

ОБЗОР ОСНОВНЫХ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЙ

Сенина А. А.

Тузовский А. Ф.

Томский политехнический университет

loff93@mail.ru

В настоящее время веб-приложения получили большое развитие в различных сферах деятельности общества. Работа веб-приложения производится посредством клиент-серверной технологии, где клиентом является браузер, а в качестве сервера выступает веб-сервер. Для начала приведем общую схему работы веб-приложений, которая описана ниже на рисунке 1.

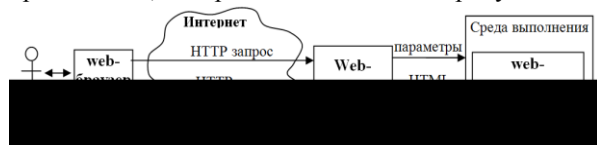


Рис. 1. Принцип работы веб-приложения

Из рисунка 1 видно, что клиент, обращаясь к веб-браузеру, отправляет HTTP-запрос по определенному URL адресу, указывающему на некоторый динамический ресурс, а именно само веб-приложение. Далее сервер формирует на основе веб-приложения HTML-страницу, которая посредством браузера отображается клиенту. Из описания схемы можно сделать вывод, что основная работа веб-приложения осуществляется на стороне сервера.

В настоящий момент существует множество технологий, реализующих логику веб-приложений на стороне сервера. В данной статье будут рассмотрены самые популярные из них.

Первой широко используемой технологией стала CGI (Common Gateway Interface), которая особенно применима для создания динамических веб-страниц и служит для обеспечения связи между клиентом (веб-браузером) и веб-сервером. Данная технология представляет собой набор правил, следуя которым, программа способна выполняться на различных серверах операционных системах. В соответствии с технологией CGI, HTTP запрос, содержащий ссылку на динамическую страницу, поступая на веб-сервер, генерирует новый процесс и запускает нужную прикладную программу. В результате реализации данной архитектуры возникает возможность реализации интерактивной составляющей работы веб-страницы, а именно обмен данными между элементами веб-страницы. Технология CGI позволяет использовать любой язык программирования, способный работать с устройствами ввода/вывода. Также в разработке веб-приложения можно использовать CGI скрипты, например Python, Perl, Tcl и т. д. Если в CGI программе содержатся скрипты, то при ее выполнении вызывается script engine (интерпретатор скриптов), которому передаются данные HTTP запроса и имя файла, содержащего

запрашиваемый скрипт. После выполнения данного скрипта программой клиенту возвращается сформированная HTML страница. В числе достоинств данной технологии стоит отметить ее независимость от клиентской части веб-приложения.

Несмотря на то, что технология CGI позволяет достаточно просто формировать динамическим образом информацию в веб-сети, она имеет значительные недостатки. Одним из главных недостатков является производительность. Причиной этого низкой производительности является сам процесс обработки HTTP запроса: для каждой обработки такого запроса веб-сервер генерирует новый процесс, который заканчивает свою работу только после завершения программы, что является достаточно трудоемким и при наличии множества таких процессов, начинается конкуренция за ресурсы оперативной памяти.

Следующей технологией, которая довольно широко распространена, стала Java Servlets или просто сервлеты. Данная технология позволяет решить проблему производительности путем выполнения всех запросов в одном процессе путем распределения их по потокам внутри процессов. Это означает, что программный код сервлета должен быть потоко-безопасным. Также плюсом использования сервлетов является их независимость от платформы, потому что они выполняются на виртуальной Java-машине. Java Servlets обладает широким функционалом, который достигим благодаря большому количеству библиотек. Сервлет-контейнер запускает сервлет, который является классом Java, при первом же обращении к нему или при специальном указании.

Java Servlet Pages (JSP) - это надстройка над технологией Java Servlets, разработанной ранее компанией Sun Microsystems, что означает, что их архитектуры взаимосвязаны. Данная технология обеспечивает быструю и упрощенную разработку web-приложений с применением шаблонного подхода. Шаблоны страниц JSP представляют собой шаблоны HTML страниц и крайне схожи с шаблонами ASP и PHP.

