

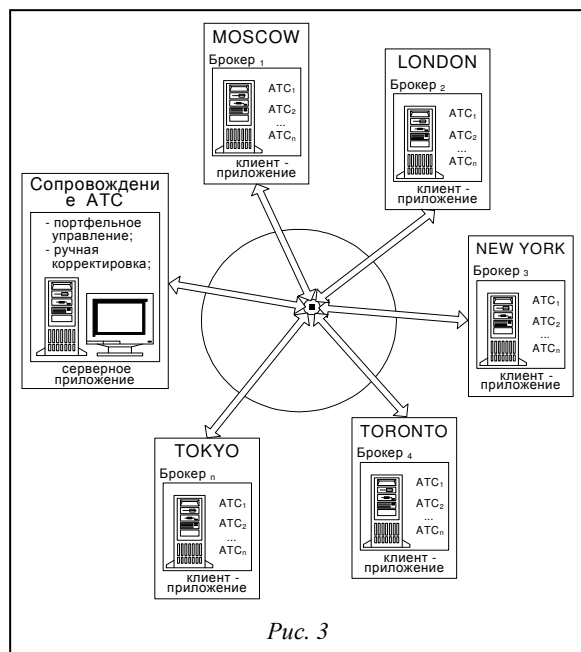


Рис. 3

вающие АТС, обмениваются с клиентскими приложениями через установленный протокол связи основополагающими данными и скриптами (программами, реализующими логику АТС).

Интернет-распределенная архитектура клиент-серверных АТС позволит принципиально изменить эффективность управления и распределения депозитов по АТС и брокерам, добиться глубокой диверсификации биржевых активов, избавиться от временного фактора – задержки связи, сократить

сопутствующие издержки, повысить живучесть и надежность всей системы в целом. Такой вид рисков, как отсутствие связи, будет сведен к минимуму – АТС устанавливаются на специально выделенном компьютере самого брокера. Диверсификация рисков станет возможной не только по инструментам и АТС, но и по странам. В случае сбоя или выхода из строя одного клиентского приложения остальные продолжают торговлю.

Для реализации подобной архитектуры требуются соответствующие инфраструктура и сервис, которые только начинают развиваться в данном направлении.

Торговые роботы являются следующим эволюционным этапом развития автоматической торговли, продуктом информационных технологий и новой экономики. Они способны привести к революции в биржевой индустрии, но заменить непосредственное участие человека в процессе биржевой торговли пока не смогут.

Литература

1. NYSE Rule 80A.40(b) Index Arbitrage Trading Restrictions. (<http://www.nyse.com/glossary/1042235995760.html>).
2. Federal Register (USA): April 17, 2007. – Vol. 72. – Num. 73. – pp. 19225–19227. (From the Federal Register Online via GPO Access. DOCID:fr17ap07-88.// wais.access.gpo.gov).
3. Чеботарев Ю. Торговые роботы на российском фондовом рынке. – М.: Омега-Л, 2006. – 136 с.
4. Беляев А., Евтушенко С. Принципы построения механических торговых систем. // Валютный спекулянт. – 2006. – № 6.
5. Яшин С. Базовые принципы создания торговых автоматов. // Там же. – № 5.

СОЗДАНИЕ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ АНАЛИЗА ПОСЕЩАЕМОСТИ САЙТА

Ф.Ф. Юсифов

(Институт информационных технологий НАН Азербайджана, г. Баку, farhadyusifov@gmail.com)

Ключевые слова: эффективность работы сайта, счетчик посещений, бесплатные продукты (системы), аналитический сервис, регистрационные журналы сервера.

Получение и обработка статистических данных о посетителях сайта – важная задача для определения эффективной маркетинговой стратегии и оптимального использования Web-сервера. Тщательный анализ журналов сайта может помочь в выработке четкой стратегии развития [1].

Тысячи больших и малых предприятий, устанавливая собственные Web-серверы, неизбежно сталкиваются с оценкой эффективности работы своего Web-сайта. Можно отметить, что, конечно же, на выбор подхода к оценке эффективности Web-сайта непосредственно влияет его назначение. Например, реализуя собственные товары или услуги, продавец должен с помощью своего Web-сайта предоставить соответствующую информа-

цию как можно большему числу потенциальных покупателей и убедить его посетителей приобрести их. Продавая рекламное пространство, необходимо максимально увеличить число посетителей Web-сайта, чтобы они как можно чаще видели размещенную на нем рекламу. В зависимости от характера бизнеса потребуется различная информация о посещениях. При продаже собственных товаров (услуг) наиболее важными будут сведения о том, как посетители используют Web-сайт. Вооружившись такой информацией, можно своевременно внести необходимые изменения и привлечь внимание дополнительного числа потенциальных клиентов к продукции, убедиться в том, что они ознакомились с предложениями.

