

АНАЛИЗ ПРИМЕНЕНИЯ ТЕХНОЛОГИЙ PHP И ASP.NET ПРИ РАЗРАБОТКЕ ПРОГРАММНЫХ ПРИЛОЖЕНИЙ

А.Д.АСКАРОВА, Ж.Э.КАЗЫБАЕВА
[E.mail. ksucta@elcat.kg](mailto:ksucta@elcat.kg)

Берилген макалада web приложения жасалуучу эки технологиянын салыштыруусу каралган

В статье сравниваются две технологии для создания web-приложений PHP и ASP.NET.

In this article comparison of two technologies for creation of web of the PHP and ASP.NET appendices is considered

На сегодняшний день у разработчиков веб-сайтов есть богатый выбор языков и технологий для создания сайта: Perl, PHP, ASP, ASP.Net, JSP, Coldfusion... Самыми распространенными сегодня являются PHP и ASP.Net.

PHP (Hypertext Preprocessor) – язык для написания серверных сценариев (скриптов). Интерпретатор языка бесплатен, с открытым исходным кодом, созданы версии для различных веб-серверов – прежде всего, для Apache и Internet Information Services. Актуальная на сегодняшний день версия интерпретатора – PHP5. В последнее время появилась встроенная поддержка PHP в Internet Information Services 7-й версии (поставляется с Windows Server 2008) – ранее PHP можно было использовать только как внешнее расширение /1/.

ASP.Net – технология, предназначенная для создания веб-сайтов, веб-сервисов и приложений, разработанная корпорацией Microsoft. Актуальная сегодня версия – 4. Стоит заметить, что ASP.Net кардинально отличается от своего предшественника – ASP. Принцип работы ASP.Net-приложения совершенно иной /2/.

Как можно сравнивать эти две технологии, ведь принцип их работы кардинально отличается? Тем не менее, эти технологии решают одну задачу – создание веб-сайтов, и неважно, какими средствами это достигается.

У ASP.Net в отличие от PHP, принцип работы совершенно иной – каждая страница состоит из двух файлов: файла с HTML-разметкой и ASP.Net-контролами (файл с расширением Aspx) и файла серверной логики (расширения vb или cs). Идея такова: в файле разметки – только разметка (фактически, описание внешнего вида страницы), в файле логики – только логика (программный код). И хотя можно и добавлять программный код в файл разметки, и в файле логики формировать HTML-разметку (фактически, использовать подход, аналогичный PHP), поступать так не рекомендуется – нарушается общая идея – отделение внешнего вида страницы от логики.

В ASP.Net попытка Microsoft применить при разработке Web-приложений те же принципы, что и при разработке Windows-приложений. Например, на страничке создается кнопка, для которой пишется обработчик события, который при нажатии, например, меняет текст у какого-нибудь серверного контроля, и так далее. Но такая модель не совсем применима при разработке веб-приложений. Протокол HTTP – это сеансовый протокол, работающий по принципу запрос-ответ. А Windows-приложения работают непрерывно. Поэтому для реализации аналогичного подхода пришлось применять множество абстракций и приемов. Достоинством их является то, что значительно ускоряется процесс разработки веб-приложения. Но если программист начинает изучение веб-разработки с ASP.Net – весьма вероятно, что он не сразу поймет истинные (физические) принципы работы веб-приложений. Например, если нужно получить от

посетителя сайта какие-нибудь данные и обработать их (например, записать в базу данных), PHP-разработчику приходится создать html-форму, соответствующий скрипт, который получит от нее данные, обработает их и вернет посетителю ответ также в виде html-разметки – так и работают все веб-приложения: запрос-ответ. Если аналогичную задачу решает ASP.Net-разработчик, то форма за него уже создана, он лишь добавляет поле для ввода и кнопку, на которую прикрепляет обработчик события, в котором выполняются нужные действия по обработке введенных данных.

Очень похоже на то, как бы задача решалась в Windows-приложении. И начинающий разработчик может и не знать, что на самом деле при нажатии на кнопку реально произойдет отправка данных с формы, они будут автоматически обработаны, затем будет выполнена его логика, после чего автоматически будет выполнено формирование ответной HTML-разметки, которую получит посетитель.

Иными словами, начиная изучение сферы веб-приложений с PHP, программист раньше поймет и увидит реальные принципы работы Web, чем программист ASP.Net, от которого эти самые принципы пытаются скрыть. Принципы работы PHP гораздо проще, нежели ASP.Net. Гораздо проще понять его принципы работы PHP, чем сложную модель ASP.Net

PHP сам по себе является языком программирования. Используется Си-подобный синтаксис, отсутствует типизация переменных, в 5-й версии значительно улучшена поддержка объектно-ориентированного программирования.

При создании веб-приложений ASP.Net можно использовать любой язык из платформы .Net – C# или Visual Basic.Net. Доступна вся функциональность библиотеки .Net Framework /3/.