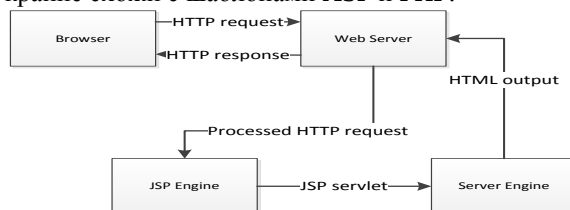


Рис. 2. Архитектура работы JSP технологии

Отличием данной технологии от других похожих является то, что при обращении к странице код, находящийся внутри тегов не интерпретируется, а предварительно компилируется в Java Servlet, то есть так, как если бы он располагалась внутри сервлета. Данная процедура производится один раз при первом запуске к странице или сервлет-контейнера, т.к. выполнение вышеописанной операции является трудоемкой. Так же, как и Java Servlets, JSP не привязана к конкретной программной или аппаратной платформе. JSP гармонично сочетает шаблонную реализацию страниц и все имеющиеся преимущества платформы Java.

Новейшей технологией разработки web-приложений является .NET технология, разработанная компанией Microsoft. Платформа .NET значительно упростила процесс разработки приложений и повысила надежность кода. Стали доступными функции автоматического управления временем жизни объектов, обработка исключений и их отладка, в наличии появились библиотеки, нейтральные к языкам программирования. Набор стандартных базовых классов обеспечивают разработчику доступ к сервисам платформы при использовании любых языков программирования, совместимых с .NET. Common Language Runtime совместно с базовыми классами составляют основу платформы .NET и предлагает разработчикам высокоуровневые сервисы, такие как ADO.NET (усовершенствованный ADO, используемой SOAP и XML с целью обмена данными), ASP.NET (новое поколение ASP, дающий возможность использовать любой язык программирования, совместимый с .NET) и Windows Forms и Web Forms (классы, реализующие локальные и web-ориентированные приложения). Расширенная версия архитектуры платформы ASP.NET представлена на рисунке 3.

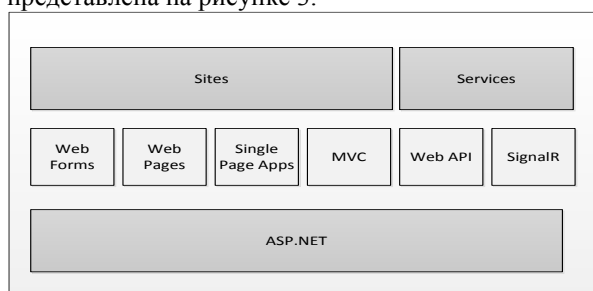


Рис. 3. Архитектура платформы ASP.NET

Компилирование исходного кода происходит по следующей схеме: создается код на промежуточном языке (Microsoft Intermediate Language). В отличие от старой версии, где компилятор создавал машинный код, данный вид компиляции позволяет скомпилированному файлу выполняться на платформе любого процесса. Новые возможности ASP.NET соответствуют современным требованиям. Вот лишь некоторые из них: большой набор библиотек, языковая

независимость платформы, новые пути обработки ошибок и т.п.

В результате рассмотрения вышеперечисленных технологий разработки веб-приложений можно сформировать следующие обобщенные характеристики платформ. В первую очередь, выделяется выполнение запросов клиентской части веб-приложения отдельно при помощи генерации программы, обрабатывающей данные запросы. Описанный подход присущ выполнению CGI-скриптам. Следующей не менее важной характеристикой является повторное выполнение запущенного процесса без повторного запуска программы, отвечающей за обработку процесса. Данная функция реализована в технологиях Java Servlets. Широко распространен шаблонный подход при разработке веб-приложений. При загрузке веб-страницы шаблон автоматический заполняется содержимым, что демонстрируют технологии ASP, JSP и PHP.

Основные характеристики технологий разработки веб-приложений описаны в таблице 1. В терминах данной таблицы «+» - поддержка в недостаточной степени, «+» - поддержка в полной мере.

Таблица 1. Характеристики технологий разработки мобильных веб-приложений

	PHP	Java Servlets	JSP	ASP.NET
Независимость от платформы	±	+	+	±
Производительность	±	±	±	+
Поддержка языков реализации	±	+	+	+
Простота использования	±	±	+	+
Наличие необходимых библиотек	+	+	+	+
Разделение дизайна и логики	±	±	±	+

Таблица 1 была составлена по результатам исследования характеристик технологий разработки веб-приложений, а также с учетом особенностей каждой из платформ, что позволяет произвести конструктивный анализ существующих решений.

Каждая из вышеперечисленных платформ имеет возможности и ограничения в индивидуальном порядке, а также свою собственную область применения, что предоставляет разработчику широкий выбор инструментов разработки.

ЛИТЕРАТУРА

1. А. Ф. Тузовский. Проектирование интернет приложений. – М.: Изд-во Томского политехнического университета, 2010. – 200с.