ФОРМАЛЬНЫЕ МЕТОДЫ ВЕРИФИКАЦИИ ЦЕЛОСТНОСТИ МАКРОСТРУКТУРЫ WEB-САЙТОВ

М.Р. Когаловский, Е.Н. Ефимова, Т.А. Рыбина, В.Б. Брахин
Институт проблем рынка РАН
(опубликована в журнале "Программирование", Российская Академия наук,
Издательство "Наука - МАИК/Интерпериодика", 4, 2000).
e-mail: kogalov@cemi.rssi.ru

Аннотация

Рассматривается проблема верификации макроструктуры Web-сайта, информационные ресурсы которого представлены средствами языка HTML. Под макроструктурой сайта мы понимаем структуру взаимосвязей его ресурсов, основанную на гиперссылках. Предлагаемый подход предусматривает инвентаризацию информационных ресурсов сайта, частичный синтаксический анализ представленных на нем HTML-файлов, выявление имеющихся в них гиперссылок и, тем самым, восстановление его макроструктуры (графа гиперсвязей ресурсов). Описание макроструктуры сайта помещается в реляционную базу данных и анализируется далее формальными методами, основанными на реляционной алгебре и теории графов, с целью обнаружения неактуальных ("висячих") гиперссылок, ресурсов, на которые не указывает ни одна внутренняя гиперссылка, а также гипертекстовых страниц, недостижимых из домашней страницы сайта. В работе обсуждаются также возможности использования данного подхода для Web-сайтов, созданных средствами XML-технологий, кратко рассматривается реализованный прототип системы верификации HTML-сайтов. Работа частично поддержана грантом РГНФ 96-02-12016.

1 Введение

Одной из важных задач, связанных с разработкой и поддержкой Web-сайта, является обеспечение целостности структуры его ресурсов. Будем различать далее *микроструктуру* сайта - структуру содержания отдельных его страниц (документов) - и *макроструктуру*, т.е. структуру взаимосвязей информационных ресурсов сайта, основанную на гиперссылках. Методы, предложенные в этой работе, служат для исследования целостности макроструктуры.

Верификации целостности макроструктуры сайта сводится к анализу существования гиперссылок, указывающих на каждый из ресурсов сайта, а также существования целевых ресурсов гиперссылок и навигационных путей от домашней страницы к каждой другой странице сайта. Более точная формулировка задачи приведена в разд. 3.

Рассматриваемая здесь проблема аналогична проблеме *целостности по ссылкам* (*Referential Integrity*) в области баз данных. Однако, в отличие от гипермедийных информационных систем, основанных на Web-технологиях, в традиционных системах баз данных (реляционного типа) имеются специальные механизмы автоматической поддержки целостности по ссылкам. Для их активизации достаточно декларировать соответствующие ограничения в схеме базы данных средствами описания данных, и поддержка таких ограничений будет автоматически осуществляться системными механизмами. При этом проверка рассматриваемых ограничений осуществляется динамически, непосредственно в процессе выполнения операций манипулирования данными в базе данных.

Ограничения целостности по ссылкам для Web-сайтов не могут проверяться в процессе порождения новых информационных ресурсов, поскольку язык HTML не обладает средствами декларации таких ограничений, и они не могут поддерживаться Web-серверами при помещении на них новых HTML-страниц. Web-технологии, основанные на языке HTML, оперируют слабоструктурированными данными [1], для которых отсутствуют возможности типизации и интенсионального определения данных. Поэтому спецификация каких-либо абстрактных ограничений целостности макроструктуры сайтов и использование автоматических механизмов их поддержки оказываются невозможными. По существу, ограничения целостности по ссылкам представляются здесь неявно. При корректном проектировании сайта предполагает-

ся, что если имеется ссылка, то должен существовать и целевой ее ресурс, если существует ресурс, то на него должна быть хотя бы одна ссылка. Однако на практике эти ограничения часто нарушаются как для ссылок на ресурсы самого сайта в результате ошибок Web-мастера, так и для ссылок на ресурсы других сайтов в связи с автономностью сайтов в среде WWW. Эти ситуации обнаруживаются лишь непосредственно в процессе навигации и доступа пользователей к ресурсам сайтов.

