные свойства включают:

- синтаксис XML;
- возможность индексирования, поиска и выделения текста);
- возможность применения стилевого управления презентацией графики (CSS);
- включение векторной и растровой графики с применением фильтров (эффектов);
- возможность свободного смешивания описания графики с текстовой и иной информацией в рамках одного XML-файла.

SVG имеет статус свободно распространяемой рекомендации W3C. В настоящее время уже доступны программные средства для создания, редактирования и просмотра данных в формате SVG, а также набор тестов для верификации соответствия данных и продуктов выпущенной рекомендации.

8. О выставке

В выставке принимали участие около 60 компаний, которые представили свои решения, продукты и услуги. Некоторые из них давно известны как поставщики решений в области SGML, задолго до появления на свет XML: SoftQuad, Adobe, Arbortext, Advent 3B2, Omnimark Technologies и другие.

Как обычно, демонстрировались новые версии флагманских продуктов компаний: Компания SoftQuad показывала XMetaL 2.0, Adobe демонстрировала новый Framemaker + SGML, на стенде Omnimark Technologies раздавались CD-ROM с Omnimark5 для Windows и Linux, Micrografx показывала свои новые средства создания технических иллюстраций.

Интересно появление большого количества продуктов, ориентированных не только на разработчиков ПО или авторов XML-документов, но и на создание развитых систем с использованием XML: порталов, систем электронной коммерции, систем цифровой публикации. Фирма Enigma демонстрировала систему автоматизированного построения цепочек «поставщик-заказчик». На стенде компании Kinecta работала система Interact — платформа для построения систем агрегирования информационных потоков. Корпорация eXcelon показывала business-to-business сервер, технологическое ядро системы электронной торговли. Выставка наглядно иллюстрировала прогнозы, согласно которым рынок корпоративного и межкорпоративного ПО будет испытывать быстрый и непрерывный рост в ближайшие пять лет. AMR Research, например, прогнозирует объем роста порядка 60% ежегодно.

XML и связанные с ним стандарты

Краткий справочник для руководителя

Кен Холман

СОДЕРЖАНИЕ

Синтаксически корректный XML (Well-formed XML)

Структурно корректный XML (Valid XML)

XML схема

Схематрон (Schematron)

Пространства имен

Расширяемый гипертекстовый язык разметки XHTML

Иерархические стили (CSS)

Объектная модель документа (DOM)

Упрощенный программный интерфейс для XML (Simple API for XML, SAX)

Упрощенный протокол доступа к объектам XML (SOAP)

Язык описания путей XML (XPath)

XSL и XSLT

Связывание документов со стилями

Языки XML для ссылок и связей (XPointer и XLink)

Тематические карты XML (Topic Maps)

Шаблон описания ресурсов (Resource Definition Framework, RDF)

Запросы XML (XML Query)

Цифровые подписи XML

Масштабируемая векторная графика (Scalable Vector Graphic, SVG)

Язык математической разметки (Mathematics Markup Language, MathML)

Протокол беспроводного доступа к приложениям (WAP)

Предисловие переводчика

В последнее время технологии, основанные на XML, переживает этап взрывного развития. Появляется множество продуктов, ориентированных на XML или поддерживающих его тем или иным образом; вокруг XML формируются разнообразные стандарты. Разобраться в этом калейдоскопе стандартов, рекомендаций, аббревиатур подчас нелегко даже инженерам, не говоря уже о руководителях, которым, тем не менее необходимо ориентироваться в тенденциях технологического развития. Предлагаемый их вниманию справочник поможет расставить по местам кусочки мозаики, которую представляет из себя мир XML.

Справочник состоит из «словарных статей». После заголовка каждой статьи и, возможно, расшифровки акронима следуют ссылки на официальные материалы. Далее перечислены основные особенности рассматриваемого понятия, снабженные комментариями и иногда примерами. Краткость описаний иногда компенсируется их непонятностью.

G. Ken Holman является директором по технологиям фирмы Crane Softwrights Ltd., канадской корпорации, предлагающей клиентам по всему миру услуги обучения программирования OmniMark, языкам DSSSL и XSL/XSLT, а также услуги анализа информационных систем, базирующихся на SGML и XML. Г-н Holman в настоящее время является представителем Канады в подкомитете ISO, отвечающем за семейство SGML стандартов, приглашенным экспертом W3C, автором книги «Практическая трансформация при помощи XSLT и XPath» (ISBN 1-894049-04-7), а также частый докладчик на соответствующих конференциях. До основания Crane, г-н Holman провел 13 лет в качестве разработчика и консультанта. Дополнительную информацию о компании Crane Softwrights вы можете получить по адресу: <http://www.CraneSoftwrights.com/links/info-jet.htm>.

Синтаксически корректный XML (Well-formed XML)

XML - Extensible Markup Language (расширяемый язык разметки)

- <http://www.w3.org/XML>
- <http://www.w3.org/TR/REC-xml>
- <http://www.w3.org/XML/xml-19980210-errata>
- <http://www.xml.com/axml/axml.html> - аннотированная версия
- является синтаксисом для структурированной информации
- задает правила для хорошо организованной иерархической информации
- модель документа неявно выражена в иерархии данных
- идентифицирует информацию с помощью разметки, к которой можно программно обращаться
- дает возможность передавать, принимать и обрабатывать через WWW информацию, размеченную произвольным образом, так, как это сейчас делается с HTML
- разработан для простоты реализации и совместимости с SGML и HTML

