



5. The Trellis Dynamic Infrastructure Optimization Platform [Электронный ресурс]. URL: <http://www.emersonnetworkpower.com/en-US/trellis/Documents/Brochures/TRELLIS-BRO-EN.pdf> (время доступа: 22.08.2013)
6. Romonet: The Holy Grail of Data Center TCO [Электронный ресурс]. URL: <http://www.romonet.com> (время доступа: 22.08.2013)
7. Srinivasan V. et al. Energy-aware task and interrupt management in linux // Ottawa Linux Symposium. 2008.
8. Ивашко Е. Е., Румянцев А. С., Чухарев А. Л. Задача прогнозирования нагрузки для повышения энергетической эффективности вычислительного кластера // Параллельные вычислительные технологии (ПаВТ'2013): труды международной научной конференции (г. Челябинск, 1-5 апреля 2013 г.). Челябинск, Издательский центр ЮУрГУ, 2013. С. 363-370.

В.В. Козлов

ВЫСОКОПРОИЗВОДИТЕЛЬНЫЕ НАТИВНЫЕ СЕРВЕРНЫЕ WEB ТЕХНОЛОГИИ

(Самарский государственный архитектурно-строительный университет)

Распространенные WEB технологии (PHP, ASPX, NodeJs, Python, ...) являются скриптовыми и удовлетворительно работают при низких нагрузках на сервер. При повышении нагрузки требуется горизонтальное масштабирование серверного оборудования, то есть создания серверных ферм, что влечет за собой значительное удорожание поддержки таких сайтов. Таким образом, можно выделить основные недостатки скриптовых WEB языков:

- 1) низкая производительность на один сервер – предопределено динамической структурой скриптовых языков программирования;
- 2) снижение эффективности кеширования контента и значительное увеличение трудоемкости разработки при горизонтальном масштабировании системы (увеличении количества серверов).

Технология Java выходит за рамки скриптовых языков. Однако использование Java локализовано в так называемом Enterprise секторе для корпоративных приложений, а самым массовым WEB языком, в настоящее время, является PHP.

Одними из наиболее крупных потребителей и движущей силой PHP технологии являются FaceBook и V Kontakte (Россия). Однако, даже перед этими компаниями достаточно давно остро встал вопрос недостаточной производительности применяемых ими технологий.

Рассмотрим историю решения вопроса производительности информационных систем FaceBook [1] (состояние вопроса на 2009 год). В 2010 компания опубликовала технологию HipHop [2], суть которой состояла в трансляции языка PHP в код на C++ с последующей его компиляцией, что позволило поднять



производительность до 5-50 раз. В настоящее время (конец 2013 года) FaceBook экспериментирует с технологиями виртуальной машины PHP (компиляция в байт-код) [3], позволяющей ускорить выполнение PHP скрипта до 9 раз.

Другой крупной компанией, применяющей подобный подход является российская компания Vkontakte. В компании разработали технологию kPHP (kitten PHP, 2013 год) [4]. Технология, аналогично тому как это сделано в FaceBook, осуществляет компиляцию PHP кода в код на языке C++ и последующую машинную трансляцию. По утверждениям компании Vkontakte их результат подхода существенно превосходит результат компании FaceBook.

Можно продолжать приводить подобные примеры. Однако, при трансляции PHP была принесены жертвы, например: пришлось отказаться от всего ООП языка PHP, отказаться от средств динамического программирования (инструкция eval).

Для преодоления этого в марте 2014 году компания FaceBook выложила релиз нового языка WEB программирования - Hack [5]. Особенность языка является его статическая типизация и выполнение на виртуальной машине PHP, представленной FaceBook ранее.

Таким образом, ведущие Internet компании мира ускоряют свои WEB приложения за счет применения нативного кода, который непосредственно будет взаимодействовать с клиентом.

Автором предлагается подход по непосредственному использованию нативного кода для WEB скриптов. Основная проблема разработки WEB приложений на C++ это неосведомленность программистов, и следовательно малая распространенность подходящих инструментов. В настоящее время таких инструментов разработано большое количество. Вот некоторые из них:

- 1) C++ server page [6] – аналог PHP по способу использования путем встраивания в HTML код (последнее обновление 2008 год);
- 2) CppCMS [7] – последнее обновление: середина 2013 года;
- 3) A C++ WebToolkit [8] – немецкая разработка, позволяющая полностью абстрагироваться от знания браузерных WEB технологий и писать GUI WEB приложение как обычное desktop приложение (последнее обновление март 2014 года) ;
- 4) C++ dynamite for the web [9] – аналог PHP по способу использования путем встраивания в HTML код (последнее обновление 2014 год);
- 5) High-speed C++ MVC Framework for Web Application [10] – японская быстро развивающаяся WEB технология, базирующаяся на библиотеки Qt.