Верификация целостности макроструктуры Web-сайтов является важной функцией Web-мастера. Регулярное ее выполнение при изменениях ресурсов сайта позволяет в значительной мере уменьшить количество случаев возникновения раздражающих пользователей отказов в получении запрашиваемых ими информационных ресурсов, дает возможность обнаружить избыточные ресурсы сайта.

В данной работе предлагается подход к решению задачи верификации макроструктуры Web-сайтов, основанных на HTML-технологии, который не требует для этой цели непосредственного обхода элементов структуры сайта по гиперссылкам. Используя программные средства, реализующие рассматриваемый подход, Web-мастер может легко исследовать макроструктуру сайта после каждой модификации его содержания и тем самым постоянно поддерживать ее целостность.

Заметим, что особую проблему составляет поддержка актуальности ссылок на внешние ресурсы. Ресурсы этого рода не контролируются Web-мастером данного сайта, и при отсутствии регулярного мониторинга их состояния такие ссылки часто могут становиться неактуальными.

Предлагаемый подход более подробно рассматривается в следующих разделах работы.

2 Основные понятия

Прежде всего, введем некоторые понятия, которые мы будем использовать далее наряду с общеупотребительными в области World Wide Web (см., например, [2, 3, 4]).

Будем рассматривать *Web-сайт* как совокупность взаимосвязанных гипертекстовых (гипермедийных) ресурсов Web, обладающих единством содержания.

Информационные ресурсы Web-сайта - это содержащиеся в нем файлы допустимых для HTML-среды форматов (HTML, TXT, PDF, ZIP, GIF, JPEG, CLASS и др.), а также идентифицируемые якорными точками (тег $< A\ \ NAME = ... >$) фрагменты HTML-файлов. Взаимосвязи ресурсов сайта обеспечиваются *гиперссылками*, содержащимися в его HTML-файлах. Каждая гиперссылка задает бинарную ориентированную связь между исходным и целевым ресурсами. *Исходным ресурсом* при этом всегда является HTML-файл, содержащий данную ссылку. *Целевым ресурсом* ссылки может быть не только ресурс данного сайта (*внутренний ресурс*), но и ресурс какого-либо другого сайта или других информационных служб Internet - FTP, Telnet, Gopher и т.д. (*внешний ресурс*). В соответствии с этим, будем различать *внутренние* и *внешние* гиперссылки.

Только такие ресурсы Web-сайтов, которые могут потенциально являться исходными или целевыми для каких-либо гиперссылок в HTML-файлах, принимаются во внимание в данной работе. К их числу не относятся, например, ресурсы систем баз данных или других приложений, доступные пользователям сайта через CGI-интерфейс Web-сервера, на котором поддерживается данный сайт, или иным образом. Такие ресурсы сайта будем назвать *скрытыми*.

Будем далее различать две составные части гиперссылки - имя гиперссылки и ее значение. *Именем гиперссылки* будем называть ту ее часть (обычно отражающую смысл целевого ресурса), которую пользователь видит на экране, когда браузер воспроизводит на экране дисплея содержимое запрашиваемой HTML-страницы. Точнее говоря, имя гиперссылки - это строка, содержащаяся между открывающим и закрывающим тегом гиперссылки. В случае, если роль имени ссылки играет какой-либо графический образ, то будем далее использовать в качестве имени такой ссылки имя соответствующего графического файла. *Значением гиперссылки* будем называть заданный в ней URL целевого ресурса. Именно благодаря тому, что значение гиперссылки может оставаться скрытым от пользователя, осуществляющего навигацию в гиперсреде WWW, распределенность этой среды является для него *прозрачной*.

Гиперссылку будем называть *актуальной* в данный момент, если целевой ресурс, заданный ее значением (URL), действительно в этот момент существует и доступен. В противном случае ссылка называется *неактуальной*. Поддержка актуальности внутренних ссылок - процесс, полностью контролируемый администратором данного Web-сайта. Что касается внешних ссылок, то, как уже отмечалось, Web-мастер может лишь осуществлять мониторинг их актуальности, поскольку внешние информационные ресурсы сайта автономны и им не контролируются. Нарушение актуальности ги-

перссылки требует от Web-мастера исключения ее вхождений в HTML-страницах данного сайта, изменения должным образом ее значения либо переименования файла целевого ресурса.

