**Языки серверных скриптов**

Александр Качанов

В последнее время Web-узлы перестают быть просто набором страниц и картинок. Все чаще создаются узлы, которые взаимодействуют с посетителем, выполняют какие-то действия, собирают данные, ведут подсчеты, самонастраиваются под предпочтения посетителя. Вcе меньше Web-узлы становятся похожими на статическую библиотеку страниц, и превращаются в целые программные системы, связанные с базами данных. Все чаще можно встретить объявления Web-студий, приглашающих помимо Web-дизайнеров и администраторов еще и Web-программистов.

Для того, чтобы простая HTML-страница ожила, существует несколько технологий. Первое деление связано с местом осуществления этого оживления: на сервере или на клиенте. Далее деление ведется по методу программирования: с использованием интерпретируемых скриптов или откомпиллированных программ. Вышеизложенное можно представить в виде следующей таблицы:

|  |  |  |  |
| --- | --- | --- | --- |
| Сторона клиента | | Сторона сервера | |
| Скрипты | Программы | Скрипты | Программы |
|  JavaScript   JScript   VBScript |  Plug-ins   ActiveX   Java applets |  Perl   Tcl   командная оболочка (shell) |  отдельные CGI-программы   модули сервера   ISAPI/NSAPI   ActiveX   Java servlets |

Серверные скрипты, о которых речь пойдет в данной статье, относятся к серверной части, но стоят особняком, так как сочетают в себе свойства как скриптовых языков, так и программ.

Основное отличие от CGI-скриптов (которые пишутся чаще всего на языке Perl, а иногда на Tcl и даже на языке командных оболочек bash/tcsh) и от CGI-программ (которые пишутся на языках высокого уровня от C/C++ до Basic-a) – это то, что в CGI-программах вы сами пишете выводимый HTML-код, а используя язык серверного скрипта, вы встраиваете свою программу в готовую HTML-страницу, используя открывающий и закрывающий теги.

Отличие серверного скрипта от JavaScript, VBScript или JScript, как можно догадаться, состоит в том, что серверный скрипт выполняется на сервере, а клиенту передается результат работы, тогда как в JavaScript-код полностью передается на клиентскую машину и только там выполняется.

Ваша страница в этом случае представляет собой смесь HTML-кода с кодом программы. Процесс схож с тем, как работает SSI (Server Side Includes). Web-сервер настраивается таким образом, что файлы-страницы с определенным расширением (asp/php/cfm) не отправляются сразу клиенту, а перед отправкой пропускаются через обработчик, который производит разбор страницы и выполняет операции, в ней записанные. Таким образом клиенту выдается не та страница, которая хранится на сервере, а страница созданная на лету по результатам выполнения программы. Как видите, связка состоит из двух звеньев: скриптов в HTML-странице и обработчика, который представляет собой либо отдельную программу, либо для увеличения скорости работы встраивается в виде модуля в Web-сервер.

Преимущества:

Независимость от платформы. В большинстве случаев скрипты, написанные для одной платформы, можно перенести на другую платформу с минимальными изменениями или даже вообще без них. Главное, чтобы под данную платформу существовал "обработчик". В последнее время разработчики языков серверных скриптов стали уделять большее внимание переносимости и стараются выпускать обработчики под как можно большее количество платформ.

Открытость кода. Работая со скриптами, вы всегда можете быть уверены в том, что полностью контролируете выполнение программы. В отличие от готовых откомпиллированных программ вам нет нужды задаваться вопросом, а не исполняет ли программа какие-либо запрещеные действия? Имея исходный текст, вы всегда можете внести в программу такие изменения, которые в полной мере отвечают именно вашим специфическим запросам.

Недостатки:

Скорость работы. Несмотря ни на что, рассматриваемые нами языки - являются интерпретируемыми. Это значит, что они в большей степени уступают по скорости работы откомпиллированным CGI-программам.