Пример синтаксически корректного XML:

```
<?xml version="1.0"?>
<weather>
  <current>
    <temp scale="F">72</temp>
    <pressure>1005</pressure>
    <humidity>43</humidity>
  </current>
  <min>
    <temp scale="F">65</temp>
    <pressure>998</pressure>
    <humidity>38</humidity>
  </min>
  <max>
    <temp scale="F">78</temp>
    <pressure>1010</pressure>
    <humidity>43</humidity>
  </max>
</weather>
```

Структурно корректный XML (Valid XML)

XML - Extensible Markup Language (расширяемый язык разметки)

- <http://www.w3.org/XML>
- <http://www.w3.org/TR/REC-xml>
- <http://www.w3.org/XML/xml-19980210-errata>
- <http://www.xml.com/axml/axml.html> - аннотированная версия
- является синтаксисом для задания модели структурированной информации
- модель документа явно задана с помощью набора определений
- позволяет инструментарию накладывать ограничения на структуру вводимой информации в рамках заданной модели
- позволяет инструментарию проверять соответствие структуры информации заданной модели
- ориентирован на иерархическую структуру (без ограничений на содержание)

```
<?xml version="1.0"?>
<!DOCTYPE weather [
<!ELEMENT weather ( current, ( min, max )? )>
<!ELEMENT current ( temp, pressure, humidity )>
<!ELEMENT min ( temp, pressure, humidity )>
<!ELEMENT max ( temp, pressure, humidity )>
<!ELEMENT temp ( #PCDATA )>
<!ATTLIST temp scale ( C | F ) #REQUIRED>
<!ELEMENT pressure ( #PCDATA )>
<!ELEMENT humidity ( #PCDATA )>
]>
<weather>
  <current>
    <temp scale="F">72</temp>
    <pressure>1005</pressure>
    <humidity>43</humidity>
  </current>
  <min>
    <temp scale="F">65</temp>
    <pressure>998</pressure>
    <humidity>38</humidity>
  </min>
  <max>
    <temp scale="F">78</temp>
    <pressure>1010</pressure>
    <humidity>43</humidity>
  </max>
</weather>
```

XML схема

- <http://www.w3.org/TR/xmlschema-1/> - структуры
- <http://www.w3.org/TR/xmlschema-2/> - типы данных
- словарь для спецификации структуры различных словарей
 - определяет и описывает классы XML-документов
- определяет структурные, синтаксические и количественные ограничения и отношения между:
 - элементами
 - атрибутами
 - их содержимым

- типами данных
- помогает интерпретировать смысл и использовать документы
- предоставляет значения по умолчанию для атрибутов и элементов
- имеет XML-синтаксис
 - документ-схема содержит представление элементов схемы
 - задает иерархические конструкции
 - может включаться в экземпляр документа
 - то есть не обязана быть отдельным документом
 - в отличие от определения типа документа — Document Type Definition — может быть создана стандартными средствами редактирования XML

Не пыгается предоставить все мыслимые возможности

- некоторые приложения все равно должны осуществлять свои собственные семантические проверки

Часть 1: Структуры

- определяет содержание XML-схем и их компонентов
- определяет применение XML-схем к XML-документам
- определяет иерархическое отношение между составляющими структурами элементами
- задает иерархии определений типов

Часть 2: Типы данных

- тип значений атрибутов и текста
 - например, целое, дата, строка, булевская переменная, число с плавающей точкой, число двойной точности, десятичное и т.д.
 - неделимые и агрегатные типы
 - примитивные и генерируемые типы
 - встроенные и определяемые пользователем типы
- язык, предназначенный для использования с XML-схемами и другими рекомендациями XML
 - вероятно, включая XSL и RDF
- контроль типов данных
 - обеспечивает надежность информационного обмена
- возможность расширения прикладными типами данных
 - специализированными
 - производными от стандартизованных типов
- типизированные значения атрибутов и элементов