Информационный ресурс сайта будем называть *несвязанным*, если он не является целевым ни для каких гиперссылок в этом сайте. Если следовать этому определению, к числу несвязанных ресурсов иногда может быть отнесена домашняя страница сайта. Однако это - вырожденный случай, не представляющий практического интереса.

Строго говоря, несвязанность ресурса не может трактоваться как его избыточность, поскольку всегда возможен доступ к таким ресурсам непосредственно по их URL, который может быть предоставлен, например, в результате обращения пользователя к поисковой машине WWW. Тем не менее, разработчики сайтов обычно стремятся не допускать наличия такого рода ресурсов.

В соответствии с существующими традициями проектирования, предполагается, что каждый Web-сайт имеет единственную точку входа - *домашнюю страницу*. Ее URL рассматривается как адрес сайта. Каждый HTML-файл (за исключением, возможно, файла-носителя домашней страницы) должен быть *достижимым* из домашней страницы с помощью навигационных операций по гиперсвязям ресурсов, т.е. должен существовать хотя бы один путь к нему по гиперссылкам из домашней страницы.

Теперь мы можем ввести понятие целостной макроструктуры сайта. Макроструктура данного сайта является *целостной*, если все содержащиеся в нем гиперссылки актуальны, он не содержит несвязанных ресурсов, и все его HTML-файлы достижимы из домашней страницы.

Для дальнейшего рассмотрения важно сделать следующее замечание. Мы предполагаем, что совокупность внутренних ресурсов сайта представляет собой множество всех файлов, содержащихся в некотором поддереве дискового каталога Web-сервера. Поэтому для задания ресурсов сайта достаточно задать соответствующий подкаталог корневого каталога сервера.

3 Постановка задачи и общий подход

Пусть для некоторого Web-сайта известны его URL, например, $SiteAddr$, подкаталог каталога Web-сервера $SitePath$, содержащий информационные ресурсы сайта, а также имя HTML-файла его домашней страницы $SiteHome$.

Задача заключается в том, чтобы проверить целостность макроструктуры данного сайта и диагностировать обнаруженные случаи ее нарушения, не прибегая при этом к непосредственному обходу структуры сайта по навигационным путям, образуемым гиперссылками, которые содержатся в принадлежащих ему HTML-файлах.

Предлагаемый общий подход к решению указанной задачи сводится к следующему.

Прежде всего проводится инвентаризация информационных ресурсов сайта. После этого осуществляется частичный синтаксический анализ содержимого HTML-файлов и выявляются имеющиеся в них гиперссылки. Тем самым восстанавливается в явном виде макроструктура сайта (граф гиперсвязей его ресурсов). Сведения о ней помещаются в реляционную базу данных. Далее с помощью формальных методов, основанных на реляционной алгебре [5, 6] и теории графов [7, 8], осуществляется собственно верификация целостности макроструктуры сайта. Выявляются имеющиеся неактуальные ссылки, несвязанные ресурсы, а также недостижимые из домашней страницы HTML-файлы, если они имеются.

Методы решения этой задачи рассматриваются в следующем разделе.

4 Формализация предлагаемого подхода

Процесс верификации макроструктуры сайта состоит из описанных ниже этапов.

4.1 Восстановление макроструктуры сайта

Для восстановления макроструктуры анализируемого сайта прежде всего необходимо осуществить инвентаризацию его информационных ресурсов. (В соответствии с нашим предположением, к их числу относятся все файлы, содержащиеся в соответствующем данному сайту подкаталоге дискового каталога Web-сервера, на котором поддерживается рассматриваемый сайт.) С этой целью в реляционной базе данных системы верификации строится отношение $RES(P, FN, FE, A)$, в котором каждому файлу сайта или фрагменту HTML-файла, идентифицируемому якорной точкой (тегом $< A\ NAME = ... >$), соответствует отдельный кортеж, а атрибуты этого отношения имеют следующий смысл :

P - путь доступа к данному файлу

FN - имя файла

FE - тип файла (расширение имени)

A - имя якорной точки фрагмента файла.