Здесь со значительным перевесом выигрывает ASP.Net.

Во-первых, по возможностям PHP далеко позади от .Net-языков. Например, хотя в PHP5 (который вышел лишь летом 2004 года) и была значительно улучшена поддержка объектно-ориентированного программирования, но она по-прежнему слабее, чем в .Net, где языки (по крайней мере, C# и VB.Net) являются полностью объектными. В них нельзя создать просто функцию – только метод класса, нельзя создать просто переменную – только атрибут класса и т.п. В PHP нет перечислений (enum), нет событий («ручное создание» нельзя считать событиями, так как фактически встроенной возможности программирования с использованием модели событий нет), нет возможности перегружать методы (вследствие отсутствия типизации в языке) – это лишь некоторые неприятные особенности, с которыми приходится сталкиваться, используя PHP после серьезных объектно-ориентированных языков. В PHP нет поддержки многопоточности, есть проблемы с поддержкой Unicode.

Во-вторых, PHP – гораздо менее строгий язык, чем .Net-языки. В первую очередь – из-за отсутствия типизации. Переменные объявлять необязательно. В результате – множество опасных конструкций, которые не скомпилируются даже в компиляторе Си, в PHP будут выполнены с непредсказуемым результатом. Некоторые из ошибок (например, неинициализированные переменные) можно увидеть, если включить режим отображения всех ошибок (error_reporting=all), но об этом нужно знать и помнить. Если человек начинает изучение программирования с PHP, то далеко не факт, что он будет информирован о необходимости «дополнительной настройки». Рано или поздно возникнет такая ситуация, когда пропущенная ошибка приведет к серьезным последствиям (взлом, некорректность работы и т.п.). В .Net-языках такое сделать не получится: пока не будет исправлена ошибка – программа не скомпилируется.

Бесплатность PHP – одно из его важнейших преимуществ перед другими языками программирования. Общеизвестно, что весь LAMP-стек (Linux Apache Mysql PHP) бесплатен. Бесплатны Unix-подобные операционные системы, бесплатен веб-сервер Apache, бесплатен интерпретатор PHP и бесплатна база данных MySQL (за исключением ее коммерческого использования, в этом случае лицензией предусмотрена оплата за базу данных). Со средами разработки ситуация иная. Например, Zend Studio – платный продукт. За Zend Encoder, который позволяет защитить приложение путем его компиляции, тоже нужно платить /4/.

Кроме того Express-версия Visual Studio полностью бесплатна и обладает довольно широкими возможностями. Также бесплатна и Express-версия SQL-сервера (которая имеет весьма несущественные ограничения, вроде поддержки баз только до 4 Гб, но если у вас проект настолько масштабный, что этого объема не хватает – скорее всего, и средства в него вкладываются серьезные, на фоне которых стоимость платной версии может оказаться невысокой). Проводится множество акций, в рамках которых можно получить бесплатно операционные системы, среды разработки и т.п. Реализуется множество студенческих программ, программ сотрудничества с вузами, научными центрами и т.д. Одним словом, за последние несколько лет Microsoft сделала свои продукты значительно более доступными /4/.

Традиционно считается, что PHP больше подходит для маленьких проектов, а ASP.Net – для больших. Но, как известно, под PHP существует множество шаблонизаторов и фреймворков (Smarty, Zend Framework и др.), которые приближают его к ASP.Net с точки зрения архитектуры приложения (отделяется программный код от HTML-разметки, присутствует подобие ASP.Net-контролов и т.п.). И примеров крупномасштабных проектов, написанных на PHP, достаточно: vkontakte.ru, photofile.ru и т.п. В то же время и на ASP.Net можно легко и быстро (возможно, даже быстрее, чем на PHP) создать простой веб-сайт.

Используя любую из описанных технологий, можно создать современные информационные продукты, отвечающие требованиям современных информационных технологий.

Список литературы

1. **«PHP5. Практика создания web-сайтов»**, Кузнецов, Симдянов, Голышев, «БХВ-Петербург», 2005.
2. http://www.ti-net.ru/publikatsii/saity_na_aspnet/
3. ASP.NET 3.5, компоненты AJAX и серверные элементы управления для профессионалов: Роб Камерон, Дэйл Михалк. – СПб.: Вильямс, 2009. – 608 с.
4. <http://www.php.ru/>
5. Microsoft ASP.NET 2.0 с примерами на C# 2005 для профессионалов: Мэтью Мак-Дональд, Марио Шпушта. – СПб.: Вильямс, 2007. – 1408 с.
6. AJAX и PHP. Разработка динамических веб-приложений: Кристиан Дари, Богдан Бринзаре, Филип Черчез-Тоза, Михай Бусика. – СПб.: Символ-Плюс, 2009. – 336 с.