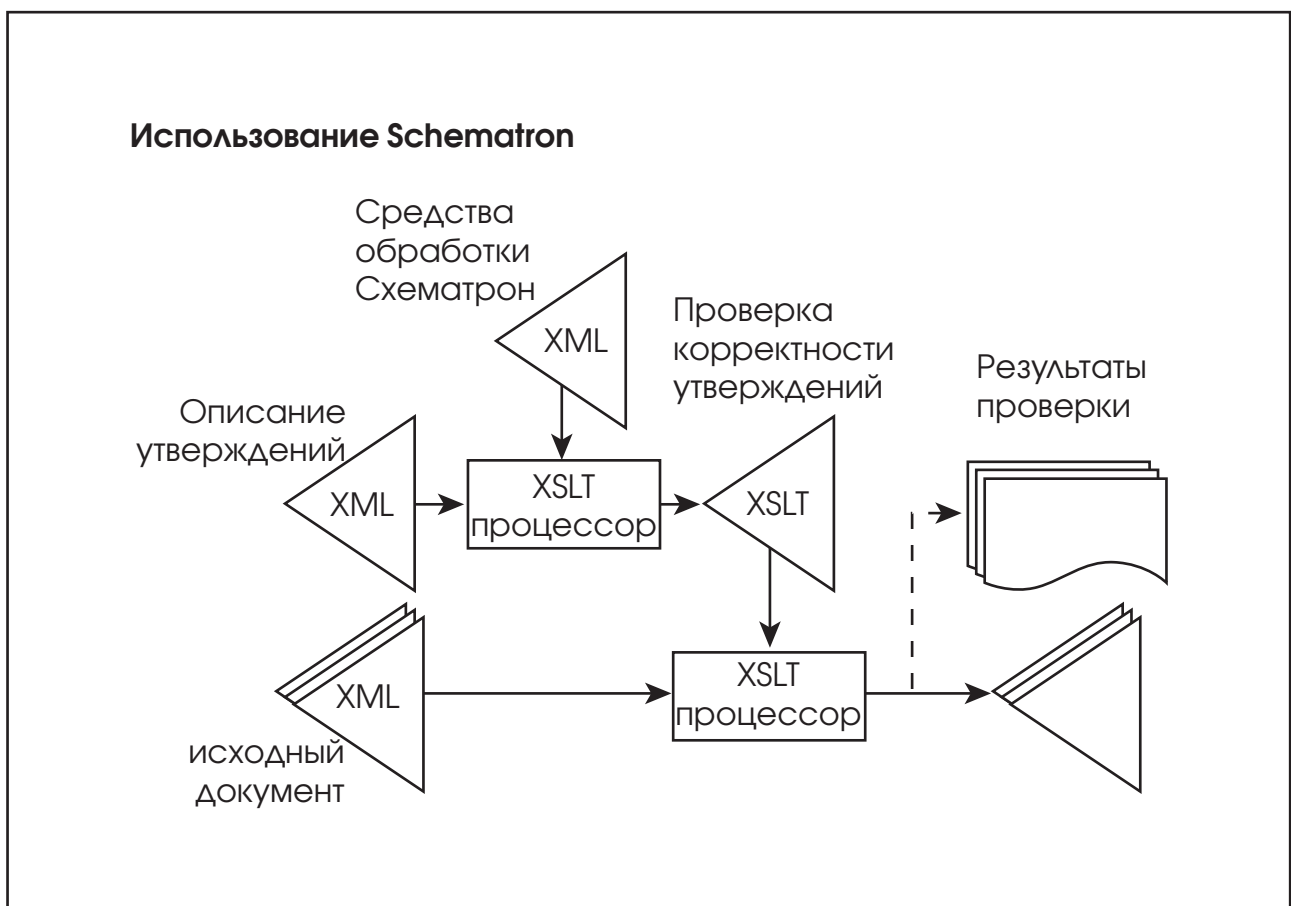
Схематрон (Schematron)

- <http://www.ascc.net/xml/resource/schematron/schematron.html>
 - разработан Rick Jelliffe, Academia Sinica Computing Centre
- язык для проверки правильности утверждений о данных, использующийся для верификации соответствия данных бизнес-правилам
- подход к верификации экземпляров документов, отличающийся от использования схем
 - использование схемы основано на грамматике
 - процесс верификации сверяет экземпляр с грамматикой
 - отношения родитель/потомок описаны в модели содержимого
 - схематрон основан на наборе утверждений, описывающих ожидаемое содержимое экземпляра
 - схематрон проверяет экземпляр на соответствие набору правил
 - правила могут включать произвольные отношения наследования и ветвления
 - возможно кодирование бизнес-правил, относящихся к структурной целостности экземпляра

- при формулировании правил используются выражения языка XPath
- предоставляет возможность связывать правила друг с другом
 - правила могут зависеть от истинности или ложности других правил
 - схемы, основанные на грамматике, не предоставляют таких возможностей

Пример экземпляра структурно-корректного XML, которому Схематрон необходим для проверки бизнес-правил:

```
<?xml version="1.0"?>
<!DOCTYPE thing [
<!ELEMENT thing ( a | b )>
<!ATTLIST thing content ( a | b ) #REQUIRED>
<!ELEMENT a (#PCDATA)>
<!ELEMENT b (#PCDATA)>
]><thing content="a"><b>test</b></thing>
```



Примечания:

- Существуют различные Схематроны, обеспечивающие различные формы отчетов
- Результат использования Схематрона представляет собой стиль для проверки и отчета
- Стиль можно использовать для любого количества проверок без регенерации

Пространства имен

Обеспечивают разделение словарей элементов и атрибутов

- <http://www.w3.org/TR/REC-xml-names>
- <http://www.megginson.com/docs/namespaces/namespace-questions.html>

- описывает простой метод для квалификации имен элементов и атрибутов, используемых в XML-документах
- позволяет в данном документе использовать одно и тоже имя типа элемента из различных словарей
- рассмотрим два словаря, каждый из которых определяет тип элемента, обозначаемый как «<set>», причем с разной семантикой
 - в языке разметки SVG элемент <set> позволяет присвоить значение в заданном контексте
 - в языке разметки MathML <set> ссылается на набор конструкций, представляющих собой множество
- для любого документа, в котором необходимо смешивать элементы двух словарей, может возникнуть необходимость использования элементов с одинаковыми именами
 - без пространства имен приложение окажется не в состоянии определить, какая из конструкций используется
- префикс в имени элемента определяет пространство имен, в котором этот элемент определен
 - <svg:set>
 - <math:set>
- составные имена лексически анализируются как имена XML
 - использование двоеточия определено рекомендацией по пространству имен

Ассоциация значений URI

- префиксы типов имен элементов ассоциируются с идентификаторами ресурсов (URI) вне зависимости от существования какого-либо ресурса по этому URI
- примеры:
 - `xmlns:svg="http://www.w3.org/2000/svg-20000303-stylable"`
 - `xmlns:math="http://www.w3.org/1998/Math/MathML"`
- доменом URI владеет использующая его организация
- соблюдение определенных правил предотвращает конфликты URI при ассоциировании URI
- не предполагается наличия какой бы то ни было информации по данному URI
 - заметим, хотя это и за рамками рекомендации по пространству имен, что в рекомендации RDF, тем не менее, принято обращение за информацией по указанному URI.
- согласно рекомендации URI используется исключительно для устранения неоднозначности между идентичными неквалифицированными элементами различных словарей