Кортежи со значениями этого атрибута, отличными от неопределенного (Null-Value), соответствуют якорным точкам и записываются в базу данных позднее, в процессе синтаксического анализа содержимого HTML-файлов при обнаружении в них этих якорных точек. В остальных кортежах этот атрибут имеет неопределенное значение.

Теперь можно приступить к восстановлению макроструктуры сайта. Для этого необходимо обнаружить все гиперссылки, которые содержатся в HTML-файлах сайта. С этой целью проводится частичный синтаксический анализ содержимого всех HTML-файлов, зарегистрированных в отношении RES. Для каждой гиперссылки, встретившейся в процессе анализа, создается один кортеж в отношении $Lnk(SP, SFN, LN, Pr, SA, TP, TFN, TFE, TA)$ базы данных. Атрибуты этого отношения имеют следующий смысл:

SP - путь доступа к HTML-файлу сайта, содержащему данную гиперссылку

SFN - имя исходного HTML-файла ссылки

LN - имя данной ссылки

Pr - протокол доступа к целевому ресурсу

SA - адрес сервера целевого ресурса ссылки

TP - путь доступа к целевому ресурсу на сервере

TFN - имя файла целевого ресурса ссылки

TFE - тип этого файла (расширение имени)

TA - имя якорной точки, идентифицирующей целевой фрагмент HTML-файла гиперссылки.

Кроме того, как указывалось выше, выявленные в процессе синтаксического анализа HTML-файлов якорные точки регистрируются в отношении RES, где для каждой из них порождается один кортеж.

Построенные таким образом отношения RES и Lnk полностью представляют макроструктуру рассматриваемого сайта, и можно приступить к ее анализу.

4.2 Выявление неактуальных внутренних ссылок

Для решения этой проблемы сначала с помощью реляционной операции селекции (σ) декомпозируем отношение Lnk на два, одно из которых $ILnk$ будет содержать только внутренние ссылки, а другое $ELnk$ - только внешние:

$ILnk = \sigma_{SA=SiteAddr}(Lnk)$;

$ELnk = \sigma_{SA\neq SiteAddr}(Lnk)$.

Теперь можно с помощью реляционных операций внутреннего соединения ($\bowtie$), проекции (π) и разности ($-$) отношений построить новое отношение $NoLnk$ с той же схемой, что и у Lnk, содержащее сведения о неактуальных внутренних ссылках в сайте:

$$NoLnk = ILnk - \pi_{AtL}(ILnk \underset{Crit}{\bowtie} RES),$$

где:

AtL=$\{SP, SFN, LN, Pr, SA, TP, TFN, TFE, TA\}$ – множество атрибутов отношения $ILnk$,

$Crit$=$(TP$=P&TFN=FN&TFE=FE&TA=$A)$ - критерий соединения отношений в формуле вычисления отношения $NoLnk$.

Актуальность обнаруженных внешних ссылок, представленных в отношении $ELnk$, может быть проверена для каждой из них только с помощью попытки непосредственного доступа к ее целевому ресурсу.

4.3 Нахождение несвязанных внутренних ресурсов

Для нахождения несвязанных ресурсов сайта строится отношение $UnRef$ с той же схемой, что и у отношения RES, в котором каждому несвязанному внутреннему ресурсу сайта соответствует один кортеж:

$$UnRef = RES - \pi_{AtR}(RES \underset{Crit}{\bowtie} ILnk),$$

где AtR=$\{P, FN, FE, A\}$ - множество атрибутов отношения RES, а $Crit$ имеет тот же смысл, что и ранее в формуле для отношения $NoLnk$.

4.4 Нахождение ресурсов, недостижимых из домашней страницы

Если представить макроструктуру анализируемого сайта в виде ориентированного графа (орграфа), вершины которого соответствуют его гипертекстовым файлам, а дуги - связывающим их гиперссылкам, то рассматриваемая проблема сводится к хорошо известной - к построению матрицы достижимости для орграфа [7, 8].

Напомним, что две вершины графа находятся в бинарном отношении достижимости, если в графе существует путь из первой вершины во вторую. Матрица достижимости орграфа описывает указанное отношение на множестве его вершин. Она представляет собой матрицу смежности транзитивного замыкания данного орграфа. Единичные элементы в некоторой строке этой матрицы соответствуют тем вершинам графа, которые достижимы из вершины, соответствующей данной стро-

ке. Один из возможных алгоритмов построения матрицы достижимости предложен в [8].