Однако все описанные технологии достаточно сложны в изучении и имеют свои особенности. Требовалась простая в обслуживании, только серверная технология, позволяющая писать сервлеты в виде простых консольных приложений без специфики WEB, то есть узкоспециализированная, нишевая технология с низким порогом вхождения.

Разработанная технология использования произвольного нативного кода для сервлетов имеет много общего с технологиями CGI [11], но отличается от нее большей абстракцией от специфики WEB.



Для реализации разработанной технологии был разработан HTTP сервер, основанный на принципах:

- 1) переносимость кода основного модуля между различными операционными системами, включая мобильные;
- 2) требование «нуль-администрирования» - максимально простое развертывание на целевой системе;
- 3) высокая производительность;
- 4) модульное расширение с возможностью использования любого языка программирования, позволяющего создавать исполняемые модули, и не требующее от разработчика знания протокола HTTP;
- 5) прозрачная для разработчика дополнительных модулей авторизация с поддержкой авторизации по нескольким таблицам пользователей и встроенная система генерации сессий;
- 6) возможность встраивания всей полученной системы в популярные Web сервера (например, Nginx).

В настоящий момент реализованы методы «GET» и «POST» [12], а также поддерживается код продолжения передачи «HTTP/1.1 100 Continue» для обеспечения корректного продолжения приема данных, передаваемых через «POST»; необходимости в реализации всех HTTP методов (PUT, DELETE, ..) не было.

Серверная часть написана на языке C++ с использованием кроссплатформенной библиотеки Qt. Серверная часть является многопоточной и каждый HTTP запрос обрабатывается в отдельном потоке.

Модульное расширение реализуется через общение сервера приложений и модулей через стандартный поток ввода-вывода и передачу стандартизированной информации через системные переменные. Таким образом, модуль, расширяющий систему, представляет собой простое консольное приложение, получающее информацию стандартными операторами ввода и отвечающее через стандартный поток вывода. При этом модуль представляет собой классическое консольное приложение и может быть запущен и отлажен вне Web окружения. Данные авторизации и сессии передаются посредством установки соответствующих системных переменных окружения.

Встроенная аутентификация проводится на основе имени клиента, задаваемого в виде «имя@группа», где в случае отсутствия параметра «группа» сначала проводится расширение имени до полного по специальной таблице. Затем на основе параметра «группа» динамически формируется SQL запрос к соответствующей таблице и таким образом реализуется возможность разделения групп пользователей по различным таблицам. Это необходимо для формирования индивидуального набора полей для каждой из групп пользователей, а также в связи с большим числом пользователей. Для разгрузки базы данных вся информация кешируется в приложении и повторно из базы не запрашивается.

Для облегчения программирования клиентской части (Jscript) сервер имеет ряд дополнительных встроенных функций, например, агрегирования файлов JS скриптов и передача их единым массивом с гарантированным порядком



и возможностью проверки их корректной загрузки. Необходимость этой серверной функции обусловлена тем, что в современных браузерах для ускорения реализован многопоточный метод запроса всех ссылок из HTML файла и в случае наличия взаимосвязей JS функций возможны коллизии. Кроме того, не гарантируется, что к моменту выполнения JS на странице, все JS ресурсы будут подгружены.

На базе созданной среды WEB разработки были реализованы следующие проекты:

- 1) «Мониторинг реализации проектов по обеспечению формирования системы взаимодействия университетов и учреждений общего образования по реализации общеобразовательных программ старшей школы, ориентированных на развитие одаренных детей» - реализован по контракту с Минобрнауки РФ в 2011-2013 годах;
- 2) проект «ОДАРМОЛ», представляющий собой WEB сайт поддержки индивидуального руководства учеными вузов творческих работ учащихся школ – создан в 2013 году по заказу ассоциации строительных вузов России;
- 3) система тестирования СГАСУ, которая была применена в период зимней сессии 2013/14 учебного года.