Выбор префикса произволен и может быть любым лексически разрешенным именем

- выбранное имя не обязано соответствовать назначению элемента (хотя это и повышает ясность разметки)
- никакая рекомендация не накладывает ограничений на выбор имен
- процессор XML заменяет префикс на ассоциированный URI:
 - `<{http://www.w3.org/2000/svg-20000303-stylable}set>`
 - `<{http://www.w3.org/1998/Math/MathML}set>`
 - в вышеприведенном скобки "{ " и "}" служат только для выделения ассоциированного URI
 - заметим, что, поскольку символы "/" не являются разрешенными в рамках лексических правил имен, то прямое использование URI в метках XML-документа не допускается
- префикс представляет собой просто синтаксическое сокращение, позволяющее не использовать длинные отличительные строчки

Расширяемый гипертекстовый язык разметки XHTML

- <http://www.w3.org/TR/xhtml1>
- расширение HTML с использованием XML 1.0
- воспроизводит, включает в себя и расширяет словарь HTML 4
- соответствует XML и работает в пользовательских агентах, поддерживающих HTML 4
- файлы XHTML могут обрабатываться XML-инструментами

- стилевые процессоры
- приложения

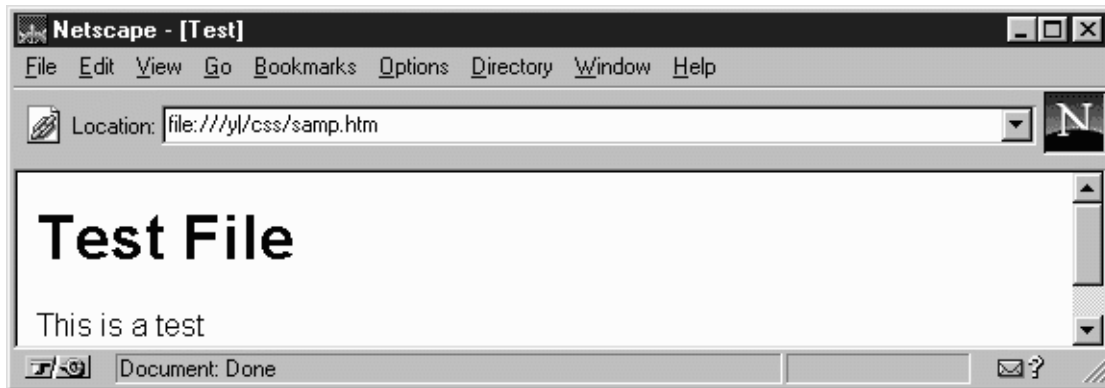
Иерархические стили (CSS)

- <http://www.w3.org/TR/REC-CSS1>
- <http://www.w3.org/TR/REC-CSS2>
- снабжают Web-документы (HTML и XML) информацией о форматировании
- не имеют средств манипуляции документом (структурой и содержимым)
- обеспечивают только «украшение» документа
 - сопоставляют элементам стилистическую информацию
 - предоставляют простые средства для задания текстовых префиксов или суффиксов элементов
 - управляют обработкой пробелов
 - обеспечивают перекрытие и прозрачное наложение прямоугольных областей
- поддерживают различные типы представления информации
 - свойства отображения на символьном дисплее
 - свойства табличного представления
 - свойства звукового представления для работы вслепую
 - незрячие пользователи
 - мобильные пользователи
- не мешают работе навигаторов, не поддерживающих CSS
- позволяют рабочей группе разрабатывать общую модель форматирования Web-документов

Пример HTML:

```
<html>
<head>
  <Title>Test</title>
  <style type="text/css">
    H1    { color: green; text-align: right }
    .info { color: red }
  </style>
</head>
<body>
  <h1>Test File</h1>
  <p class="info">This is a test</p>
</body>
</html>
```





Пример XML:

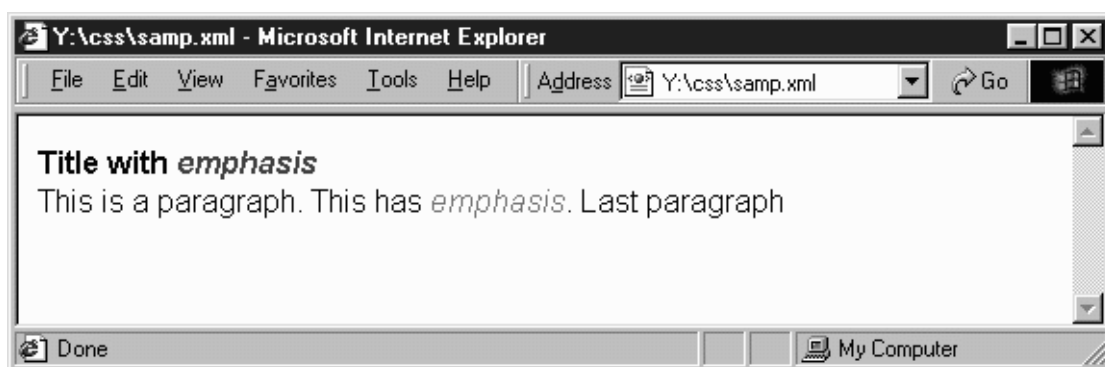
Файл samp.css

```
EMPH      { color: red; display: inline; font-style:italic }
PARA, TITLE { font-family: arial, sans-serif; display: block }
TITLE     { font-weight: bold }
TITLE EMPH { color: blue }
```