Поскольку рассматриваемый здесь подход предполагает использование реляционной базы данных, мы полагаем, что более естественно воспользоваться для указанных целей иным алгоритмом, который эффективно реализуется в терминах реляционной алгебры. Рассмотрим предлагаемый нами алгоритм.

Пусть D - отношение, представляющее множество имен всех файлов, содержащих гипертекстовые ресурсы данного сайта. Как нетрудно видеть, отношение, которое содержит кортежи, соответствующие этим файлам, можно построить из отношения RES с помощью суперпозиции реляционных операций селекции, проекции и исключения дубликатов кортежей.

Пусть далее SRC - отношение, которое содержит подмножество всех кортежей отношения $ILnk$, соответствующих гиперссылкам, целевыми ресурсами которых являются HTML-файлы, отличные от файла - исходного ресурса ссылки. Это отношение строится из Lnk с помощью операции селекции и исключения дубликатов кортежей.

При этих условиях проблема анализа достижимости страниц сайта решается следующим итеративным алгоритмом (записанным на смеси языка Паскаль с нотацией реляционной алгебры):

$D := adom^*(\pi_{P,FN}\ (\sigma_{FE="html"}(RES)))$;

$SRC :=$

$adom^*(\sigma_{TFE="html"\&\neg(SP=TP\&SFN=TFN)}(ILnk))$;

$x := 0$;

$CUR :=< SP : SitePath,\ SFN : SiteHome >$;

while $x < card(CUR)$ *do*

begin

$x := card(CUR)$;

$CUR_1 := \pi_{SRC.TP,SRC.TFN}\ (CUR \underset{CritJ}{\bowtie} SRC)$;

$CUR := adom^*(CUR \cup CUR_1)$

end;

где:

$card(R)$ - функция, возвращающая текущее количество кортежей в отношении R,

$adom^*(R)$ - актуальный составной домен отношения R, включающий все его атрибуты,

$CritJ =$

$(CUR.SP{=}SRC.SP\&CUR.SFN{=}SRC.SFN)$ - критерий соединения в формуле вычисления отношения CUR_1,

σ, π, $\bowtie$, $\cup$ - символы реляционных операций селекции, проекции, внутреннего соединения и объединения отношений.

В результате исполнения этого алгоритма будет фактически построена единственная нужная нам строка матрицы достижимости для исходного графа, а именно строка, соответствующая его вершине, которая представляет домашнюю страницу исследуемого Web-сайта. Нетрудно видеть, что число возможных итераций не превышает числа гипертекстовых страниц сайта, т.е. заведомо не превышает $card(D)$.

Теперь вычислим отношение D^* следующим образом:

$D^* := D{-}CUR$.

Каждый кортеж этого отношения указывает некоторый HTML-файл, не достижимый из домашней страницы сайта. Наша задача, таким образом, решена.

5 Верификация макроструктуры XML-сайтов

В настоящее время наряду с массовой разработкой Web-сайтов с использованием языка HTML интенсивно развиваются технологии, основанные на новом языке разметки - *Extensible Markup Language (XML)* [9], спецификации которого недавно приобрели статус стандарта W3C (*World Wide Web Consortium*), и на его инфраструктуре [10]. В этой связи возникает вопрос о возможности использования предложенного здесь подхода к верификации макроструктуры XML-сайтов.

Прежде всего, нужно отметить, что использование XML-технологий открывает радикально новые возможности поддержки целостности микроструктуры сайта. Декларации *Document Type Definition (DTD)* в языке XML позволяют описывать структурные свойства XML-документов. При этом структура документа определяется как последовательность элементов и/или иерархий элементов определяемых в документе типов.

Используя эти метаданные, различные приложения XML, например, программы-браузеры, могут контролировать целостность структуры отдельных XML-документов перед публикацией их на Web-сервере, а также на стадии просмотра. DTD играет при этом роль, аналогичную роли схемы базы данных в системах баз данных.

Более развитые средства описания структуры и других свойств XML-документов обеспечивают разрабатываемые W3C спецификации языка опре-

деления схемы для XML-документов [11, 12].