В настоящее время крупный корпоративный сайт не может обойтись обычным счетчиком посещений, так как администратору сайта необходимо контролировать множество параметров и улучшать качество обслуживания *Web*-ресурсов. Среди контролируемых параметров – общее число хитов и хостов для сайта в целом, число посещений с поисковых систем, IP-адреса и время посещения сайта каждым из посетителей, ключевые слова (по которым сайт был найден), тип и версия операционной системы и браузера посетителя и др. [1,2].

Существует ряд бесплатных продуктов (известные системы *Analogue* или *WebStat*), как правило, реализованных в виде *Java*-апплета, читающего в режиме реального времени журнал сайта и отображающего на *html*-странице некоторое фиксированное количество отчетов. Эти решения предполагают относительно небольшой размер журнала и скромные потребности пользователей [1,3].

Также можно отметить аналитический сервис, который предоставляется на сайте третьей фирмы компаниям, предпочитающим не эксплуатировать программное обеспечение, а арендовать чужие вычислительные ресурсы. Пользователь располагает на своем сайте специальный скрипт-шпион (*spider*), который получает информацию от браузера посетителя и посылает его на сайт поставщика сервиса, где информация записывается в общую базу данных. При входе на личную страницу этого сайта пользователь сервиса может выполнять параметризованные запросы и получать отчеты заданной формы. Одним из популярных примеров такого сервиса на Западе является *WebTrends* компании *NetIQ*, а в России – *SplyLog* одноименной компании [1].

Популярность WWW продолжает расти, и поэтому есть потребность в разработке методики и инструментальных средств для обнаружения и обработки информации. Основная задача программных средств для анализа *Web*-трафика – извлечение полезной информации из регистрационных журналов сервера.

Анализ посещаемости сайта – процедура сбора информации об основных источниках трафика, о географии посетителей, выявление популярных страниц. Это позволяет определить сильные и слабые стороны сайта с точки зрения посетителей и выработать советы по улучшению его содержания и структуры.

Существуют три основных способа идентификации пользователей [1]:

- по IP-адресу компьютера посетителя – у данного способа максимальные недостатки по сравнению с остальными, поскольку сайт могут посетить несколько пользователей с одним и тем же IP-адресом, например, работающие через прокси-сервер;

- по файлам *cookies* – небольшим файлам с данными, которые *Web*-сервер при его посещении через браузер оставляет на компьютере пользователя; таким образом, во время следующего визита сервер знает, что данный пользователь уже был его посетителем ранее (основные недостатки обусловлены тем, что файлы *cookies* идентифицируют именно браузер пользователя, а не конкретного человека);

- при обязательной регистрации пользователей – в этом случае при посещении сайта или обращении к одному из сервисов сайта пользователь вводит свои имя и пароль, и система в течение всего визита может однозначно идентифицировать его; этот способ допускает меньше всего погрешностей при подсчете пользователей и их повторных визитов, но, к сожалению, применим лишь к отдельным сервисам.

Основными методами сбора информации о поведении посетителей на сайте являются анализаторы и использование статистики, получаемой при анализе лог-файлов сервера [1].

При проведении исследований необходимы специальные инструменты для анализа трафика. В связи этим можно отметить, что время, когда инструмент (счетчик) представлял собой простейший скрипт на платформе PHP или Perl, и файл, в котором хранилось число посещений, увеличиваемое на единицу при обращении к страницам сайта, на данный момент довольно примитивны. Теперь системы анализа посещаемости сайта являются серьезными *Web*-приложениями, требующими тщательной разработки.

Сайты крупных и даже некоторых небольших компаний, как правило, динамически генерируют *Web*-контент из содержимого базы данных. К сожалению, это осложняет анализ поведения посетителя сайта, поскольку большинство программ для анализа регистрационных журналов воспринимают их как ссылки на одну и ту же страницу. Необходимо убедиться в том, что приобретаемый для анализа регистрационных журналов инструмент позволит определять указанные параметры. Анализируя строки запросов целиком, можно получить более детальные отчеты.