для файла samp.xml

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/css" href="samp.css"?>
<INFO>
  <TITLE>Title with <EMPH>emphasis</EMPH></TITLE>
  <P>This is a paragraph.</P>
  <P>This has <EMPH>emphasis</EMPH>.</P>
  <P>Last paragraph</P>
</INFO>
```

Результат:



Объектная модель документа (DOM)

- <http://www.w3.org/TR/REC-DOM-Level-1>
- <http://www.w3.org/TR/DOM-Level-2>
- ориентированный на работу с деревом документа, не зависящий от платформы и языка программирования интерфейс, позволяющий программам динамически разбирать и изменять содержимое, структуру и стиль документов; содержит:

- модель объектов для XML и HTML
- стандартный интерфейс для манипулирования объектами
- строится на трех слоях функциональности
 - ядро
 - базовые интерфейсы
 - интерфейсы для работы с XML
 - расширенные интерфейсы для работы с HTML
- осуществляет узаконенное разграничение полномочий между программистом и производителем
- программист использует стандартный интерфейс и не нуждается в знании конкретной реализации производителя
- производитель поддерживает стандартный интерфейс ко внутренним структурам данных без необходимости открывать закрытую информацию

Основные принципы:

- документ представляет собой иерархию узлов
- узлы можно собирать в список узлов, упорядоченный в естественном порядке документа
- узлы можно собирать в неупорядоченные множества
- конкретная реализация интерфейса может строиться на основе документов дерева, списки и множества узлов
- программист может манипулировать деревьями узлов, списками и множествами узлов

Пример определения интерфейса:

```
interface Attr : Node {
    readonly attribute DOMString name;
    readonly attribute boolean specified;
    attribute DOMString value;
};
```

- name - имя атрибута
- specified - указатель на то, что значение атрибута явно задано (true) или использует умолчание (false)
- value - значение атрибута, доступное для изменения

Реализация на языке Java:

```
public interface Attr extends Node {
    public String getName();
    public boolean getSpecified();
    public String getValue();
    public void setValue(String value);
}
```

Упрощенный программный интерфейс для XML (Simple API for XML, SAX)

- <http://www.megginson.com/SAX>
- разработан сообществом участников списка рассылки XML-DEV под руководством Дэвида Меггинсона (David Megginson)
- представляет собой потоково-ориентированный независимый от платформы и языка интерфейс, позволяющий программам обрабатывать данные в формате XML
 - используется классическая схема обработки событий
 - конструкции XML иницируют события, определенные в интерфейсе
 - программные функции реагируют на события и осуществляют необходимую обработку
- позволяет программам не держать в памяти иерархию XML-документа
 - приложения могут использовать интерфейс SAX для того, чтобы построить иерархию XML-документа из входного потока данных

- в настоящий момент разрабатывается SAX 2

Свободно распространяемые реализации базируются на:

- Java
- Python
- COM
- Perl
- C++
- множестве XML-процессоров, включающих интерфейс SAX

Примеры событий, реализованных в SAX:

- начало документа
- начало элемента
- буквенные знаки
- конец элемента
- конец документа

Упрощенный протокол доступа к объектам XML (SOAP)

- <http://search.ietf.org/internet-drafts/draft-box-http-soap-01.txt>
- определяет механизм удаленных вызовов процедур (RPC) с использованием синтаксиса XML
- реализует клиент-серверное взаимодействие по сети
- представляет собой стандартный протокол вызова объектов, построенный на стандартах Internet
 - Транспортный уровень - HTTP
 - Использование XML для кодирования
 - запросов вызова
 - ответов
- реализует расширяемый протокол
- реализует расширяемый формат передаваемой информации
 - использование словаря SOAP для адресных данных
 - включение в передаваемой информации пользовательских словарей
- реализует простые типы данных для значений, базирующихся на типах данных XML-схемы
 - включая массивы, составные типы и другие агрегаты простейших типов

Язык описания путей XML (XPath)

- <http://www.w3.org/TR/xpath>
- система адресации, идентифицирующая одну или несколько позиций в иерархии документа
- общая семантика и синтаксис для адресации в любых основанных на XML-приложениях
 - функциональность, необходимая для XSLT и XPointer
- компактный не XML-синтаксис
 - пригоден для использования в качестве значений атрибутов XML-документов:
 - `select="id('start')//question[@answer='y']"`
 - выбор всех элементов question, атрибут answer которых имеет значение "y" и которые являются потомками того элемента в данном документе, чей уникальный идентификатор имеет значение "start"

Отдельная рекомендация W3C

- <http://www.w3.org/TR/xpath>
- модель данных для представления XML-документа в виде дерева узлов
- механизм адресации информации, находящейся в узлах дерева

- язык выражений, для работы с булевыми, цифровыми и строковыми значениями, а также со значениями узлов
- ядро, над которым надстраивается функциональность, необходимая для XSLT или XPointer

XPath не является языком запросов

- одним из аспектов организации запросов является адресная информация, которую требуется найти
- другие аспекты организации запросов включают работу с адресуемой информацией до того, как возвращается ответ на запрос
- XPath используется только для адресации компонентов экземпляра XML-документа и сам по себе не обеспечивает никаких традиционных возможностей для запросов (хотя, возможно, будет рассмотрен в качестве схемы адресации теми, кто создает такие системы).

XSL и XSLT

XSL (Extensible Style Language) - расширяемый язык стилей

XSLT (XSL Transformations) - язык стилей для преобразования XML-документов

- <http://www.w3.org/TR/xslt>
- <http://www.w3.org/TR/xsl>
- два отдельных словаря XML
 - словарь трансформаций, позволяющий обработать входные документы и представить их на выходе в требуемом виде
 - включает модель обработки, которая позволяет осуществлять нелинейный проход по данным (идентичную модели обработки DSSSL и отличающуюся от CSS)
 - использует XPath как базовый язык выражений
 - добавлены специальные расширения, описывающие организацию связей
 - может трансформировать XML в XML, HTML, HTML + CSS, HTML + DOM + CSS или обычный текст
- словарь форматирования для задания способа внешнего представления документа
 - применяет семантику стилей уже существующих стандартов
 - DSSSL (Document Style and Semantics Specification Language -x ISO/IEC-10179)
 - CSS (Cascading Style Sheets - W3C)

Рассмотрим исходный файл, включающий элемент <note>

- элемент note должен быть отображен полужирным курсивом и заключен между двумя горизонтальными линиями, в виде отдельного абзаца, и снабжен префиксом "Note: "

Генерация форматизирующих объектов, соответствующих лексическим и синтаксическим правилам XML:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/XSL/Transform/1.0"
                xmlns:fo="http://www.w3.org/XSL/Format/1.0">

<xsl:import href="main.xsl"/>

<xsl:template match="note">
  <fo:display-rule/>
  <fo:block font-posture="italic" font-weight="bold">
    <xsl:text>Note: </xsl:text>
    <xsl:apply-templates/>
  </fo:block>
  <fo:display-rule/>
</xsl:template>

</xsl:stylesheet>
```