При создании ресурсов сайта в среде XML для формирования его структуры могут использоваться структурообразующие средства самого языка XML (спецификации DTD), а также гиперссылки и указатели, связывающие между собой XML-документы и/или фрагменты документов. Для декларации гиперссылок и указателей предусматривается использование разрабатываемых в настоящее время W3C языков XLink [13] и XPointer [14].

Спецификации XML DTD и внутридокументные указатели, описанные на языке XPointer, - это средства образования микроструктуры XML-документа, и они не имеют отношения к обсуждаемым здесь проблемам. Макроструктура сайта формируется средствами языка XLink и указателей языка XPointer, целевой ресурс которых не принадлежит XML-документу, в котором эти указатели определены. Рассмотрим функциональные возможности этих средств несколько подробнее.

Указатели в языке XPointer аналогичны ссылкам на якорные точки в языке HTML в том смысле, что целевыми ресурсами определяемых ими связей являются фрагменты документов. Однако идентификация этих фрагментов основана на их структурных свойствах - типе элемента, относительном положении его экземпляра в документе и т.д. В то время, как в HTML-документах целевые ресурсы - фрагменты документов - явно идентифицированы с помощью тегов $< A \quad NAME = ... >$, в XML-документах они явным образом не выделены. Поэтому для применения предложенных нами методов пришлось бы регистрировать в списке внутренних ресурсов сайта на этапе их инвентаризации не только все XML-файлы сайта, но и все экземпляры элементов XML-документов. Это необходимо, поскольку пока не выделены все связи, определенные в XML-документах сайта (что делается уже на следующем этапе), нам неизвестно, какие из экземпляров элементов и каких документов являются целевыми ресурсами этих связей. Очевидно, что такой подход был бы неэффективным. Как нам представляется, задачу исследования связей с целевым ресурсом - фрагментом документа целесообразно решать средствами синтаксического анализа содержания документа, а не предложенными здесь формальными методами.

Рассмотрим теперь гиперсвязи, формирующие макроструктуру сайта, которые поддерживаются средствами языка XLink, т.е. связи между документами. В соответствии с указанным выше, будем далее рассматривать только случай, когда целевым ресурсом связи является полный документ, а не его фрагмент.

В языке XLink предусматривается три вида гиперсвязей между XML-документами - простые ссылки, расширенные ссылки и групповые расширенные ссылки. Каждая из них представляется в XML-документе связующим элементом соответствующего типа.

Простая ссылка функционально полностью аналогична гиперссылке в языке HTML. Она имеет единственный целевой ресурс. *Расширенная ссылка* декларирует, по существу, некоторую совокупность целевых ресурсов и порядок их обхода, иначе говоря, связь вида "1 :N" с упорядочением целевых ресурсов. *Групповая расширенная ссылка* также представляет собой в общем случае множественную связь вида "1 :N". Она отличается от расширенной ссылки фактически лишь тем, что предписывает XML-процессору иной режим обработки ссылки. Для наших целей важно отметить, что актуальность как расширенной, так и групповой расширенной ссылки, эквивалентна актуальности N простых ссылок, которые являются их компонентами. Кроме того, все декларируемые ими навигационные маршруты между ресурсами образуются, в конечном счете, этими простыми ссылками. Поэтому для анализа связности ресурсов сайта и достижимости их нужно, как и в случае HTML-сайта, попрежнему исследовать множества простых ссылок.

Таким образом, для XML-сайтов можно утверждать следующее. Все макроструктурные их связи могут быть попрежнему выявлены с помощью частичного синтаксического анализа. Этот анализ несколько усложняется, по сравнению с языком HTML, поскольку XML обладает существенно более развитой функциональностью в структурообразовании и, соответственно, использует более сложный синтаксис. Далее, проверка актуальности всех макроструктурных связей между полными документами сводится к проверке актуальности бинарных связей между ними. Более того, все существующие навигационные пути между ресурсами сайта также определяются бинарными связями. Поэтому соответствующее подмножество макроструктуры сайта, очевидно, может быть описано данными, представляемыми в рассмотренной выше базе данных, которая используется для случая HTML-сайтов. Следовательно, для решения задач верификации целостности указанного подмножества макроструктуры сайта можно использовать описанный выше формальный аппарат.