Опыт применения созданной платформы WEB разработки на нативном коде показал ее высокую эффективность, высокую скорость разработки и простоту при ее эксплуатации.

Литература

1. «High Performance at Massive Scale: Lessons Learned at Facebook» [Электронный ресурс] - Режим доступа: <http://idleprocess.wordpress.com/2009/11/24/presentation-summary-high-performance-at-massive-scale-lessons-learned-at-facebook/> - Загл. с экрана.
2. «PHP Creator Asks, Is Facebook's HipHop Just a 'Nifty Trick'?» [Электронный ресурс] - Режим доступа: <http://www.nytimes.com/external/readwriteweb/2010/02/03/03readwriteweb-php-creator-asks-is-facebooks-hiphop-just-a-90057.html> - Загл. с экрана.
3. «Facebook invents a PHP virtual machine» [Электронный ресурс] - Режим доступа: <http://news.techworld.com/applications/3461086/facebook-invents-a-php-virtual-machine/> - Загл. с экрана.
4. «kPHP от ВКонтакте» [Электронный ресурс] - Режим доступа: <http://habrahabr.ru/post/187214/> - Загл. с экрана.
5. «Hack: a new programming language for HHVM» [Электронный ресурс] - Режим доступа: <https://code.facebook.com/posts/264544830379293/hack-a-new-programming-language-for-hhvm/> - Загл. с экрана.
6. «C++ server page» [Электронный ресурс] - Режим доступа: <http://www.micronovae.com/CSP.html> - Загл. с экрана.
7. «C++CMS» [Электронный ресурс] - Режим доступа: <http://cppcms.com/wikip/ru/page/main> - Загл. с экрана.
8. «A C++ WebToolkit» [Электронный ресурс] - Режим доступа: <http://www.webtoolkit.eu/wt/download> - Загл. с экрана.



9. The Tntnet project C++ dynamite for the web / <http://www.tntnet.org/> - Загл. с экрана.
10. High-speed C++ MVC Framework for Web Application <http://www.treefrogframework.org/> - Загл. с экрана.
1. RFC 3875, «The Common Gateway Interface (CGI) Version 1.1» [Текст], October 2004
11. RFC 2616, «Hypertext Transfer Protocol -- HTTP/1.1» [Текст], June 1999

Л.В. Логанова

ВЫБОР МАТЕМАТИЧЕСКОЙ МОДЕЛИ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА С УЧЕТОМ ОСОБЕННОСТЕЙ АРХИТЕКТУРЫ СУПЕРКОМПЬЮТЕРА СЕРГЕЙ КОРОЛЁВ

(Самарский государственный аэрокосмический университет)

Введение. При решении задач математического моделирования основной интерес исследователей, прежде всего, связан с разработкой или выбором эффективного численного метода. Однако, зачастую при выборе численного метода, в тени остаются вопросы, связанные с архитектурой аппаратных средств и программным обеспечением, используемым для его реализации. Традиционно эти вопросы в литературе рассматривались отдельно [1-4]. Исключение составляют, пожалуй [5-6].

Однако в последние годы появились работы, в которых совместно рассматриваются математические основы параллельных численных методов и параллельных вычислительных систем (ВС). Для параллельных ВС особенно актуальной становится проблема согласованности алгоритмов со структурой таких систем. Эффективная реализация алгоритма на выбранной параллельной вычислительной системе может быть не обеспечена в результате недостаточной загрузки устройств, а также недостаточного учета особенностей коммуникационной сети, соединяющей их. Вместе с тем, еще 70-е годы прошлого столетия академиком Г.И. Марчуком было сформировано соответствующее направление исследований, получившее название «отображение проблем вычислительной математики на архитектуру вычислительных систем». Данное направление включает в себя: структурный анализ алгоритмов, изучение математических особенностей архитектур вычислительных систем, автоматизацию процессов распараллеливания программ, разработку и исследование параллельных численных методов [7].

В настоящее время с активным внедрением в повседневную практику GP GPU интерес к данной тематике многократно возрос. При этом для совместного исследования параллельных алгоритмов и ВС необходимо, чтобы средства их описания были унифицированы. Для построения математической модели алгоритма, реализуемого на суперкомпьютере «Сергей Королев», приведем несколько нотаций представления математических моделей алгоритмов и ВС.