Генерация HTML, соответствующего лексическим и синтаксическим правилам SGML:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/XSL/Transform/1.0">

<xsl:output method="html"/>

<xsl:import href="main.xsl"/>

<xsl:template match="note">
  <hr/>
  <p><i><b>
    <xsl:text>Note: </xsl:text>
  <xsl:apply-templates/>
    </b></i></p>
  <hr/>
</xsl:template>

</xsl:stylesheet>
```

Связывание документов со стилями

- <http://w3.org/TR/xml-stylesheet>
- задает связь одного или нескольких стилей с данным XML-документом
- использует те же псевдоатрибуты и семантику, что и в HTML 4:
 - <LINK REL="stylesheet">
 - <LINK REL="alternate stylesheet">

Вспомогательная разметка:

- не является частью структурной разметки объекта, поэтому производится при помощи инструкций обработки, а не основной разметки

Типичный пример использования:

```
<?xml-stylesheet href="fancy.xsl" type="text/xsl"?>
<?xml-stylesheet href="normal.css" type="text/css"?>
```

Менее типичные, но тоже разрешенные примеры:

```
<?xml-stylesheet alternate="yes" title="small"
  href="small.xsl" type="text/xsl"?>
```

- указывает процессору альтернативный стиль, который можно активизировать дополнительными внешними данными