Что касается исследования связей между фрагментами XML-документов, то использование нашего подхода для этих целей представляется неэффективным по указанным выше причинам.

6 Реализация прототипа

Для реализации описанного подхода авторами разработан прототип программной системы, использующий реляционную СУБД Clipper и алгоритмическим путем осуществляющий анализ и верификацию макроструктуры HTML-сайтов.

Помимо решения задачи собственно верификации целостности структуры сайта с помощью методов, описанных в разд. 4, прототип диагностирует некоторые обнаруженные в HTML-файлах синтаксические ошибки, генерирует список встретившихся адресов электронной почты, формирует HTML-страницу, содержащую полный список имеющихся в сайте внешних ссылок. Актуальность таких ссылок автоматически в прототипе не проверяется, однако Web-мастер может, используя сформированную страницу, проверить далее существование и доступность нужных внешних ресурсов вручную с помощью обычного Web-браузера, последовательно "прозванивая" внешние ссылки.

Поскольку рассматриваемый прототип реализован на платформе IBM PC в среде MS-DOS, он не работает с "длинными" именами файлов. Кроме того, при реконструкции макроструктуры сайта не анализируются некоторые сравнительно редко используемые конструкции языка HTML, которые могут содержать в общем случае структурообразующие элементы.

В последнее время появились некоторые коммерческие и свободно доступные программные средства, предназначенные для поддержки работы Web-мастера (см., например, [15]), которые позволяют синтезировать по запросу графическое представление структуры сайта и тем самым дают возможность визуального ее анализа. Какие-либо публикации о реализации решения рассматриваемой в данной работе проблемы формальными методами с автоматической диагностикой нарушений целостности макроструктуры сайта нам неизвестны.

Список литературы

[1] Florescu D., Levy A., Mendelzon A. *Database Techniques for the World-Wide Web: A Survey.* SIGMOD Record, Vol. 27, No. 3, September 1998. Есть рус. пер.: Флореску Д., Леви А., Мендельсон А. *Технологии баз данных для World-Wide Web: обзор.* СУБД, 4-5/1998.

[2] Клименко С., Уразметов В. *Internet. Среда обитания информационного общества.* - Протвино: Российский центр физико-технической информатики, 1995.

[3] Рассохин Д., Лебедев А. *World Wide Web - Всемирная Информационная Паутина в сети Internet.* - М.: Химический факультет МГУ, 1997.

[4] Храмцов П. *Лабиринт Internet. Практическое руководство.* - М.: Электронинформ, 1996.

[5] Мейер Д. *Теория реляционных баз данных* /Пер. с англ. под ред. М.Ш. Цаленко. - М.: Мир, 1987.

[6] Ульман Дж. *Основы систем баз данных* /Пер. с англ. с предисл. и под ред. М.Р. Когаловского. - М.: Финансы и статистика, 1983.

[7] Берж К. *Теория графов и ее применения* /Пер. с фр. под ред. И.А. Вайнштейна. - М.: ИЛ, 1962.

[8] Евстигнеев В.А. *Применение теории графов в программировании.* - М.: Наука, Гл. ред. физ.-мат. литературы, 1985.

[9] *Extensible Markup Language (XML) 1.0.* W3C Recommendation 10-February-1998. http://www.w3.org/TR/1998/REC-xml-19980210.

[10] Булах Е., Кузина И., Храмцов П. *Развитие стека спецификаций W3C или гносеология XML.* Открытые системы, 5-6, 1999.

[11] *XML Schema Part 1: Structures.* W3C Working Draft 5, November 1999. http://www.w3.org/TR/1999/WD-xmlschema-1-19991105.

[12] *XML Schema Part 2: Datatypes.* W3C Working Draft 5, November 1999. http://www.w3.org/TR/1999/WD-xmlschema-2-19991105.

[13] *XML Linking Language (XLink).* WWWC Working Draft, 26 July 1999. http://www.w3.org/1999/07/WD-xlink-19990726.

[14] *XML Pointer Language (XPointer).* W3C Working Draft, 9 July 1999. http://www.w3.org/1999/07/WD-xpointer-19990709.

[15] Кеплер Ф. *Стратегия управления сервером Web.* LAN/Журнал сетевых решений, Октябрь 1998.