Приведем сведения, которые должна собирать система сбора и анализа статистической информации по посещаемости сайта [1,2,4]:

- количество хостов (уникальных IP-адресов), хитов (общее число загрузок страниц) как по сайту в целом, так и по каждой странице;
- статистика по используемым посетителями операционным системам и браузерам;
- информация по IP-адресам и времени последнего посещения сайта каждым из посетителей;
- информация по количеству посетителей, пришедших на сайт с поисковых систем (*Rambler*, *Google*, *Yahoo!* и др.);

- количество посещений сайта со страниц, на которых расположены ссылки на сайт.

Для построения системы на платформе PHP потребуются, как известно, несколько таблиц в базе данных *MySQL*. Отметим, что необходим учет статистической информации для каждой страницы сайта, число которых не ограничено, необходима таблица, в которой будет храниться информация о каждой из страниц, участвующих в сборе статистической информации.

Рассмотрим один из основных модулей создания системы анализа посещаемости сайта на платформе PHP.

```

// Определение число посетителей с уникальными IP-
адресами за последние сутки
$query = "SELECT count(distinct ip) FROM ip
        WHERE putdate > NOW() - INTERVAL 1 DAY $tmp";
$num = mysql_query($query);
if (!$num) puterror("Ошибка при обращении к базе IP-
адресов...");
$total = mysql_result($num,0);
// Определение числа страниц
$number = (int)($total/$pnumber);
if((float)($total/$pnumber)-$number != 0) $number++;
.....
// Вывод ip-адреса
?>





```

Скрипт выводит таблицу, *\$nnumber* строк и столбцы: IP-адреса, хост и время последнего обращения.

Для тщательного анализа посещаемости сайта на основе некоторых довольно популярных рекомендаций *Web*-разработчиков можно сделать следующие выводы.

– По статистике, о посетителе известно только одно – его IP-адрес, но он ни о чем не говорит аналитику. Поэтому можно добавить модуль, который запрашивает по IP-адресу имя хоста. Однако это не всегда дает точную информацию, например, сервер посетителя может находиться в домене *com* и при этом располагаться в любой стране.

– Более точный способ получить код страны посетителя из его IP-адреса – выполнить запрос к одной из баз данных в сети, хранящей зарезервированные диапазоны IP-адресов. Для получения кода города или региона можно выполнять запрос к региональным серверам. Это важнейший вид анализа для глобального бизнеса – международного или в масштабах страны.

– Из строки ссылки для поисковых машин выделяются поисковые слова, что позволяет анализировать области целенаправленных интересов посетителей, а также соответствие этим интересам мета-тегов и контента сайта.

– Можно получать отчеты о количестве пользователей, зашедших на сайт, по ссылкам с других сайтов и с поисковых машин. Из полной ссылки на ссылающуюся страницу можно выделить тип ссылающегося сервера: поисковая машина, *News Groups*, ссылающийся сайт и собственный сайт. В таблицу журнала добавляется поле «Категория ссылающегося сервера», и создается связанный с ним справочник, также добавляется поле, в которое записывается имя сервера, вычлененное из ссылки.

- Можно отфильтровать категорию ссылающегося сайта, при этом страницы будут показывать маршрут движения посетителя.

Как отмечено ранее, сделать сайт более привлекательным и полезным для пользователей или потенциальных заказчиков может детальный анализ регистрационного журнала сайта с применением профессиональных *Web*-анализаторов. Однако многие администраторы, и без того перегруженные информацией, испытывают значительные трудности, интерпретируя данные регистрационных журналов *Web*-серверов. С этой целью для повышения эффективности работы *Web*-сайтов необходимо применение *Web*-анализаторов, способных принести компании максимум пользы. Для создания таких интеллектуальных *Web*-анализаторов могут использоваться методы *Data mining*.

Можно отметить, что с начала существования WWW разработчики Web-серверов придерживаются традиционного формата представления данных о регистрируемом сервером трафике. Для регистрации используются четыре основных файла: *access log* (журнал регистрации доступа), *error log* (журнал регистрации ошибок), *referrer log* (журнал ссылок) и *agent log* (журнал агентов) [5]. Комбинации этих журналов могут варьироваться, но именно они являются единственным источником информации о трафике. Используя данные этих файлов, можно тщательно анализировать пользователей сайта.