```
<?xml-stylesheet href="#style1" type="text/xsl"?>
```

- указывает процессору на стиль, расположенный в указанном месте в теле документа

Языки XML для ссылок и связей (XPointer и XLink)**Связь описывает отношение:**

- ассоциацию между двумя компонентами
- адреса ассоциируемых конструкций
- смысл отношения

Две рекомендации W3C:

- разработаны для совместной работы и достижения двух указанных целей (обеспечения ссылок и связей)
- XPointer
 - <http://www.w3.org/TR/xptr>
 - определяет синтаксис, позволяющий задавать адреса внутри XML-документа
 - определяет ссылки на элементы, символьные строки и другие части XML-документа
 - адресует множества узлов или индивидуальные узлы
 - расширяет механизмы, существующие в рамках XPath
 - пригоден для использования в URI в виде идентификатора фрагмента
 - независим от использования в качестве указателя ссылки или для других специальных целей
- XLink
 - <http://www.w3c.org/TR/xlink>
 - определяет синтаксис, позволяющий описывать связи между адресуемыми частями информации
 - декларирует существование отношения между двумя или более объектами данных или частями объектов данных
 - позволяет создать простые (однаправленные) гиперссылки
 - позволяет создать и более сложные (многоэлементные и типизированные) ссылки
 - включает семантические и поведенческие атрибуты
 - существует мнение, что различные типы поведения должны определяться на более высоком уровне, чем данный, как это было сделано в средствах форматирования

Богатое наследие:

- базируются на концепциях, успешно разработанных и используемых в рамках:
 - TEI (Text Encoding Initiative)
 - HyTime (Hypermedia and Time-based Structuring Language - ISO/IEC-10744)

Основная разметка:

- конструкции не являются второстепенными по отношению к разметке документа
- размечаются при помощи элементов и атрибутов, в отличие от вспомогательных конструкций, таких, как ассоциация со стилями, использующей инструкции обработки
- связи между двумя или большим количеством единиц информации является частью сущности описываемой информации

Тематические карты XML (Topic Maps)

- <http://www.infoloom.com/topmap.htm>
- <http://www.TopicMaps.com>
- <http://www.TopicMap.com>
- архитектура определения нотаций для структурирования информационных ресурсов путем определения тем и их взаимосвязей
 - местонахождение
 - группирование адресуемых информационных объектов
 - ассоциации
 - отношения между темами
 - аспекты
 - пары «свойство/значение» у информационных объектов и их компонентов
 - области определения
 - ограничения на ассоциации (из соображений согласованности и осмысленности)
- основаны на стандарте ISO/IEC 13250:1999 Topic Maps
- в свою очередь, основанном на ISO/IEC 10744:1997 HyTime

- переработаны для реализации в XML
- технология адресации на базе XPointer

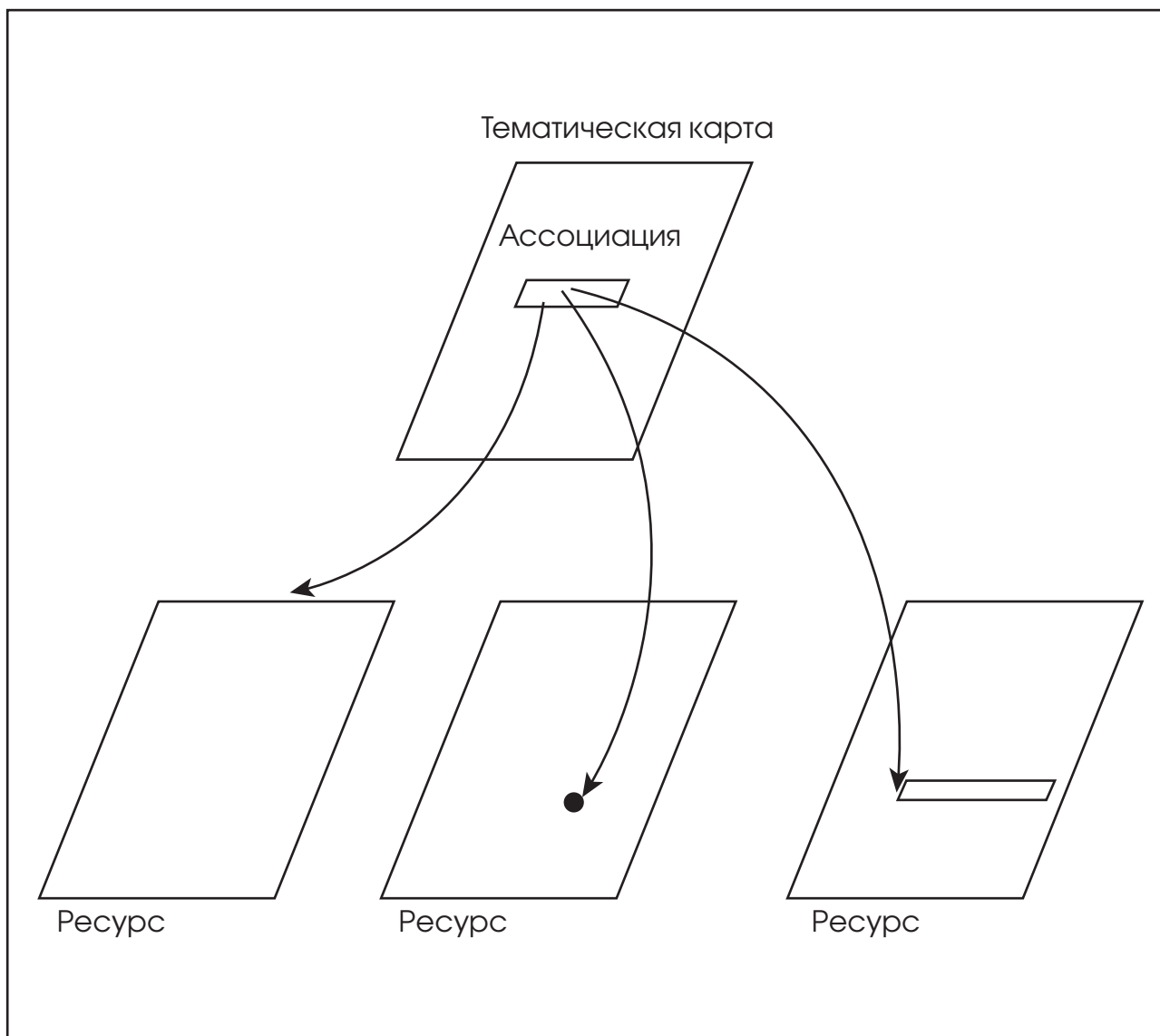
Сосуществующие модели областей знаний

- идеальны для навигации по концептуальным областям и темам
 - используются для связывания информации, распределенной по множеству ресурсов разнообразных типов
- ориентированы на конечного пользователя информации — человека
 - идеальны для указателей, словарей, энциклопедий, систем контекстной помощи
- обычно являются дополнением к массиву информации
 - расположены отдельно от самой информации
 - конкретный массив информации не изменяется от наложения тематических карт

Архитектура, основанная на:

- средствах моделирования данных, определяющих словари на основе архитектуры тематических карт
- инструментах навигации, позволяющих подстраиваться под произвольные словари за счет работы на уровне базовой архитектуры тематических карт

Пример документа, описывающего навигацию:



Шаблон описания ресурсов (Resource Definition Framework, RDF)

- <http://www.w3.org/RDF>
- <http://www.w3.org/TR/PR-rdf-schema>
- <http://www.w3.org/TR/REC-rdf-syntax>
- обобщенный способ работы с метаданными (информации об информации)
- ориентирован на программное обеспечение в качестве конечного потребителя информации
 - позволяет осуществлять автоматическую обработку Web-ресурсов
 - поиск
 - каталогизацию
 - генерацию иерархических карт сайтов
- модель данных RDF напоминает диаграмму отношений объектов
 - классы и их свойства
- словарь описания схемы с XML-синтаксисом
 - пользователь определяет язык описания ресурсов как схему, специфичную для конкретного приложения
 - атрибуты и отношения элементов на Web
 - ресурсы и их свойства

Запросы XML (XML Query)

- <http://www.w3.org/TR/xmlquery-req>
- создается модель данных для XML-документов
 - используется информационный набор XML
 - включена поддержка пространств имен
- создается набор операторов запроса к данным
 - для заданного набора документов
 - для одного документа
 - для набора простых и сложных величин
- обеспечивается независимость от любой стратегии вычислений
 - полная декларативность
- обеспечивается поддержка простых и сложных типов данных XML-схемы
- обеспечивается поддержка ссылок внутри и между XML-документами
- обеспечивается поддержка реальных и виртуальных документов
 - может работать над временными результатами трансформаций документов, не представленными в статической форме

Цифровые подписи XML

- <http://www.w3.org/TR/xmlsig-requirements>
- <http://www.w3.org/TR/xmlsig-core/>
- XML-синтаксис для создания и представления цифровой подписи
- процедуры для вычисления и верификации подписей
 - методы ссылок на наборы ресурсов
 - алгоритмы
 - ключевая информация и управление ключами
- гибкие методы подписи
 - вложенные цифровые подписи содержания одного или нескольких ресурсов, содержащиеся в том же или в отдельном XML-документе
 - автономные цифровые подписи данных, не содержащихся в XML-документе с цифровой подписью

- основные сервисы цифровой подписи
 - целостность данных («Эти данные не были изменены и они полны»)
 - аутентификация («Эти данные поступили от меня»)
 - невозможность отказаться от присланного документа («Эти данные поступили от Вас»)

Масштабируемая векторная графика (Scalable Vector Graphic, SVG)

- <http://www.w3.org/TR/SVG/>
- <http://www.svgcentral.com/>
- словарь для представления двумерных графических объектов
 - общая семантика с различным прикладным синтаксисом
 - ориентированность на обмен данными:
 - атрибуты отображения, заданные в экземпляре документа
 - гарантированная совместимость
 - специально не включает возможности задания стилей вместе с данными
 - XML-альтернатива форматам обмена графикой, таким, как Encapsulated PostScript (EPS)
 - пригодность для стилизации:
 - атрибуты отображения, не заданные в экземпляре документа
 - на отображение могут влиять любые языки стилей:
 - внешние стили
 - внутренние стили
 - встроенные блоки описания стилей
 - наследование стилевых свойств
 - использование DOM позволяет динамически изменять существующие экземпляры данных SVG
 - результат может быть получен как результирующий словарь после XSL трансформации исходных XML-данных

Основные типы конструкций:

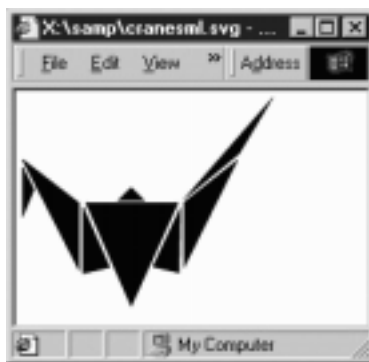
- геометрический вектор
- изображение
- текст

Неограниченные области применения:

- построение диаграмм
- визуализация информации
- картография

Простой пример с многоугольниками:

```
<?xml version="1.0"?>
<svg width="175" height="145" >
  <g style="stroke:black; fill:black" >
    <polygon points=" 5, 50, 5, 81, 12, 64" />
    <polygon points=" 5, 45, 41,116, 41, 73" />
    <polygon points=" 44, 76, 44,119, 61,115" />
    <polygon points=" 46, 73, 75,140,105, 73" />
  </g>
  <g stroke="black" fill="black">
    <polygon points="107, 76,106,119, 89,115" />
    <polygon points="144, 45,109,116,109, 73" />
    <polygon points="145, 41,167, 4,109, 71" />
    <polygon points=" 66, 72, 75, 63, 84, 72" />
  </g>
</svg>
```



Язык математической разметки (Mathematics Markup Language, MathML)

- <http://www.w3.org/TR/REC-MathML>
- <http://www.w3.org/TR/MathML2>
- словарь для описания математической нотации
 - описывает структуру и внешнее представление
 - 28 типов элементов
 - традиционные математические конструкции
 - основные наборы символов
 - средства для построения выражений
 - описывает математическую семантику
 - 75 типов элементов
 - кодирование нижележащих математических концепций
 - охватывает неоднозначную презентацию различных концепций
 - может обрабатываться и вычисляться
 - поддерживает (хотя и не в полном объеме) арифметику, алгебру, логику и операции отношений, вычисления, теорию множеств, последовательности, тригонометрию, статистику, линейную алгебру
 - полностью определяет взаимодействие семантики и презентации
- каталоги объектов, представляющие расширенный набор знаков

Протокол беспроводного доступа к приложениям (WAP)

- <http://www.wapforum.org/>
- словарь, ориентированный на доставку и представление информации в мобильных приложениях
 - язык разметки WML
 - производный от HTML
 - включает язык скриптов WML
- управление отображением, оптимизированное для маленьких экранов
 - масштабируемость от однострочных до графических дисплеев
- средства навигации, оптимизированные для работы только одной рукой
- средства обеспечения безопасности
- клиент-серверная архитектура
 - простой «тонкий» пользовательский агент (микро-навигатор)
 - шлюз WAP управляет и направляет информационные потоки

Jet Info
ИНФОРМАЦИОННЫЙ БЮЛЛЕТЕНЬ

Издается с 1995 года

Издатель: компания Джет Инфо Паблишер

Главный редактор: Дмитриев В.Ю. (vlad@jet.msk.su)
Технический редактор: Антонов А.Н. (silver@jet.msk.su)
Россия, 103006, Москва, Краснопролетарская, 6
тел. (095) 972 11 82, 972 13 32
факс (095) 972 07 91
e-mail: JetInfo@jet.msk.su
<http://www.jetinfo.ru>



Подписной индекс по каталогу Роспечати

32555