Основные принципы построения *Web*-анализатора можно предложить в следующей форме: инициализация, настройка параметров, чтение лог-файлов, обработка данных, семантический анализ, вывод результатов.

В заключение отметим, что область анализа *Web*-трафика и создание лог-анализаторов не теряют своей актуальности. Хорошим показателем может служить возросшее число научно-исследовательских работ в этой области. При этом особое внимание уделяется созданию интеллектуальных систем для анализа *Web*-трафика и идентификации пользователей.

Литература

1. Рабин Д. Изучайте журналы посещений. // Сети и системы связи. – № 1, 2005 (http://ccc.ru/magazine/depot/05_01/read.html?0201.htm).

2. Iváncsy R., Vajk I. Different Aspects of Web Log Mining. // 6th International Symposium of Hungarian Researchers on Computational Intelligence. Budapest, 2005.

Марков Р. WtSpy – считаем и контролируем трафик почтовых и прокси-серверов. // Системный администратор,

август, 2005.

4. Когтзолл Д. PHP 5. Полное руководство. – М.: Издат. дом, 2006.

5. Юсифов Ф.Ф. Извлечение знаний из Интернет с использованием лог-файлов. // Телекоммуникации. – 2006. – № 8.

ПРОГРАММНАЯ МОДЕЛЬ КОМПЛЕКСНОЙ ДИАГНОСТИКИ СОСТОЯНИЯ ПАЦИЕНТА

А.М. Унакафов

(Технологический институт Южного федерального университета, г. Таганрог, anton@rista.ru)

Ключевые слова: диагностический комплекс, разработка модели, алгоритмические блоки, режимы функционирования.

В настоящее время в науке, в том числе в медицине, все шире используются информационные технологии. В частности, интенсивно развивается комплексная диагностика состояния пациента с помощью специализированных программно-аппаратных комплексов.

Рассмотрим характерные проблемы, возникающие при разработке программных моделей комплексной диагностики состояния пациента:

- правильная формализация метода диагностики,
- автономность диагностического комплекса,
- простота и удобство интерфейса.

Была поставлена задача создания программной модели, позволяющей решить указанные проблемы, для реализации в рамках программно-аппаратного комплекса «Измеритель RR&T». В качестве метода комплексной диагностики выбран метод вариационной пульсометрии, основанный на анализе variability сердечного ритма [1].

Комплекс состоит из следующих структурных частей.

1. Аппаратная часть – автономное устройство «Измеритель RR&T» на основе микроконтроллера ATMEGA32.

2. Программа для аппаратной части комплекса обеспечивает в автономном режиме комплексную диагностику состояния пациента, прием данных с датчиков электрокардиограммы (ЭКГ)/фотоплетизмограммы (ФПГ) и протоколирование их. Результаты диагностики должны выводиться на экран устройства и при необходимости пересылаться на COM-порт компьютера. Интерфейс должен позволять пользователю легко управлять работой устройства.

3. Компьютерная программа позволяет считывать данные из памяти устройства, выводить их на монитор и сохранять на компьютере для последующей обработки методами анализа variability сердечного ритма, отличными от реализованного в устройстве.

При разработке модели учитывалось то, что реализовываться она будет на микроконтроллере ATMEGA32 (8-разрядный AVR-микроконтроллер

с расширенной RISC-архитектурой). Программирование на языках высокого уровня для микроконтроллеров имеет ряд особенностей:

- ограниченность в ресурсах памяти (для ATMEGA32 – 32 Кб энергонезависимой, перепрограммируемой (до 10 000 циклов перезаписи) Flash-памяти; 1 Кб энергонезависимой EEPROM-памяти (до 100 000 циклов перезаписи); 2 Кб энергозависимой памяти) и, как следствие, необходимость проведения оптимизации по объему кода программы;

- низкая разрядность контроллера;
- относительно низкая производительность (рабочая частота – до 8 МГц) и, как следствие, необходимость проведения оптимизации по времени выполнения;

- невозможность применения стандартных оптимизаторов.

Основные алгоритмические блоки модели показаны на рисунке.

В алгоритмическом блоке диагностики собраны действия по обеспечению работы метода ва-

